

Universität Karlsruhe (TH)
Fakultät für Informatik
Institut für Betriebs- und Dialogsysteme
Prof. Dr. Hartmut Prautzsch



Rekonstruktion farbiger Objekte aus strukturiert beleuchteten Ansichten

Diplomarbeit von

黎
顥

Hao Li

Juni 2005
(aktualisiert: Juli 2005)

Betreuer: Prof. Dr. H. Prautzsch, Dipl.-Inform. R. Straub

Erklärung

Hiermit erkläre ich, die vorliegende Arbeit selbständig erstellt und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt zu haben.

Karlsruhe, den 30. Juni 2005

In Memoriam Patris Mei
Dr. Li Chuan-Tseng (1947–2004)

Danksagung

An dieser Stelle möchte ich mich ausdrücklich bei Herrn Dipl.-Inform. Raphael Straub für seine intensive und nicht zu übertreffende Betreuung, die er bis zur letzten Stunde mit hoher Verantwortung übernommen hat, danken.

Weiterhin möchte ich Herrn Prof. Dr. rer. nat. Hartmut Prautzsch danken, dass ich die Freiheit bekommen habe, in den Gebieten zu forschen, die mich interessiert haben. Seine wertvolle Unterstützung hat es ermöglicht, meine Ziele schneller zu erreichen, als ich es mir vorstellen konnte. Weiterhin möchte ich Herrn Prof. Dr. Ing. Lars Linsen für die ausführlichen Auskünfte zu den Flächendarstellungen von Punktwolken, die eines der Kernprobleme dieser Arbeit sind, danken.

Für die zusätzlichen Korrekturen und Verbesserungsvorschlägen, sowie Hinweise auf Unstimmigkeiten möchte ich meinen Dank an Herrn Wolfgang Globke richten.

Mein besonderer Dank gilt meiner Schwester Jin Li, die für die Beschaffung der menschlichen Schädel, die als Testobjekte in dieser Arbeit verwendet wurden, verantwortlich war. Ebenso danke ich Herrn Janczak vom Institut für Anatomie und Zellbiologie der Universität Heidelberg für die Erlaubnis zur Verwendung dieser Objekte.

Für ihr Interesse und die Zusammenarbeit bei der Rekonstruktion von Freiformflächen, möchte ich ebenfalls Herrn Prof. Dr. rer. nat. Klaus Meierer und Herrn Dipl.-Ing. Udo Beyer vom Institut für Grundlagen der Gestaltung der Fakultät für Architektur herzlich danken.

Bei der Thomas-Gessmann-Stiftung, der Markel-Stiftung und der Richard-Winter-Stiftung bedanke ich mich für die großzügige Förderung meines Studiums. Hierbei möchte ich zusätzlich Herrn Prof. Dr. es-sciences physiques Jacques Calmet und Herrn Prof. Dr. rer. nat. Enrico Leuzinger danken.

Nicht zuletzt möchte ich mich bei meiner Mutter, meiner unglaublich lieben Freundin Lee Yuanshan und bei allen meinen Freunden aus Saarbrücken für ihre unendliche Geduld und Nachsicht bedanken. Diese Arbeit möchte ich meinem verstorbenen Vater widmen, der mein Interesse an der Wissenschaft geweckt hat.

Kurzfassung

Die Erfassung und Rekonstruktion von dreidimensionalen Objektoberflächen ist ein intensiv erforschtes Gebiet, welches einen wichtigen Bestandteil vieler Anwendungen darstellt. Um einen breiteren Einsatz von 3D-Scanverfahren zu ermöglichen, müssen Methoden entwickelt werden, die mittels einfachster Hardware hochqualitative Flächen rekonstruieren. Insbesondere muss, für die Flexibilität des Verfahrens, die Schnelligkeit und die Robustheit der Erfassungen von komplexen Objekten gewährleistet werden.

Diese Arbeit beschreibt ein vollständiges Scansystem, das nur mit Hilfe eines Projektors und einer Digitalkamera, durch Projektion von farbigem strukturier-tem Licht, die Oberflächengeometrie eines Objekts erfasst und ein hochauflösendes Dreiecksnetz rekonstruiert, das frei von Ausreißern und Rauschen ist. Dazu werden neue Methoden präsentiert, die nur drei Aufnahmen benötigen und dennoch robust gegenüber hochfrequenten Selbstverdeckungen und schlecht reflektierenden Oberflächen sind.

Für die Erfassung der Objektgeometrie werden farbige Streifensequenzen projiziert, die den detektierten Kanten der Aufnahmen zugeordnet werden, um durch optische Triangulierung eine Punktwolke zu erzeugen. Insbesondere haben diese Streifensequenzen lokale, sowie globale Eigenschaften, die die Eindeutigkeit einer vorgegebenen Teilsequenzgröße garantieren. Dadurch können Falschzuordnungen leichter erkannt und beseitigt werden. Die Zuordnung wird dadurch gelöst, dass die eindeutig kodierten Streifenübergänge anhand der Streifenmitten, korrigiert durch einen Ballungsprozess, zugeordnet werden. Zusätzlich wird in dieser Arbeit, ein Flächenrekonstruktionsschritt in der Rekonstruktionspipeline integriert. Wichtige A-Priori-Kenntnisse vorheriger Schritte werden ausgenutzt, um ein topologisch konsistentes Dreiecksnetz zu erzeugen. Im Gegensatz zu den meisten Verfahren, die nur von der reinen Geometrie der Punktwolke ausgehen, kommt das vorgestellte Verfahren auch mit unregelmäßigen Punktdichten zurecht.

Das resultierende Verfahren ist zur Erzeugung hoch detaillierter Flächenrekonstruktionen von farbigen Objektoberflächen geeignet. Experimentelle Ergebnisse werden anhand einer Vielzahl von Beispielen demonstriert und es werden insbesondere die Vorteile gegenüber traditionellen Zuordnungsmethoden, wie etwa die des dynamischen Programmierens, dargestellt.

Abstract

The acquisition and reconstruction of 3D objects is an active area of research. In order to make 3D scanning technology commonly available to the public, it is important to develop techniques with minimal hardware requirement and yet produce high quality surfaces. Flexibility, speed and robustness of the acquisition is also of critical importance.

In this project, we present a complete scanning system that only requires a projector and a digital camera. The shape of an object is acquired through optical triangulation using a novel color structured light pattern. A high resolution triangle mesh, which is purged from outliers and noise, is then generated. New methods that make use of additional local spatial continuity assumption are introduced in order to be robust against high frequency occlusions and inappropriate reflections. Only three shots are needed for the reconstruction from each view.

In the acquisition process, uniquely encoded stripe boundaries are projected onto the object. The acquired stripe boundaries are detected in the photographs before they are corresponded with the projected ones. This generates a point cloud after optical triangulation between the corresponded data. Our stripe sequences have local and global uniqueness properties of a predefined subsequence length. This makes wrongly corresponded stripe boundaries, that yield outliers, easily detectable for annihilation. In addition, we use the colors located on the medial axes of the acquired stripes for a more reliable correspondence. Although these acquired colors are closer to the projected ones, they are highly susceptible to spatial discontinuities and occlusions. We present a method to overcome this by clustering those medial axes w. r. t. their corresponding stripe boundaries and by considering the median values of the clusters. An interscanline labeling algorithm is used for the clustering. This way, we are able to partly shift the labeling problem between projected and acquired data to a labeling between two acquired data, which is similar to stereo vision correspondence.

The reconstruction of the surface is tightly integrated into the reconstruction pipeline. It is therefore possible to exploit crucial a-priori-knowledge of previous stages to generate a topologically consistent triangle mesh, reusing the assumed local space coherence. In contrast to most algorithms that approximate the surface only from an unorganized point cloud, the proposed method is insensitive to irregularly sampled points.

Finally, we have successfully demonstrated the quality of our results on a large variety of objects that contain many occlusions and have poor reflection properties. Advantages (in terms of robustness) over previous works, such as those based on dynamic programming, are experimentally shown.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Anwendungen	3
1.2	Verwandte Arbeiten	5
1.3	Zielsetzung	13
1.4	Rekonstruktionspipeline	18
2	Beschränkungen des Scansystems	23
2.1	Geometrische Unzulänglichkeiten	24
2.2	Beschränkungen der Optik	30
2.3	Beschränkungen des Emitters	42
2.4	Beschränkungen des Sensors	49
3	Erzeugung der projizierten Muster	61
3.1	Kohärenzannahmen	62
3.2	Musterentwurf	66
3.3	Algorithmus zur Mustererzeugung	76
4	Streifenzuordnung	83
4.1	Vorverarbeitung der Aufnahmen	83
4.2	Kantendetektion	93
4.3	Zuordnungsverfahren	100
5	Flächenrekonstruktion	117
5.1	Optische Triangulierung	117
5.2	Dreiecksnetzerzeugung	123
5.3	Glättung	131

6	Ergebnisse	141
6.1	Implementierung	141
6.2	Experimente	144
7	Schlusswort	155
7.1	Zusammenfassung	155
7.2	Ausblick	156
	Literaturverzeichnis	168
	Lebenslauf	169

Kapitel 1

Einleitung

Während sich die traditionelle Computergraphik hauptsächlich mit den Aufgaben der Modellierung von 3D-Objekten, deren Animierung und Visualisierung beschäftigt, hat sich in den letzten Jahren gezeigt, dass das Interesse zur automatischen Erfassung und Rekonstruktion von Objekten aus der realen Welt stark zugenommen hat. Vor allem der starke Wettbewerb in der Unterhaltungsindustrie hat dazu geführt, dass viele ausgereifte Verfahren direkt in der Graphikhardware umgesetzt werden. Das erklärt, weshalb die Leistungen heutiger Endverbraucher-Graphikhardware sich schneller entwickeln als das *Moore'sche Gesetz*¹[Moore '65], das derzeit immer noch seine Gültigkeit in der Halbleiterentwicklung hat. Demzufolge ist auch die Nachfrage nach der Visualisierung von noch komplexeren und detaillierteren Szenen gestiegen. Obwohl eine Vielzahl von CAD-Methoden entwickelt worden sind, um das Modellieren realistischer Objekte zu erleichtern, ist die Erstellung komplexer Modelle immer noch eine mühsame und aufwändige Arbeit. Beispielsweise werden heutzutage immer noch viele talentierte Künstler benötigt und meist Monate gebraucht, um kurze Animationssequenzen in digitalen Zeichentrickfilmen fertigzustellen, woraus hohe Produktionskosten entstehen. Die Tatsache, dass die verwendeten virtuellen Objekte vorwiegend von der realen Welt inspiriert oder direkt übernommen werden, motiviert die Aufgabe, diese Lücke in der traditionellen Computergraphik durch einen Erfassungs- und Rekonstruktionsschritt zu überbrücken. Die Geometrie und Oberflächeneigenschaften eines Objekts werden üblicherweise mittels eines 3D-Scanners erfasst, während Bewegungen durch Motion-Capture-Verfahren gewonnen werden.

Gegenwärtige Systeme bestehen meistens aus speziellen und teuren Hardwareanfertigungen, die viele Anwendungsbereiche daran hindern, von der Scan-Technologie zu profitieren. Fortschritte im Bereich des Maschinensehens (engl. *computer vision*) haben in den letzten Jahren gezeigt, dass handelsübliche Digitalkameras eine

¹Moore's Gesetz besagt, dass die Transistorendichte in integrierten Schaltungen sich alle 18 Monate verdoppelt.

robuste Erfassung zur Digitalisierung von hochwertigen 3D-Objekten ermöglichen. Viele Verfahren liefern bereits hervorragende Messungen und Rekonstruktionen bezüglich Robustheit und Genauigkeit. Jedoch sind diese Verfahren meistens noch sehr experimentell und benötigen erfahrene Benutzer zur Eingabe von Parametern, die überdies Kenntnis des Verfahrens voraussetzen. Außerdem werden sehr oft Bedingungen bei den Erfassungen vorausgesetzt, die nicht immer in der Praxis möglich sind, etwa die Abdunklung der Szene, um mögliche Auswirkungen von Lichtquellen, die unerwünscht sind, zu vermeiden.

In dieser Arbeit wird eine vollständige Pipeline zur Erfassung und Rekonstruktion der Oberfläche von komplexen farbigen Objekten mittels optischer Triangulierung vorgestellt. Das Scansystem besteht nur aus einer Fuji Finepix S2 Pro Digitalkamera, einem ASK M2 DLP-Videoprojektor und einem Rechner (siehe Abb. 1.1). Es wird darauf geachtet, dass ein hoher Grad an Automatisierung vorhanden ist, um auch nicht erfahrenen Benutzern das Scannen zu ermöglichen. Unter Berücksichtigung des zugrunde liegenden Systems werden neue Verfahren eingeführt, die im Zusammenhang mit der Objektrekonstruktion derzeit nicht bekannt sind.

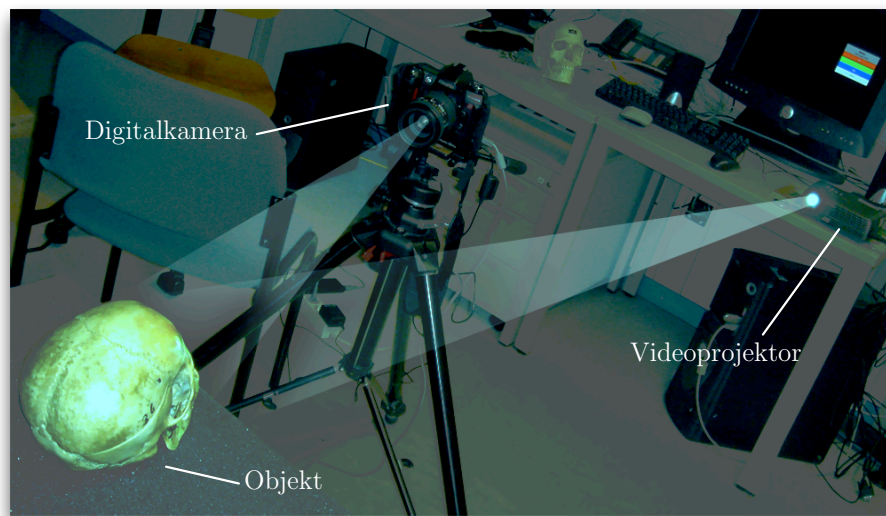


Abbildung 1.1: Unser 3D-Scanner besteht nur aus einer Digitalkamera und einem Videoprojektor. Die von der Kamera und dem Projektor aus sichtbaren Bereiche werden als Fläche einer Ansicht rekonstruiert. Um das komplette Objekt einzuscannen, kann entweder das Objekt oder das Scansystem bewegt werden, damit die restlichen Ansichten erfasst werden. Letztendlich werden alle Ansichten zu einem zusammenhängenden Objekt verschmolzen.

Bereits etablierte Verfahren reduzieren die Erfassungszeit, indem sie Videoprojektoren ausnutzen, um ein bestimmtes Muster, das *strukturierte Licht*, auf das Objekt zu projizieren und gleichzeitig mehrere Punkte auf der Oberfläche abzutasten. Wir zeigen, dass wir in der Lage sind, die Aufnahmegeschwindigkeit durch ein im *Zeitmultiplex kodierte Farbstreifenmuster* drastisch zu beschleunigen, ohne bemerkbare Verluste an Robustheit und Genauigkeit zu erhalten. Außerdem hat das neuartige Projektionsmuster die Eigenschaft, dass es unempfindlich ist gegenüber Uneindeutigkeiten aufgrund von hochfrequenten Objektoberflächen. Da das Aussehen der projizierten Farbstreifen bekannt ist, liegt die Hauptschwierigkeit darin, die von der Objektoberfläche reflektierten Streifen wiederzuerkennen. Es wird in dieser Arbeit ein effizientes, durch Ballung unterstütztes Verfahren vorgestellt, das die gewünschte Zuordnung mittels lokaler Korrelation ermittelt. Im Gegensatz zu den meisten Arbeiten zur 3D-Rekonstruktion, die sich nur mit einem Teilproblem der gesamten Rekonstruktionspipeline beschäftigen, entwerfen wir ein komplettes Scansystem. Folglich sind wir in der Lage, wichtige A-priori-Informationen auszunutzen, die in vielen Arbeiten nach der Ausgabe verloren gehen. Gleichzeitig können wichtige Annahmen vorausgesetzt werden, die für eine zuverlässige Rekonstruktion ausschlaggebend sind. Beispielsweise können wir bei der Oberflächenrekonstruktion aus den strukturierten Abtastungen die topologische Struktur wesentlich korrekter abschätzen, als aus einer nicht zusammenhängenden Punktwolke, deren Konnektivität nur aus der Lage und Dichte der Punkte gewonnen werden kann.

In diesem Kapitel beschreiben wir zunächst zur Motivation einige wichtige Anwendungen, die heutzutage von dem Einsatz von 3D-Scansystemen erheblich profitieren können (Abschnitt 1.1). Darum geben wir im Abschnitt 1.2 einen Überblick über verschiedene Methoden, die wichtige Beiträge im Bereich 3D-Erfassung erbracht haben und zeigen, wodurch unsere Arbeit sich von diesen unterscheidet. Insbesondere werden Vor- und Nachteile kurz dargestellt. Anschließend werden die Zielsetzungen, die zum Entwurf des Systems erforderlich sind, erläutert (Abschnitt 1.3). Die Umsetzung des Scanverfahrens wird im Abschnitt 1.4 in Form einer Pipeline veranschaulicht, wodurch die Gliederung dieser Arbeit skizziert wird.

1.1 Anwendungen

In vielen Bereichen haben sich 3D-Erfassungssysteme als wichtiger Bestandteil durchgesetzt. Früher waren die Hauptanwendungen lediglich die Modellinspektion und Reverse Engineering in der Fertigungsindustrie [Várady et al. '97]. Ebenfalls wurden diese zur Unterstützung der selbständigen Navigation in der Robotik vielseitig eingesetzt. Heutzutage beobachtet man eine stark anwachsende Nachfrage nach hoch detaillierten 3D-Modellen in der Computergraphik, wodurch sich hier die Rekonstruktion der vollständigen Geometrie zur Visualisierung, gegenüber ein-

fachen Tiefenmessungen, hervorhebt. Ein Auswahl moderner Anwendungen wird im Folgenden vorgestellt:

- **Robotik:** Zur autonomen Roboterführung in unerforschten Gebieten werden Umgebungen mittels Sensoren (meist mit niedriger Auflösung) erfasst, um Hindernisse zu überwinden. Zusätzlich werden oft Stereokameras montiert, um gleichzeitig die Geometrie des Geländes zu rekonstruieren, wie zuletzt etwa bei den halbautonomen Erkundungsfahrzeugen der unbemannten NASA-Marsmissionen.
- **Unterhaltungsindustrie:** In der Filmindustrie sind Spezialeffekte mittels computergenerierter Bilder bereits zum Standard geworden. Dennoch sind heutige Bilderzeugungsverfahren und Animationstechniken noch nicht in der Lage, komplexere Objekte, wie z.B. Schauspieler, in hoher Auflösung zu ersetzen. In vorhersehbarer Zukunft werden vollständige und realistisch synthetisch erzeugte Filme möglich sein. 3D-Erfassungssysteme und Modelldatenbanken würden dann eine führende Rolle spielen. Zusätzlich bildet die Videospielebranche einen wichtigen Bestandteil der Unterhaltungsindustrie. Aufgrund ihrer wirtschaftlichen Lukrativität und der hohen Konkurrenz hat sie, wie kein anderes Anwendungsgebiet, einen An Schub für den Fortschritt in der Computergraphik ausgelöst. Auch hier werden Modelle aus der realen Welt oft gebraucht.
- **Elektronischer Handel:** Im Zeitalter des Internets bieten bereits viele Online-Geschäfte interaktive Darstellungen ihrer Produkte. Obwohl viele Standards zur 3D-Visualisierung entwickelt worden sind, wie etwa die *Virtual Reality Modeling Language* (VRML), sind vor allem Modelle beliebt, die mittels *bildbasierter Modellier- und Bilderzeugungsverfahren* (engl. *image-based modelling and rendering*) [Levoy & Hanrahan '96, Gortler et al. '96, Buehler et al. '01, Szeliski & Shum '97] konstruiert wurden (z. B. Quicktime VR [Chen '95]). Im Gegensatz zu echten 3D-Modellen, haben sie den Vorteil, einfacher, schneller und kostengünstiger erzeugt zu werden. Die Komplexität der Rekonstruktion ist in diesem Fall unabhängig von der Komplexität der zu scannenden Geometrie. Der Nachteil ist natürlich, dass die Anzahl der Ansichten stark begrenzt ist, der Speicherverbrauch sehr hoch ist und die Daten viel Redundanz enthalten. Flexible und kostengünstige Lösungen zur 3D-Rekonstruktion sind notwendig, damit die Darstellung durch echte 3D-Modelle zum Zuge kommen kann.
- **Medizin:** Zur Unterstützung von Diagnosen und Planung von chirurgischen Eingriffen werden oft CT- und MRI-Scanner verwendet, um Bilder vom Inneren des menschlichen Körpers zu erzeugen. Die Ausgaben solcher Erfassungstechniken sind räumlich diskrete Messwerte, die bei einer Rekonstruktion als volumetrische Daten behandelt werden. Als ebenso wichtig stellt sich

der Einsatz von Oberflächenmesstechniken dar. Prothesen können individuell an den Patienten angepasst werden, wenn die genauen Abmessungen der entsprechenden Körperteile vorliegen. Weiterhin werden in der rechner- und sensorgestützten Chirurgie hochgenaue 3D-Oberflächendaten des Patienten verwendet, um Operationen in kritischen Bereichen, wie etwa im Kopf- oder Herzbereich, sicherer zu machen.

- **Sicherheit:** Als Beitrag zur Sicherheit wurden bereits biometrische Identifikationstechnologien in vielen Systemen eingeführt, wie z. B. Authentifizierungen mittels Fingerabdruck. Um potentielle Fälschungen zu erschweren können sichtgestützte Systeme durch 3D-Objektextraktion einen Datenbankabgleich durchführen, um die Authentizität sicherzustellen. Weiterhin können, zur Überwachung von öffentlichen Einrichtungen und des Verkehrs, Rekonstruktionen räumlicher Szenen erstellt werden. Vergleichende Untersuchungen haben bereits ergeben, dass dreidimensionale Messungen Vorteile gegenüber zweidimensionalen Bildanalysen haben.
- **Erhaltung von Kulturgütern:** Für Kunsthistoriker erschließen eingescannte Modelle von Skulpturen neue Wege, Techniken berühmter Bildhauer aus vergangener Zeit zu studieren. Außerdem kann der Originalzustand unvollständiger und zerstörter Relikte rekonstruiert und in Form von digitalen Museen visualisiert werden. Insbesondere für die Computergraphik bieten solche Projekte die Möglichkeit, mit interessanten Modellen zu arbeiten, deren Topologie meistens sehr einfach ist und deren Oberflächen für Scanner sehr gut zugänglich sind. Deswegen haben solche Projekte häufig zum Fortschritt der Scantechnologie beigetragen, wie bereits im Rahmen des „Digital Michelangelo“-Projekts von der Stanford University und der University of Washington [Levoy et al. '00], des „Pietà“-Projekts von IBM [Rushmeier et al. '98, Bernardini et al. '02] und des „Great Buddha“-Projekts von der University of Tokyo [Miyazaki et al. '01] demonstriert worden ist. Weiterhin werden aufgrund des wachsenden Interesses an der Erfassung von Beleuchtungs- und Oberflächenmaterialeigenschaften (*Photometrie*) derzeit Forschungen im Rahmen des „ViHAP3D“-Projekts, geleitet vom Max-Planck-Institut für Informatik in Saarbrücken, angestellt.

1.2 Verwandte Arbeiten

Da 3D-Scanverfahren in den letzten 15 Jahren enorm an Bedeutung gewonnen haben, sind ebenfalls eine Reihe von wichtigen Arbeiten zu diesem Thema entstanden. Ein ausführlicher Überblick verschiedener Methoden, sowie der Einsatz von unkonventioneller Hardware wird in [Curless et al. '00] gegeben. Wie in

[Várady et al. '97] und [Curless '97] beschrieben, werden diese Verfahren üblicherweise in *taktile* und *nicht-taktile* Verfahren eingeteilt (siehe Abb. 1.2).

Taktile Verfahren

Obwohl sie hochgenaue Messungen liefern und auch Objekte mit transparenter Oberfläche abtasten können, sind taktile Verfahren, wie etwa Koordinatenmessmaschinen oder Gelenksysteme, aufgrund von aufwändigen manuellen Eingaben und der langsamen Geschwindigkeit ungeeignet zur Erfassung komplexer Objekte. Neben den hohen Anschaffungskosten der Hardware besteht außerdem die Gefahr, zerbrechliche Objekte durch Berührungen mit dem Erfassungssystem zu beschädigen.

Nicht-taktile Verfahren

Nicht-taktile Verfahren bieten dagegen die Möglichkeit, aus sicherer Entfernung schnellere Erfassungen, sowie einen höheren Grad an Automatisierung zu erzielen. Aufgrund der speziellen Hardwareanforderungen und der Inflexibilität bei magnetischen und akustischen Scantechnologien, beschränken wir uns ausschließlich auf optische Verfahren. Hierbei unterscheidet man hauptsächlich zwischen *passiven* und *aktiven* Verfahren.

Passive Verfahren

Passive Scanverfahren benutzen ausschließlich *Sensoren* und bieten somit die Möglichkeit, unbemerkt eine Szene zu erfassen. Dennoch ist eine zuverlässige Rekonstruktion stark abhängig der Struktur der Szene, wie in [Pollefeys '99] ausführlich beschrieben wird. Beispielsweise müssen bei Stereoverfahren Koordinaten desselben Merkmals auf zwei oder mehreren Aufnahmen gefunden werden, um die Tiefeninformation zu ermitteln [Dhond & Aggarwal '89, Faugeras '93, Ma et al. '04]. Verdeckungen aufgrund unterschiedlicher Kameraansichten, merkmalsarme Oberflächen, sowie mehrdeutige Merkmale und andere Fehlerquellen erschweren dieses Korrespondenzproblem. Ein experimenteller Vergleich verschiedener Lösungsansätze für Stereokorrespondenzen wird in [Szeliski & Zabih '00] gegeben. Nicht von diesen Unzulänglichkeiten betroffen sind die silhouettenbasierten Rekonstruktionsverfahren aus [Laurentini '94, Fautz '03]. Diese volumetrischen Ansätze haben den Nachteil, sehr viele Aufnahmen zu benötigen und es kann maximal nur die *visuelle Hülle* rekonstruiert werden, d. h. Tiefeninformationen werden bei konkaven Bereichen falsch ermittelt. In [Vedula et al. '98, Li et al. '02] werden Hybridverfahren untersucht, um die Nachteile beider Ansätze zu mindern.

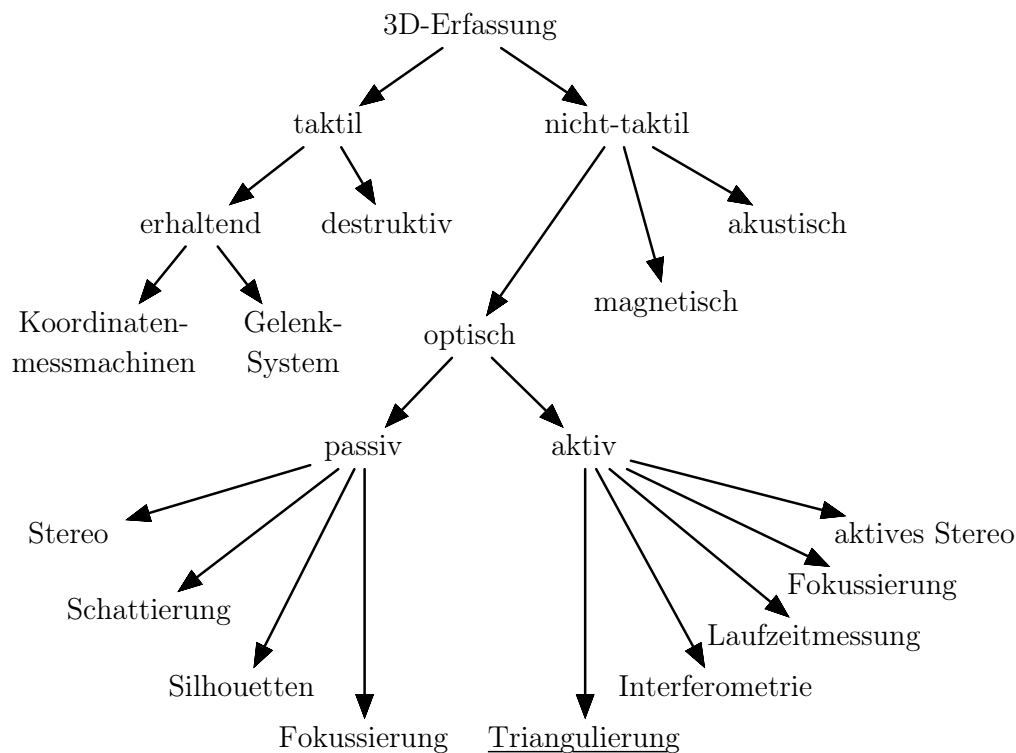


Abbildung 1.2: Taxonomie von 3D-Erfassungsverfahren (angelehnt an [Várady et al. '97, Curless '97]). In dieser Arbeit wird ein auf optischer Triangulierung basierendes Scanverfahren vorgestellt.

Aktive Verfahren

Aktive Verfahren hingegen erfassen das Objekt, indem kontrollierte elektromagnetische Wellen (z. B. strukturiertes Licht) aus einem *Emitter* projiziert werden und die Auswirkungen des Objekts auf diese Wellen (z. B. Oberflächenreflexion) mittels *Sensoren* erfasst werden (z. B. durch Fotografie). Dabei wird das Korrespondenzproblem stark vereinfacht, da ein von vornherein bekanntes Signal leichter wieder zu erkennen ist. Demzufolge haben die meisten aktiv-optischen Verfahren die Eigenschaft, hochgenaue und sehr dichte Abtastungen zu erreichen. Angesichts dieser Tatsache sind sie besonders für robuste Erfassungen und Rekonstruktionen komplexer Modelle geeignet. Eine Reihe von unterschiedlichen Methoden zur aktiven Erfassung werden in [Besl '88] behandelt.

Laserbasierte optische Triangulierungsverfahren

Zu den gängigsten Methoden zur Erfassung der Oberflächengeometrie gehören die optischen Triangulierungsverfahren mit kohärentem Licht, d. h. mit Laser-

Emittern, wie etwa die professionellen Scansysteme von Konica-Minolta oder Cyberware [Konica-Minolta '03, Cyberware '99], die heutzutage immer noch eine recht kostspielige Angelegenheit sind (ca. 40 000 €). Obwohl Projektionen mit kohärentem Licht eine große Reichweite haben und äußerst unempfindlich gegenüber farbigen Objekten und Umgebungsbeleuchtungen sind, neigen sie dazu, auf rauen Oberflächen und bei Oberflächenvariationen ungleichmäßig zu zerstreuen (laser speckle), was zu einer ungenauen Messung führt. Um diese Ungenauigkeiten zu überwinden, entwickelten Curless und Levoy in [Curless & Levoy '95, Curless '97] ein Verfahren, die so genannte *Raum-Zeit-Analyse* (engl. *space time analysis*), auf Kosten von zusätzlichen Eingabebildern, die über die Zeit aufgenommen werden. Zur Beschleunigung des Scanvorgangs wird oft ein Laserstreifen anstelle von einem Punkt projiziert, um gleichzeitig mehrere Punkte abzutasten. Damit eine vollständige Abtastung aus einer Ansicht gewonnen werden kann, wird z. B. ein vertikaler Streifen in horizontaler Richtung bewegt, dessen Abtastdichte durch die Geschwindigkeit der Bewegung und durch die Bildfrequenz der Aufnahme bestimmt wird [Kanade et al. '91]. Daher sind für Laser-Scansysteme nur Videokameras sinnvoll, die heutzutage aufgrund ihrer niedrigeren Auflösung (bei einigen sogar mit Zwischenzeilenabtastung) und aufgrund von Kompressionsartefakten noch nicht an die Qualität von digitalen Fotokameras herankommen.

Optische Triangulierungsverfahren mit strukturiertem Licht

Mit dem Ziel, Oberflächenerfassungen zu beschleunigen, wurde eine Vielzahl von Verfahren entwickelt, die ein bestimmtes strukturiertes Muster auf das Objekt projizieren. Folglich besteht die Möglichkeit, das Objekt gleichzeitig mit mehreren Streifen abzutasten. Aus Kosten- und Flexibilitätsgründen werden üblicherweise Videoprojektoren (ursprünglich Diaprojektoren) eingesetzt. Die optische Triangulierung mittels Kamera und Projektor wird in Abb. 1.3 zur Veranschaulichung zweidimensional und dreidimensional dargestellt. Im 2D-Fall erzeugt ein Punkt auf der Projektorbildebene mit dem zugehörigen Augpunkt eine Projektionsgerade. Aus der Sicht der Kamera wird der vom Projektionsstrahl getroffene Punkt auf die Kamerabildebene abgebildet und spannt eine Sichtgerade mit dessen Augpunkt auf. Die Position des abgetasteten Punkts ist gerade der Schnittpunkt beider Geraden. Analog ist, im 3D bei einer Streifenprojektion, ein abgetasteter Punkt der Schnittpunkt einer projizierten Ebene und einer Sichtgeraden. Bei der gleichzeitigen Projektion mehrerer Merkmale, besteht die Herausforderung darin, die Muster so zu entwerfen, dass die Merkmale in den Aufnahmen korrekt wiederzuerkennen sind. Nur auf diese Weise kann eine Triangulierung mit den korrekten Merkmalen erfolgen, die oft aufgrund von Unzulänglichkeiten, auf die in Kap. 2 näher eingegangen wird, ihre Eindeutigkeit in der Aufnahme verlieren. Dieses Problem wird auch als *Zuordnungs-* oder *Korrespondenzproblem* bezeichnet. Eine Klassifizierung, sowie ein qualitativer Vergleich verschiedener Strategien zum Entwurf der projizierten Muster wurde neulich in [Salvi et al. '04] vorgestellt.

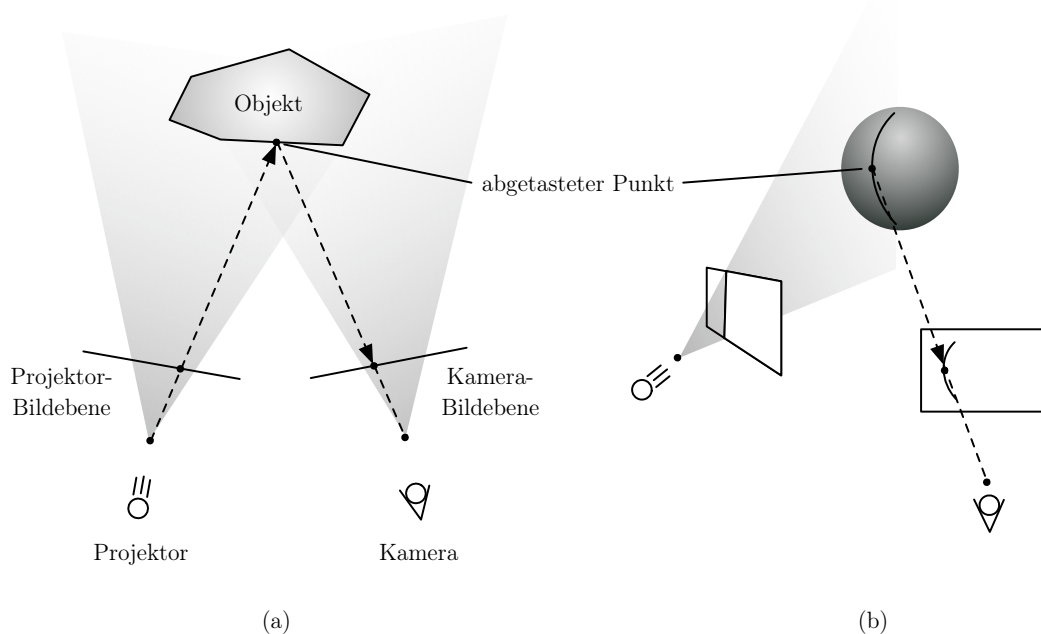


Abbildung 1.3: Optische Triangulierung mittels Kamera und Projektor. (a) Zweidimensionaler Fall: Die Position des abgetasteten Punktes ist gerade der Schnittpunkt der Sicht- bzw. Projektionsgeraden der Kamera und des Projektors, die durch deren Augpunkte und den zugehörigen Bildern des vom Projektionsstrahl getroffenen Punktes auf der Bildebene erzeugt werden. (b) Analog ist im Dreidimensionalen bei einer Streifenprojektion ein abgetasteter Punkt der Schnittpunkt einer projizierten Ebene und einer Sichtgeraden.

Erfassung unter Annahme der Zeitkohärenz

Indem weiterhin eine statische Szene angenommen wird (*zeitliche Kohärenzannahme*), reduzieren Posdamer und Altschuler die Anzahl der Aufnahmen mittels einer sequentiellen Projektion von binärkodierten Schwarz-Weiß-Streifen [Posdamer & Altschuler '82]. Mit dieser Methode, können über die Zeit n Streifen aus $\log n$ Mustern eindeutig kodiert werden. Aufgrund von potentiellen lokalen Oberflächenunstetigkeiten können zu feine Streifen von der Kamera aus nicht sichtbar sein. Mittels einer *Gray-Kodierung* der Streifensequenzen² generierten Sato und Inokuchi in [Sato & Inokuchi '85] breitere Streifen in den projizierten Mustern. Durch eine rekursive Unterteilung werden die Streifenübergänge schrittweise lokalisiert, d. h. dass in der k -ten Aufnahme alle Streifenübergänge aus den Aufnahmen $0, \dots, k - 1$ mitberücksichtigt werden. Mit dem Argument, dass bei Schwarz-Weiß-Beleuchtungen, hellere Streifen breitere Streifen erzeugen und da-

²Die Gray-Kodierung ist eine Kodierung mit der Eigenschaft, dass der Hammingabstand von aufeinander folgenden Zahlen immer eins ist.

durch die Genauigkeit verloren geht, verwenden Rocchini et al. [Rocchini et al. '01] rote und blaue Streifen mit einem ein Pixel breiten grünen Übergang. Demzufolge haben die Streifen an jeder Stelle dieselbe Lichtintensität. Um wiederum mit noch weniger Aufnahmen auszukommen, verallgemeinern Caspi et al. die Gray-Codes auf Farben [Caspi et al. '98]. Damit die Streifen auch auf farbigen Objekten erkennbar bleiben, wird zusätzlich eine Farbkalibrierung des Systems durchgeführt. Eine andere zeitkohärente Kodierung mit Schwarz-Weiß-Streifen wird in [Hall-Holt & Rusinkiewicz '01, Rusinkiewicz '01, Rusinkiewicz et al. '02] vorgestellt. Hier werden alle Streifenübergänge durch ihre zugehörigen benachbarten Streifenfarben aus 4 sequentiellen Aufnahmen identifiziert. Man spricht in diesem Fall von einer *Annahme einer lokalen Zeitkohärenz*. Dadurch können ebenfalls Streifensequenzen erzeugt werden, die lang genug sind, um größere Flächen zu erfassen. Im Gegensatz zu den vorherigen Verfahren, wurden diese Muster für die interaktive 3D-Rekonstruktion entwickelt, d. h. das Objekt wird vor einer Videokamera langsam bewegt und dessen Rekonstruktion in Echtzeit vervollständigt. Da die meisten Videokameras eine begrenzte Bildwiederholungsrate haben und das Objekt bewegt wird, müssen hier zusätzliche Restriktionen eingefügt werden, damit die Streifenübergänge so oft wie möglich an allen Stellen sichtbar bleiben. Eine weitere Studie einer echtzeitfähigen 3D-Erfassung wird, in Zusammenhang mit unstrukturierten Beleuchtungen und einem aktiven Stereosystem, in [Zhang et al. '03] gegeben. Solche Systeme werden, aufgrund ihrer Ungenauigkeiten, nicht für die Rekonstruktion hochqualitativer 3D-Modelle, sondern für die Ansteuerung und Animation von vormodellierten Modellen verwendet, wie neulich in [Zhang et al. '04] demonstriert worden ist.

Erfassung unter Annahme einer idealen Reflexion

Mit nur zwei Eingabebildern projizieren Carrihill und Hummel einen Farbverlauf von schwarz nach weiß und ein weißes Bild [Carrihill & Hummel '85]. Dadurch kann in jedem Pixel des Kamerabildes das Verhältnis der Lichtintensität beider Bilder berechnet werden. Dieses Verhältnis bestimmt den zugehörigen projizierten Pixel. In diesem Fall muss angenommen werden, dass das Objekt die Grautöne ideal reflektiert (*ideale Reflexionsannahme*). Aufgrund der hohen Rauschanfälligkeiten in den Messungen wurde dieses Verfahren in [Chazan & Kiryati '95] in Kombination mit Streifenbildern untersucht.

Erfassung unter Annahme der Raumkohärenz

Einen weiteren Ansatz für minimale Bildeingaben stellen die auf *räumlicher Kohärenz* basierenden Verfahren dar. In [Proesmans et al. '96] werden zweidimensionale Gitter projiziert und durch Abzählen der Gitterlinien korrespondiert. Obwohl bei diesem Verfahren nur ein einziges Eingabebild ausreicht, muss hier sichergestellt werden, dass alle Gitterlinien von der Kamera aus sichtbar sind, da es sonst zu ei-

ner fehlerhaften Triangulierung führt. Die Annahme, dass der sichtbare Bereich ein einziges zusammenhängendes Flächenstück darstellt, wird auch als *globale räumliche Kohärenzannahme* bezeichnet. Um auch Objektoberflächen mit Selbstverdeckungen erfassen zu können (siehe Kap. 2), wurden eine Reihe von Verfahren vorgeschlagen, die die projizierten Merkmale mittels benachbarter Merkmale korrespondieren (*lokale Raumkohärenz*). Sie werden oft in Zusammenhang mit Farbkodierungen und/oder wenigen zusätzlichen Eingabebildern realisiert. Um farbkodierte Muster leichter zu identifizieren, ist es wichtig, bei farbigen Objekten die Anzahl der projizierten Farben zu begrenzen. Boyer und Kak verwendeten in ihrem so genannten *Crystal-Fitting-and-Growing-Algorithmus* beispielsweise nur die Primärfarben Rot, Grün und Blau [Boyer & Kak '87]. In diesem Verfahren werden iterativ jeweils die längsten Untersequenzen von korrespondierten Streifenübergängen erkannt. In [Zhang et al. '02] stellen Zhang et al. fest, dass dieses Verfahren relativ empfindlich gegenüber Falschzuordnungen ist und zu vielen Ausreißern und Löchern bei der Rekonstruktion führt. Einen ähnlichen Ansatz verfolgten Davies und Nixon in [Davies & Nixon '98], indem sie auf Kosten eines komplexeren Segmentierungsprozesses, hexagonal angeordnete Punkte projizieren. Hier werden insbesondere zweidimensionale Nachbarschaften berücksichtigt. Aufgrund der nur lokalen Betrachtung der Nachbarschaften, sind diese Zuordnungsalgorithmen besonders empfindlich gegenüber lokal unvollständig erkannten Streifen (aufgrund von Selbstverdeckungen und Unstetigkeiten) und schlecht reflektierenden Oberflächen, die zu einer schwierigen Erkennung führen.

Zu den Ansätzen, die die gesamte Zuordnung (z. B. alle Streifenzuordnungen) als Ganzen betrachten gehören die *globalen Korrespondenzalgorithmen*, wie etwa die Verfahren mittels dynamischem Programmieren, die sich bereits im Bereich der passiven Stereoerfassung etabliert haben [Ohta & Kanade '85]. Im Zusammenhang mit strukturiertem Licht stellten Chen et al. in [Chen et al. '97] ein aktives Stereoerfassungssystem vor, das mit Hilfe von zwei Kameras farbig projizierte Streifen mittels dynamischem Programmieren korrespondiert. Auf Kosten einer zusätzlichen Kamera liegt der Vorteil darin, dass bei der Projektion von Farbstreifen deutlich identifizierbare Merkmale auf dem Objekt reflektiert werden und dadurch sich das Stereokorrespondenzproblem stark vereinfacht. In diesem Fall ist eine Kamera-Kamera-Zuordnung tatsächlich zuverlässiger als eine Projektor-Kamera-Zuordnung, da schlecht reflektierende Oberflächen sich gleichermaßen auf die Aufnahmen beider Kameras auswirken. Letztendlich wurde in [Zhang et al. '02] das Zuordnungsverfahren mittels dynamischem Programmieren auf pure Projektor-Kamerasysteme übertragen. Ein Problem, das sich beim dynamischen Programmieren feststellen lässt, ist dass angenommen wird, dass in jedem Zeilendurchgang die Reihenfolge der projizierten Streifen mit den erfassten Streifen übereinstimmt (*Monotonizitätsannahme*). Mittels eines Mehrpassverfahrens umgehen Zhang et al. diese Einschränkung. Zusätzlich wurden in diesem Verfahren Farbstreifen pro-

jiziert, die aus einer so genannten *De-Bruijn-Sequenz*³ erzeugt werden. Die Streifen dieser Muster können aus sehr langen Streifensequenzen identifiziert werden, obwohl sie mehrmals vorkommen. Indem durch die De-Bruijn-Sequenz garantiert wird, dass sehr kleine Untersequenzen nur höchstens einmal vorkommen, kann die Zuordnung mittels dynamischem Programmierens aus einer lokalen Nachbarschaft erfolgen. Hier wurde insbesondere ein Verfahren entwickelt, das nur ein einziges Eingabebild zur Rekonstruktion benötigt und daher für die Erfassung von beweglichen Objekten geeignet ist. Unzufriedenstellend bei diesem Verfahren sind vor allem Bereiche an den Rändern des Objekts, wie in [Li et al. '04, Li '04b] experimentell herausgefunden wurde. Wenn längere Streifensequenzen nicht erfasst werden, kommt es oft vor, dass es zu starken Ausreißern kommt, da aufgrund von Mehrdeutigkeiten, mehrere optimale Lösungen gefunden werden können. Li et al. reduzierten diesen Effekt, indem sie hinter dem Objekt eine Ebene stellten um auch die Streifen zu erfassen, die nicht auf das Objekt treffen. Weiterhin erweiterten sie das Verfahren mit einer vernetzungsbasierten Nachverarbeitung und einer robusteren subpixelgenauen Kantendetektion, um weitere Ausreißer zu eliminieren. Der große Nachteil bei den globalen Zuordnungsverfahren liegt allerdings darin, dass bei einer Überdetektion von Kanten (oft bei hochauflösenden Aufnahmen), die nicht zu den projizierten Streifen gehören, diese weiterhin zugeordnet werden, da sie die globale Bewertungsfunktion weiter optimieren. Auch bei Kanten mit schwachen Kontrasten ist dieses Phänomen zu beobachten. Dieses führt zu hohen Rauschanteilen in der triangulierten Punktwolke.

Erfassung unter Annahme von Raum- und Zeitkohärenz

Wir präsentieren in dieser Arbeit ein innovatives Hybridverfahren mittels strukturiertem Licht, das sowohl die Zeitkohärenz als auch die lokale Raumkohärenz annimmt. Obwohl wir annehmen, dass wir eine statische Szene haben, benötigen wir nicht mehr als drei Aufnahmen des Objekts, darunter zwei mit Farbstreifen beleuchtet und eine mit einem weißen Bild beleuchtet für die Farbkorrektur und Texturerfassung. Um möglichst viele eindeutige Streifen projizieren zu können, identifizieren wir diese über die benachbarten Farben an jedem Streifenübergang, ähnlich wie bei [Hall-Holt & Rusinkiewicz '01], nur dass hier zur Erzeugung von längeren Streifensequenzen farbige Streifen eingesetzt werden. Zusätzlich haben unsere projizierten Streifensequenzen die Eigenschaft, dass alle Streifenfarben innerhalb einer lokalen Umgebung eindeutig sind und vor allem, dass keine Paare von Streifenfarben innerhalb dieser Umgebung gleichzeitig in einer anderen nicht überlappenden Umgebung derselben Größe liegen. Dadurch können bei hochfrequenten Selbstverdeckungen und nicht zusammenhängenden Objektflächen Mehrdeutigkeiten vermieden werden, um eine robustere Rekonstruktion zu erzielen. Im Ge-

³In der Kombinatorik ist eine De-Bruijn-Sequenz k -ter Ordnung eine zirkuläre Sequenz der Länge k^n mit der Eigenschaft, dass jede Untersequenz der Länge n eindeutig ist.

gensatz zu den globalen Korrespondenzalgorithmen, wird hier zusätzlich die lokale Kohärenz als Unterstützung verwendet, damit Falschzuordnungen, im Sinne einer Nachverarbeitung, detektiert und entfernt werden können. Aufgrund der Tatsache, dass Projektoren nur auf einer einzigen Ebene (Schärfenebene) scharf fokussieren können, beobachten wir bei hochauflösenden Aufnahmen, dass die Kantenübergänge oft sehr schwache Kontraste haben. Deswegen verwenden wir für die Zuordnung die benachbarten Streifenmitten anstelle eine lokalen Umgebung eines Streifenübergangs. Die Verfahren mit Streifenprojektionen, die uns derzeit bekannt sind, betrachten in der Kameraaufnahme alle Zeilen unabhängig von einander. Wir zeigen in dieser Arbeit, dass eine zuverlässigere Korrespondenz erzielt werden kann, indem zwischen den Zeilen stark korrelierende lokale Umgebungen der Streifenübergänge geballt werden. Anschließend erzeugen wir ein Dreiecksnetz aus der optisch triangulierten Punktwolke unter Ausnutzung des vorliegenden Scansystems und der Zuordnungsinformationen. Wir demonstrieren, dass unsere Methode bei ungleichmäßigen Abtastungen, wie sie bei hochauflösenden Erfassungen mittels strukturiertem Licht vorkommen, erhebliche Vorteile gegenüber anderen Ansätzen, die aus einer reinen Punktwolke die Fläche rekonstruieren, hat.

1.3 Zielsetzung

Ziel dieser Arbeit ist es, aus sehr wenigen Aufnahmen (in der Praxis sind es drei) einer strukturiert beleuchteten Ansicht die vollständige sichtbare Oberfläche aus dieser Perspektive zu rekonstruieren. Die Aufgabenstellung kann grob in einen Erfassungsschritt und in einen Flächenrekonstruktionsschritt aufgeteilt werden. In diesem Abschnitt geben wir kurz eine formale Spezifikation der Aufgabenstellung beider Schritte. Anschließend werden die Beiträge, die mit dieser Arbeit erreicht worden sind, dargestellt.

Formale Spezifikation der Zielsetzung

Es soll bei der Erfassung mittels eines gegebenen Projektor-Kamera-Systems, wie in Abb. 1.3, aus den Eingabebildern eine Punktwolke rekonstruiert werden. Insbesondere kalibrieren wir zunächst das System, damit eine optische Triangulierung erfolgen kann, d. h. wir müssen die internen (Brennweite, Pixelgrößen, ...) und externen Parameter (Lage und Orientierung) für Kamera und Projektor abschätzen (siehe Kap. 5). Die projizierten Muster werden von der Objektoberfläche reflektiert und verzerrt auf die Kamerabildebene abgebildet. Dadurch induziert das System, zusammen mit der unbekanntenen Objektoberfläche, eine Transferfunktion, die wir $f := f_{p,\Theta}$ nennen. Wir beschreiben die Oberfläche durch eine kompakte Teilmenge $\mathcal{S} \subset \mathbb{R}^3$ (streng genommen eine kompakte topologische 2-Mannigfaltigkeit). Die Ausgabe von f wird durch eine Menge von Projektor-Kamera-Parametern Θ und die Menge der projizierten Muster $\mathcal{P} = \{p_1, \dots, p_n\}$ bestimmt. Im Idealfall wird ei-

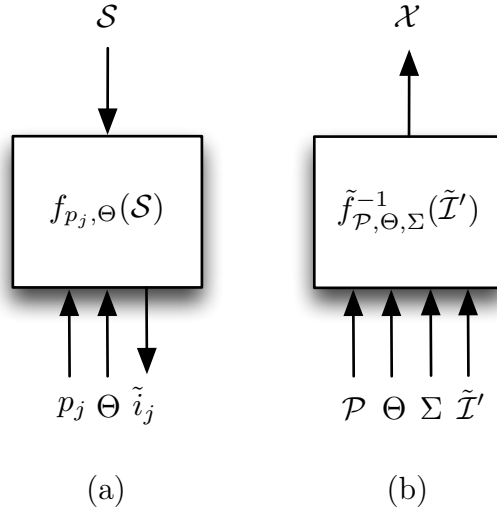
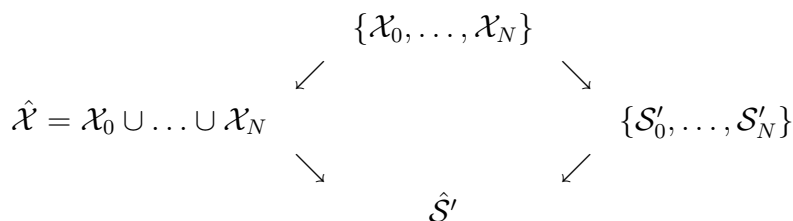


Abbildung 1.4: Das inverse Problem des Erfassungsschrittes. (a) f modelliert die Transferfunktion des Objekts im Triangulierungssystem. Eine Menge von projizierten Mustern $\mathcal{P} = \{p_1, \dots, p_n\}$ wird durch die unbekannte Objektoberfläche \mathcal{S} reflektiert und eine Menge von verzerrten Aufnahmen $\tilde{\mathcal{I}} = \{\tilde{i}_j\}$ wird ausgegeben. (b) Die Aufgabe besteht darin, ein Verfahren \tilde{f}^{-1} mit einer Menge von Eingabeparametern Σ zu finden, um eine Abtastung \mathcal{X} der Oberfläche zu rekonstruieren, deren Abweichung zu \mathcal{S} möglichst gering sein soll.

ne Menge von Aufnahmen $\mathcal{I} = \{i_1, \dots, i_n\}$ ausgegeben mit $i_j = f_{p_j, \Theta}(\mathcal{S})$. Aufgrund von Messfehlern, die im Scansystem und in der erfassten Szene ihren Ursprung haben, und dadurch, dass eine ganze Dimension durch die 3D-nach-2D-Projektion verloren geht, wird eine Menge von verrauschten Aufnahmen $\tilde{\mathcal{I}} = \{\tilde{i}_1, \dots, \tilde{i}_n\}$ gemacht. Demzufolge ist es i. Allg. nicht möglich, die genaue Umkehrfunktion f^{-1} zu bestimmen, ohne zusätzliche Beschränkungen festzulegen, wie es bei den meisten Computer-Vision-Problemen üblich ist [Ma et al. '04]. Die Aufgabe besteht nun darin, ein Verfahren \tilde{f}^{-1} mit einer Menge von Eingabeparametern $\Sigma = \{\sigma_1, \dots, \sigma_m\}$ zu entwerfen, das die ideale Umkehrfunktion so gut wie möglich abschätzt. Falls $\Sigma = \emptyset$, so handelt es sich um ein vollautomatisches Verfahren. Um dieses Problem zu erleichtern, reduzieren wir das Rauschen und führen eine Farbkorrektur durch, so dass wir eine Menge von vorverarbeiteten Bildern $\tilde{\mathcal{I}}' = \{\tilde{i}'_1, \dots, \tilde{i}'_n\}$ erhalten. Die Ausgabe des Verfahrens liegt als Punktwolke \mathcal{X} , mit $\mathcal{X} = \tilde{f}_{\mathcal{P}, \Theta, \Sigma}^{-1}(\tilde{\mathcal{I}}')$, vor. Zur Veranschaulichung zeigt Abb. 1.4 ein Modellsystem des *inversen Problems* zum Erfassungsschritt.

Aus der gegebenen Punktwolke \mathcal{X} soll nun eine Fläche \mathcal{S}' rekonstruiert und somit eine Visualisierung der erfassten Oberfläche ermöglicht werden. Es muss ein \mathcal{S}' gefunden werden, das \mathcal{S} so gut wie möglich approximiert. In dieser Arbeit wird die

Fläche \mathcal{S}' als Dreiecksnetz repräsentiert, welches, für eine glatte Flächenvisualisierung, mittels Gouraud- oder Phong-Schattierungsverfahren dargestellt wird. Diese Aufgabe ist eng verbunden mit der Problemstellung, aus einer angeordneten (*registrierten*) Menge von Erfassungen $\{\mathcal{X}_0, \dots, \mathcal{X}_N\}$ aus N unterschiedlichen Ansichten eine einzige zusammenhängende 2-Mannigfaltigkeit $\hat{\mathcal{S}}' \subset \mathbb{R}^3$ zu *integrieren*. Dazu betrachten wir die im folgenden Diagramm dargestellten zwei Möglichkeiten:



Die erste Möglichkeit eine Fläche zu rekonstruieren besteht darin, aus den Punktwolken \mathcal{X}_n durch Vereinigungsoperatoren eine einzige Punktwolke $\hat{\mathcal{X}}$ zusammenzusetzen, um daraus hinterher eine approximierende Fläche $\hat{\mathcal{S}}'$ zu erzeugen. Diese Methode wurde beispielsweise in [Pulli et al. '97, Rusinkiewicz '01] für die Punktwolkenintegration von 3D-Scandaten eingesetzt und in [Linsen '01] für die Modellierung von Oberflächen in Punktwolkendarstellungen in Zusammenhang mit CSG-(Constructive Solid Geometry)-Verfahren ausführlich untersucht.

Der andere Weg ist zunächst zu jeder erfassten Punktwolke \mathcal{X}_i die zugehörige approximierende Fläche \mathcal{S}'_i zu erzeugen, um die Menge $\{\mathcal{S}'_0, \dots, \mathcal{S}'_N\}$ zu erhalten, die anschließend in $\hat{\mathcal{S}}'$ integriert wird. Während bei der ersten Methode der Integrations Schritt trivial ist, kann hier eine korrekte Flächenrekonstruktion nur dann erfolgen wenn die integrierte Punktwolke $\hat{\mathcal{X}}$ eine genügend hohe und gleichmäßige Dichte hat, frei von Ausreißern ist und das Signal-zu-Rauschverhältnis niedrig genug ist.

Kriterien zum Entwurf des Rekonstruktionsverfahrens

Entscheidend beim Entwurf von \tilde{f}^{-1} sind die Kriterien, die erfüllt werden müssen, um die Überlegenheit dieses Verfahrens gegenüber anderen hervorzuheben. Da diese Kriterien stark vom gewünschten Anwendungsfall abhängen, legen wir Wert auf ein schnelles, flexibles und handhabbares Scansystem, welches mittels einfacher Hardware hochdetaillierte Dreiecksnetze erzeugt, die den Ansprüchen heutiger Computergraphikanwendungen genügen. Die Hauptbeiträge dieses Scanverfahrens zeichnen sich durch folgende Eigenschaften aus:

- **Hohe Auflösung:** Wir nutzen die volle Bildauflösung der Digitalkamera und des Projektors aus. Entweder um große Objektoberflächen abzudecken oder um so viele Details wie möglich auf nahen Objektoberflächen zu erfassen. Dadurch bleiben beispielsweise, bei einer starken Vergrößerung in

interaktiven Visualisierungen, selbst die feinsten Details komplexer Objekte, wie etwa kleine Selbstverdeckungen und Löcher erhalten. Bei traditionellen Modellierverfahren ist dies ein zeitaufwändiger Prozess. Vor allem durch die weiter ansteigenden Leistungen heutiger Graphikhardware, gewinnen geometrisch komplexere Objekte gegenüber Objekten, die mittels Texturierung ihre visuelle Qualität erhalten, immer mehr an Bedeutung. Vielversprechend sind bevorstehende Technologien, wie etwa High-Definition-Projektoren (kurz HD-Projektoren) und die weiterhin stark anwachsende Auflösung bei Digitalkameras.

- **Schnelle Erfassung:** Inspiriert durch den Trend, die Anzahl der Eingabebilder auf nur wenige zu reduzieren, stellen wir in dieser Arbeit ein Verfahren vor, das nur drei Aufnahmen benötigt, d. h. $|\tilde{\mathcal{I}}| = 3$. Für eine größtmögliche Schärfe der Aufnahmen, schließen wir, auf Kosten einer längeren Belichtungszeit, die Blende des Objektivs. Ein automatisierter Prozess überträgt anschließend die Bilder von der Kamera zum Rechner für die Weiterverarbeitung. Da eine Aufnahme durch die Belichtungszeit bestimmt wird und die Übertragungsgeschwindigkeit der unkomprimierten Bilder begrenzt ist, sind schnelle Erfassungen besonders von Interesse. Außerdem besteht potenziell die Möglichkeit, wie in [Zhang et al. '02, Rusinkiewicz '01], mit Videokameras bewegliche Objekte zu erfassen.
- **Robustheit:** Dadurch, dass wir mehrere reflektierte Streifen aus wenigen Bildern identifizieren wollen und wir annehmen müssen, dass die Reflexionseigenschaften der Oberfläche nicht ideal sind, kann es durchaus zu Falschzuordnungen bzw. Ausreißern nach der optischen Triangulierung kommen. Nachfolgende Schritte, wie etwa die Registrierung und die Flächenrekonstruktion, sind besonders empfindlich gegen Ausreißer. Um Falschzuordnungen bestmöglich vorzubeugen, entwickeln wir ein farbiges Projektionsmuster und einen Korrespondenzalgorithmus, der die Zeitkohärenz ausnutzt, um eine eindeutige Kodierung einer hohen Anzahl von Streifen zu erzeugen. Außerdem verwenden wir die lokale Raumkohärenz, um eine Robustheit bei lokalen Selbstverdeckungen und Unstetigkeiten zu gewährleisten. Insbesondere muss auch ein stark abweichendes $\tilde{\mathcal{I}}$ toleriert werden.
- **Genauigkeit:** Die Genauigkeit eines Scansystems wird von vielen Faktoren bestimmt. Daher werden eine genaue Farb- und Geometriekalibrierung des Systems, sowie ideale Reflexionseigenschaften vorausgesetzt. Wie auch bei [Curless '97] angedeutet, ist selbst bei einem laserbasierten Verfahren mit Raum-Zeit-Analyse eine Gewährleistung der Genauigkeit fraglich. Obwohl die Rekonstruktion eines geometrisch konsistenten Modells wichtiger ist als die absolute Genauigkeit, benutzen wir, wenn die Möglichkeit besteht, eine subpixelgenaue Lokalisierung der Streifen. In Anlehnung an

[McIvor & Valkenburg '97] wurden diese Verfahren in [Li et al. '04] vorgestellt (siehe Kap. 4).

- **Effiziente Zuordnung:** Da es sich bei uns nicht um ein Echtzeitverfahren handelt, spielt die Geschwindigkeit der Aufnahme eine wichtigere Rolle als die Offline-Verarbeitung der erfassten Daten. Dennoch ist es sinnvoll, dass die Algorithmen innerhalb einer vernünftigen Zeit terminieren. Aufgrund der hohen Anzahl von Streifen die projiziert werden, ist die Laufzeit einer naiven Implementierung der korrelationsbasierten Zuordnung recht problematisch. In dieser Arbeit wird eine effiziente Datenstruktur eingeführt, um den Rechenaufwand der Zuordnung drastisch zu reduzieren. Folglich ist dieses Rekonstruktionsverfahren auch für heutige PCs geeignet.
- **Automatisierbarkeit:** Während wir in dieser Arbeit demonstrieren, dass wir in der Tat sehr gute Rekonstruktionen erzielen, ist eine andere Frage die Handhabbarkeit des Verfahrens. Bei den meisten Verfahren, die in Abschnitt 1.2 vorgestellt worden sind, werden viele Eingaben implizit vorausgesetzt, die nur durch Kenntnis der eingesetzten Algorithmen bestimmt werden können. Außerdem müssen oft wiederholt unterschiedliche Eingabewerte ausprobiert werden, um eine optimale Lösung zu erzielen. Deswegen muss ein Verfahren entwickelt werden, dass eine möglichst geringe Anzahl von Eingaben benötigt und auch bei nicht optimalen Eingaben zu einer guten Rekonstruktion führt, d. h. $|\Sigma|$ muss minimiert werden und es müssen auch stark abweichende σ_k toleriert werden.
- **Flexible kostengünstige Hardware:** Aufgrund der begrenzten maximalen Brennweiten der Optik heutiger Projektoren, haben die projizierten Streifenbreiten eine untere Grenze. Es ist i. Allg. nicht möglich, auf die Präzision der laserbasierten Scanverfahren mit Raum-Zeit-Analyse heranzukommen, die in [Curless '97] präsentiert worden sind. Bei Objekten ab einer bestimmte Größe sind diese kleinen Details i. Allg. nicht mehr von herausragender Bedeutung, da sie schnell zu einer sehr hohen Datenkomplexität führen und diese nur schwierig handhabbar sind. Wir glauben deswegen, dass aus Kostengründen die Nutzung einfacher Hardware und die Verlagerung der Verfahren in die Software eine höhere Akzeptanz für viele Anwendungen erzielen kann. Andererseits können viele Objekte (z. B. aufgrund ihrer Masse) nicht bewegt werden. Dabei ist es wichtig, dass der Scanner große räumliche Freiheitsgrade hat, damit auch schwer zugängliche Bereiche der Oberfläche erfasst werden können. Dieses Verfahren profitiert somit auch von der Entwicklung immer kleinerer Projektoren und Digitalkameras.
- **Integrierte Flächenrekonstruktion:** Die Rekonstruktion der Objektoberfläche aus Punktwolken, die in vielen Arbeiten oft als unabhängiger Schritt betrachtet wird, integrieren wir mit in unsere Rekonstruktionspipeline. In

der Scangeometrie und durch die Zuordnung sind wichtige Informationen über die Topologie des zu rekonstruierenden Dreiecksnetzes implizit enthalten. Aufgrund von vielen nicht detektierten Streifen und Falschzuordnungen würde eine einfache Betrachtung der Nachbarschaften durch die Zuordnung zu vielen Löchern führen. In dieser Arbeit wird ein Flächenrekonstruktionsalgorithmus vorgestellt, der auch bei unvollständigen Erfassungen die gegebene Topologieinformation ausnutzt, um die Punktwolkenkonnektivität korrekt abzuschätzen. Wir zeigen, dass bei stark ungleichmäßigen Abtastungen dieses Verfahren unabdingbar ist. Außerdem wird die Dreiecksnetzerzeugung wie auch in [Li et al. '04, Curless '97] dazu verwendet, um weitere Ausreißer und Rauschen in der Abtastung zu eliminieren (siehe Kap. 5). Dieses Flächenrekonstruktionsverfahren kann insbesondere auch für Punktwolken verwendet werden, die mit anderen optischen Triangulierungsverfahren erfasst worden sind, solange die Zuordnungsinformation nicht verloren gegangen ist.

1.4 Rekonstruktionspipeline

Nachdem wir im vorherigen Abschnitt die Zielsetzungen der 3D-Rekonstruktion festgelegt haben, beschreiben wir hier zunächst die Hauptschritte einer typischen 3D-Rekonstruktion mittels optischer Triangulierung, um die Beiträge dieser Arbeit zum gesamten Rekonstruktionsprozess zu veranschaulichen. Anschließend stellen wir eine detaillierte Pipeline vor, die, ausgehend von einem realen Objekt, ein geglättetes Dreiecksnetz rekonstruiert.

Ablauf einer vollständigen Objektrekonstruktion

Während die Erfassung mittels optischer Triangulierung durch eine Vielzahl von Verfahren technisch realisiert werden kann (siehe Abschnitt 1.1), ist es möglich eine typische Pipeline zu entwerfen, die allgemein gültig ist, wie in Abb. 1.5 skizziert wird. Dieselben Schritte können auch in [Levoy et al. '00] identifiziert werden.

Ein solcher modularisierter Entwurf vereinfacht den Einsatz und den Vergleich unterschiedlicher Verfahren zu jedem Schritt. Die vorgeschlagene Pipeline ist jedoch weit davon eindeutig zu sein. Wie auch im Abschnitt 1.3 angedeutet, können je nach Verfahren auch unterschiedliche Reihenfolgen sowie Überlappungen zwischen den Schritten auftreten. Eine selbstverständliche Voraussetzung einer optischen Triangulierung ist ein geometrischer Kalibrierungsprozess (Kalibrierung). Hier werden, wie bereits zuvor erwähnt, die externen und internen Parameter des Projektors und der Kamera bestimmt (siehe Kap. 5). Eine Möglichkeit die Kamera zu kalibrieren wäre, einige Aufnahmen einer Kalibrierebene in unterschiedlichen Lagen zu machen. Diese Kalibrierebene wird beispielsweise mit einem Schachbrettmuster texturiert und es werden die Eckpunkte über alle Aufnahmen zugeordnet.

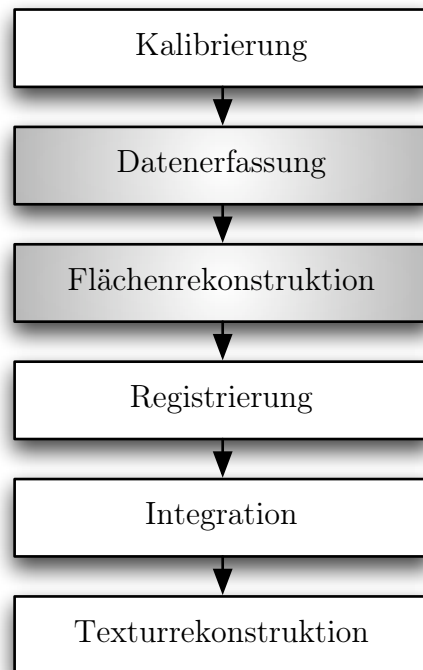


Abbildung 1.5: Typische Schritte einer vollständigen 3D-Rekonstruktion mittels optischer Triangulierung. In dieser Arbeit werden insbesondere die grau markierten Schritte behandelt.

Ausgehend von dieser Zuordnung gewinnt Zhang in [Zhang '00] alle internen und externen Parameter der Kamera. Insbesondere sind die externen Parameter der Kamera bezüglich des Schachbrettmusters gegeben. Dadurch ist es möglich den Projektor zu kalibrieren, indem zusätzliche Kalibrieremuster auf diese Kalibrierebene projiziert werden. Es kann somit zwischen dieser Kalibrierebene und der Projektorbildebene zugeordnet werden und durch Wiederverwenden des Verfahrens von Zhang die Projektorparameter gewonnen werden. Weitere bekannte Kalibrierverfahren werden in [Tsai '87, Heikkila & Silven '97] angegeben. In dieser Arbeit gehen wir von einem kalibrierten Kamera-Projektor-System aus und werden die Kalibrierung des Scansystems nicht mehr weiter behandeln.

Danach besteht die Aufgabe darin, aus den Aufnahmen des strukturiert beleuchteten Objekts eine Punktwolke aus dieser Perspektive zu erfassen (Datenerfassung). Anschließend rekonstruieren wir eine Fläche, die durch ein Dreiecksnetz dargestellt wird, aus dieser Punktwolke (Flächenrekonstruktion). Folglich erhalten wir aus mehreren Aufnahmen unterschiedlicher Ansichten des Objekts eine Menge von rekonstruierten Flächen der jeweiligen Perspektive.

Im Hinblick auf die Rekonstruktion einer vollständigen Oberfläche, liegt es nun nahe, die Dreiecksnetze dieser verschiedenen Ansichten in ein einziges Koordinatensystem zusammenzuführen (Registrierung). In einigen Arbeiten wurde speziell

angefertigte Hardware (Tracking-Systeme) untersucht, die die verschiedenen Lagen der Sensor-Emitter-Paare verfolgen. Aufgrund der Ungenauigkeiten und der hohen Kosten sind viele Arbeiten entstanden, die mittels korrespondierender Merkmale in den überlappenden Bereichen der erfassten Flächen versuchen, die Registrierung zu lösen. Zu den populärsten halbautomatischen Methoden gehören die Iterative-Closest-Point-(ICP)-Algorithmen [Besl & McKay '92, Chen & Medioni '92], wodurch iterativ eine euklidische Bewegung ermittelt wird, um die Distanz zwischen den Punkten aus paarweise teilüberlappenden Flächen, im Sinne der kleinsten Quadrate zu minimieren. Bei diesem Verfahren wird eine grobe manuelle Anordnung vorausgesetzt. Für einen detaillierten Einblick und Vergleich verschiedener ICP-Varianten verweisen wir auf eine Studie von Rusinkiewicz und Levoy [Rusinkiewicz & Levoy '01]. Nennenswert ist in diesem Zusammenhang auch die Rekonstruktionspipeline von Bernardini et al. [Bernardini & Rushmeier '02, Bernardini et al. '01, Bernardini et al. '02], die eine Registrierung der rekonstruierten Flächen mittels beleuchtungsinvarianter Oberflächenfarben (photometrischer Daten) unterstützt. Da das automatische Registrierungsproblem ein unabhängiges und an sich recht komplexes Problem ist, sprengt es den Rahmen dieser Arbeit und wir erlauben uns, diesen Schritt zu überspringen.

Zusätzlich zu den redundanten überlappenden Bereichen müssen wir aufgrund von Kalibrier- und Messungenauigkeiten davon ausgehen, dass nach einer Registrierung Durchdringungen und Abweichungen zwischen den Flächen auftreten. Es liegt nahe, alle rekonstruierten Flächen in eine einzige konsistente Fläche zu integrieren (Integration). Eine Möglichkeit, die in [Turk & Levoy '94] entwickelt worden ist, besteht darin, die überlappenden Bereiche zu entfernen und hinterher entlang den Rändern die Flächen zusammenzunähen. Im Anschluss werden die Daten der Überlappungen wieder gewichtet eingeführt, um nahtlose Übergänge zwischen den Flächen zu erhalten. Der Vorteil von diesem Verfahren ist, dass größtenteils die erfasste Flächengeometrie erhalten bleibt.

Alternativ wurde in [Curless & Levoy '96] ein volumetrischer Ansatz vorgestellt. Hier werden die registrierten Flächen in einem 3D-Gitter implizit durch Distanzfunktionen für jede Ansicht dargestellt und inkrementell, in überlappenden Bereichen, mit Gewichten gemittelt. Die gesuchte Fläche ist letztendlich die Nullstellenmenge dieser Distanzfunktion und daraus wird eine so genannte *Isofläche* extrahiert. Insbesondere hat dieses Verfahren den Vorteil, robuster zu sein. Es ist beliebig oft möglich, mit weiteren Erfassungen die integrierte Fläche zu aktualisieren und es werden nur Fehler, die entlang der Sichtgeraden der Kameraansichten entstehen, in Betracht gezogen. Ein weiterer Vorteil dieser Methode ist die Möglichkeit Löcher zu füllen, um eine wasserdichte Oberfläche zu rekonstruieren. Nach einer Integration erhalten wir ein geometrisch vollständig rekonstruiertes Dreiecksnetz. Ein Nachteil allerdings ist der hohe Speicherverbrauch und Rechenaufwand.

Auch den Integrationsschritt überspringen wir in dieser Arbeit, da er schon für sich eine ausführliche Untersuchung benötigt und bereits eine Reihe von Implementierungen existieren, die im Internet zur Verfügung stehen [Curless '01].

Anschließend können wir die vorher durch Projektion eines weißen Bilds gewonnene Aufnahme zur Rekonstruktion einer beleuchtungsinvarianten Textur verwenden, die letztendlich auf das Dreiecksnetz projiziert und an den überlappenden Bereichen überblendet wird, um einen nahtlosen Texturübergang zu erhalten. Wir beschränken uns in dieser Arbeit auf die Rekonstruktion von nicht texturierten Dreiecksnetzen einzelner Ansichten. Demzufolge überspringen wir auch die Flächenintegration und die Texturrekonstruktion.

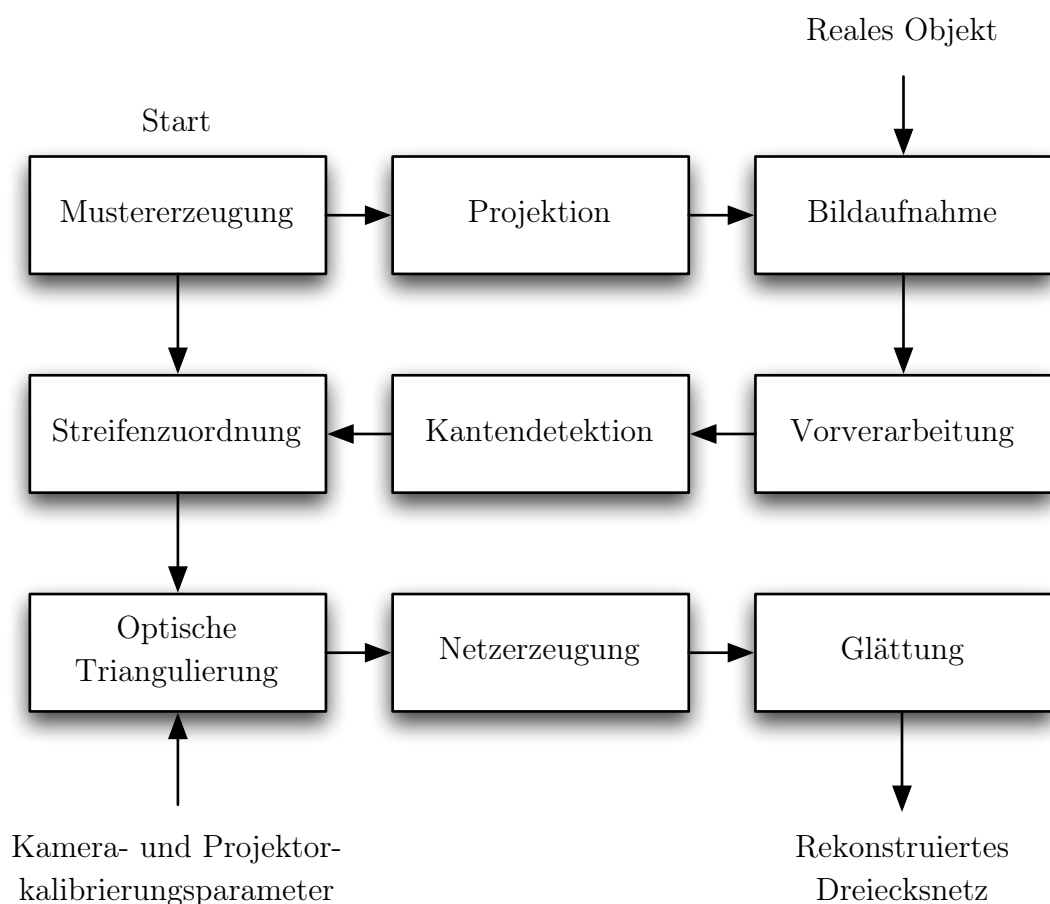


Abbildung 1.6: Datenfluss der vorgestellten 3D-Rekonstruktion.

Datenerfassungs- und Flächenrekonstruktionspipeline

Der Ablauf unserer 3D-Rekonstruktion wird detailliert in Form eines Datenflusses in Abb. 1.6 veranschaulicht. Wir beginnen mit der Erzeugung einer Menge von sorgfältig entworfenen Mustern (i. d. R. zwei) und projizieren diese nacheinander

auf das zu scannende Objekt (Kap. 3). Dabei wird das strukturiert beleuchtete Objekt aus einer anderen Perspektive fotografiert. Zusätzlich wird eine weitere Aufnahme des von weißem Projektorlicht beleuchteten Objekts erstellt. Ein Vorverarbeitungsschritt sorgt zunächst dafür, dass das Objekt aus den Hintergründen der Aufnahmen extrahiert wird (Kap. 4). Weiterhin werden die Farben der strukturiert beleuchteten mittels der mit weißem Licht beleuchteten Aufnahmen korrigiert. Außerdem mindern wir das Rauschen in den Aufnahmen mittels eines Farbmedianfilters. Daraufhin werden die Streifenübergänge mit einer Kantendetektion lokalisiert. Da die projizierten Muster auch Streifenübergänge enthalten können, die nicht auf allen Bildern zu sehen sind, müssen wir alle erfassten Streifenübergänge aus allen Aufnahmen integrieren. Nachdem wir die Streifenübergänge lokalisiert haben, korrespondieren wir diese mit den projizierten Streifenübergängen durch einen Zuordnungsalgorithmus. Folglich können wir aus den korrespondierten projizierten und erfassten Streifenübergangspaaren mittels optischer Triangulierung und der zugehörigen Kalibrierungsparameter eine Punktwolke erzeugen (Kap. 5). Diese Punktwolke enthält in der Praxis weiterhin viele Ausreißer und Rauschen. Durch A-priori-Information der Zuordnung und der Scangeometrie können wir jedoch daraus ein topologisch korrektes Dreiecksnetz rekonstruieren. Zum Schluss verwenden wir diese Vernetzung, um die restlichen Ausreißer zu eliminieren, sowie das Rauschen mittels eines iterativen Glättungsverfahrens zu reduzieren. Die Ergebnisse werden in Kapitel 6 präsentiert.

Kapitel 2

Beschränkungen des Scansystems

Der Entwurf von Algorithmen zu inversen Computer-Vision-Problemen setzt ein grundlegendes Verständnis der Szenengeometrie und der Entstehung von aufgenommenen Bildern voraus. Da eine genaue physikalische Modellierung oft zu komplex und nicht umsetzbar ist, müssen eine Reihe von Annahmen gemacht werden, um das Auftreten gewisser Störungen zu minimieren. Die Aufgabe besteht nun darin, einen vernünftigen Kompromiss zwischen physikalischer Korrektheit und mathematischer Einfachheit zu finden. Durch die Hinzunahme zusätzlicher Annahmen ist das entworfene Rekonstruktionssystem nur noch unter bestimmten Bedingungen einsetzbar. Diese Beschränkungen werden in diesem Kapitel für die optische Triangulierung mit Kamera und Projektor behandelt.

Die Erfassung von Objekten mittels optischer Triangulierung ist die Analyse von projiziertem Licht, welches vom Objekt reflektiert und anschließend auf der Bildebene des Sensors aufgenommen wird. Deswegen liegt es nahe, das Lichtsignal von der Erzeugung bis zur Aufnahme zu verfolgen (siehe Abb. 2.1), um potentielle Störquellen zu beschreiben. Wenn die Beschränkungen einer Komponente nicht beachtet werden, kann es zu gravierenden Fehlrekonstruktionen kommen.

Die Analyse der Beschränkungen, die hier präsentiert wird, ist weit von der Vollständigkeit entfernt und hängt stark von den abstrahierten physikalischen Modellen ab. Dennoch glauben wir, die wichtigsten Unzulänglichkeiten des Systems, die uns bei den Experimenten begegnet sind sowie aus verwandten Arbeiten bereits bekannt sind, vorgestellt zu haben. Wir beginnen zunächst mit der Beschreibung der Scankonfiguration, um die Schwierigkeiten zu schildern, die bei der Erfassung mit Projektor und Kamera auftreten. Hier werden insbesondere die externen Hardwareeigenschaften und die Auswirkungen der Geometrie des Objekts auf die Genauigkeit der Triangulierung behandelt. Im Anschluss untersuchen wir die Probleme, die aufgrund der eingesetzten Hardware entstehen. Dazu gehören die internen Hardwareeigenschaften, wie etwa die Optik und die spezifischen Einschränkun-

gen des Emitters (Projektor) und des Sensors (Kamera-CCD). Geometrische und signaltheoretische Beschreibungen des Scansystems stehen hier im Vordergrund.

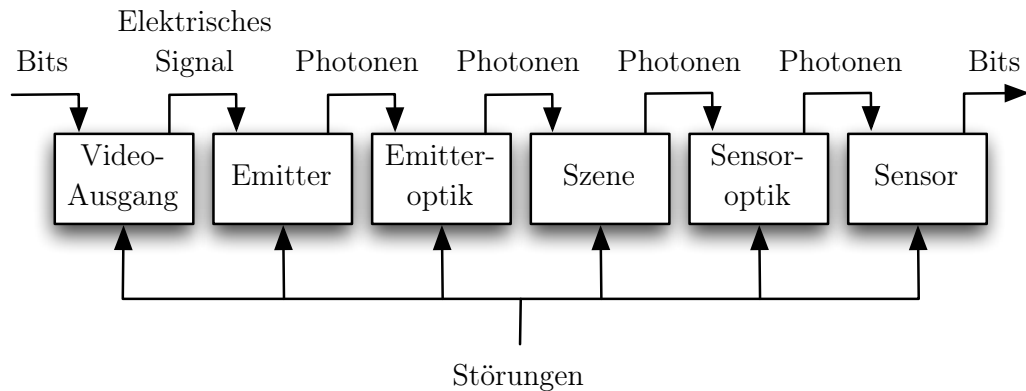


Abbildung 2.1: Modell der Informationsübertragung eines Projektor-Kamera-Systems. Jede Komponente wird durch Störprozesse beeinflusst.

2.1 Geometrische Unzulänglichkeiten

Externe Scankonfiguration

Bei Triangulierungssystemen werden zu Beginn Emitter und Sensor so positioniert und orientiert, dass das projizierte strukturierte Licht vom Sensor aus sichtbar ist. In dieser Arbeit gehen wir von einem schnellen und flexiblen Kalibrierungsprozess aus, weswegen wir uns leisten können, je nach Bedarf nach jeder Aufnahme die Hardware in eine neue Lage zu bringen, während viele Scansysteme ihre Erfassungsansichten auf einer vorprogrammierten Bewegung des Objekts oder des Sensors beschränken (z. B. Laserscanner mit Drehteller). Obwohl wir nicht mehr auf einer festen externen Parametrisierung angewiesen sind, sind die Freiheitsgrade jedoch begrenzt, wie es in diesem Abschnitt erläutert wird. Weiterhin untersuchen wir, welche Auswirkungen die Geometrie des Objekts auf die Genauigkeit der optischen Triangulierung hat. Insbesondere werden hier die wichtigsten Störquellen vorgestellt, die durch die Szene und die externen Eigenschaften der Kamera und des Projektors hervorgerufen werden (siehe Abb. 2.1).

Damit ein Oberflächenpunkt, der einen Strahl des Projektors reflektiert, trianguliert werden kann, muss dieser auf der Bildebene der Kamera sichtbar sein. In

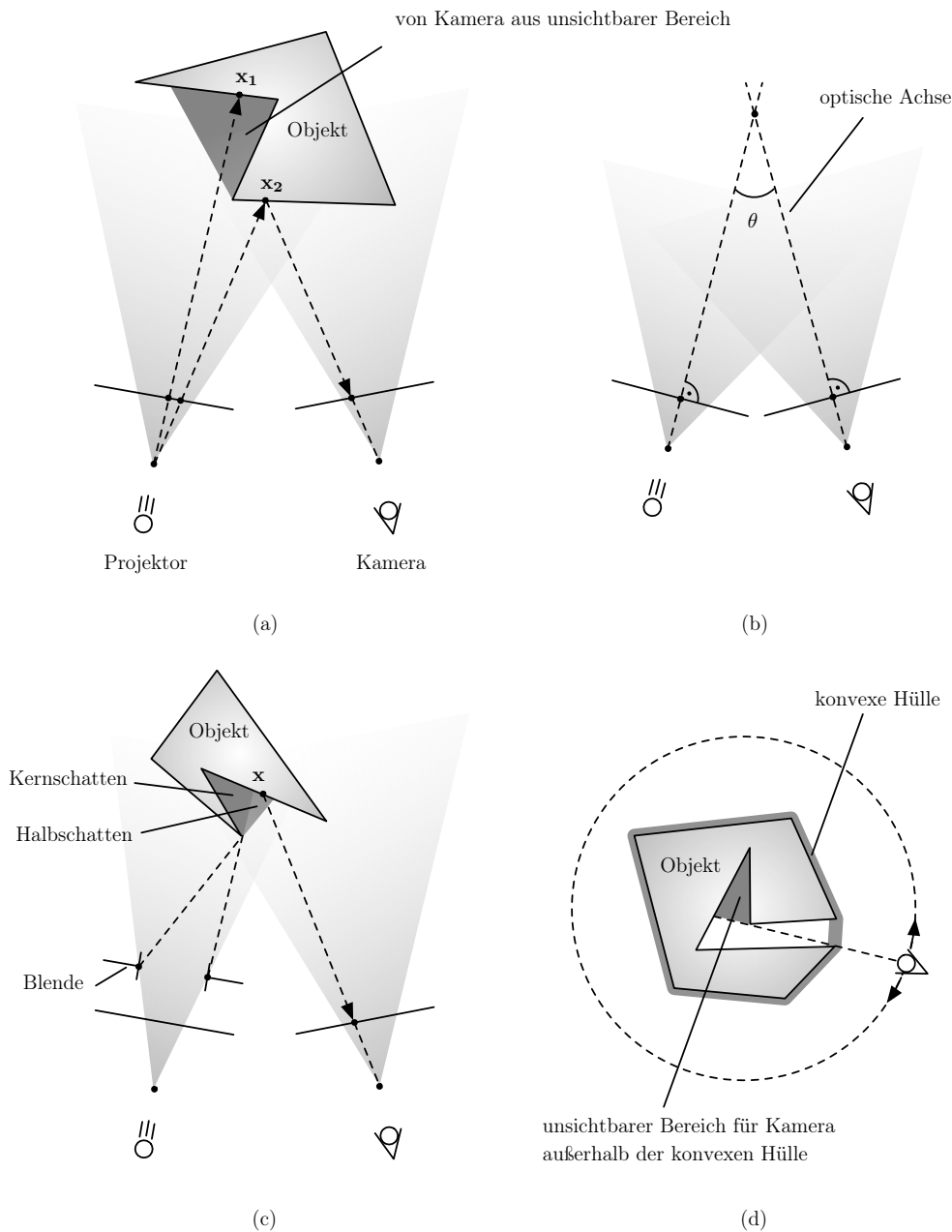


Abbildung 2.2: Sichtbarkeitseinschränkungen. (a) Aufgrund der Verdeckung kann x_1 von der Kamera aus nicht erfasst werden. (b) Der Triangulierungswinkel θ ist der Winkel zwischen den optischen Achsen der Kamera und des Projektors. In der Praxis wählt man $|\theta| \in [15^\circ, 30^\circ]$. (c) Durch die Projektorblende kann der Projektor nicht mehr als Punktlichtquelle betrachtet werden und es wird durch Diffraktion des Lichts ein weicher Schatten geworfen. Wenn x sich im Halbschatten befindet, ist es auf der Kamerabildebene schwer erkennbar. (d) Oft können Objekte nur von außerhalb der konvexen Hülle eines Objekts erfasst werden. Dadurch entstehen nicht zugängliche Bereiche.

Abb. 2.2a beispielsweise sieht man, dass der projizierte Punkt \mathbf{x}_1 durch die Verdeckung des Objekts nicht erfasst werden kann, während \mathbf{x}_2 , sowohl aus Sicht des Projektors als auch der Kamera, sichtbar ist und sich dessen Position als Schnittpunkt von zwei Sichtgeraden bestimmen lässt. Eine direkte Folgerung ist es, zu versuchen, den Winkel θ , zwischen den optischen Achsen von Kamera und Projektor (siehe Abb. 2.2b), so klein wie möglich zu halten. Wir bezeichnen θ auch als *Triangulierungswinkel*. Das Problem bei zu kleinen Winkeln ist allerdings, dass kleine Fehler in der Kamerabildebene zu großen Fehlern beim triangulierten Punkt im Raum führen. In der Praxis wählen wir einen Triangulierungswinkel $|\theta| \in [15^\circ, 30^\circ]$. Ein weiteres Sichtbarkeitsproblem entsteht durch die Tatsache, dass wegen der Öffnung der Projektorblende der Projektor keine ideale Punktlichtquelle ist. Dadurch wird bei einer Verdeckung durch Diffraktion des Lichts ein weicher Schatten geworfen. Es entsteht ein völlig verdeckter Schattenbereich (*Kernschatten*) und ein weicher Übergangsbereich (*Halbschatten*), wie in Abb. 2.2c veranschaulicht wird. Die Kernschattenbereiche sind, ähnlich wie bei der Kamera, verdeckte Bereiche, die das projizierte Licht nicht reflektieren und keine Triangulierung ermöglichen. Falls sich ein projizierter Punkt \mathbf{x} im Halbschatten befindet, ist sein Bild auf der Kamerabildebene stark gestört und nur noch schwierig zu erkennen. Weiterhin kann es durchaus vorkommen, dass Objekte nur von außerhalb ihrer konvexen Hülle erfassbar sind, weil etwa die Optik des Projektors nur aus einem Mindestabstand scharf fokussieren kann oder es z. B. aus Sicherheitsgründen beim Scannen von Kulturgütern in [Levoy et al. '00] nötig war. Als Hilfsmittel werden oft Spiegel eingesetzt, um ein Muster auch an engen Stellen scharf zu projizieren [Fasano et al. '03]. Abbildung 2.2d zeigt den Bereich, der für Kameraansichten außerhalb der konvexen Hülle unsichtbar bleibt.

Auswirkungen der Objektgeometrie auf die Erfassungsgenauigkeit

Eine weitere Frage ist, inwiefern die Geometrie des Objekts die Genauigkeit der optischen Triangulierungen mit Kamera und Projektor beeinflusst. Eine ausführliche Untersuchung für traditionelle Laser-Emitter wird in [Curless '97] gegeben. Es wird üblicherweise angenommen, dass der Querschnitt der Lichtintensität eines projizierten Laserstrahls normalverteilt ist. Während man bei Laserprojektionen nach einem „Zentrum“ eines eindimensionalen (wegen der zeilenweisen Abtastung) Impulses auf der Kamerabildebene sucht, detektieren wir bei unseren Streifenprojektionen die Streifenübergänge mittels Kantendetektion (siehe Kap. 4).

Wir zeigen in diesem Abschnitt, dass die Auswirkungen der Objektgeometrie auf einem reflektierten Streifenübergang denjenigen auf einem reflektierten Laserimpuls entsprechen und potentielle Störquellen für die Genauigkeit in der Triangulierung darstellen. Anschließend vergleichen wir die Lokalisierung von Streifenübergängen mittels Kantendetektionsverfahren mit einer Lokalisierung im Sinne einer

traditionellen Erwartungswert-Analyse bei Laser-Impulsen. Ohne Einschränkung der Allgemeinheit betrachten wir die projizierten Streifen nur in einem Farbkanal.

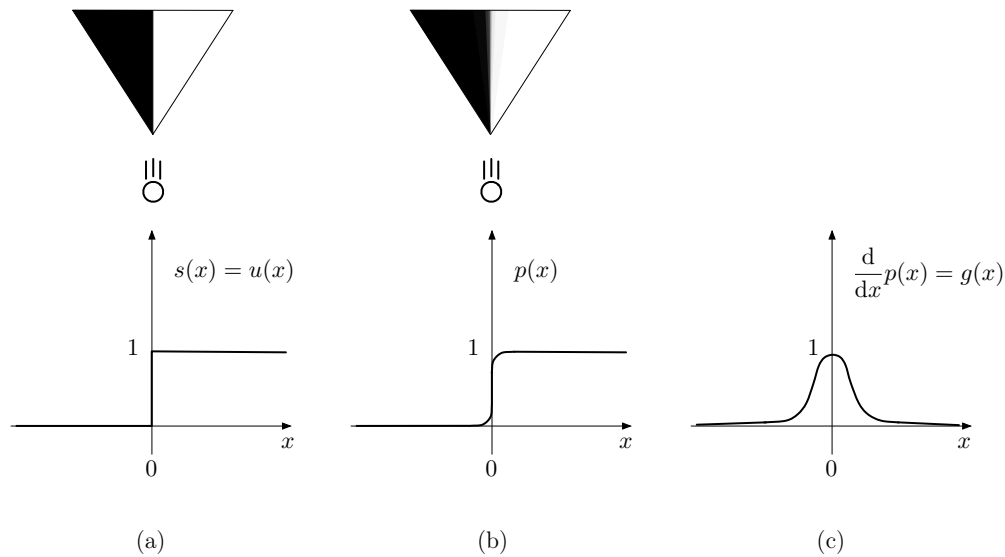


Abbildung 2.3: (a) Die ideale Streifenprojektion $s(x)$ wird durch die Heaviside-Funktion $u(x)$ beschrieben. (b) Aufgrund von Lichtinterferenzen nehmen wir einen weichen Streifenübergang im projizierten Signal $p(x)$ an, der durch die Faltung von $s(x)$ mit einer Gauß-Funktion entsteht. (c) Die Ableitung von $p(x)$ ist wiederum eine Gauß-Funktion, die zur Lokalisierung des genauen Streifenübergangs verwendet wird. Weiterhin dient $\frac{d}{dx}p(x)$ zur Beschreibung der Auswirkungen der Geometrie auf die Genauigkeit der optischen Triangulierung.

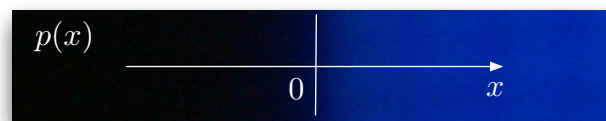


Abbildung 2.4: Die Aufnahme eines projizierten Streifenübergangs entspricht, nach unserer Annahme, einer Glättung der Heaviside-Funktion durch Faltung mit einer Gauß-Funktion.

Sei $u(x)$ die so genannte *Heaviside-Funktion*, die in Form einer *verallgemeinerten Funktion* (auch *Distribution* genannt) folgendermaßen definiert wird:

$$u(x) = \begin{cases} 0 & , \text{ für } x < 0 \\ \frac{1}{2} & , \text{ für } x = 0 \\ 1 & , \text{ für } x > 0 \end{cases} .$$

Dadurch können wir ein ideales Streifenübergangssignal $s(x)$ je nachdem entweder durch $s(x) = u(x)$ oder $s(x) = 1 - u(x)$ beschreiben. Wir beschränken uns auf das Erstere, wie in Abb. 2.3a veranschaulicht wird.

Weder bei LCD- noch bei DLP-Projektoren sind die Pixel auf der Projektorbildebene unendlich dicht beieinander und es treten somit Trennlinien im projizierten Licht auf (siehe Kapitel 2.3). Einfachheitshalber nehmen wir an, dass der Projektor ein Streifenübergangssignal $p(x)$ projiziert, das aufgrund von Lichtinterferenzen einen weichen Streifenübergang hat. Die in Abb. 2.4 gezeigte Aufnahme eines projizierten Streifenübergangs von Schwarz nach Blau auf einer Ebene bestätigt die Korrektheit dieser Annahme. Demzufolge beschreiben wir $p(x)$ durch die Faltung von $s(x)$ mit einer Gauß-Funktion (siehe Abb. 2.3b):

$$p(x) := g(x) * s(x) = g(x) * u(x) = \int_{-\infty}^{\infty} g(\tau)u(x - \tau)d\tau \quad (2.1)$$

mit

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}}e^{-x^2/2\sigma^2} .$$

Die genaue Position x_c des projizierten Streifenübergangs in $p(x)$ wird durch die Position charakterisiert, wo der Betrag der Ableitung von $p(x)$ ein lokales Maximum aufweist, d. h.

$$x_c = \arg \max_x \left\{ \left| \frac{d}{dx} p(x) \right| \right\} .$$

Aus Gleichung 2.1 folgt

$$\begin{aligned} \frac{d}{dx} p(x) &= \frac{d}{dx} \left(\int_{-\infty}^{\infty} g(\tau)u(x - \tau)d\tau \right) \\ &= \int_{-\infty}^{\infty} g(\tau) \frac{d}{dx} (u(x - \tau)) d\tau \\ &= g(x) * \frac{d}{dx} u(x) . \end{aligned} \quad (2.2)$$

Die Ableitung der Heaviside-Funktion $u(x)$ definiert die *Dirac-Delta-Funktion* $\delta(x)$ (auch *Impulsfunktion* genannt):

$$\frac{d}{dx}u(x) = \lim_{T \rightarrow 0} \frac{1}{T} (u(x) - u(x - T)) =: \delta(x) = \delta(-x) \quad . \quad (2.3)$$

Üblicherweise identifiziert man in der Signaltheorie $\delta(x)$ als Grenzfall eines Rechteckimpulses der Breite T und der Höhe $1/T$ für $T \rightarrow 0$ an der Stelle $x = 0$ und es gilt

$$\int_{-\infty}^{\infty} \delta(x) dx = 1 \quad . \quad (2.4)$$

Letztendlich folgt aus den Gleichungen 2.2, 2.3 und 2.4 und der Tatsache, dass $\delta(\tau - x) = 0$, für alle $\tau \neq x$:

$$\begin{aligned} \frac{d}{dx}p(x) &= g(x) * \delta(x) \\ &= \int_{-\infty}^{\infty} g(\tau) \delta(x - \tau) d\tau \\ &= \int_{-\infty}^{\infty} g(\tau) \delta(\tau - x) d\tau \\ &= g(x) \int_{-\infty}^{\infty} \delta(\tau - x) d\tau \\ &= g(x) \quad . \end{aligned}$$

Demzufolge können wir, wie in Abb. 2.3c, aus dem projizierten Streifenübergang $p(x)$ die projizierte Gauß-Funktion $\frac{d}{dx}p(x)$ identifizieren. Dies entspricht dem Gauß-verteiltern Querschnitt eines projizierten Laserstrahls.

Im Anschluss wird das reflektierte Signal von $p(x)$ (bzw. $\frac{d}{dx}p(x)$) auf die Kamerabildebene abgebildet und wir erfassen zeilenweise ein durch das Objekt gestörtes Signal $f(x)$ (bzw. $\frac{d}{dx}f(x)$). Die Aufgabe bestimmt nun darin, aus $f(x)$ das zu $p(x)$ gehörende „Zentrum“ zu lokalisieren. Bei Laserprojektionen werden diese üblicherweise durch den Erwartungswert der Verteilung ermittelt, d. h. das Zentrum in der Erfassung x_m ist gegeben durch:

$$x_m = \frac{\int_{-\infty}^{\infty} x \frac{d}{dx}f(x) dx}{\int_{-\infty}^{\infty} \frac{d}{dx}f(x) dx} = \frac{1}{f(x)} \int_{-\infty}^{\infty} x df(x) \quad .$$

Im Idealfall, würde die Reflexion an einer ebenen Oberfläche zu einer korrekten Lösung führen, wie es in Abb. 2.5a dargestellt wird. Wie bereits erwähnt, lokalisieren wir den genauen Streifenübergang mittels Kantendetektion, d. h. es wird nach einem maximalen lokalen Kontrast in $f(x)$ bzw. nach einem lokalen Maximum des Betrages von $\frac{d}{dx}f(x)$ gesucht:

$$x_e = \arg \max_x \left\{ \left| \frac{d}{dx} f(x) \right| \right\} .$$

Die Genauigkeit einer Triangulierung hängt also von der Lokalisierungsmethode der Streifenübergänge und von der Geometrie des Objekts, welche Störungen auf dem Sensor hervorruft, ab. In Abb. 2.5b erkennt man, dass der reflektierte Streifenübergang teilweise verdeckt ist und nicht vollständig auf die Kamerabildebene abgebildet wird. Eine Triangulierung würde sowohl bei einer Kantendetektion, als auch bei einer Lokalisierung mittels Erwartungswert zu einem abweichenden Punkt führen. Abbildung 2.5c skizziert den Fall, dass der Übergang am Rand der Objektoberfläche nicht vollständig reflektiert wird. Das reflektierte Streifenübergangssignal $f(x)$ hat einen zusätzlichen sprunghaften Übergang und es entsteht, aufgrund des hohen Kontrasts bei der Betrachtung von $\frac{d}{dx}f(x)$, ein neues lokales Maximum, das wiederum zu einer falschen Triangulierung führt. Hier ist ebenfalls eine korrekte Lokalisierung mittels Erwartungswert nicht möglich. Abbildung 2.5d beschreibt den Fall, dass das projizierte Licht auf eine Ecke der Objektoberfläche trifft. Man beobachtet auf der Kamerabildebene eine nach rechts verschobene Verteilung des reflektierten Lichts. Während man bei der Bestimmung des Erwartungswertes ein falsches Zentrum x_m lokalisiert, ist eine Kantendetektion mittels lokalem Maximum in diesem Fall unempfindlich gegenüber den Auswirkungen der Ecke.

2.2 Beschränkungen der Optik

Das Erfassungssystem dieser Arbeit durchläuft grundsätzlich zwei optische Systeme, die Optik des Projektors und der Kamera. Das Projizieren eines Musters auf das Objekt kann als inverse Bildaufnahme einer Kamera gesehen werden. Während das Lochkamera-Modell ein ideales optische Modell darstellt (siehe Kap. 5), so ist es in der Praxis nicht umsetzbar, da es eine unendlich kleine Blendenöffnung voraussetzt, die kein Licht durchlassen würde. Weiterhin gilt, dass bei zu großen und kleinen Blenden (wegen Diffraktion) ein unscharfes Bild entsteht. Im Sinne des *Huygens-Fresnelschen Prinzips*¹ liegt es nahe, die Zerstreung des Lichts mittels

¹Das Huygens-Fresnelsche Prinzip besagt, dass jeder Punkt einer optischen Wellenfront einen Ausgangspunkt einer neuen Welle, der sog. Elementarwelle, darstellt. Die Ausbreitung der gesamten Wellenfront ist die Überlagerung der Elementarwellen.

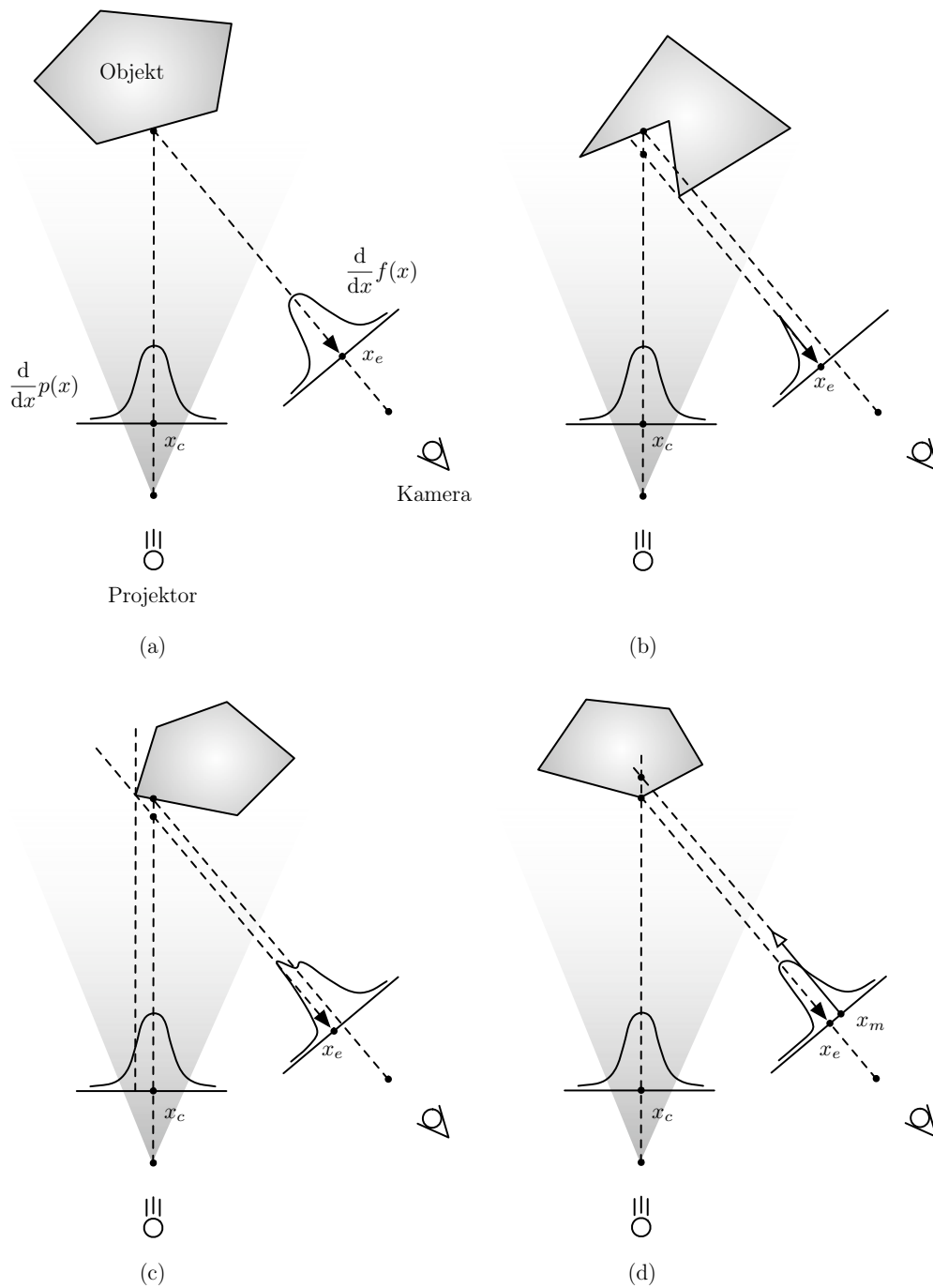


Abbildung 2.5: Die Auswirkung der Oberflächengeometrie auf die Triangulierung wird durch das projizierte Signal $\frac{d}{dx}p(x)$ und das erfasste Signal $\frac{d}{dx}f(x)$ beschrieben.

eines optischen Systems wieder auf einer Bildebene zu fokussieren. Diese werden üblicherweise durch den Einsatz einer Vielzahl von Linsen, Blenden und Spiegeln realisiert und stellt, wie in Abb. 2.1, die Schnittstelle zwischen Projektor und Szene bzw. zwischen Kamera und Szene dar. Die Optik gehört zu den internen Eigenschaften einer Kamera bzw. eines Projektors.

In der Realität findet ein komplexes Zusammenspiel von mehreren Linsen unterschiedlicher Oberflächenkrümmungen statt, um optische Komponenten mit einstellbaren Brennweiten (z. B. Zoomobjektiven) zu ermöglichen und um Abweichungen von idealen Abbildungen zu minimieren. Zur Vereinfachung bietet das *dünne Linsenmodell* eine geeignete geometrische Approximierung eines korrekt fokussierten optischen Systems.

Bei der aktiv-optischen Triangulierung ist eine zuverlässige Erkennung, sowie eine hochgenaue Lokalisierung der projizierten Streifen stark von der Optik abhängig. Zusätzlich zur Tatsache, dass nur ein begrenzter Bereich (*Sichtfeld*) sichtbar ist, und nur auf einer bestimmten Ebene (*Schärfenebene*) scharf fokussiert werden kann, treten Verzerrungen, sowie Abweichungen in der Fokussierung (sog. *Aberrationen*) hervor, die aufgrund von *paraxialen Näherungen* entstehen. Während die meisten professionellen Kamerasysteme mit komplexeren Objektiven ausgestattet sind, und dadurch Aufnahmen mit minimalen Abweichungen produzieren, müssen bei heutigen Projektoren Einschränkungen der Optik hingenommen werden. Der Grund dafür ist, dass bei Projektoren üblicherweise eine hohe Lichtleistung im Vordergrund steht. Ein lichtstarkes Objektiv kann aber nur durch eine begrenzte Anzahl von Linsen erreicht werden.

Geometrische Optik und das dünne Linsenmodell

In diesem Abschnitt führen wir grundlegende Konzepte der geometrischen Optik ein, um einen Überblick über die Beschränkungen und die Störquellen des Linsensystems zu geben. Insbesondere leiten wir durch eine Reihe von Annahmen das dünne Linsenmodell her und geben eine kompakte Beschreibung der Transformation des Lichts in diesem Modell. Für einen ausführlichen Einblick in die Optik verweisen wir auf [Hecht '01].

Die Eigenschaften von transparenten optischen Materialien werden durch deren Brechungsindex

$$n = \frac{c_0}{c} = \frac{\lambda_0}{\lambda}$$

beschrieben, wobei c_0 die Geschwindigkeit des Lichts² (bzw. λ_0 die Wellenlänge) im Vakuum und c die Geschwindigkeit (bzw. λ die Wellenlänge) des Lichts im betrachteten Medium ist. Sei $n_1 \approx 1$ der Brechungsindex der Luft und $n_2 \approx 1,5$ derjenige von Glaslinsen.

²Die Lichtgeschwindigkeit im Vakuum beträgt ca. 299 792 458 m s⁻¹.

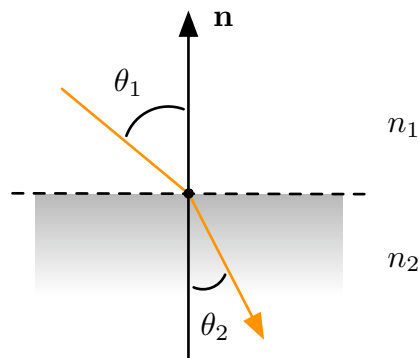


Abbildung 2.6: Das einfallende Lichtstrahl mit Winkel θ_1 zur Normalen \mathbf{n} wird um einen Winkel θ_2 gebrochen.

Wie in Abb. 2.6 veranschaulicht, verhalten sich der Winkel θ_1 des einfallenden Lichtstrahls und derjenige des gebrochenen Lichtstrahls θ_2 gemäß dem *Snellschen Brechungsgesetz*:

$$n_1 \sin \theta_1 = n_2 \sin \theta_2 \quad \Leftrightarrow \quad n_1 \sum_{i=0}^{\infty} \frac{(-1)^i}{(2i+1)!} \theta_1^{2i+1} = n_2 \sum_{i=0}^{\infty} \frac{(-1)^i}{(2i+1)!} \theta_2^{2i+1} .$$

Indem angenommen wird, dass die Winkel θ zwischen dem Lichtstrahl und der optischen Achse an jeder Stelle sehr klein sind ($\theta \ll 1$ rad), können wir die folgende *Näherung 1. Ordnung (paraxiale Näherung)* verwenden:

$$n_1 \theta_1 \approx n_2 \theta_2 \quad . \quad (2.5)$$

Um das optische System zu beschreiben, liegt es nun nahe, die Ausbreitung des Lichts zu jedem Zeitpunkt durch dessen Winkel θ und dessen Abstand x zur optischen Achse darzustellen. Der beschriebene Raum ist daher ein zweidimensionaler Vektorraum $\mathbb{R} \times \mathbb{R}_{\geq 0}$, dessen Elemente $\mathbf{x} = [\theta, x]$ 2-Tupel sind.

Wie in Abb. 2.7 skizziert, breiten sich die Lichtstrahlen in Medien mit konstantem Brechungsindex geradlinig aus (*Transmissionsgesetz*).

Die Ausbreitung über eine Distanz l wird durch die paraxial approximierete 2×2 -Matrix T_l wie folgt beschrieben:

$$\underbrace{\begin{bmatrix} \theta' \\ x' \end{bmatrix}}_{\mathbf{x}'} \approx \underbrace{\begin{bmatrix} 1 & 0 \\ l & 1 \end{bmatrix}}_{T_l} \underbrace{\begin{bmatrix} \theta \\ x \end{bmatrix}}_{\mathbf{x}} .$$

Insbesondere gilt $\theta' = \theta$ und $x' \approx x + l \cdot \theta$, da $\theta \approx \tan \theta$ für kleine θ .

Aus technischen Gründen werden Linsen meistens mit einer sphärischen Oberfläche (mit konstanter Oberflächenkrümmung ungleich Null) hergestellt. Demzufolge

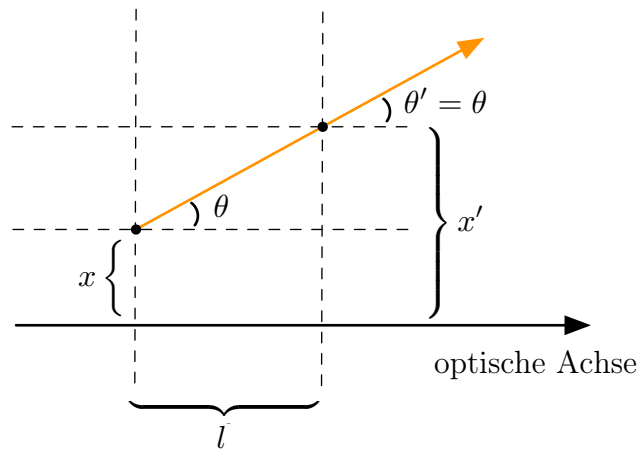


Abbildung 2.7: Geradlinige Ausbreitung des Lichts in einem Medium mit konstantem Brechungsindex.

betrachten wir die Strahlenbrechung (*Refraction*) zwischen zwei Medien unterschiedlicher Brechungsindizes, die durch die Oberfläche einer Kugel getrennt sind (siehe Abb. 2.8). Es findet beim Übergang vom optisch dünneren Medium Luft ($n_1 \approx 1$) in das dichtere Medium Glas ($n_2 \approx 2$) eine Refraktion (im Abstand $x_1 = x_2$) in Richtung der optischen Achse statt.

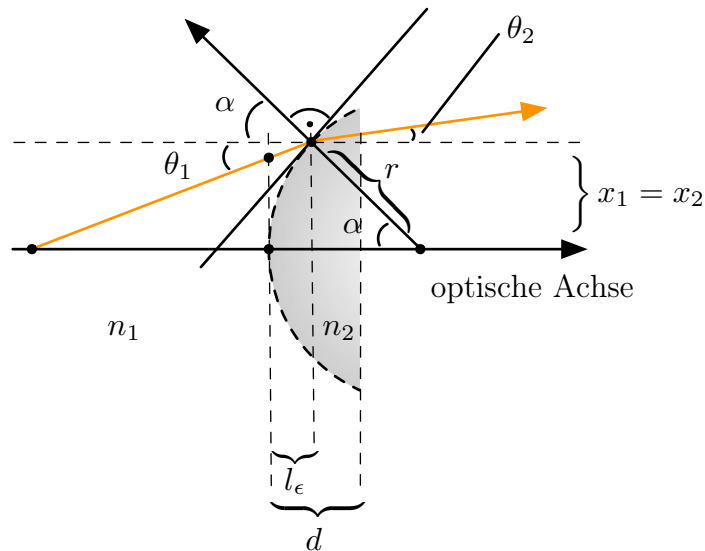


Abbildung 2.8: Die Strahlenbrechung einer Linse wird durch die Refraktion auf einer sphärischen Oberfläche beschrieben.

Sei α der Winkel zwischen der optischen Achse und der Verbindungsgeraden zwischen dem Auftreffpunkt des Lichtstrahls auf der Linse und dem Zentrum der Kugel mit Radius r . Weiterhin sei d die Dicke der Linse. Für dünne Linsen gilt: $d \ll r$. Folglich können wir den Abstand $l_\epsilon < d$, zwischen dem Auftreffpunkt eines Lichtstrahls projiziert auf die optische Achse und dem Schnittpunkt der optischen Achse mit der Kugeloberfläche, ignorieren.

Daraus schließen wir, dass α sehr klein ist:

$$\alpha \approx \sin \alpha = \frac{x_1}{r} = \frac{x_2}{r} \quad .$$

Aus der Gleichung 2.5 folgt, dass paraxial approximierte Brechungsgesetz auf einer sphärischen Oberfläche:

$$n_1 \left(\theta_1 + \frac{x_1}{r} \right) \approx n_2 \left(\theta_2 + \frac{x_2}{r} \right)$$

und wir erhalten

$$\theta_2 \approx \frac{n_1}{n_2} \theta_1 - \frac{1}{n_2} \left(\frac{n_2 - n_1}{r} \right) x_1 \quad . \quad (2.6)$$

Die Lichtstrahlen des Bilds eines Objekts, das sich im Unendlichen befindet, verlaufen parallel zur optischen Achse. Für den Winkel vor der Refraktion gilt für diese Strahlen $\theta_1 = 0$. Ein Bündel solcher Lichtstrahlen schneidet sich durch die Refraktion einer sphärischen Linse in einem gemeinsamen Punkt (*Brennpunkt*), der sich auf der optischen Achse befindet. Wie in Abb. 2.9 dargestellt, ist der Abstand vom Brennpunkt zur Linsenebene, die senkrecht auf der optischen Achse steht, die *Brennweite* f . Der Winkel zwischen gebrochenem Lichtstrahl und optischer Achse kann wie folgt angenähert werden:

$$\theta_2 \approx \tan \theta_2 = \frac{x_1}{f} \quad . \quad (2.7)$$

Aus den Gleichungen 2.6, 2.7 und der Definition der *Brechkraft* p einer sphärischen Oberfläche folgt:

$$p := \frac{(n_2 - n_1)}{r} \approx \frac{1}{f} \cdot n_2 \quad . \quad (2.8)$$

Wie auch bei der geradlinigen Ausbreitung des Lichts können wir die Refraktion mit einer sphärischen Oberfläche mit Brechkraft³ p durch die 2×2 -Matrix M_p beschreiben:

$$\underbrace{\begin{bmatrix} \theta_2 \\ x_2 \end{bmatrix}}_{\mathbf{x}_2} = \underbrace{\begin{bmatrix} \frac{n_1}{n_2} & -\frac{p}{n_2} \\ 0 & 1 \end{bmatrix}}_{M_p} \underbrace{\begin{bmatrix} \theta_1 \\ x_1 \end{bmatrix}}_{\mathbf{x}_1} \quad .$$

³Die Brechkraft einer sphärischen Oberfläche wird in *Dioptrie* gemessen ($1 \text{ dpt} = 1 \text{ m}^{-1}$).

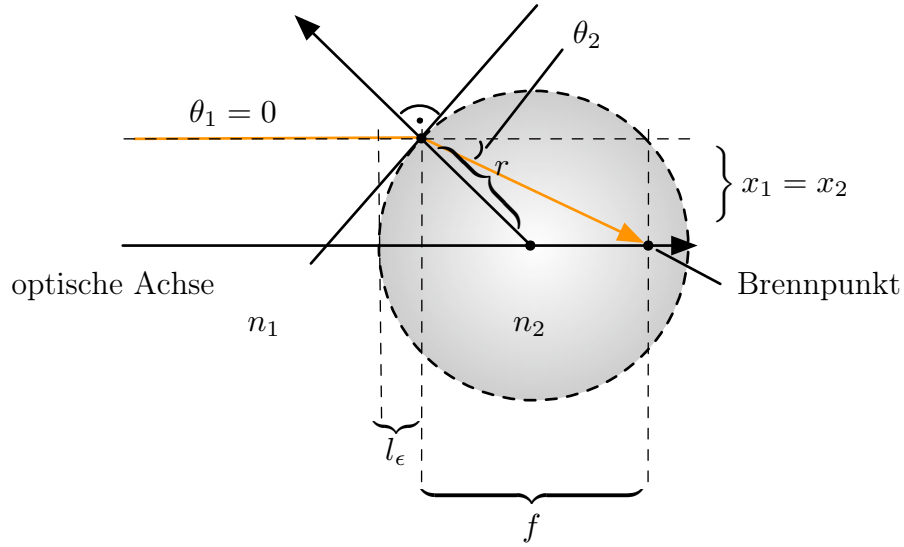


Abbildung 2.9: Alle zur optischen Achse parallelen Strahlen treffen sich im Brennpunkt. Der Abstand zwischen Linse und Brennpunkt definiert die Brennweite f .

Bei einer dünnen bikonvexen Linse tritt der Lichtstrahl auf der einen Seite einer sphärischen Linsenoberfläche mit Radius r ein und auf der anderen Seite einer sphärischen Linsenoberfläche mit Radius r' wieder aus. Während der Eintritt ein Übergang von einem Medium mit Brechungsindex n_1 nach n_2 darstellt, so ist dies für den Austritt gerade umgekehrt. Da es sich außerdem um eine dünne Linse handelt, können wir die geradlinige Ausbreitung des Lichts innerhalb der Linse ignorieren. Die gesamte Refraktion ist eine Nacheinanderausführung von M_p und $M_{p'}$ mit $p' = \frac{(n_1 - n_2)}{r'}$, die folgendermaßen beschrieben wird:

$$\begin{aligned}
 \underbrace{\begin{bmatrix} \theta_2 \\ x_2 \end{bmatrix}}_{\mathbf{x}_2} &= \underbrace{\begin{bmatrix} \frac{n_2}{n_1} & -\frac{p'}{n_1} \\ 0 & 1 \end{bmatrix}}_{M_{p'}} \underbrace{\begin{bmatrix} \frac{n_1}{n_2} & -\frac{p}{n_2} \\ 0 & 1 \end{bmatrix}}_{M_p} \underbrace{\begin{bmatrix} \theta_1 \\ x_1 \end{bmatrix}}_{\mathbf{x}_1} \\
 &= \begin{bmatrix} 1 & -\frac{p}{n_1} - \frac{p'}{n_1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_1 \\ x_1 \end{bmatrix} \\
 &= \begin{bmatrix} 1 & -\frac{1}{n_1}(p + p') \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_1 \\ x_1 \end{bmatrix} .
 \end{aligned} \tag{2.9}$$

Sei $p_{\text{dünn}}$ die Brechkraft einer dünnen Linse. Aus Gleichung 2.10 mit $n_1 = 1$ ergibt sich die so genannte *Linsefmachergleichung*:

$$p_{\text{dünn}} = p + p' = (n_2 - 1) \left(\frac{1}{r} - \frac{1}{r'} \right) . \quad (2.10)$$

Die Refraktion einer dünnen Linse ist durch folgende Matrix gegeben:

$$M_{\text{dünn}} := \begin{bmatrix} 1 & -\frac{p_{\text{dünn}}}{n_1} \\ 0 & 1 \end{bmatrix} .$$

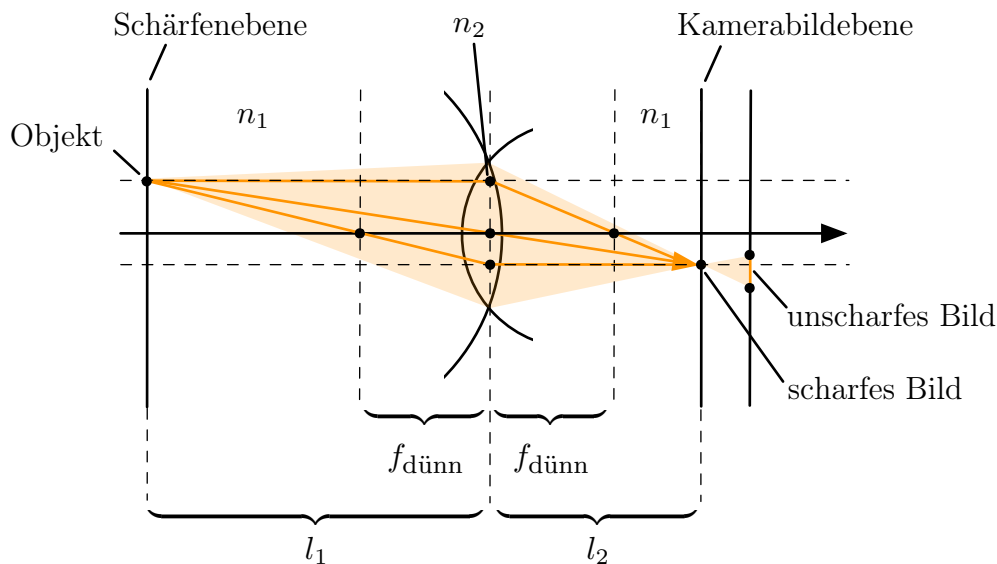


Abbildung 2.10: Das Strahlenbündel eines Objektpunkts wird durch die dünne Linse auf die Kamerabildebene genau dann auf einen scharfen Punkt abgebildet, wenn der Objektpunkt auf der Schärfenebene liegt.

Abbildung 2.10 zeigt den Strahlenverlauf des Lichts eines Objektpunkts durch ein optisches System mit einer dünnen Linse. Bevor das Licht von der Linse gebrochen wird, breitet es sich geradlinig aus. Beim Austritt aus der Linse findet ebenfalls eine geradlinige Ausbreitung zwischen Linse und Kamerabildebene statt. Insgesamt ergibt sich:

$$\underbrace{\begin{bmatrix} \theta_2 \\ x_2 \end{bmatrix}}_{\mathbf{x}_2} = \underbrace{\begin{bmatrix} 1 & 0 \\ l_2 & 1 \end{bmatrix}}_{T_{l_2}} \underbrace{\begin{bmatrix} 1 & -\frac{p_{\text{dünn}}}{n_1} \\ 0 & 1 \end{bmatrix}}_{M_{\text{dünn}}} \underbrace{\begin{bmatrix} 1 & 0 \\ l_1 & 1 \end{bmatrix}}_{T_{l_1}} \underbrace{\begin{bmatrix} \theta_1 \\ x_1 \end{bmatrix}}_{\mathbf{x}_1} .$$

Sei $f_{\text{dünn}} = \frac{n_1}{p_{\text{dünn}}}$ die Brennweite der dünnen Linse. Daraus folgt:

$$\mathbf{x}_2 = \begin{bmatrix} 1 - \frac{l_1}{f_{\text{dünn}}} & -\frac{1}{f_{\text{dünn}}} \\ l_1 + l_2 - \frac{l_1 l_2}{f_{\text{dünn}}} & 1 - \frac{l_2}{f_{\text{dünn}}} \end{bmatrix} \mathbf{x}_1 \quad . \quad (2.11)$$

Aus der Gleichung 2.10 sehen wir, dass ein Lichtstrahl, der von einem Punkt auf dem Objekt ausgeht und mit $x_1 = 0$ auf die Linse trifft, nicht gebrochen wird. Dieser Strahl wird auch *Hauptstrahl* genannt. Er schneidet sich mit dem Lichtstrahl, der mit $\theta_1 = 0$ auf die Linse auftrifft, in einem Punkt. Wenn die Kameraebene durch diesen Punkt geht, ist das Bild dieses Objekts scharf fokussiert. Mit dem Strahlensatz folgt nach einer einfachen Rechnung die *Abbildungsgleichung für dünne Linsen*:

$$\frac{1}{f_{\text{dünn}}} = \frac{1}{l_1} + \frac{1}{l_2} \quad (2.12)$$

$$\Leftrightarrow l_1 + l_2 - \frac{l_1 l_2}{f_{\text{dünn}}} = 0 \quad . \quad (2.13)$$

Mit Hilfe von Gleichung 2.13 lässt sich die Ausbreitung des Lichts (siehe Gleichung 2.11) vereinfachen:

$$\underbrace{\begin{bmatrix} \theta_2 \\ x_2 \end{bmatrix}}_{\mathbf{x}_2} = \begin{bmatrix} 1 - \frac{l_1}{f_{\text{dünn}}} & -\frac{1}{f_{\text{dünn}}} \\ 0 & 1 - \frac{l_2}{f_{\text{dünn}}} \end{bmatrix} \underbrace{\begin{bmatrix} \theta_1 \\ x_1 \end{bmatrix}}_{\mathbf{x}_1} \quad .$$

Man sieht, dass der Abstand x_2 zur optischen Achse auf der Kamerabildebene unabhängig vom Einfallswinkel θ_1 des Lichts auf die Linse ist. Folglich schneiden sich alle durch die Linse refraktierten Lichtstrahlen, die von einem Objektpunkt auf der Schärfenebene mit festem Abstand x_1 zur optischen Achse ausgehen, in einem Punkt mit Abstand x_2 zur optischen Achse auf der Kamerabildebene. Die Zerstreuung des Lichts von diesem Objektpunkt wird also auf der Kamerabildebene nur dann scharf abgebildet, wenn der Abstand zwischen Objektpunkt und Linse l_1 beträgt.

Diese Einschränkung wirkt sich doppelt auf unser Scansystem aus. Während der Videoprojektor das Streifenmuster nur auf eine bestimmte Schärfenebene scharf projiziert, werden nur Objektpunkte, die auf einer anderen Schärfenebene liegen, scharf auf die Kamerabildebene abgebildet. Das Erfassen der reflektierten Streifen auf Objekten mit großen Tiefenunterschieden ist daher sehr schwierig. Laserbasierte Emittoren hingegen können über größere Distanzen hinweg einen scharfen Punkt auf das Objekt projizieren. Da wir in dieser Arbeit Objektoberflächen mit hohen Details erfassen wollen, können wir weit abweichende Tiefen ignorieren, da die Abtastdichte der Oberfläche mit der Entfernung des Objekts abnimmt.

Blendeneinstellung und Tiefenschärfe

Gegenüber dem Projektor, hat man bei der Kamera den Vorteil, die Kontrolle über die Blende des Objektivs zu haben. Dadurch reduzieren wir die Unschärfe in der Erfassung von Objektpunkten, die sich nicht auf der Schärfenebene der Kamera befinden.

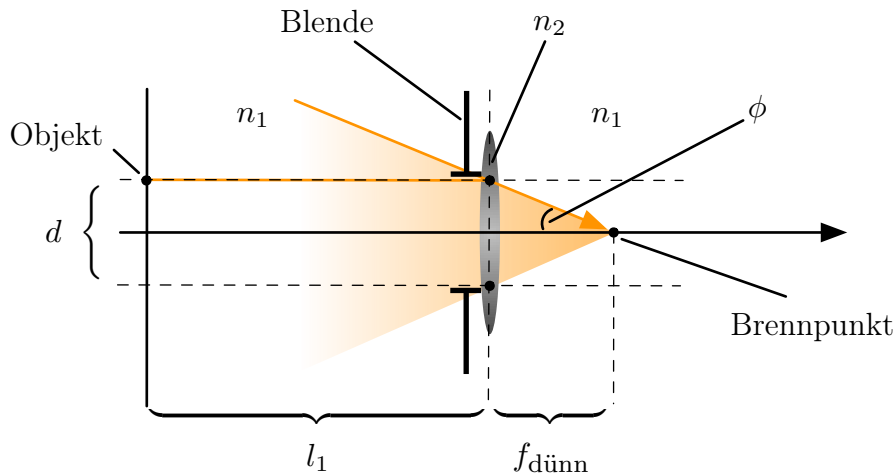


Abbildung 2.11: Blende mit Öffnungsdurchmesser d und Halböffnungswinkel ϕ .

Eine zirkuläre Blende, wie in Abb. 2.11, definiert einen Sichtkegel mit Halbwinkel ϕ , wodurch das Blickfeld 2ϕ (engl. *field of view*) entsteht. Das Maß an Licht, das durch die Blende erfasst wird, ist durch die *numerische Apertur* A_N gegeben:

$$A_N := n \cdot \sin \phi \quad ,$$

wobei n der Brechungsindex des Mediums zwischen Linse und Kamerabildebene ist. Insbesondere gilt $A_N = \sin \phi$ für $n = n_1 \approx 1$. Die Einstellung der Blende geschieht allerdings über die so genannte *Blendenzahl* (engl. *f-number*), die mit κ bezeichnet wird. Sie wird als das Verhältnis zwischen der Brennweite $f_{\text{dünn}}$ und dem Durchmesser der Blendenöffnung d definiert und steht folgendermaßen im Zusammenhang mit der numerischen Apertur:

$$\kappa := \frac{f_{\text{dünn}}}{d} = \frac{1}{2 \tan \phi} = \frac{1}{2 \tan(\arcsin(A_N))} \approx \frac{1}{2A_N} \quad .$$

Je kleiner die Blendenzahl, desto größer ist die numerische Apertur, sowie die Lichtstärke der Optik.

Um die Auswirkung der Blendenzahl auf die Schärfe der Objektpunkte, die sich nicht auf der Schärfenebene befinden, darzustellen, definieren wir gemäß Abb. 2.12 die *Tiefenschärfe* $\Delta l_1 := l_1^{\text{max}} - l_1^{\text{min}}$ eines optischen Systems wobei l_1^{max} die maximale und l_1^{min} die minimale Entfernung des Objektpunkts zur Linse ist und der erfasste Zerstreuungskreisradius ρ innerhalb eines Schwellwertes liegt.

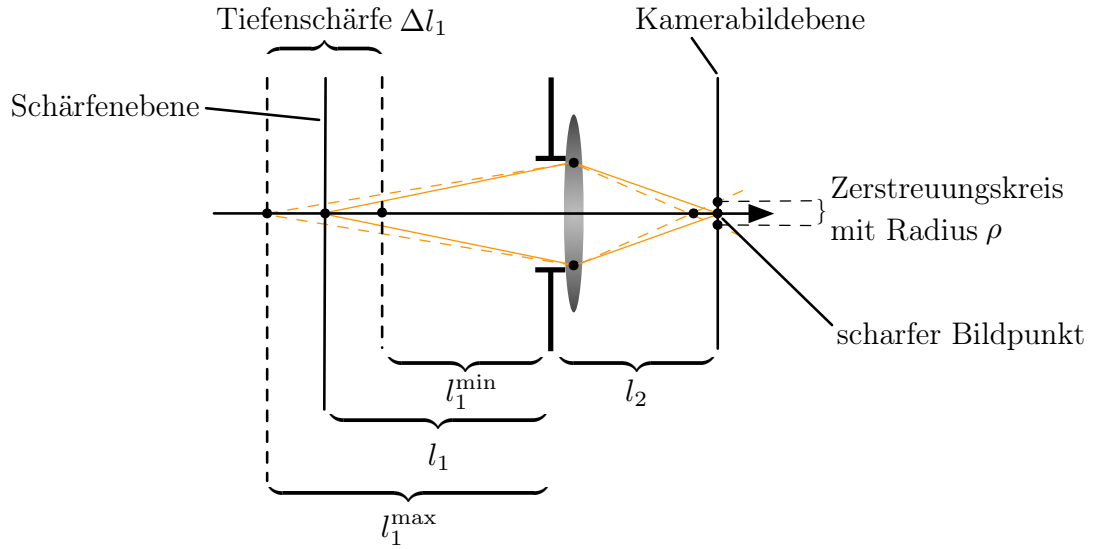


Abbildung 2.12: Ein Objektobjekt, der auf einer senkrechten Ebene zur optischen Achse und Abstand l_1^{\max} zur Linse liegt, wird unscharf als Zerstreuungskreis mit Radius ρ auf die Kamerabildebene abgebildet. Die Tiefenschärfe Δl_1 definiert einen Toleranzbereich, indem vor und hinter der Schärfenebene ausreichend scharf abgebildet wird, d. h. ρ innerhalb eines Schwellwerts liegt.

Es gilt für ein gegebenes ρ :

$$l_1^{\max} = \frac{l_1}{1 - 2\rho(\kappa) \frac{l_1 - f_{\text{dünn}}}{f_{\text{dünn}}^2}} \quad \text{und} \quad l_1^{\min} = \frac{l_1}{1 + 2\rho(\kappa) \frac{l_1 - f_{\text{dünn}}}{f_{\text{dünn}}^2}} .$$

Sei $m = \frac{x_2}{x_1} = \frac{l_2}{l_1}$ die *Lateralvergrößerung*, d. h. das Verhältnis zwischen Bildgröße und Objektgröße. Aus der Gleichung 2.12 folgt die Beziehung:

$$l_1 = \left(\frac{1}{m} + 1 \right) f_{\text{dünn}}$$

und wir erhalten die Tiefenschärfe:

$$\begin{aligned} \Delta l_1 &= l_1^{\max} - l_1^{\min} \\ &= 4\rho\kappa \frac{(m+1)}{m^2} \cdot \frac{1}{1 - \left(\frac{2\rho\kappa}{mf_{\text{dünn}}} \right)^2} \\ &\approx 4\rho\kappa \frac{(m+1)}{m^2} , \end{aligned} \tag{2.14}$$

da $\rho \ll f_{\text{dünn}}$. Es lässt sich aus Gleichung 2.14 direkt folgern, dass die Tiefenschärfe Δl_1 bei gleichem Zerstreuungskreisradius ρ erhöht werden kann, indem entweder die Optik mit einer höheren Blendenzahl oder mit einer kleineren Vergrößerung

eingestellt wird. Außerdem ist der Zerstreuungskreisradius ρ proportional zum Durchmesser d der Blendenöffnung. Als obere Grenze für ρ kann z. B. die Größe eines Pixels auf dem Kamera-CCD gewählt werden. Da wir Streifen projizieren und uns nur für eine eindimensionale Kantendetektion interessieren, müssen wir einen hohen Kontrast zwischen zwei Pixeln garantieren. In der Praxis minimieren wir die Blendenöffnung, indem wir $\kappa = 22$ (meistens als „F 22“ bezeichnet) nehmen, da es die höchste Blendenzahl ist, die unsere Kamera hat. Bei einer Verkleinerung der Blende, muss entsprechend eine längere *Belichtungszeit* (engl. *exposure time*) eingestellt werden, um eine ausreichende Helligkeit in der Aufnahme zu erzielen. Außerdem finden die Aufnahmen oft in einem abgedunkelten Raum statt, um Einflüsse durch andere Lichtquellen zu vermeiden. Je nachdem wie dunkel das Material des zu scannenden Objekts ist, werden in der Praxis Belichtungszeiten von 3 s für dunkle und $\frac{1}{2}$ s für helle Objekte benutzt.

Das hergeleitete paraxial approximierte dünne Linsenmodell beschreibt eine geometrisch aberrationsfreie (d. h. abbildungsfehlerfreie) Optik. Insbesondere gilt, dass sich alle von einem auf der Schärfeebene liegenden Punkt ausgehenden Strahlen exakt in einem Punkt auf der Kamerabildebene schneiden. Die Berücksichtigung von *Näherungen 3. Ordnung* und höher, d. h. $\sin \theta \approx \theta - \frac{\theta^3}{3!}$, führt zu Abweichungen und Unschärfen die auch unter *Seibel-Aberrationen* bekannt sind, wie etwa *sphärische Aberrationen*, *Komas*, *Astigmatismen*, *Petzval-Krümmungen* und *Verzerrungen*. Diese geometrischen Aberrationen werden durch Kombination von unterschiedlichen Linsen in den Objektiven der Kamera und des Projektors stark reduziert und werden in dieser Arbeit nicht weiter berücksichtigt.

Chromatische Aberration

Nichtsdestotrotz haben wir insbesondere bei der Optik des Projektors eine andere Art von Aberration, die sog. *chromatische Aberration* (Farblängsfehler), beobachtet die aufgrund des Dispersionseffekts von Lichtwellen in Linsen auftreten. Genauer gesagt, hängt der Brechungsindex der Linse von der Wellenlänge des Lichts ab, wie in Abb. 2.13 veranschaulicht wird.

Sei $f_{\text{dünn}}^{\max}$ die Brennweite und $p_{\text{dünn}}^{\max}$ die Brechkraft für den Brechungsindex n^{\max} einer Wellenlänge λ^{\max} . Entsprechend sei $f_{\text{dünn}}^{\min}$ und $p_{\text{dünn}}^{\min}$ für den Brechungsindex n^{\min} einer Wellenlänge λ^{\min} definiert. Aus der Gleichung 2.10 folgt:

$$f_{\text{dünn}}^{\max} = \frac{1}{p_{\text{dünn}}^{\max}} = \frac{1}{(n^{\max} - 1)} \left(\frac{rr'}{r' - r} \right)$$

und

$$f_{\text{dünn}}^{\min} = \frac{1}{p_{\text{dünn}}^{\min}} = \frac{1}{(n^{\min} - 1)} \left(\frac{rr'}{r' - r} \right) \quad .$$

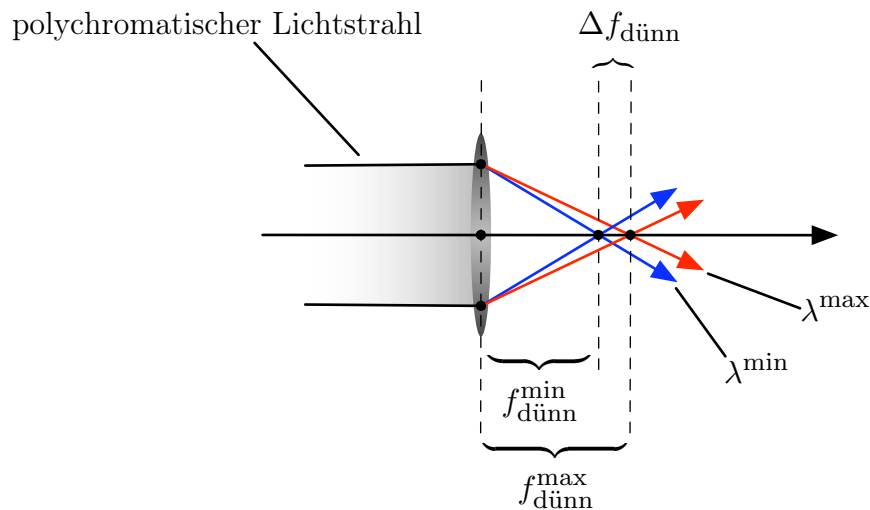


Abbildung 2.13: Hier trifft ein polychromatischer Lichtstrahl auf die Linse, sodass es zu Dispersion kommt. Die längeren Lichtwellen (z. B. rot) haben eine längere Brennweite als die kürzeren (z. B. blau).

Die *laterale chromatische Aberration* oder der *Farbvergrößerungsfehler* $\Delta f_{\text{dünn}}$ ist die Differenz der Brennweite von zwei Lichtstrahlen unterschiedlicher Wellenlänge λ^{\max} und λ^{\min} :

$$\begin{aligned} \Delta f_{\text{dünn}} &= f_{\text{dünn}}^{\max} - f_{\text{dünn}}^{\min} \\ &= \left(\frac{1}{n^{\max} - 1} - \frac{1}{n^{\min}} \right) \left(\frac{r r'}{r' - r} \right) \\ &= \frac{n^{\min} - n^{\max}}{(n^{\max} - 1)(n^{\min} - 1)} \left(\frac{r r'}{r' - r} \right) \end{aligned}$$

Die chromatische Aberration wirkt sich äußerst stark bei der Projektion von farbigen Streifenmustern aus, da diese meistens *polychromatisch* sind, d. h. aus Farben, die aus Licht mit mehreren Wellenlängen bestehen, zusammengesetzt sind. Es entstehen dadurch verfälschte Farben, die eine zuverlässige Streifenzuordnung erschweren. Diese Aberration kann leider nur mit Hilfe von zusätzlichen achromatischen Linsen vollständig korrigiert werden, die bei üblichen Projektoren nicht vorhanden sind. In Abb. 2.14 werden dazu zwei Beispiele gezeigt. Laser-Emitter hingegen projizieren ein kohärentes Licht mit einer einzigen Wellenlänge und sind nicht von diesem Problem betroffen.

2.3 Beschränkungen des Emitters

Wie bereits erwähnt, projizieren wir die Streifenmuster mittels eines portablen ASK M2 DLP-Videoprojektors, um anschließend durch optische Triangulierung

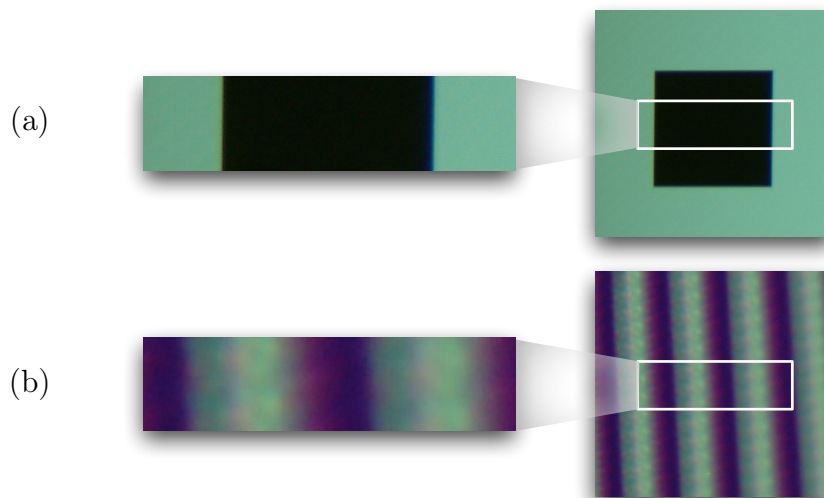


Abbildung 2.14: (a) Bei der Projektion eines schwarzen Vierecks mit weißem Hintergrund auf eine weiße Ebene können wir deutlich an den Farbübergängen eine rötliche oder eine bläuliche Farbverfälschung beobachten. (b) Ein weiteres Beispiel einer Projektion von Schwarz-Weiß-Streifen, die jeweils eine Breite von zwei Pixeln haben. Auch hier ist die chromatische Aberration deutlich zu erkennen.

aus den erfassten Streifenübergängen die Tiefeninformation zu gewinnen. Videoprojektoren haben im Gegensatz zu Diaprojektoren den Vorteil, dass die Position der Streifenübergänge auf der Projektorbildebene exakt vorgegeben werden kann. Somit ist auch gewährleistet, dass die Streifenkanten, die sich bei mehreren Aufnahmen an derselben Stelle befinden, exakt in jeder Aufnahme übereinstimmen. Der Einsatz von heutigen Videoprojektoren ruft allerdings eine Reihe von Unzulänglichkeiten hervor, die in diesem Abschnitt aufgezeigt werden. Wir beginnen mit einem kurzen Einblick in die DLP-Technologie [Hornbeck & Nelson '88], um deren Hauptproblematik zu schildern. Anschließend zeigen wir aus einer signaltheoretischen Perspektive, inwiefern die Projektion von Streifenmustern die Diskretisierung von hochfrequenten Objektoberflächen beeinträchtigt und leiten die Entstehung eines eindimensionalen Aliasing-Problems her.

DLP- und LCD-Videoprojektoren

DLP-Projektoren (Digital Light Processing) bieten im Vergleich zu traditionellen LCD-Projektoren (Liquid Crystal Display) wichtige Vorteile für eine Streifenprojektion mit hohem Signal-zu-Rauschverhältnis. DLP-Projektoren basieren auf CMOS-Halbleiterarrays (Complementary Metal Oxide Semiconductor), die aus winzigen ($16 \mu\text{m}^2$), in einer Matrix angeordneten Spiegeln bestehen, die auch DMDs (Digital Micromirror Device) genannt werden. Jeder dieser Spiegel stellt

ein Pixel dar und kann durch eine elektrostatische Ansteuerung mit bis zu einer Frequenz von 1 kHz zwischen einem Winkel von $+10^\circ$ (Pixel an) und -10° (Pixel aus) geneigt werden. Eine Halogen-Metall dampflampe erzeugt ein Licht, das durch eine Kondensator-Linse auf das DMD trifft. Je nachdem, welchen Winkel ein Spiegel gerade hat, wird entweder das Licht auf die Projektoroptik reflektiert oder auf ein lichtabsorbierendes Material abgelenkt.

Um verschiedene Grautöne zu erzeugen, wird ein entsprechendes Verhältnis der Anzahl der angeschalteten und der ausgeschalteten Zeitintervalle pro Sekunde reguliert. Diese Regulierung erfolgt durch *Modulation der Impulsbreite* (engl. *pulsewidth modulation*) [Hornbeck '95]. Das i -te Bit eines 8-Bit-Steuersignals $\mathbf{s} = [s_0, \dots, s_7] \in \mathbb{F}_2^8$ stellt eine Zeitdauer von $\frac{2^i}{2^8-1} t_{\text{vft}}$ eines angeschalteten Pixels dar, wobei t_{vft} die gesamte Integrationszeit eines projizierten Signals (*video field time*) ist. t_{vft} wird also in 8 unterschiedliche Zeitintervalle unterteilt, wobei das LSB (Least Significant Bit) für das kürzeste und das MSB (Most Significant Bit) für das längste Intervall steht. Die gesamte Zeitdauer eines angeschalteten Pixels innerhalb von t_{vft} wird gegeben durch $t_{\text{on}} = \sum_{i=0}^7 s_i \frac{2^i}{2^8-1} t_{\text{vft}}$. Folglich können 2^8 äquidistante Grauwerte erzeugt werden. Anschließend wird außerdem ein Bit-Aufspaltungsverfahren (bit-split) verwendet, um t_{on} gleichmäßiger über die Zeit zu verteilen.

Für die Erzeugung von Farben durchläuft das vom DMD reflektierte Licht ein Farbrad, das aus drei Farbfiltern besteht (Rot, Grün und Blau). Beispielsweise wird ein cyanfarbiges Licht projiziert, indem die zugehörigen Pixel ausgeschaltet sind, während sich der rote Filter auf der optischen Achse des Projektors befindet. Es können je nachdem 24-Bit-(8-Bit pro Primärfarbe) oder 30-Bit-(10-Bit pro Primärfarbe)-Farben projiziert werden.

DLP-Projektoren basieren auf reiner Reflexion des Lichts und haben daher eine höhere Lichtstärkeneffizienz als LCD-Projektoren, deren Licht durch die Polarisationsfilter und Flüssigkristallmoleküle teilweise absorbiert werden. Durch die Projektion mit hellerem Licht können wir entsprechend die Belichtungszeit der Kamera verkürzen.

Weitere Vorteile von DMDs gegenüber LCDs sind die uniformen und kleineren Abstände ($1 \mu\text{m}$) zwischen den Pixeln auf dem Chip, die bei einer fokussierten Projektion sichtbar werden. Abbildung 2.15 zeigt den signifikanten Unterschied von REM-(Rasterelektronenmikroskop)-Aufnahmen beider Technologien. Durch die Anordnung der Elektroden und der Transistoren entstehen bei LCDs größere horizontale Lücken zwischen den Pixeln, wodurch der Füll-Faktor stark beeinträchtigt wird. Diesen Effekt bezeichnet man oft als *Hühnerdraht-Effekt* (engl. *chicken wire effect*) und kann nicht vollständig beseitigt werden, wie in Abb. 2.16 bei einer DLP-Projektion zu sehen ist. Diese Lücken bilden Trennlinien in den Aufnahmen und stören vor allem die Kantendetektion von Streifen, die im projizierten Bild breiter als ein Pixel sind. DLP-Projektoren können insgesamt kontrastreichere und

schärfere Streifenmuster erzeugen als herkömmliche LCD-Projektoren und sind für die Rekonstruktion mit strukturiertem Licht besser geeignet.

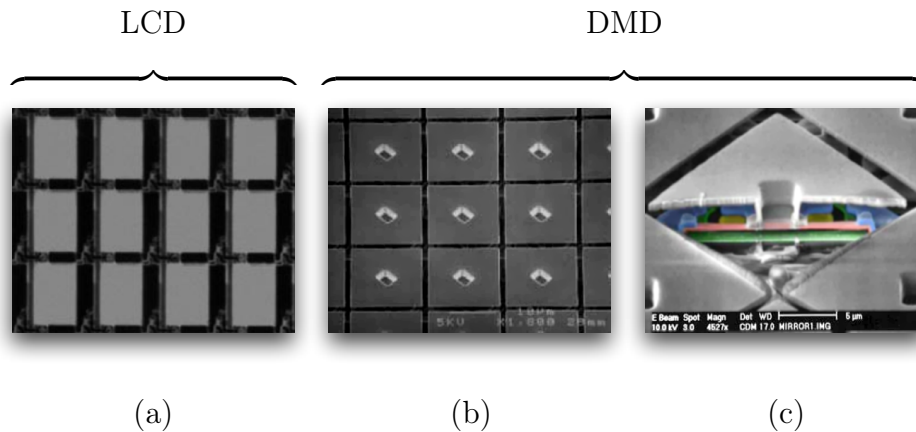


Abbildung 2.15: REM-Aufnahmen (aus [Hornbeck '96, Yoder '97]) eines Poly-Si-LCDs (a), eines DMDs (b) und dessen Querschnitt (c). DMD-Pixel werden durch uniformere und kleinere Abstände getrennt als LCDs. Der Hühnerdraht-Effekt wird dadurch reduziert und der Füll-Faktor auf ca. 90 % erhöht.

DVI- und VGA-Videoschnittstelle

Wie in Abb. 2.1 dargestellt wird, beginnt die Informationsübertragung eines Projektor-Kamera-Systems noch vor der Projektion mit der Übertragung des Videosignals zwischen der Graphikkarte und dem Projektor. Es hat sich herausgestellt, dass es besonders wichtig ist, die digitale DVI-(Digital Visual Interface)-Schnittstelle [DDWG '99] anstelle der analogen 15-Pin-VGA-Schnittstelle zu verwenden. Das Problem bei der analogen Signalübertragung ist, dass eine D/A-Umwandlung vor und eine A/D-Umwandlung nach der Übertragung stattfindet. Die Digitalisierung eines analogen Signals erfolgt typischerweise mittels einer diskreten Abtastung und eines Quantisierungsprozesses. Dadurch entstehen, zusätzlich zu den Interferenzen bei der Analogübertragung, Aliasing-Effekte⁴ und Quantisierungsfehler. Eine DVI-Übertragung umgeht die unnötige Umwandlung in analoge Signale und verwendet das standardisierte TMDS-Protokoll für eine höhere Robustheit gegen elektromagnetische Interferenzen. Außerdem erhöht die Übertragung eines Taktsignals in einem separaten Kanal die Toleranz gegenüber Bitverschiebungen. Die Aufnahmen in Abb. 2.16 zeigen den qualitativen Unterschied zwischen einem VGA- und einem DVI-Videosignal bei der Projektion desselben

⁴Unerwünschte Aliasing-Effekte tauchen bei Abtastungen auf, wenn ein hochfrequentes Signal mit einem tieffrequenten Signal verwechselt wird. Diese werden üblicherweise durch Faltung mit einem Tiefpassfilter oder mit einer höheren Abtastfrequenz im Sinne des Nyquist-Kriteriums reduziert. Antialiasing ist ein ungelöstes Problem [Blinn '98].

Streifenmusters auf demselben Projektor. Es ist eindeutig zu erkennen, dass eine digitale Signalübertragung ein saubereres Projektionsmuster ergibt.

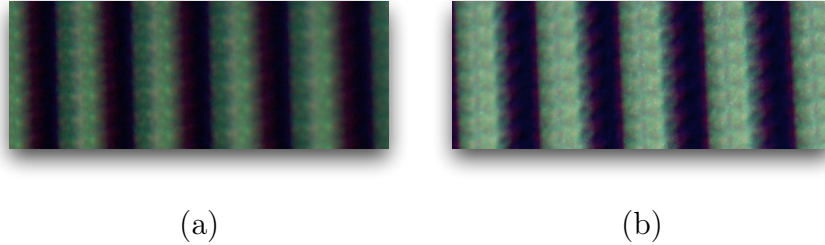


Abbildung 2.16: Aufnahmen einer Streifenprojektion (zwei Pixel breite Schwarz-Weiß-Streifen) mit einem DLP-Projektor: In (a) mit einem analogen VGA-Signal und in (b) mit einem digitalen DVI-Videosignal.

Signalanalyse der inversen Projektorabtastung

Der Projektor hat eine maximale Bildauflösung von 1024×768 Pixel und kann maximal 1024 vertikale Streifen projizieren bzw. mit 1023 Streifenübergängen abtasten. Da wir für die optische Triangulierung nur die projizierten Streifen erfassen wollen, können wir die Projektion eines Streifenbildes als eine Abtastung des Objekts betrachten. Der Projektor ist also eine inverse Kamera. Folglich liegt es nahe, die Einschränkungen dieser Diskretisierung zu untersuchen.

Bei dieser inversen Abtastung werden keine Bildintensitäten erfasst, sondern virtuelle Tiefenwerte zur Objektoberfläche betrachtet, die alleine mit dem Projektor nicht messbar sind. Im Gegensatz zur traditionellen Bildabtastung, diskretisieren wir hier anstelle des erfassten Bildes die Objektoberfläche. Dazu betrachten wir die Abbildung eines Objekts auf die Projektorbildebene als ein zweidimensionales kontinuierliches *Tiefenbildsignal* (engl. *range map*) $f_c : \mathbb{R}^2 \rightarrow \mathbb{R}$ mit $f_c(\mathbf{x}) \in \mathbb{R}$ den Abstand zwischen dem abgetasteten Punkt auf der Objektoberfläche und dem Schnittpunkt \mathbf{x} des Sichtstrahls mit der Projektorbildebene. Ohne Einschränkung der Allgemeinheit setzen wir einfachheitshalber $f_c(\mathbf{x}) = 0$ wenn die Objektoberfläche in \mathbf{x} nicht sichtbar ist bzw. kein projiziertes Licht reflektiert wird.

Eine zweidimensionale Bildabtastung $f_s : \mathbb{R}^2 \rightarrow \mathbb{R}$ mit Abtastschrittweite in x -Richtung l_x und in y -Richtung l_y ist gegeben durch

$$f_s(\mathbf{x}) = f_c(\mathbf{x}) \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - ml_x, y - nl_y) \quad ,$$

wobei $\delta(\mathbf{x})$ eine zweidimensionale Impulsfunktion ist. Bei der Projektion von vertikalen Streifen mit Pixelbreite l_x erhalten wir eine infinitesimale vertikale Schrittweite $l_y = dl_y$:

$$\begin{aligned}
f_s(\mathbf{x}) &= f_c(\mathbf{x}) \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - ml_x, y - ndl_y) \\
&= f_c(\mathbf{x}) \sum_{m=-\infty}^{\infty} \delta(x - ml_x) \sum_{n=-\infty}^{\infty} \delta(y - ndl_y) \\
&= f_c(\mathbf{x}) \sum_{m=-\infty}^{\infty} \delta(x - ml_x) \int_{-\infty}^{\infty} \delta(y) dy \\
&= f_c(\mathbf{x}) \sum_{m=-\infty}^{\infty} \delta(x - ml_x) \\
&= f_c(\mathbf{x}) \cdot i(\mathbf{x}) \quad .
\end{aligned} \tag{2.15}$$

Wie erwartet erhalten wir eine eindimensionale Streifenübergangsabtastung f_s auf der zweidimensionalen Projektorbildebene mit $i(\mathbf{x}) := i(x) = \sum_{m=-\infty}^{\infty} \delta(x - ml_x)$, die eindimensionale *Kammfunktion* der Abtastung. Die Streifenübergänge auf dem Objekt werden in der Kamerabildebene über das lokale Maximum der Ableitung nach x detektiert (siehe Abschnitt 2.1). Insbesondere kann die exakte Position eines lokalen Maximums als eine Lokalisierung eines Impulses betrachtet werden.

Es liegt nun nahe, eine Spektralanalyse des abgetasteten Signals f_s zu untersuchen. Dazu verwenden wir den *Hilbertraum* $L^2(\mathbb{R}^2) = \{f : \mathbb{R}^2 \rightarrow \mathbb{R}_{\geq 0} \mid \iint_{\mathbb{R}^2} |f(\mathbf{x})|^2 d\mathbf{x} < \infty\}$, der aus der Menge der zweidimensionalen Signale mit endlicher Energie besteht, damit deren *Fourier-Integraltransformationen* konvergieren. Die zweidimensionale Fourier-Transformation eines Signals f wird wie folgt definiert:

$$\begin{aligned}
\mathcal{F} : L^2(\mathbb{R}^2) &\rightarrow L^2(\mathbb{R}^2) \\
f(\mathbf{x}) &\mapsto \mathcal{F}\{f(\mathbf{x})\} = \hat{f}(\boldsymbol{\nu}) = \iint_{\mathbb{R}^2} f(\mathbf{x}) e^{-2i\pi(\boldsymbol{\nu}^t \mathbf{x})} d\mathbf{x} \quad ,
\end{aligned}$$

wobei $\boldsymbol{\nu} = [\nu_1, \nu_2]^t$ der entsprechende Frequenzvektor ist.

Sei nun $\hat{i}(\boldsymbol{\nu})$ die Fourier-Transformation der Kammfunktion $i(\mathbf{x})$. Es gilt:

$$\begin{aligned}
\hat{i}(\boldsymbol{\nu}) &= \iint_{\mathbb{R}^2} i(\mathbf{x}) e^{-2i\pi(\boldsymbol{\nu}^t \mathbf{x})} d\mathbf{x} \\
&= \iint_{\mathbb{R}^2} i(x) e^{-2i\pi\nu_1 x} e^{-2i\pi\nu_2 y} dx dy \\
&= \int_{-\infty}^{\infty} i(x) e^{-2i\pi\nu_1 x} dx \int_{-\infty}^{\infty} e^{-2i\pi\nu_2 y} dy \\
&= \hat{i}(\nu_1) \mathcal{F}\{1\} \quad .
\end{aligned} \tag{2.16}$$

Hier gilt $\mathcal{F}\{1\} = \delta(\nu)$. Intuitiv können wir leicht erkennen, dass sich die eindimensionale konstante Funktion 1 nicht über den Raum ändert und deren Fourier-Transformierte folglich keine Sinus- oder Kosinuskomponente bei Frequenzen ungleich Null enthalten kann. Eine Spektralzerlegung dieser Funktion kann demzufolge nur eine Ursprungsimpulsfunktion erzeugen. Weiterhin gilt mit Hilfe des Verschiebungssatzes für Fourier-Transformationen

$$\hat{i}(\nu) = \frac{1}{l_x} \sum_{m=-\infty}^{\infty} \delta\left(\nu - m \frac{1}{l_x}\right) = \nu_x \sum_{m=-\infty}^{\infty} \delta(\nu - m\nu_x) \quad ,$$

wobei $\nu_x := \frac{1}{l_x}$ die *Abtastfrequenz* in x -Richtung ist. Durch Einsetzen folgt aus Gleichung 2.16 für die Fourier-Transformierte der Kammfunktion:

$$\hat{i}(\boldsymbol{\nu}) = \delta(\nu_2) \nu_x \sum_{m=-\infty}^{\infty} \delta(\nu_1 - m\nu_x) \quad .$$

Mit Hilfe der Faltungsregel für Fourier-Transformationen erhalten wir aus Gleichung 2.15 die Fourier-Transformation des abgetasteten Tiefenbildsignals:

$$\begin{aligned} \hat{f}_s(\boldsymbol{\nu}) &= \hat{f}_c(\boldsymbol{\nu}) * \hat{i}(\boldsymbol{\nu}) \\ &= \hat{f}_c(\boldsymbol{\nu}) * \left(\hat{i}(\nu_1) \cdot \delta(\nu_2) \right) \\ &= \iint_{\mathbb{R}^2} \hat{f}_c(\nu_1 - \xi_1, \nu_2 - \xi_2) \cdot \left(\hat{i}(\xi_1) \cdot \delta(\xi_2) \right) d\xi_1 d\xi_2 \\ &= \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} \hat{f}_c(\nu_1 - \xi_1, \nu_2 - \xi_2) \cdot \delta(\xi_2) d\xi_2 \right) \cdot \hat{i}(\xi_1) d\xi_1 \\ &= \int_{-\infty}^{\infty} \hat{f}_c(\nu_1 - \xi_1, \nu_2) \cdot \hat{i}(\xi_1) d\xi_1 \\ &= \int_{-\infty}^{\infty} \hat{f}_c(\nu_1 - \xi_1, \nu_2) \cdot \nu_x \sum_{m=-\infty}^{\infty} \delta(\xi_1 - m\nu_x) d\xi_1 \quad . \end{aligned} \quad (2.17)$$

Um die Auswirkungen dieser Streifenübergangsabtastung zu analysieren, betrachten wir zunächst die Normfunktion $|\hat{f}_c(\nu_1)|$. Sie beschreibt die Amplitude einer bestimmten Frequenz der Fourierzerlegung und ist durch die Wurzel der *Spektraldichte* $\Phi_c(\nu_1) = \hat{f}_c(\nu_1) \hat{f}_c^*(\nu_1)$ definiert (mit der komplex Konjugierten $\hat{f}_c^*(\nu_1)$ von $\hat{f}_c(\nu_1)$). Wir nehmen an, dass $|\hat{f}_c(\nu_1)|$ bis zu einer maximalen Frequenz $\nu_x^{(c)}$ ungleich Null ist und somit einen endlichen Träger besitzt.

Für das abgetastete Bildsignal, erhalten wir entsprechend

$$|\hat{f}_s(\nu_1)| = \left(\hat{f}_s(\nu_1) \hat{f}_s^*(\nu_1) \right)^{\frac{1}{2}} \quad ,$$

wobei wir aus Gleichung 2.17 sehen, dass die Normfunktion nur von der Abtastfrequenz ν_x abhängig ist. Abbildung 2.17 zeigt, dass wenn die maximale horizontale

Bildsignalfrequenz $\nu_x^{(c)}$ größer als $\frac{\nu_x}{2}$ ist, die Normfunktion $|\hat{f}_s(\nu_1)|$ aufgrund von Überlappungen der Amplituden von \hat{f}_c gestört wird. Diese Störung ist auch unter der Bezeichnung *Aliasing-Effekt* bekannt und eine fehlerfreie Rekonstruktion (bzw. Interpolation) von $f_c(\mathbf{x})$ aus $\hat{f}_s(\nu_1)$ über die inverse Fouriertransformation ist unmöglich. Insbesondere wirkt sich der Aliasing-Effekt bei der inversen Abtastung auf die Geometrie des Objekts, und nicht wie bei der Bilderfassung auf das Bild, aus. Die Abtastung wird also vom *Shannonschen Abtasttheorem* mit *Nyquistfrequenz* $\nu_x^{(n)}$ begrenzt. Für eine vollständige Abtastung des Tiefenbildsignals muss somit das folgende Kriterium gelten:

$$\nu_x^{(c)} < \frac{\nu_x}{2} = \nu_x^{(n)} \quad .$$

Der vertikalen Maximalfrequenz $\nu_x^{(c)}$ sind dagegen keine Grenzen gesetzt, da die vertikale Abtastfrequenz $\nu_y = \frac{1}{l_y}$ unendlich hoch ist. Bei hochdetaillierten Objekten können die horizontalen Aliasing-Effekte durch die Projektorstreifenabtastung nicht vollständig behoben werden. Die Reduktion dieser Störungen kann nur durch eine kleinere Abtastschrittweite l_x erreicht werden, wobei diese nicht kleiner als eine Pixelbreite des projizierten Bildes sein kann. Wie in Abbildung 2.18 veranschaulicht wird, ist eine Erfassung der reflektierten Streifen durch die Streifenübergangsabtastung beschränkt. In der Regel kann nur eine Untermenge der abgetasteten Punkten erfasst werden, da ein Teil davon, aufgrund von Verdeckungen aus der Kameraperspektive, nicht sichtbar ist.

2.4 Beschränkungen des Sensors

Die Erfassung der reflektierten Streifenmuster mit einer professionellen Fuji Finepix S2 Pro Spiegelreflex-Digitalkamera ist gemäß Abb. 2.1 der letzte Schritt des Triangulierungssystems, der physikalische Unzulänglichkeiten hervorruft. Bei der optischen Triangulierung mit hochfrequenten farbigen Streifenmustern ist eine hochqualitative Erfassung mit niedrigem Rauschen unabdingbar. Jede Störung im Bild könnte zu einer potentiellen Fehlinterpretation führen, die entsprechend zu Ungenauigkeiten, Rauschen und Ausreißer in der Rekonstruktion führen. Eine korrekte Konfiguration und das Verständnis des Sensors steht in diesem Abschnitt im Vordergrund. Wir konzentrieren uns zunächst auf die Methoden der CCD-Bilderfassung und auf die Einschränkungen des Sensorchips. Daraufhin zeigen wir die Vorteile des eingesetzten Super-CCD-Chips und dessen Geometrie. Im Anschluss wird die Problematik der Farberfassung im Zusammenhang mit der Interpolation von Bayer-Muster-Abtastungen behandelt. Zum Schluss präsentieren wir, ähnlich wie im vorherigen Abschnitt, eine Signalanalyse der Erfassung von bereits durch Streifenprojektion abgetasteten Oberflächen und deren Auswirkung auf die Rekonstruktion mit optischer Triangulierung.

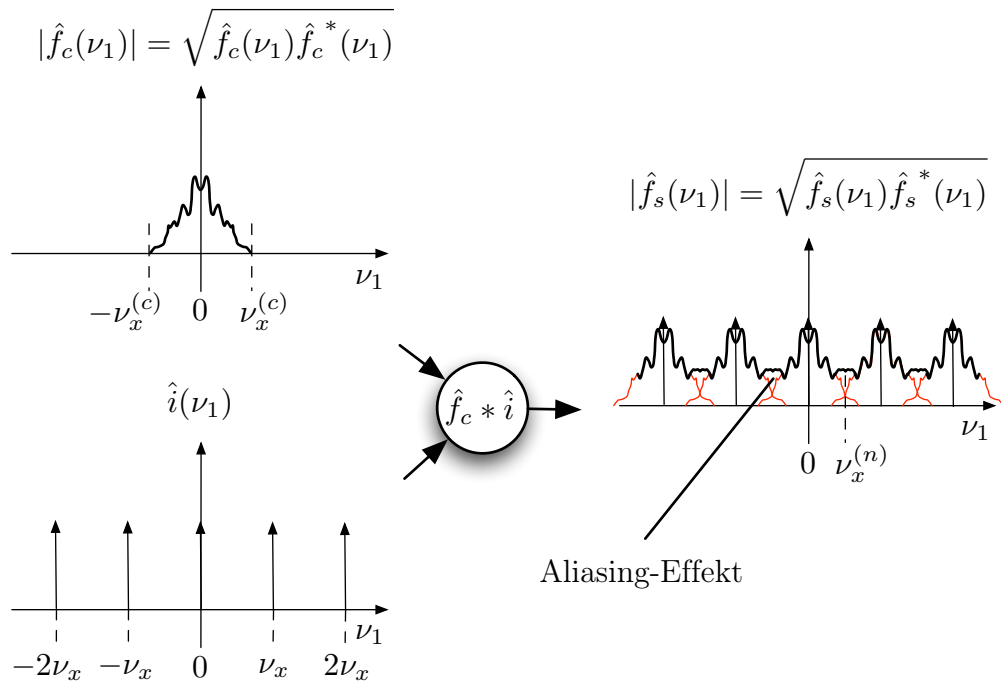


Abbildung 2.17: Die Abbildung des Objekts auf die Bildebene hat eine maximale horizontale Tiefenbildsignalfrequenz von $\nu_x^{(c)}$, wie wir es im Frequenzspektrum $|\hat{f}_c(\nu_1)|$ sehen können. Die Fouriertransformation der Kammfunktion $\hat{i}(\nu)$ ist, wie man sieht, wieder eine Kammfunktion mit einer Abtastfrequenz ν_x . Bei der Abtastung von \hat{f}_c durch Faltung mit \hat{i} erhalten wir den Graph von $|\hat{f}_s(\nu_1)|$. Wir erkennen die Aliasing-Effekte, die aus den Überlappungen der Amplituden von \hat{f}_c entstehen (in rot eingezeichnet).

Methoden der CCD-Bilderfassung

Eine Reihe von unterschiedlichen Sensoren wurden bereits für optische Triangulierungssysteme eingesetzt. Während für punktweise Abtastungen meist einzeilige (oder eindimensionale) CCDs ausreichen, benötigen wir für Lichtstreifenabtastungen (bei uns sogar mehrere gleichzeitig) zweidimensionale, in einer Matrix angeordnete CCD-Sensoren.

Bei der digitalen Bilderfassung mit zweidimensionalen CCD-Arrays werden Photonen, die durch die Optik auf den CCD gelenkt werden und genügend Energie besitzen, auf Fotodioden absorbiert. Hierbei ist zu bemerken, dass bei Fotodioden im Gegensatz zu anderen Fotodetektoren der *innere lichtelektrische Effekt* ausgenutzt wird, d. h. Photonen bewirken in Halbleitern die Erzeugung freier Ladungsträgerpaare beim Auftreffen auf einem pn-Übergang. Die Ladungspaare erzeugen einen Stromfluss und wir erhalten insgesamt ein elektrisches Signal. Es besteht ein linearer Zusammenhang zwischen der produzierten Menge an Ladungen und

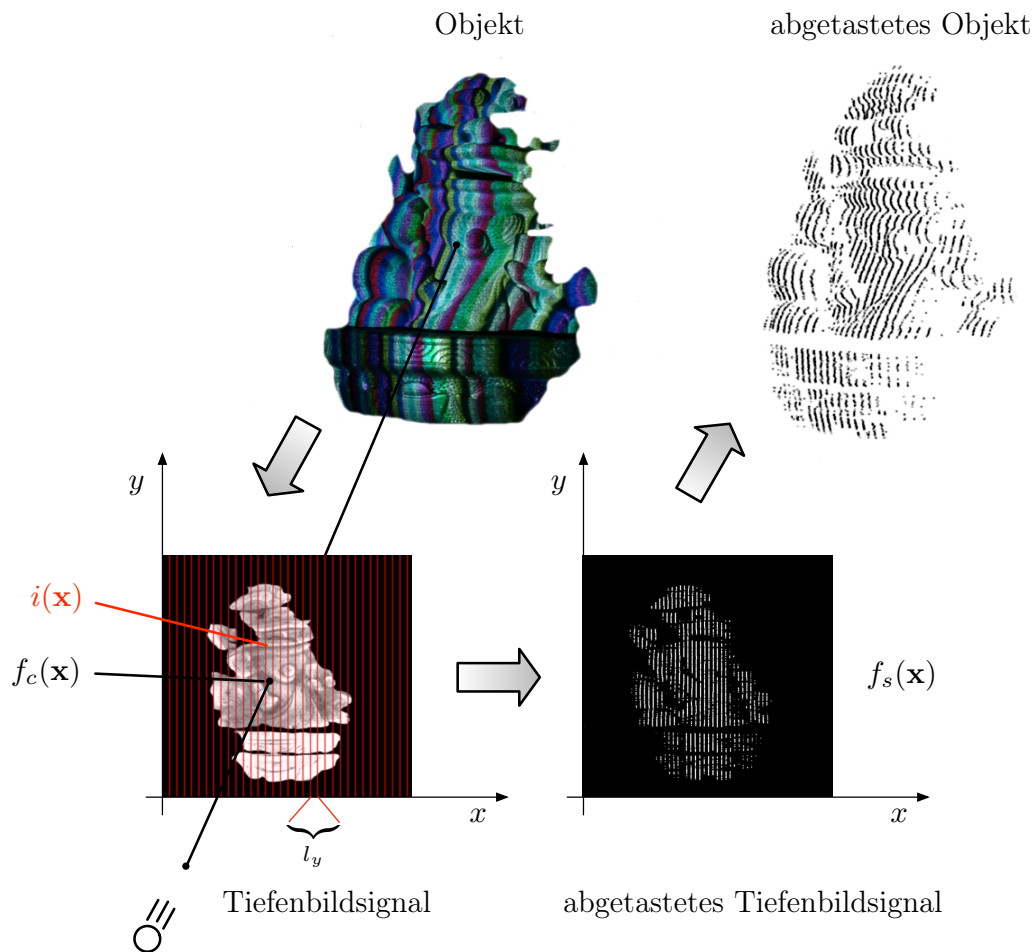


Abbildung 2.18: Die Beschränkung der Abtastung des Projektors wird als inverse Abtastung gesehen. Das Objekt wird auf die Projektorbildebene abgebildet und es werden die vom Projektor alleine nicht messbaren Tiefenwerte betrachtet. Daraufhin findet eine Streifenübergangsabtastung mit $i(\mathbf{x})$ auf das kontinuierliche Tiefenbildsignal $f_c(\mathbf{x})$ statt. Da die horizontale Abtastschrittweite l_x durch die kleinste Pixelbreite begrenzt ist, erhalten wir eine hoch anisotrope Abtastung.

der Bestrahlungsstärke (engl. irradiance) des einfallenden Lichts. Im Anschluss übertragen die in Matrix angeordneten Fotodioden zeilenweise ihre Signalwerte an ein analoges Schieberegister. Dieses Register wird sequentiell ausgelesen. Anschließend findet eine Signalverstärkung und eine A/D-Konvertierung statt, um quantisierte Lichtintensitäten zu erhalten, die letztendlich ein diskretes Pixelbild erzeugen. Bei unserer Kamera werden pro Farbkanal die analogen Signale mit 12-bit-Intensitäten quantisiert und anschließend auf 8 bit interpoliert, um 24-bit-Farbbilder auszugeben. Wichtig bei dieser Arbeit ist die Ausgabe von unkomprimierten Bilddaten, da verlustbehaftete Kompressionsverfahren, wie etwa JPEG,

zu hochfrequenten Farbartefakten führen. Weiterhin verwenden wir bei unseren Aufnahmen eine Lichtempfindlichkeit (bzw. Signalverstärkung) von ISO 200.



Abbildung 2.19: Von links nach rechts: Eine Vergrößerung einer unkomprimierten Aufnahme im Dunklen mit Blendenzahl F 22, Lichtempfindlichkeit ISO 200 und einer Belichtungszeit von 3 s. Hier wird impulsartiges Rauschen beobachtet, das durch eine erhöhte Belichtungszeit entsteht.

CCD-Aufnahmen werden durch eine Reihe von Störungen beeinflusst, die ohne Nachverarbeitung nur schwierig zu vermeiden sind. Neben der möglichen Existenz von defekten Pixelzellen, die bei einer nichtperfekten Herstellung vorkommen, werden die Hauptstörquellen durch die Aufnahme selbst und durch die Anregung von zusätzlichen Elektronen auf dem CCD-Chip aufgrund einer erhöhten Betriebstemperatur (thermisches Rauschen) hervorgerufen. Insbesondere können beide Prozesse durch eine Poisson-Verteilung modelliert werden, indem man die Verteilung der erfassten Bildwerte über mehrere Aufnahmen derselben Szene betrachtet. In [Goesele '04] wurde experimentell gezeigt, dass eine Verdopplung des Rauschens in Digitalkameras bei jeder Temperaturerhöhung um 6 K zu beobachten ist. Bei längerem Betrieb ist deshalb eine entsprechende Abkühlung zu empfehlen. Insbesondere führt eine längere Belichtungszeit, wie es bei uns für die Erfassung mit hoher Blendenzahl notwendig ist, auch zu einer längeren Erhitzung des CCD-Chips. Abbildung 2.19 zeigt dieses impulsartige Rauschen, welches auch *hot pixel* genannt wird.

In letzter Zeit werden neben CCDs auch CMOS-Sensoren für Computer-Vision-Anwendungen eingesetzt. CMOS-Bildsensoren sind in der Herstellung kostengünstiger als CCDs und das Auslesen der Bildsignale geschieht für jede Fotodiode einzeln (RAM-Zugriff). Außerdem haben sie die Eigenschaft, einen höheren Dynamikumfang (engl. *dynamic range*) zu haben, als konventionelle CCDs. Der Dynamikumfang ist das Verhältnis der maximalen zur minimalen Intensität ungleich Null. Auch der Stromverbrauch ist bei CMOS-Sensoren niedriger und ein flexiblerer Einsatz als bei CCDs ist somit möglich. Der große Nachteil bei der CMOS-Technologie allerdings ist das niedrigere Signal-zu-Rauschverhältnis und die geringe Lichtempfindlichkeit gegenüber den CCD-Sensoren. Für unsere Erfassungen haben diese zwei Eigenschaften eine höhere Priorität als die anderen, da wir in einem abgedunkelten Raum arbeiten und auf korrekte Farbwerte angewiesen sind.

Geometrische Aspekte des Super-CCD-Chip-Entwurfs

Aus einer geometrischen Perspektive betrachtet, bietet die bei uns eingesetzten, um 45° rotierten achteckige 4-Phasen-Senseler-(sensor elements)-Architektur, auch Super CCD genannt [Yamada et al. '00] interessante Möglichkeiten. Die verwendete Fuji Finepix S2 Pro ist mit der Super-CCD-Technologie der 3. Generation ausgestattet. Die Entwicklung von höher auflösenden Bildsensoren setzt typischerweise eine Reduzierung der Pixelgröße voraus. Beispielsweise werden $5,6 \mu\text{m}^2$ große Pixel von 1 MPixel CCDs auf $3,2 \mu\text{m}^2$ reduziert, um eine Bildauflösung von 3 MPixel bei gleicher Chipgröße zu erhalten. Leider führt eine Reduzierung der Pixelgröße auch zu einer ungefähr proportionalen Abschwächung der Lichtempfindlichkeit und des Dynamikumfangs. Ein Weg, diese Nachteile zu überwinden, ist den Füll-Faktor der aktiven Bereiche (Fotodioden) zu vergrößern. Wie bereits erwähnt, werden die erfassten Signale zeilenweise vertikal in Schieberegister transferiert. Diese Übertragung soll synchron über die Poly-Si-Elektroden verlaufen. In Abb. 2.20a sehen wir, dass bei konventionellen ITCCDs (interline transfer CCD) die Fotodiode eine Isolierung darstellt und wir eine Extraverdrahtung als Taktgeber benötigen. Die Übertragung des Signals geschieht üblicherweise in 3-Phasen (Φ_1 bis Φ_3). Diese Extraverdrahtung stellt einen inaktiven Bereich auf dem CCD dar und reduziert somit die Lichtempfindlichkeit. Für eine verdrahtungslose Realisierung müssen alle Elektroden, die sich in derselben Phase befinden, in Kontakt bleiben. Mit einer 4-Phasen-Übertragung (Φ_1 bis Φ_4) und einer 45° Rotation auf einem PIACCD (pixel interleaved array CCD) sind alle Elektroden, die sich in derselben Übertragungsphase befinden, verbunden (siehe Kontaktpunkt in Abb. 2.20b). Das Signal wird zickzackartig in das Schieberegister übertragen. Somit kann der Füll-Faktor einzelner Pixel erhöht werden und entsprechend die Lichtempfindlichkeit. Einen weiteren innovativen Punkt stellen die achteckigen Fotodioden dar. Dadurch können bei den Super CCDs regelmäßige Abstände zwischen den diagonalen Pixelzeilen l_d und den vertikalen Pixelzeilen l_y erreicht werden, als bei den konventionellen, quadratischen Fotodioden. Es ergibt sich eine bessere Verteilung des erfassten Lichts.

Farberfassung und Interpolation von Bayer-Muster-Abtastungen

Wie auch bei unserer Digitalkamera verwenden die meisten Kameras CCDs mit Fotodioden, die jeweils nur die Intensität einer einzigen Farbe (Rot, Grün oder Blau) aufnehmen können. Dazu werden zwischen der Fotodiode und der zugehörigen Mikrolinse Farbfilter eingebaut. Die Anordnung dieser Farbfilter entspricht dem wohl etablierten Bayer-Muster [Bayer '76]. Aufgrund der Matrix-Anordnung der Pixelbilder können die drei Primärfarben nicht gleichmäßig verteilt werden. Dazu werden in Analogie zum Luma-Chroma-Prinzip⁵ für jedes rote und blaue Pixel jeweils zwei grüne Pixel verwendet. Es entsteht dadurch eine höhere Abtastfrequenz

⁵Die Zerlegung der Farben in Luma-(Helligkeit)- und Chroma-(Farb)-Werten, wie bei der YUV-Bilddarstellung, und die genauere Verarbeitung der Luma- gegenüber den Chroma-Werten,

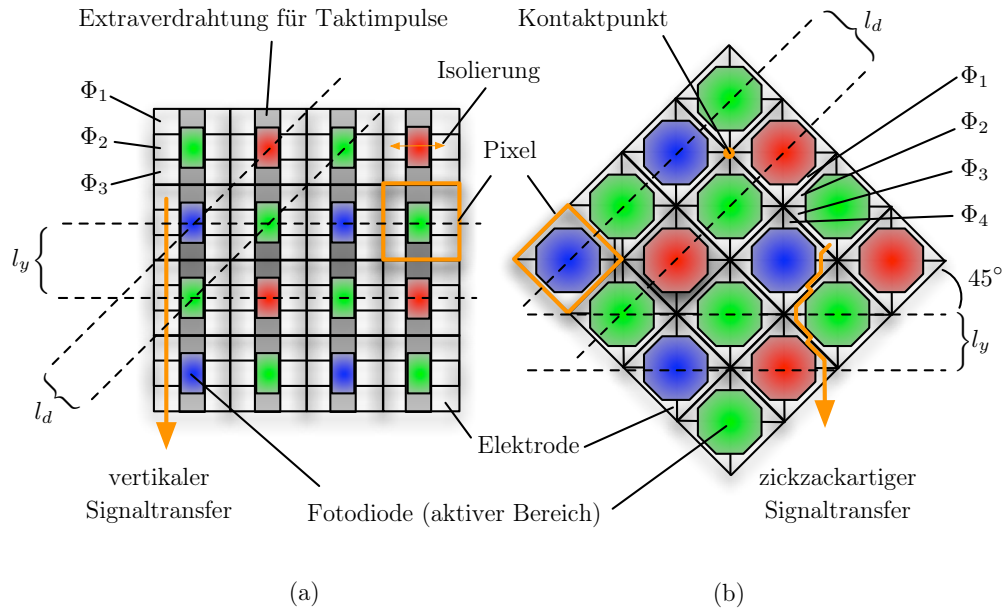


Abbildung 2.20: (a) Ein konventioneller 3-Phasen-ITCCD. (b) Der eingesetzte 4-Phasen-PIACCD. Durch Rotation um 45° sind alle Elektroden derselben Phase verbunden und eine Extraverdrahtung als Taktgeber ist überflüssig. Dadurch erhöht sich der Füll-Faktor und somit auch die Lichtempfindlichkeit und der Dynamikumfang. Die Verwendung von achteckigen Fotodioden ermöglicht außerdem eine regelmäßigere Abtastung als rechteckigen Dioden. Die Farben der Fotodioden sind, für die Erfassung von Farbbildern, im Bayer-Muster [Bayer '76] angeordnet.

für den grünen Kanal, als für die beiden anderen Kanäle und es ergibt sich eine störende räumliche Phasendifferenz der Abtastung zwischen den Farben. Daraus folgen automatisch Aliasing-Effekte. Mittels Interpolation für jede Farbe werden üblicherweise die Teilabtastungen (engl. sub sampling) vervollständigt. Sei beispielsweise $i(x, y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x - 2ml_x, y - 2nl_y)$ die regelmäßige Abtastfunktion der roten Pixel eines kontinuierlichen Bildsignals $f_c(x, y) : \mathbb{R}^2 \rightarrow [0, 1]$ mit l_x und l_y die Pixelabstände in x - und y -Richtung. Die Abtastung von $f_c(\mathbf{x})$ ist gegeben durch $f_s(\mathbf{x}) = f_c(\mathbf{x}) \cdot i(\mathbf{x})$. Eine fehlerfreie Interpolation kann nur rekonstruiert werden, wenn das Nyquist-Kriterium erfüllt ist. Obwohl es in der Praxis selten der Fall ist, können wir eine „ideale“ Interpolation als eine Rekonstruktion aus einer Abtastung $f_s(\mathbf{x})$ definieren, indem wir im Frequenzbereich $\hat{f}_s(\boldsymbol{\nu}) = \mathcal{F}\{f_s(\mathbf{x})\}$ alle Frequenzen, die größer als die Nyquistfrequenz $\nu^{(n)}$ sind, herausfiltern (Tiefpassfil-

entspricht der höheren Empfindlichkeit des menschlichen Auges bezüglich der Lichtintensität gegenüber der Farbwahrnehmung.

ter). Wir erhalten dementsprechend die interpolierte Funktion im Frequenzbereich $\hat{f}_i(\boldsymbol{\nu}) = \hat{f}_s(\boldsymbol{\nu}) \cdot r(\boldsymbol{\nu})$, wobei $r(\boldsymbol{\nu})$ die zweidimensionale *Kastenfunktion*

$$r(\boldsymbol{\nu}) = \begin{cases} 1 & , \text{ für } |\nu_1| < \nu^{(n)} \text{ und } |\nu_2| < \nu^{(n)} \\ \frac{1}{2} & , \text{ für } |\nu_j| = \nu^{(n)} \text{ und } |\nu_k| \leq \nu^{(n)} , j, k \in \{1, 2\} , j \neq k \\ 0 & , \text{ sonst} \end{cases}$$

ist. Die inverse Fouriertransformation der Kastenfunktion ist $\mathcal{F}^{-1}\{r(\boldsymbol{\nu})\} = \text{sinc}(x)\text{sinc}(y) = \frac{\sin(\pi x)\sin(\pi y)}{\pi^2 xy}$, wodurch die ideale Interpolation entsteht:

$$f_i(\mathbf{x}) = f_s(\mathbf{x}) * (\text{sinc}(x)\text{sinc}(y)) \quad .$$

Da si-Funktionen keinen endlichen Träger besitzen, werden ideale Interpolationen in der Praxis nicht implementiert.

Eine naive Approximierung für die Interpolation kann daher beispielsweise durch eine diskrete bilineare Interpolation gegeben werden. Dazu betrachten wir ein diskretes Bildsignal $f_d(x, y) : \mathbb{N}^2 \rightarrow [0, 1]$ und die diskrete Abtastungsfunktion $i_d(x, y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta_d(x-lm, y-ln)$ mit $\delta_d(\mathbf{x}) = 1$ für $\mathbf{x} = \mathbf{o}$ und $\delta_d(\mathbf{x}) = \mathbf{o}$ für $\mathbf{x} \neq \mathbf{o}$. $l \in \mathbb{N}$ ist die Schrittweite der Abtastfunktion gemessen in Anzahl der Pixel. Gemäß dem Bayer-Muster erhalten wir $l = 2$ für die Abtastung der roten und blauen Pixel. Folglich ist die diskrete Abtastung $f_s(\mathbf{x}) = f_d(\mathbf{x}) \cdot i_d(\mathbf{x})$ und die entsprechende bilineare Interpolation ist:

$$f_i(\mathbf{x}) = \frac{1}{l^2} \begin{bmatrix} f_s(\arg \min_{\mathbf{x}'} \{d(\mathbf{x}', \mathbf{x}) \mid f_s(\mathbf{x}') \neq 0, x' \leq x, y' \geq y\}) \\ f_s(\arg \min_{\mathbf{x}'} \{d(\mathbf{x}', \mathbf{x}) \mid f_s(\mathbf{x}') \neq 0, x' \geq x, y' \geq y\}) \\ f_s(\arg \min_{\mathbf{x}'} \{d(\mathbf{x}', \mathbf{x}) \mid f_s(\mathbf{x}') \neq 0, x' \leq x, y' \leq y\}) \\ f_s(\arg \min_{\mathbf{x}'} \{d(\mathbf{x}', \mathbf{x}) \mid f_s(\mathbf{x}') \neq 0, x' \geq x, y' \leq y\}) \end{bmatrix}^t \begin{bmatrix} (l-x)y \\ xy \\ (l-x)(l-y) \\ x(l-y) \end{bmatrix}$$

mit dem *City-Block-Abstand* $d(\mathbf{x}', \mathbf{x}) = |x' - x| + |y' - y|$. Die bilineare Interpolation der grünen Pixel besteht einfach aus Mittelungen von vier benachbarten Abtastungen. Obwohl die bilineare Interpolation ein effizientes und beliebtes Verfahren ist, führt sie zu einer unscharfen Rekonstruktion, da sie die Bandbreite des Signals stark begrenzt. Bei hochwertigen Digitalkameras werden meistens komplexere und proprietäre Algorithmen zur Interpolation verwendet. Für eine ausführliche Studie zu Bayer-Muster- Interpolationen verweisen wir auf [Ramanath et al. '02].

Im Hinblick auf eine Rekonstruktion ohne Interpolation, werden in [Goesele '04] 4 Aufnahmen mit jeweils versetzten Bayer-Mustern gemacht, um hinterher die erfassten Bilder aufzusummieren (das 4. Bild dient zur Rauschreduktion). In Abb. 2.21a sehen wir deutliche Farbartefakte, die durch die Interpolation der

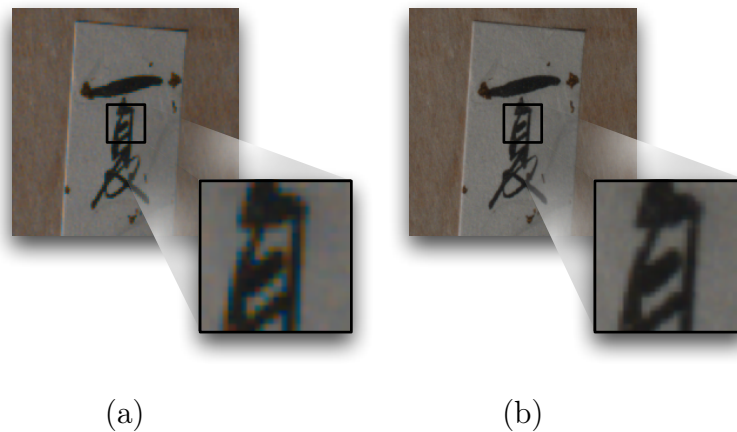


Abbildung 2.21: Ein qualitativer Vergleich zwischen einer Aufnahme mit einer Interpolation einer Bayer-Muster-Abtastung (a) und einem 4-Aufnahmen-Modus einer Jenoptik ProgRes C14 Kamera (b)[Goesele '04].

Bayer-Muster-Abtastung entstehen. Der 4-Aufnahmen-Modus einer Jenoptik ProgRes C14 Kamera erzielt hiermit eine deutlich höhere Auflösung (Abb. 2.21b). Da es bei unserem Scanverfahren auf die Schnelligkeit ankommt, beschränken wir uns auf eine Lösung mittels Interpolation. Vielversprechende Technologien, wie etwa die patentierten Foveon-X3-Sensoren [Billings '98], bieten die Möglichkeit, mehrere Farben mit einer einzelnen Pixelzelle zu erfassen.

Signalanalyse der Erfassung von streifigen Oberflächen

Bei der optischen Triangulierung kann das Projektor-Kamera-System wie ein traditionelles Stereo-System gesehen werden, wobei der Unterschied darin liegt, dass der Projektor ein Bildsignal projiziert und nicht erfasst, und somit das Objekt mit einem ideal projizierten Streifenbild abgetastet wird. Um möglichst viele Details bei der Erfassung von hochfrequenten Objektoberflächen zu erhalten, projizieren wir Streifen mit minimaler Breite, die durch die Pixelbreite nach unten begrenzt wird (siehe Abschnitt 2.3). Die Aufgabe besteht nun darin, die dicht liegenden Streifenprojektionen, die eine inverse Abtastung darstellen, auf der Objektoberfläche durch die Kamera möglichst genau wiederzuerkennen, damit eine korrekte Rekonstruktion erfolgen kann. Insbesondere kann maximal nur die durch den Projektor abgetastete streifige Oberfläche rekonstruiert werden. Die Auflösung der Digitalkamera spielt daher eine wichtige Rolle. Wir leiten in diesem Abschnitt zunächst den Begriff des Auflösungsvermögens her und zeigen den Zusammenhang der Linsenauflösung mit der CCD-Auflösung. Anschließend analysieren wir die Abtastung der Kamera in Bezug auf das Signal der Streifenübergänge der Objektoberfläche.

Unsere eingesetzte Kamera hat eine maximale effektive Bildauflösung von 3024×2016 Pixeln, die wir bei unseren Aufnahmen vollständig ausnutzen. Das maximale Auflösungsvermögen wird jedoch auch durch die Optik bestimmt. 6 MPixel Aufnahmen liegen bereits recht am maximal erreichbaren Auflösungsvermögen der Optik. Die Unterscheidung zweier sehr dicht beieinanderliegende Punkte ist also selbst bei einer analogen Film-Kamera mit derselben Optik kaum mehr möglich. Üblicherweise wird das Auflösungsvermögen eines optischen Systems durch die *Modulationsübertragungsfunktion* oder MTF-Funktion (engl. *modulation transfer function*) beschrieben. Dazu betrachten wir die Abbildung eines Objektpunkts \mathbf{x} (mit einer Intensität $\delta(\mathbf{x})$), welcher sich auf der Objektebene (die Ebene die senkrecht zur optischen Achse steht und auf der der Objektpunkt liegt) befindet, auf die Kamerabildebene. Aufgrund von Diffraction, Linsenaberration und ungenauer Fokussierung entsteht eine Intensitätsverteilung dieses Punkts auf der Kamerabildebene, die durch die zur Optik zugehörigen *Punktstreuungsfunktion* (engl. *point spread function*) $\text{PSF}(\mathbf{x})$ gegeben ist. Da die Szene i. Allg. ein nichtkohärentes Licht ausstrahlt (es wird angenommen, dass die Korrelation der elektromagnetischen Feldamplituden von zwei benachbarten Punkten gleich Null ist), summieren sich die Intensitäten von benachbarten Bildpunkten. Mit der zusätzlichen Annahme, dass eine Translation des Objektpunkts auf die Objektebene eine proportionale Translation der zugehörigen PSF auf die Kamerabildebene bewirkt (*Isoplanasie*), beschreibt das zugrundeliegende optische System insgesamt eine *lineare und verschiebungsinvariante Transferfunktion* (engl. linear shift invariant system). Sei nun $f_i(\mathbf{x})$ die Intensität des Lichts auf der Kamerabildebene und $f_o(\mathbf{x})$ die auf der Objektebene. Daraus ergibt sich folgende Beziehung:

$$f_i(\mathbf{x}) = \iint_{\mathbb{R}^2} f_o(\mathbf{x}') \text{PSF}(\mathbf{x} - \mathbf{x}') d\mathbf{x}' = \text{PSF}(\mathbf{x}) * f_o(\mathbf{x}) \quad .$$

Hier beschreibt $\text{PSF}(\mathbf{x})$ die *Impulsantwort* (engl. *impulse response*) des Systems. Entsprechend definiert die Fouriertransformation der PSF-Funktion die *optische Übertragungsfunktion* OTF (engl. *optical transfer function*). Es gilt wegen der Faltungsregel:

$$\text{OTF}(\boldsymbol{\nu}) = \mathcal{F}\{\text{PSF}(\mathbf{x})\} = \frac{\mathcal{F}\{f_i(\mathbf{x})\}}{\mathcal{F}\{f_o(\mathbf{x})\}} = \frac{\hat{f}_i(\boldsymbol{\nu})}{\hat{f}_o(\boldsymbol{\nu})} \quad .$$

Aus dieser Spektralanalyse ergibt sich demzufolge eine sinnvolle Interpretation der Informationsübertragung zwischen einem Punkt auf der Objektebene und dessen Bild auf der Kamerabildebene. Ist die OTF-Funktion gleich Null, besteht keine Übertragung und wenn sie komplexwertig ist, besteht eine Phasendifferenz. Für das Auflösungsvermögen des optischen Systems interessiert man sich allerdings ausschließlich für die Amplitude der OTF-Funktion, die gerade die MTF-Funktion definiert:

$$\text{MTF}(\boldsymbol{\nu}) = |\text{OTF}(\boldsymbol{\nu})| = \frac{|\hat{f}_i(\boldsymbol{\nu})|}{|\hat{f}_o(\boldsymbol{\nu})|} \quad ,$$

wobei $\text{MTF}(\nu)$ stets kleiner als 1 sein muss, da das optische System selbst kein Signal produziert oder verstärkt. Folglich kann die Schärfe eines optischen Systems durch die MTF-Funktion charakterisiert werden: Sie beschreibt nämlich das Vermögen, ein Signal einer bestimmten Frequenz zu übertragen. Außerdem verhält sich das optische System wie ein linearer Tiefpassfilter mit $\text{PSF}(\mathbf{x})$ als Filterfunktion. Insbesondere werden Amplituden in hochfrequenten Bereichen reduziert und wir können annehmen, dass die $\text{OTF}(\nu)$ eine hinreichend große Amplitude bis zu einer maximalen Frequenz ν_{\max} besitzt. Die Abtastfrequenz der CCD-Sensoren, die aus der Größe und Pixelauflösung des CCDs berechnet wird, sollte daher in horizontaler und vertikaler Richtung mindestens doppelt so hoch sein wie ν_{\max} . Im Falle einer idealen impulsartigen Punktstreuungsfunktion $\text{PSF}(\mathbf{x}) = \delta(\mathbf{x})$ wäre entsprechend $\text{MTF}(\nu) = 1$ und wir würden eine perfekte Übertragung mit einem perfekten Auflösungsvermögen für jede beliebige Frequenz, erhalten. Weiterhin können wir aus der chromatischen Aberration (siehe Abschnitt 2.2) folgern, dass ein grünes Signal schärfer abgebildet werden kann, als ein rotes oder blaues Signal. Dadurch findet sich sogar eine weitere Berechtigung, höhere Abtastraten bei den grünen Pixel-Sensoren im Bayer-Muster der CCDs zu verwenden. Vor allem um die Genauigkeit eines Scansystems zu bestimmen, bieten sich eine Reihe von Methoden an, um die MTF-Funktion zu messen, wie etwa der *Messerschneidentest* für digitale Erfassungen [Reichenbach et al. '91] oder die auf 3D-Scanner angepasste *Scharfe-Kanten-Technik* [Goesele '04].

Die Streifenübergänge werden in der Praxis, wie in Kap. 4 gezeigt wird, ohne weiteres mittels Kantendetektionsverfahren, nach einer Diskretisierung des erfassten Bildsignals, gewonnen. Dadurch ist eine Analyse von Streifenübergängen mit exakten kontinuierlichen Signalen nicht mehr möglich, da man auf die Diskretisierung angewiesen ist. Hier wollen wir zeigen, dass die in Abschnitt 2.3 entstandenen impulsartigen Streifenübergänge aufgrund der PSF-Funktion durch die Abbildung auf die Kamerabildebene mit übertragen werden, und somit durch die Kamera erfasst werden. Im Unterschied zu einer traditionellen Bilddiskretisierung interessieren wir uns hier für die Kameraabtastung der vom Objekt reflektierten Streifenübergänge. Hierbei betrachten wir ebenfalls die zunächst unbekanntes Tiefenwerte. Dazu übernehmen wir dieselben Bezeichnungen, die wir für die inverse Projektorabtastung in 2.3 verwendet haben (siehe Abb. 2.22).

Sei nun $g_c(\mathbf{x})$ das Tiefenbild von $f_s(\mathbf{x})$ aus der Kameraperspektive und $g_s(\mathbf{x})$ die diskrete Abtastung der Kamera. Aufgrund von Verdeckungen der originalen Oberfläche $f_c(\mathbf{x})$ stellt $g_c(\mathbf{x})$ nur eine Untermenge von $f_s(\mathbf{x})$ dar. Außerdem stellt $g_c(\mathbf{x})$ eine unregelmäßige Abtastung dar. Wir erinnern uns, dass die Lokalisierung von Streifenübergängen über die Bestimmung der lokalen Maxima der Gradienten geschieht, und dass $f_s(\mathbf{x})$ durch eine eindimensionale impulsartige Streifenabtastung gegeben ist. Aufgrund der Tiefpass-Eigenschaften des optischen Systems können wir uns bei der Kameraabtastung nicht mehr auf einen impulsartige Streifenübergang beschränken. Folglich ist die Abbildung der Streifenübergänge auf die Ka-

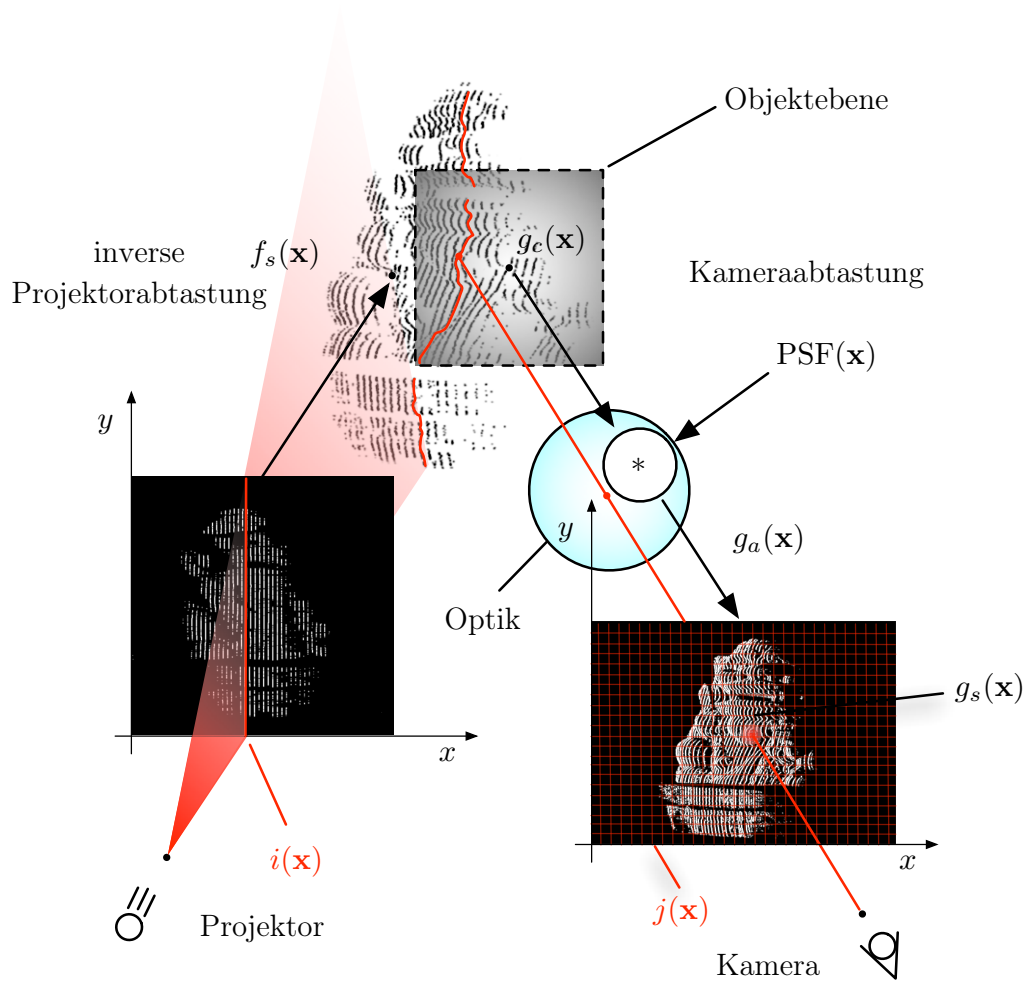


Abbildung 2.22: Signalübertragungsmodell: Auf der linken Seite findet die inverse Kameraabtastung mit Abtastfunktion $i(\mathbf{x})$ statt und es entsteht ein Tiefenbild $f_s(\mathbf{x})$. Auf der rechten Seite findet die Kameraabtastung statt. Das vorliegende Streifenübergangssignal $g_c(\mathbf{x})$ wird mit der PSF-Funktion gefaltet und wir erhalten $g_a(\mathbf{x})$. Danach wird $g_a(\mathbf{x})$ mit $j(\mathbf{x})$ abgetastet und wir erhalten die Kameraabtastung $g_s(\mathbf{x})$. Insbesondere erhalten wir am Ende eine Punktwolke.

merabildebene durch $g_a(\mathbf{x}) = \text{PSF}(\mathbf{x}) * g_c(\mathbf{x})$ gegeben. Hier stellt $g_a(\mathbf{x})$ das bestmögliche kantendetektierte Signal der Streifenübergänge dar. Intuitiv können wir bereits feststellen, dass es bei sehr dicht beieinanderliegenden Streifenübergängen aufgrund der PSF-Funktion in der Kamerabildebene zu Aliasing-Effekten kommen kann. Es liegt nun nahe, die durch $g_a(\mathbf{x})$ repräsentierte Oberfläche abzutasten. Da wir zunächst nur eine unregelmäßige Abtastung dieser Oberfläche kennen, wollen wir deren Interpolation betrachten. Bei der Kameraabtastung $g_s(\mathbf{x})$ würde gera-

de diese Interpolation abgetastet. Dazu betrachten wir die Fouriertransformation $\hat{g}_a(\boldsymbol{\nu}) = \text{OTF}(\boldsymbol{\nu})\hat{g}_c(\boldsymbol{\nu})$. In der Regel können wir bei optischen Systemen aus den MTF-Profilen eine schnell abfallende Amplitude ab einer bestimmten Frequenz feststellen. Diese Frequenz definiert die Nyquistfrequenz $\nu^{(n)}$ für eine Abtastung der zugehörigen OTF-Übertragungsfunktion. Daraus ergibt sich, dass die Bandbreite der unregelmäßigen Abtastung $\hat{g}_c(\mathbf{x})$, die in der Praxis oft höher ist als $\nu^{(n)}$, durch $\text{OTF}(\boldsymbol{\nu})$ begrenzt wird.

Wir stellen zunächst fest, dass die Bandbegrenzungseigenschaft der OTF-Funktion, ähnlich wie bei einer Multiplikation mit einer Kastenfunktion, eine Interpolation durch die inverse Fouriertransformation bewirkt. In der Tat ist diese Interpolation unnötig, da die PSF-Funktion durch die Faltung bereits eine zweidimensionale Kameraabtastung mit nulldimensionalen Impulsen und hoher Abtastfrequenz ermöglicht. Zusätzlich ist uns eine explizite Formulierung der Nyquist-Frequenz einer reinen unregelmäßigen Abtastung $g_c(\mathbf{x})$ derzeit nicht bekannt, die für eine Interpolation ohne die OTF-Funktion wichtig wäre.

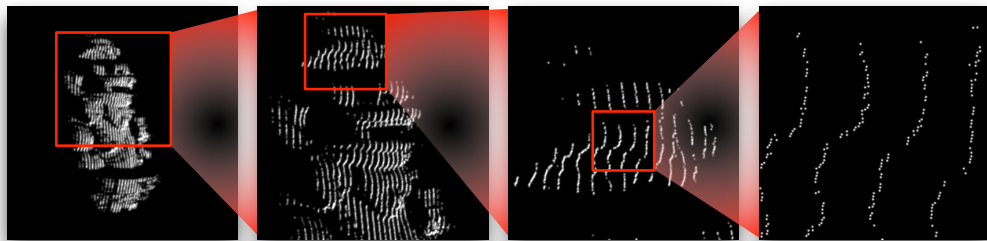


Abbildung 2.23: Von Links nach Rechts: Vergrößerungen nach einer Kameraabtastung. Hier sehen wir die Entstehung einer Punktwolke in Streifenkonstellation.

Letztendlich können wir annehmen, dass wenn die Abtastrate der Kamera hoch genug ist, nur die Aliasing-Effekte der PSF-Funktion in Erscheinung treten. Die PSF-Impulsantwort stellt somit auf dem ersten Blick eine Störquelle dar, ist aber zur gleichen Zeit auch Voraussetzung einer Kameraabtastung mit einer zweidimensionalen regelmäßigen Abtastfunktion $j(\mathbf{x}) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \delta(x-ml, y-nl)$ mit Pixelabstand l für impulsartige Streifenübergänge. Das Ergebnis ist somit das Tiefenbildsignal $g_s(\mathbf{x})$ aus der Kameraperspektive, das insbesondere eine Punktwolke in Streifenkonstellation darstellt und keine eindimensionalen Streifen. Die Streifenkonstellation der Punktwolke folgt aus der vergrößerten und höher auflösenden Aufnahme der Kameraabtastung gegenüber derjenigen der Projektorabtastung (siehe Abb. 2.23).

Kapitel 3

Erzeugung der projizierten Muster

Für die Erfassung von Oberflächen mittels optischer Triangulierung werden Streifen projiziert, um das Zuordnungsproblem zu vereinfachen. Im Gegensatz zu Stereoverfahren sind Einschränkungen aufgrund der Epipolargeometrie überflüssig. Die Rekonstruktionen sind üblicherweise robuster und genauer, da ein bekanntes Bild projiziert wird. Wie bereits erwähnt spielt die Schnelligkeit eine besonders wichtige Rolle beim Entwurf unseres Scansystems. Dazu reduzieren wir die Anzahl der Aufnahmen, indem wir mehrere Streifen gleichzeitig projizieren. Dadurch entsteht die Schwierigkeit, zwischen mehreren Streifen in der Aufnahme zu unterscheiden, um die entsprechende Triangulierung durchzuführen. Dieses Zuordnungsproblem hängt davon ab, wie das projizierte Muster identifiziert werden kann. Wie wir bereits in Kap. 1 und 2 gesehen haben, sind die Aufnahmen der reflektierten Streifen stark durch die Geometrie des Objekts und eine Reihe anderer Faktoren gestört. Nur unter Berücksichtigung bestimmter Annahmen kann eine korrekte Erfassung erfolgen. Es liegt nun nahe, ein besonders robustes Projektionsmuster zu entwerfen, das sowohl schwächere Annahmen voraussetzt, als auch gleichzeitig viel Information enthält, um die Anwendbarkeit für möglichst viele unterschiedliche Objekte zu ermöglichen und um eine möglichst hohe Auflösung zu erzielen.

Wir verwenden in diesem Kapitel die bereits eingeführten Kohärenzannahmen (Raum, Zeit und Reflexion), um die Voraussetzungen unseres Verfahrens zu beschreiben. Insbesondere wird ein Hybridverfahren vorgestellt, welches die Vorteile der uns zur Zeit bekannten Techniken ausnutzt. Anschließend geben wir eine formale Beschreibung der Problemstellung für den Entwurf des projizierten Musters, welche uns letztendlich erlaubt, einen Algorithmus für die Erzeugung des Streifenmusters zu realisieren. Das resultierende Muster ist eine im Zeitmultiplex kodierte Farbstreifensequenz, die gegenüber hochfrequenten Objektoberflächen besonders

unempfindlich ist und die eine lokale Raumkohärenz, für eine höhere Robustheit gegenüber falsch erfassten Streifenübergängen, annimmt.

3.1 Kohärenzannahmen

Die von Rusinkiewicz [Rusinkiewicz '01] eingeführte Taxonomie klassifiziert die unterschiedlichen Methoden für die optische Triangulierung mit strukturiertem Licht je nach Kohärenzannahme über den Raum, die Zeit oder die Reflexionseigenschaften der Szene. Die Anwendbarkeit des Scanners, sowie der Entwurf des Projektionsmusters und das Zuordnungsverfahren hängen lediglich von den Annahmen ab, die wir in diesem Abschnitt zusammenfassen. Ziel dabei ist es, die Annahmen so schwach wie möglich zu halten und einen Kompromiss zwischen der Qualität der Ausgabe und der Anwendbarkeit zu finden. Insbesondere handelt es sich bei unserem Verfahren um ein auf Raum- und Zeitkohärenz basierendes Verfahren (siehe Abb. 3.1), wobei auch die Reflexionseigenschaften des Objekts berücksichtigt werden müssen, da wir farbige Streifen projizieren.

Zeitliche Kohärenz

Obwohl wir auf eine schnelle Erfassung setzen, sind wir durch die lange Belichtungszeit der Kamera begrenzt, die für eine maximale Schärfentiefe im Dunkeln notwendig ist. Wir können daher, während der Aufnahmen, weder das Objekt noch das Scansystem bewegen. Die Voraussetzung einer statischen Szene während der gesamten Erfassung bezeichnen wir als globale zeitliche Kohärenzannahme. Mit jeder zusätzlichen Projektion, können wir die Anzahl der eindeutig kodierten Streifen verdoppeln und somit die Auflösung der projizierten Streifen erhöhen. Die Erfassungszeit kann i. Allg. aus der Anzahl der benötigten Aufnahmen, die mit unterschiedlichen Mustern zur Rekonstruktion einer Ansicht projiziert werden, bestimmt werden. Aus der Überlagerung der projizierten Bildsignale über die Zeit entsteht insgesamt ein im Zeitmultiplex kodiertes Streifenmuster.

Außerdem verwenden wir die zeitliche Kohärenz um eine Farbkorrektur durchzuführen, indem wir ein weißes Muster projizieren. Insgesamt benötigen wir in unserer Implementierung nur drei Aufnahmen, da die maximale Anzahl an projizierbaren Streifen durch die Projektorbildauflösung beschränkt ist. Dennoch kann die Anzahl der Projektionen beliebig erhöht werden, um eine höhere Robustheit zu erzielen.

Räumliche Kohärenz

Die räumliche Kohärenzannahme setzt eine räumliche Stetigkeit des reflektierten Musters in der Aufnahme voraus. Während bei der globalen Raumkohärenz alle projizierten Streifen in der richtigen Reihenfolge erfasst werden müssen, werden

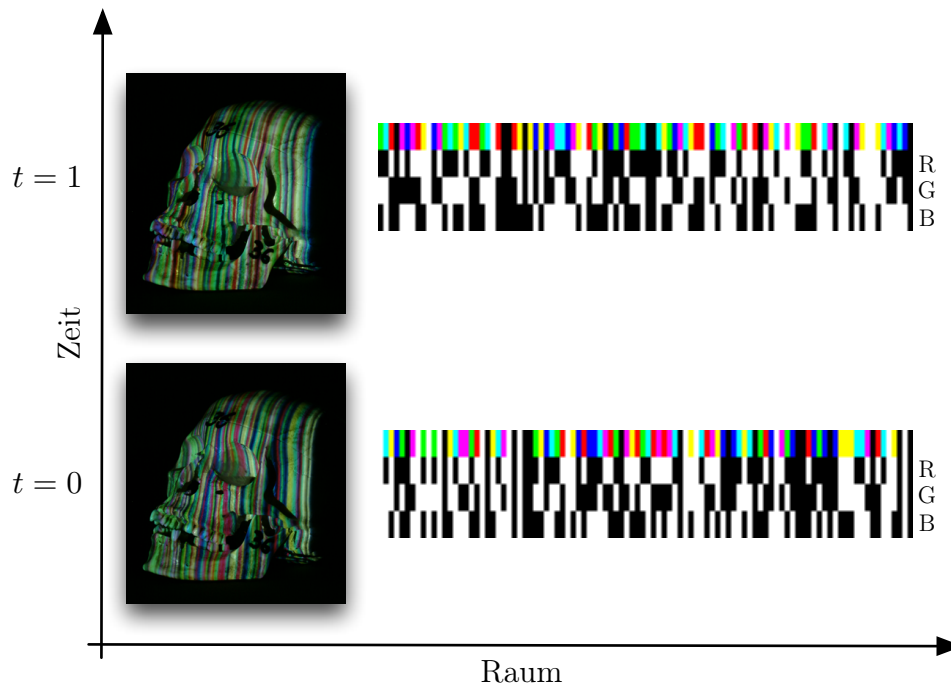


Abbildung 3.1: Das im Zeitmultiplex kodierte strukturierte Licht entsteht aus der Projektion von farbigen Streifenmustern über die Zeit. Jedes farbige Streifenbildsignal entspricht einer sequentiellen Projektion von drei Schwarz-Weiß-Streifenbildsignalen. Es genügen zwei Projektionen, für eine eindeutige Kodierung der Streifenübergänge bei der vollen Auflösung des Projektors.

bei Verfahren, die auf der lokalen Raumkohärenz basieren, die Streifen nur durch ihre lokale Nachbarschaft identifiziert. Aufgrund von Unstetigkeiten und Verdeckungen von Objekten, werden meistens nur Verfahren betrachtet, die die lokale Raumkohärenz voraussetzen.

Da wir mit der zeitlichen Kohärenz bereits die Eindeutigkeit der Streifen kodiert haben, ist die räumliche Kohärenz auf dem ersten Blick unnötig. Dennoch ist die Annahme, dass die Objektoberfläche in der Erfassung stückweise stetig ist, sinnvoll. Wir verwenden sie deshalb bei der Zuordnung und nehmen an, dass ein Streifenübergang in der Aufnahme nur dann richtig zugeordnet ist, wenn in der Aufnahme mindestens einer seiner benachbarten Streifenübergänge auch tatsächlich den benachbarten projizierten Streifenübergang darstellt. Es hat sich experimentell gezeigt, dass viele Ausreißer durch diese Vorgehensweise vermieden werden können.

Da die Streifenlokalisierung, die in Kap. 4 beschrieben wird, zeilenweise erfolgt, verwenden die uns zur Zeit bekannten, auf Streifen basierten Verfahren nur die lokale

Stetigkeit innerhalb einer einzigen Zeile, wie z. B. in [Zhang et al. '02, Li et al. '04]. Dadurch, dass wir vertikale Streifen projizieren und nicht, wie beim aktiven Stereo-Verfahren von Chen et al. [Chen et al. '97], durch die Epipolarebene eingeschränkt sind, können wir eine lokale Stetigkeit der reflektierten Streifen auch in vertikaler Projektionsrichtung annehmen. Die Annahme dieser Lokalität in vertikaler Richtung ist besonders vorteilhaft, da wir in Kap. 2 gesehen haben, dass wir eine dichtere Abtastung entlang der Streifenübergänge haben. Für eine Messung der horizontalen Raumkohärenz verwenden wir die Anzahl, der benötigten Nachbarstreifen, und für die vertikale Raumkohärenz verwenden wir die Anzahl der vertikalen benachbarten Pixel der Kameraaufnahme, die zur selben lokalen Umgebung der Oberfläche beitragen.

Einen weiteren Aspekt der räumlichen Kohärenz stellt die *Monotonizitätsannahme* dar. Wie bereits erwähnt, wird bei den Zuordnungsverfahren, die auf dem dynamischen Programmieren basieren, die Korrespondenz in mehreren Durchläufen gelöst, um auch Teilsequenzen von Streifenübergänge identifizieren zu können, die nicht in der projizierten Reihenfolge auf der Kamera aufgenommen werden. Durch die eindeutige Kodierung unserer Streifenübergänge haben wir diese Einschränkung nicht.

Reflexionsannahme

Für eine noch schnellere Erfassung projizieren wir Streifenbildsignale gleichzeitig in allen drei Farbkanälen. Bei der Projektion von farbigen Streifen ist es vor allem wichtig, dass die Farben der reflektierten Streifen auch wieder zu erkennen sind. Farbige Objekte mit hochfrequenten Texturen und schlechten Reflexionseigenschaften stellen somit eine Herausforderung dar. Die Reflexionsannahme einer Objektoberfläche besagt somit, inwieweit das Oberflächenmaterial das einfallende Licht absorbieren darf, bzw. die reflektierte Farbe und Lichtintensität verändern kann.

Es liegt nun nahe ein entsprechendes Reflexionsmodell herzuleiten, wovon unser Scansystem ausgeht. Dazu betrachten wir die so genannte *BRDF*-Funktion (*bidirectional reflectance-distribution function*), ein Spezialfall der verallgemeinerten *BSSRDF*-Funktion (*bidirectional scattering-surface reflectance distribution function*), um die Wechselwirkung zwischen Licht und Oberflächenmaterial zu beschreiben [Nicodemus et al. '92].

Die Lichtstärke pro infinitesimalem differentiellen Flächenelement, die auf einem Oberflächenpunkt \mathbf{x} auftrifft, wird durch die Bestrahlungsstärke (engl. *irradiance*) $E(\mathbf{x}, \boldsymbol{\theta})$ zum Ausdruck gebracht, wobei $\boldsymbol{\theta}$ der Einheitseinfallswinkel zur Oberflächennormale \mathbf{n} in \mathbf{x} ist. Sie wird in der Radiometrie in Wm^{-2} gemessen.

Für die Messung der Lichtintensität muss zusätzlich ein *Einheitsraumwinkel*¹ betrachtet werden, da ein infinitesimales Flächenelement allein nicht messbar ist. Dazu betrachten wir die Strahldichte (engl. *radiance*) $L(\mathbf{x}, \boldsymbol{\theta})$, die im Punkt \mathbf{x} mit einem Winkel $\boldsymbol{\theta}$ zu \mathbf{n} entweder auftrifft oder reflektiert wird. Die Strahldichte wird in $\text{Wm}^{-2}\text{sr}^{-1}$ gemessen.

Sei nun $E_1(\mathbf{x}, \boldsymbol{\theta}_1)$ die Bestrahlungsstärke und $L_1(\mathbf{x}, \boldsymbol{\theta}_1)$ die Strahldichte des einfallenden Lichts mit Einfallswinkel $\boldsymbol{\theta}_1$. Entsprechend sei $L_2(\mathbf{x}, \boldsymbol{\theta}_2)$ die reflektierte Strahldichte mit Reflexionswinkel $\boldsymbol{\theta}_2$. Die BRDF-Funktion f wird also folgendermaßen definiert:

$$f(\mathbf{x}; \boldsymbol{\theta}_1, \boldsymbol{\theta}_2) := \frac{dL_2(\mathbf{x}, \boldsymbol{\theta}_2)}{dE_1(\mathbf{x}, \boldsymbol{\theta}_1)} \quad .$$

Insbesondere beschreibt f die Wahrscheinlichkeit, dass ein Photon, das aus Richtung $\boldsymbol{\theta}_1$ auf \mathbf{x} auftrifft, in eine bestimmte Richtung $\boldsymbol{\theta}_2$ reflektiert wird. Um den Zusammenhang zwischen der projizierten Lichtintensität und der reflektierten Lichtintensität zu finden betrachten wir die Strahldichte des einfallenden Lichts L_1 . Mit Hilfe von f lässt sich nun die reflektierte Strahldichte, durch Integration über alle einfallende Richtungen Θ , folgendermaßen formulieren:

$$L_2(\mathbf{x}, \boldsymbol{\theta}_2) = \int_{\Theta} f(\mathbf{x}; \boldsymbol{\theta}_i, \boldsymbol{\theta}_2) L_1(\mathbf{x}, \boldsymbol{\theta}_i) (\mathbf{n}^t \boldsymbol{\theta}_i) d\boldsymbol{\theta}_i \quad . \quad (3.1)$$

Wie bereits beschrieben, finden unsere Erfassungen in einem abgedunkelten Raum statt. Die einzige Lichtquelle ist der Projektor und dieser kann als Punktlichtquelle betrachtet werden. Ohne die Reflexionen zwischen den Flächen zu betrachten, können wir annehmen, dass jeder Oberflächenpunkt nur aus einer Richtung bestrahlt wird. Gleichung 3.1 vereinfacht sich dadurch zu:

$$L_2(\mathbf{x}, \boldsymbol{\theta}_2) = f(\mathbf{x}; \boldsymbol{\theta}_1, \boldsymbol{\theta}_2) L_1(\mathbf{x}, \boldsymbol{\theta}_1) (\mathbf{n}^t \boldsymbol{\theta}_1) \quad .$$

Daraus können wir schließen, dass es unmöglich ist, ohne Kenntnis der BRDF-Funktion, aus einer erfassten Strahldichte L_2 die genaue zugehörige projizierte Strahldichte L_1 zu berechnen. Typischerweise werden BRDF-Funktionen nur mit relativ hohem Aufwand erfasst [Marschner et al. '99, Debevec et al. '00]. Dazu müsste die Beleuchtung der Umgebung vollständig bekannt sein. Demzufolge können wir nur mit einer radikalen Näherung dieses Modells arbeiten. Da uns die Materialeigenschaften der Objekte unbekannt sind, nehmen wir an, dass die Oberfläche aus einem homogenen Material besteht. Es ergibt sich $f(\mathbf{x}; \boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = f(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$. Weiterhin setzen wir voraus, dass das Material eine ideale diffuse Reflexion besitzt, d. h. das auftreffende Licht in alle Richtungen gleichmäßig reflektiert. Die BRDF-Funktion ist also im Sinne einer *Lambertschen Reflexion* eine konstante Funktion k und es gilt:

$$L_2(\mathbf{x}, \boldsymbol{\theta}_2) = k L_1(\mathbf{x}, \boldsymbol{\theta}_1) (\mathbf{n}^t \boldsymbol{\theta}_1) \quad .$$

¹Der Einheitsraumwinkel wird in Steradian gemessen (Abkürzung sr).

Durch diese starke Reflexionsannahme des Materials, muss das projizierte Farbspektrum drastisch reduziert werden. Die Streifen unserer Projektionsmuster haben daher in jedem Farbkanal nur die maximale bzw. minimale Lichtintensität. Das Oberflächenmaterial muss somit jeden Farbkanal reflektieren. Wir zeigen in Kap. 4, dass, durch Normalisierung mit einem weißen Projektionsbild, die Geometrie des Objekts nicht benötigt wird, um aus der reflektierten Strahldichte L_2 die projizierte Strahldichte L_1 zu schätzen. Insbesondere entsteht ein linearer Zusammenhang in jedem Farbkanal.

3.2 Musterentwurf

Nachdem wir die Kohärenzannahmen festgelegt haben, soll nun, in diesem Abschnitt, auf den Entwurf des farbigen Streifenmusters eingegangen werden, welches diese Annahmen berücksichtigt. Im Hinblick auf eine Zuordnung der projizierten und der erfassten Streifen über mehrere Aufnahmen, geben wir zunächst eine einfache Darstellung dieser Streifensequenzen. Insbesondere beinhalten diese implizit die zuvor präsentierten Kohärenzannahmen. Anschließend, zeigen wir welche Schwierigkeiten, bei der Korrespondenz, auftreten können. Diese Unzulänglichkeiten führen zu einer Reihe von zusätzlichen Kriterien, die bei der Erzeugung des Streifenmusters und bei der Zuordnung, erfüllt werden müssen.

Eindimensionales Zuordnungsproblem

Ausgehend von einem Projektor-Kamera-System, sehen wir leicht, dass das zweidimensionale Zuordnungsproblem, zwischen Streifenprojektion und Kameraerfassung, sich auf eine eindimensionale zeilenweise Zuordnung reduzieren lässt. Gemäß Abb. 3.2 sei \mathbf{x} ein Objektpunkt, der für den Projektor mit Augpunkt \mathbf{o}_p und für die Kamera mit Augpunkt \mathbf{o}_c sichtbar ist. Er wird auf der Projektorbildebene auf \mathbf{u}_p und auf der Kamerabildebene auf \mathbf{u}_c abgebildet. \mathbf{x} , \mathbf{o}_p und \mathbf{o}_c erzeugen die so genannte *Epipolarebene* \mathcal{P} , die entsprechend die Projektorbildebene in einer *Epipolargeraden* \mathcal{L}_p und die Kamerabildebene in einer anderen Epipolargeraden \mathcal{L}_c schneidet. Ohne Einschränkung der Allgemeinheit können wir annehmen, dass \mathcal{L}_c eine horizontale Zeile der Kamerabildebene darstellt. Aufgrund der Epipolareinschränkung gilt somit, dass jeder Punkt auf dieser Zeile auf einen Punkt auf \mathcal{L}_p abgebildet wird. Aufgrund von Verdeckungen wird nur in eine Untermenge $\mathcal{L}_s \subseteq \mathcal{L}_p$ abgebildet. Die Streifenübergänge schneiden \mathcal{L}_s in Punkten, die auf \mathcal{L}_c sichtbar sind. Diese Punkte werden nur einer Untermenge der projizierten Streifen zugeordnet. Insbesondere ist diese Zuordnung ein eindimensionales Problem.

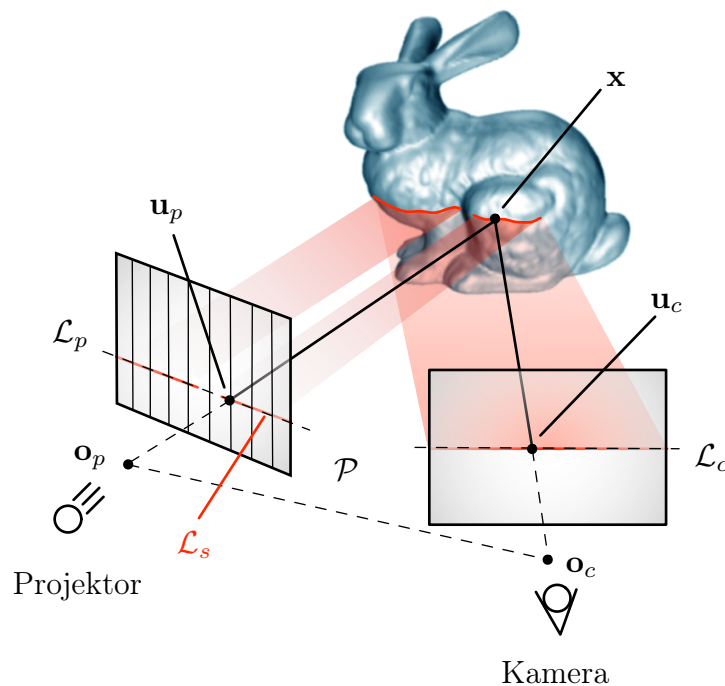


Abbildung 3.2: Durch die Epipolarbeschränkung, reduziert sich die zwei-dimensionale Zuordnung auf eine eindimensionale Zuordnung. Die erfasste Streifensequenz auf \mathcal{L}_c muss mit der projizierten Streifensequenz auf \mathcal{L}_p korrespondiert werden. Der „Stanford Bunny“ wurde aus [Levoy '05] entnommen.

Farbkodierung der projizierten Streifensequenzen

Durch die Vereinfachung des Zuordnungsproblems auf die Epipolargeraden, liegt es nun nahe, die Streifen eindeutig zu kodieren, um die Identifizierung bei der Erfassung zu ermöglichen. Insbesondere verwenden wir farbige Streifen und deren Veränderungen an den Streifenübergängen für eine schnelle Erfassung der Objektoberfläche. Wie in Abb. 3.3 veranschaulicht wird, bezeichnen wir die projizierte Streifensequenz aus N Streifen mit

$$\mathbf{P} = [\mathbf{p}_1 \dots \mathbf{p}_N] \quad ,$$

wobei \mathbf{p}_i der i -te Streifen ist.

Jeder Streifen \mathbf{p}_i wird wie folgt aus den Farben der K Projektionen zusammengesetzt:

$$\mathbf{p}_i = [\mathbf{c}_i^0 \dots \mathbf{c}_i^{K-1}]^t \quad ,$$

wobei die Farbe $\mathbf{c}_i^t = [c_{r,i}^t, c_{g,i}^t, c_{b,i}^t]^t \in [0, 1]^3$ im t -ten Projektionsbild aus den Intensitäten $c_{l,i}^t$ in jedem RGB-Kanal $l \in \{r, g, b\}$ besteht. Aufgrund der starken

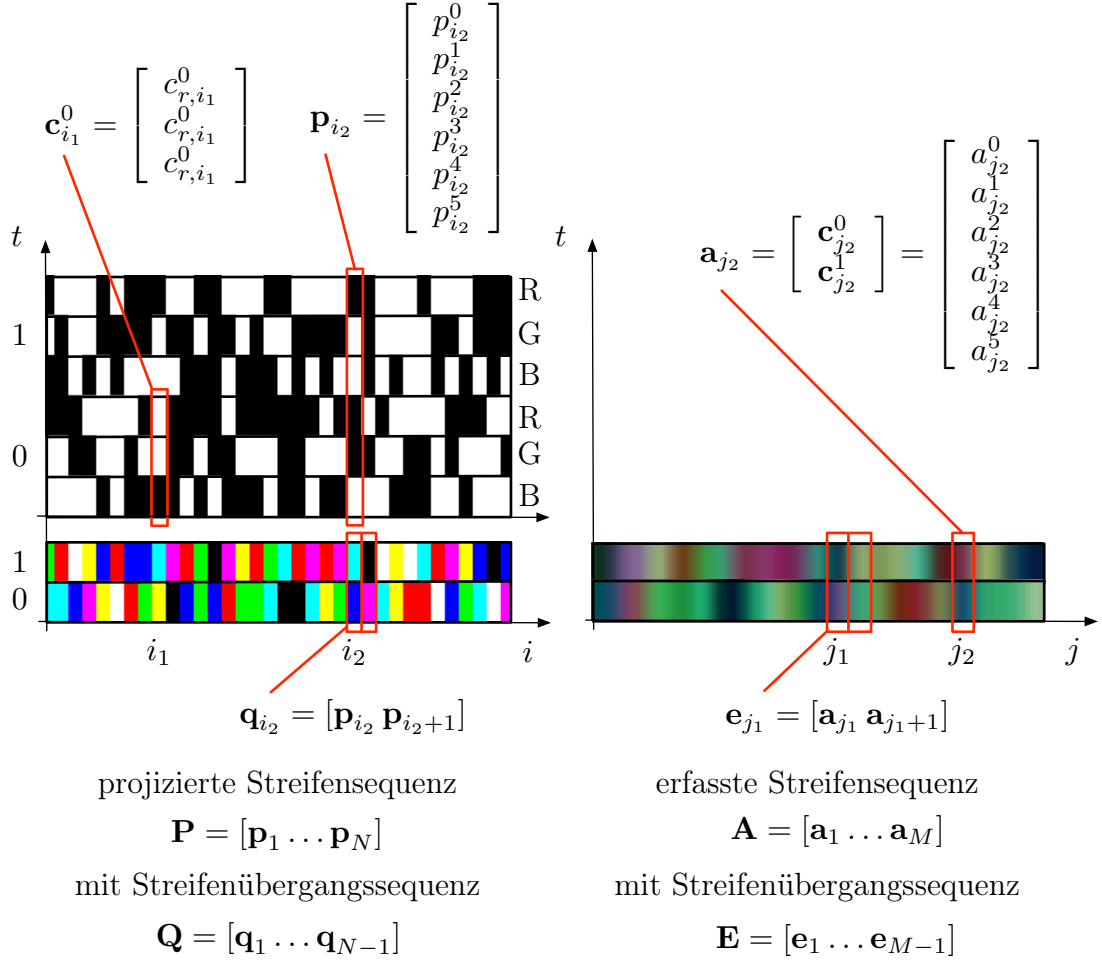


Abbildung 3.3: Farbkodierung der projizierten und erfassten Streifensequenzen. Zu jeder Sequenz betrachten wir die zugehörige Übergangssequenz.

Reflexionsannahmen, begrenzen wir das Spektrum der projizierten Farben auf die 8 Extrema des RGB-Farbraums, d. h. $\mathbf{c}_i^t \in \{0, 1\}^3$. Jedes \mathbf{p}_i enthält folglich $3K$ bit Information und lässt sich auch folgendermaßen ausdrücken:

$$\mathbf{p}_i = [p_i^0, \dots, p_i^{3K-1}]^t, \quad ,$$

mit $p_i^t \in \{0, 1\}$. Dadurch entstehen 2^{3K} verschiedene Streifenfarbkombinationen und wir bezeichnen mit \mathbf{p}_i^k die k -te Farbkombination für den i -ten Streifen.

Die Information, die wir bei dieser projizierten Sequenz verwenden, besteht, wie bereits zuvor angedeutet, aus den Streifenübergängen. Wir nennen die Streifenübergangssequenz aus $N - 1$ Übergängen

$$\mathbf{Q} = [\mathbf{q}_1 \dots \mathbf{q}_{N-1}] \quad ,$$

wobei $\mathbf{q}_i = [\mathbf{p}_i \mathbf{p}_{i+1}]$ aus beiden Nachbarn des zugehörigen Übergangs besteht.

Damit ein Übergang in der Erfassung erkennbar wird, muss vorausgesetzt werden, dass $\mathbf{p}_{i+1} - \mathbf{p}_i \neq \mathbf{o}$. Die Verwendung von Streifenübergängen als Identifikationsmerkmal hat zusätzlich den Vorteil, dass die Merkmale der Szene nicht so breit sein müssen wie eine ganzer Streifen. Außerdem ist damit eine subpixelgenaue Lokalisierung der Streifenübergänge möglich ist. Demzufolge können aus 2^{3K} eindeutig kodierte Streifen \mathbf{p}_i bis zu $(2^{3K})^2 - 2^{3K}$ eindeutig kodierte Streifenübergänge \mathbf{q}_i erzeugt werden.

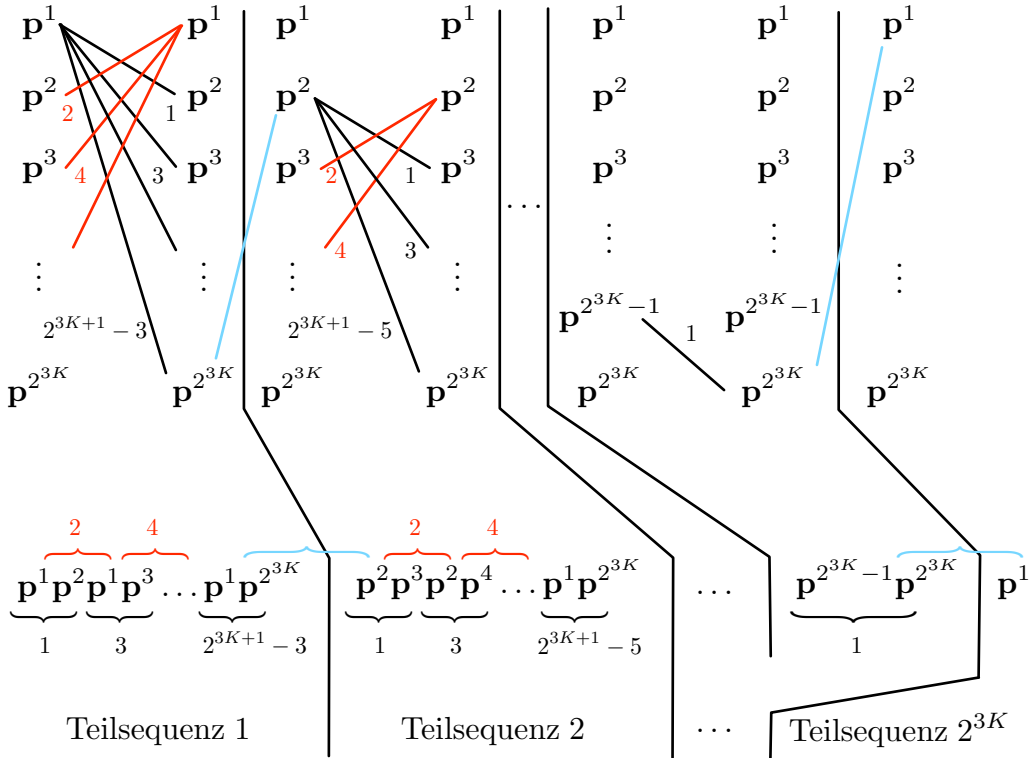


Abbildung 3.4: Naive Erzeugung einer Streifenfolge mit $N = (2^{3K})^2 - 2^{3K} + 1$ Streifen, deren Streifenübergänge alle eindeutig vorkommen. Oben: Die schwarzen Verbindungsstrecken zwischen den Farbkombinationen stellen die Streifenübergänge mit ungeraden Indizes, die roten Verbindungsstrecken diejenigen mit geraden Indizes dar. Die blauen Verbindungsstrecken stellen die Streifenübergänge zwischen den Teilsequenzen dar. Unten: Die entsprechende erzeugte Streifenfolge.

Eine entsprechende Streifenfolge mit $(2^{3K})^2 - 2^{3K}$ Streifenübergängen kann, wie in Abb. 3.4 schematisch dargestellt wird, erzeugt werden. Dazu wird die gesamte Sequenz in 2^{3K} Teilsequenzen unterteilt. Die Streifenübergänge mit ungeraden Indizes (schwarze Verbindungsstrecke des Schemas zwischen den Farbkombinationen) der k -ten Teilsequenz bestehen jeweils aus zwei unterschiedliche Farbkombinationen $\mathbf{p}_i^{k_1}$ und $\mathbf{p}_{i+1}^{k_2}$, wobei $k_1 = k$ und $k_2 \geq k + 1$. Die Streifenübergänge mit geraden Indizes (rote Verbindungsstrecke des Schemas zwischen den Farbkombinationen)

der k -ten Teilsequenz bestehen jeweils aus zwei unterschiedliche Farbkombinationen $\mathbf{p}_i^{k_1}$ und $\mathbf{p}_{i+1}^{k_2}$, wobei $k_1 \geq k + 1$ und $k_2 = k$. Die blauen Verbindungsstrecken des Schemas stellen die Streifenübergänge zwischen zwei Teilsequenzen dar.

Wir können feststellen, dass wir bei $K = 2$ Projektionen bereits eine Streifensequenz aus 4033 Streifen mit 4032 eindeutigen Streifenübergängen erzeugen können. Da unser Projektor nur 1024 vertikale Streifen projizieren kann, sind zwei Projektionsbilder völlig ausreichend. Außerdem werden in der Praxis, aufgrund von geringen Objektgrößen, meist eine deutlich niedrigere Anzahl von Streifen benötigt (oft nur 200 Streifen).

Formulierung einer optimalen Zuordnung

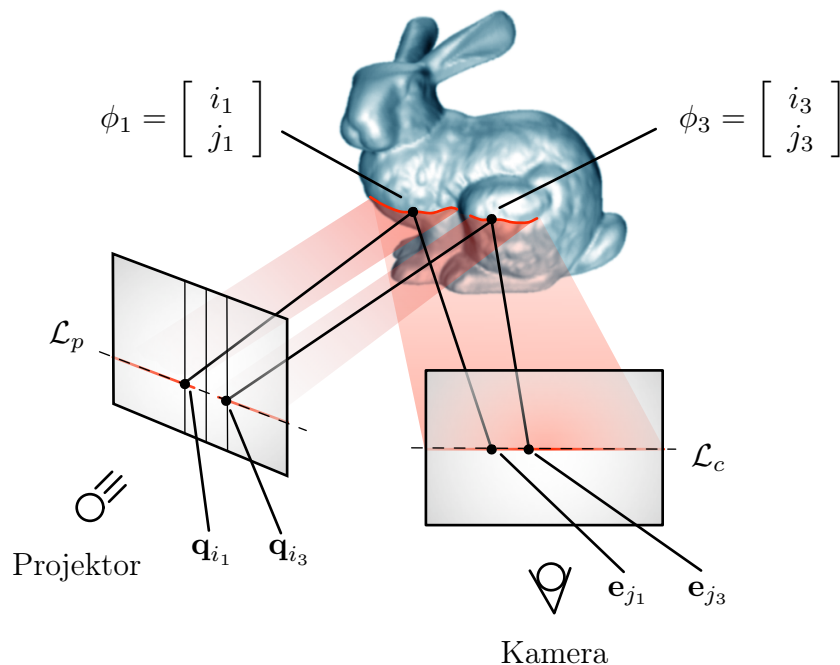


Abbildung 3.5: Bei der eindimensionalen Zuordnung wird ein projizierter Streifenübergang \mathbf{q}_{i_l} , auf einer Zeile der Kamerabildebene, in einem Streifenübergang \mathbf{e}_{j_l} erfasst. Die l -te Zuordnung wird mit $\phi_l = (i_l, j_l)$ bezeichnet.

Die projizierten Streifen werden vom Objekt reflektiert und durch die Kamera, wie in Abb. 3.5 gezeigt, zeilenweise erfasst. Wir erhalten dementsprechend eine erfasste Streifensequenz und die dadurch induzierte Streifenübergangssequenz (siehe Abb. 3.3). Analog zur Bezeichnung der projizierten Streifen, wird die erfasste Streifensequenz aus M Streifen gegeben durch:

$$\mathbf{A} = [\mathbf{a}_1 \dots \mathbf{a}_M] \quad ,$$

mit dem j -ten Streifen $\mathbf{a}_j = [\mathbf{c}_j^0 \dots \mathbf{c}_j^{K-1}]^t = [a_j^0, \dots, a_j^{3K-1}]^t \in [0, 1]^{3K}$. Daraus ergibt sich die erfasste Streifenübergangssequenz der Länge $M - 1$:

$$\mathbf{E} = [\mathbf{e}_1 \dots \mathbf{e}_{M-1}] \quad ,$$

wobei $\mathbf{e}_j = [\mathbf{a}_j \mathbf{a}_{j+1}]$.

Unser Ziel ist es, die projizierte Streifenübergangssequenz \mathbf{Q} der erfassten Streifenübergangssequenz \mathbf{E} zuzuordnen. Insbesondere wird die gesamte Zuordnung durch die Menge Φ aller einzelnen Zuordnungen ϕ_l repräsentiert:

$$\begin{aligned} \Phi &= \{\phi_1, \dots, \phi_L\} \\ &= \left\{ \begin{bmatrix} i_1 \\ j_1 \end{bmatrix}, \dots, \begin{bmatrix} i_L \\ j_L \end{bmatrix} \right\} \quad , \end{aligned}$$

wobei L die Anzahl der Paare ϕ_l ist. Es gilt $L \leq N$ und $L \leq M$. $\phi_l = [i_l, j_l]^t$ bedeutet, dass der erfasste Streifenübergang \mathbf{e}_{j_l} dem projizierten Streifenübergang \mathbf{q}_{i_l} zugeordnet ist (siehe Abb. 3.5). Ohne Einschränkung der Allgemeinheit gilt $i_1 < i_2 < \dots < i_L$ und $l_1 \neq l_2 \Rightarrow j_{l_1} \neq j_{l_2}$ für alle $l_1, l_2 = 1, \dots, L$. Insbesondere haben wir hier keine Monotonizitätsannahme, da ansonsten zusätzlich $j_1 < j_2 < \dots < j_L$ gelten müsste.

Folglich ist die Qualität einer bestimmten Zuordnung $\phi_l = [i_l, j_l]^t$ dadurch gegeben, wie stark die Intensitäten von \mathbf{e}_{j_l} und \mathbf{q}_{i_l} korrelieren, bzw. wie klein der Abstand der Intensitäten ist. Es handelt sich hier um eine kantenbasierte Bewertungsfunktion.

Wir definieren den quadratischen Abstand d_ϕ zwischen einem projizierten und einem erfassten Streifenübergang folgendermaßen:

$$d_\phi(\mathbf{q}_i, \mathbf{e}_j) := \left\| \begin{bmatrix} \mathbf{p}_i \\ \mathbf{p}_{i+1} \end{bmatrix} - \begin{bmatrix} \mathbf{a}_j \\ \mathbf{a}_{j+1} \end{bmatrix} \right\|^2 \quad (3.2)$$

$$= \sum_{k=0}^{3K-1} ((p_i^k - a_j^k)^2 + (p_{i+1}^k - a_{j+1}^k)^2) \quad . \quad (3.3)$$

Schließlich können wir die Qualität der gesamten Zuordnung $\sigma(\Phi)$ zwischen \mathbf{Q} und \mathbf{E} durch die Summe der quadrierten Abstände d_ϕ wie folgt bestimmen:

$$\sigma(\Phi) = \sum_{l=1}^L d_\phi(\mathbf{q}_{i_l}, \mathbf{e}_{j_l}) \quad .$$

Daraus ergibt sich die global optimale Zuordnung

$$\Phi^* = \arg \min_{\Phi} \{\sigma(\Phi)\} \quad . \quad (3.4)$$

Insbesondere kann diese optimale Lösung ohne zusätzliche Einschränkungen nur in $O(N^M)$ gelöst werden. Die Bestimmung einer Näherung zur optimalen Lösung Φ^* wird detailliert in Kap. 4 behandelt und es wird insbesondere ein effektiver Algorithmus vorgestellt, der mit einer gierigen Suchstrategie, die projizierten und erfassten Streifensequenzen zuordnet.

Schwierigkeiten bei der Identifikation von Streifenübergängen

Bisher können wir, mittels im Zeitmultiplex kodierten Farbstreifensequenzen, mit nur zwei Projektionen eindeutige Streifenübergänge auf die Objekt Oberfläche projizieren. Bei der Erfassung von hoch komplexen Objekten, die keine idealen Reflexionseigenschaften besitzen und viele Selbstverdeckungen und Unstetigkeiten in der Oberfläche beinhalten, würde eine naive Erzeugung von Streifensequenzen, wie in Abb. 3.4 vorgestellt wurde, zu einer hohen Anzahl von fehlerhaften Zuordnungen führen. Im Hinblick auf den Entwurf eines robusten Projektionsmusters, zeigen wir vorerst, anhand von Beispielen, wodurch die Identifikation von Streifenübergängen beeinträchtigt wird. Im Unterschied zu den, im Kap. 2 vorgestellten, Unzulänglichkeiten, werden hier die Verfälschungen von Streifen des zugehörigen Streifenübergangs behandelt (siehe Abb. 3.6). Dazu unterscheiden wir drei Fälle:

- **Nicht detektierte Streifen:** Oft kommt es vor, dass die projizierten Streifen unvollständig auf die Kamerabildebene abgebildet werden. Insbesondere Selbstverdeckungen, Schattenbereiche, Streifenübergänge mit schwachem Kontrast, zu nah beieinanderliegende Streifenübergänge und Löcher sind Ursache dieses Problems. Fehlende Streifen in der Erfassung führen zu zwei Arten von falsch identifizierten Streifenübergängen. Bei den Einen kann der Fehler festgestellt werden, entweder durch die Inkompatibilität zur lokalen Nachbarschaft oder einfach weil der Streifenübergang nicht projiziert wird. Bei den Anderen kann der Fehler nicht erkannt werden.

In Abb. 3.6a wird der projizierte Streifen \mathbf{p}_2 aufgrund einer Verdeckung nicht von der Kamera erfasst. Aus den erfassten Streifen \mathbf{a}_1 und \mathbf{a}_2 wird ein Streifenübergang \mathbf{e}_1 interpretiert, der zu keinem projizierten Streifenübergang gehört. Es handelt sich um einen falsch identifizierten Streifenübergang der detektiert werden kann. Wäre z. B. $\mathbf{p}_3 = \mathbf{p}_1$ würden wir gar keinen Streifenübergang erfassen.

In Abb. 3.6b wird eine Sequenz aus 5 Streifen projiziert, wobei der vierte projizierte Streifen \mathbf{p}_4 von der Kamera aus nicht zu sehen ist. Der erste und dritte, sowie der zweite und der fünfte projizierten Streifen haben dieselben Farben. Es werden in der Erfassung drei Streifenübergänge lokalisiert, wobei \mathbf{e}_1 und \mathbf{e}_3 nicht zu unterscheiden sind, da sie dieselben benachbarten Farben haben und sogar im projizierten Bild vorkommen. In diesem Fall können wir den falsch erkannten Streifenübergang \mathbf{e}_3 nicht detektieren und es kann zu einer Falschzuordnung kommen.

- **Zusätzlich detektierte Streifen:** Ein weiteres Problem sind zusätzlich detektierte Streifen, die nicht zu den projizierten Streifen gehören. Aufgrund von hochfrequenten kontrastreichen Texturen, Schattenbereichen und Rauschen, die durch das Scansystem entstehen (siehe Kap. 2), wird das Zuordnungsproblem erschwert.

Das Beispiel in Abb. 3.6c zeigt, wie ein Schattenbereich, der vom Projektor aus nicht sichtbar ist, durch die Kamera erfasst wird. Es wird ein Streifen \mathbf{a}_3 erfasst und es entstehen zwei zusätzliche Streifenübergänge \mathbf{e}_2 und \mathbf{e}_3 , die nicht zu den projizierten Übergängen gehören. Die Schwierigkeit liegt nun darin, zwischen einem projizierten Streifenübergang und einem erfassten Streifenübergang zu unterscheiden.

- **Farbverfälschung:** Eine schwierige Identifikation von Streifenübergängen wird auch durch schlechte Reflexionseigenschaften der Objektoberfläche und durch eine ungenaue Farbkalibrierung des Scansystems verursacht.

Wie bereits erwähnt, ist es unrealistisch von einer ideal reflektierenden Objektoberfläche auszugehen. Abbildung 3.6c zeigt die Projektion von zwei Streifen \mathbf{p}_1 und \mathbf{p}_2 , die in \mathbf{a}_1 bzw. \mathbf{a}_2 abgebildet werden. Bei diesem Beispiel handelt es sich um eine Objektoberfläche, die kurzwellige Lichtstrahlen besser reflektiert als langwellige. Die Intensitätswerte von \mathbf{a}_2 werden dadurch verfälscht und zusammen mit \mathbf{a}_1 entsteht ein inkorrekt erfasster Streifenübergang \mathbf{e}_1 .

Neue Methoden für robustere Projektionsmuster

Die oben präsentierten Unzulänglichkeiten weisen darauf hin, dass eine eindeutige Farbkodierung über mehrere Projektionsbilder alleine für eine robuste Erfassung der Objektoberfläche nicht ausreichend ist. Aufgrund der Scankonfiguration und der Geometrie des Objekts, sind fehlerhafte Streifenübergänge in der Erfassung in der Regel unausweichlich. Um die Ausreißer, die bei der optischen Triangulierung entstehen können, zu reduzieren, müssen die falschen Streifenübergänge frühzeitig erkannt und beseitigt werden. Damit inkorrekte Streifenübergänge, die sich nicht von korrekten Übergängen unterscheiden lassen, nicht vorkommen, müssen eine Reihe von Restriktionen bei der Erzeugung des Streifenmusters beachtet werden. Insbesondere stellen wir in dieser Arbeit neue Methoden, für eine gezielte Vorbeugung von nicht-identifizierbaren fehlerhaften Streifenübergängen, vor. Es ist uns derzeit kein vergleichbares Verfahren bekannt, welches Streifenmuster erzeugt, die alle Eigenschaften unserer Muster besitzen. Wir haben experimentell ermittelt, dass diese Eigenschaften signifikante Vorteile gegenüber dem Verfahren von [Zhang et al. '02] bringen, welches ein De-Bruijn-Muster projiziert.

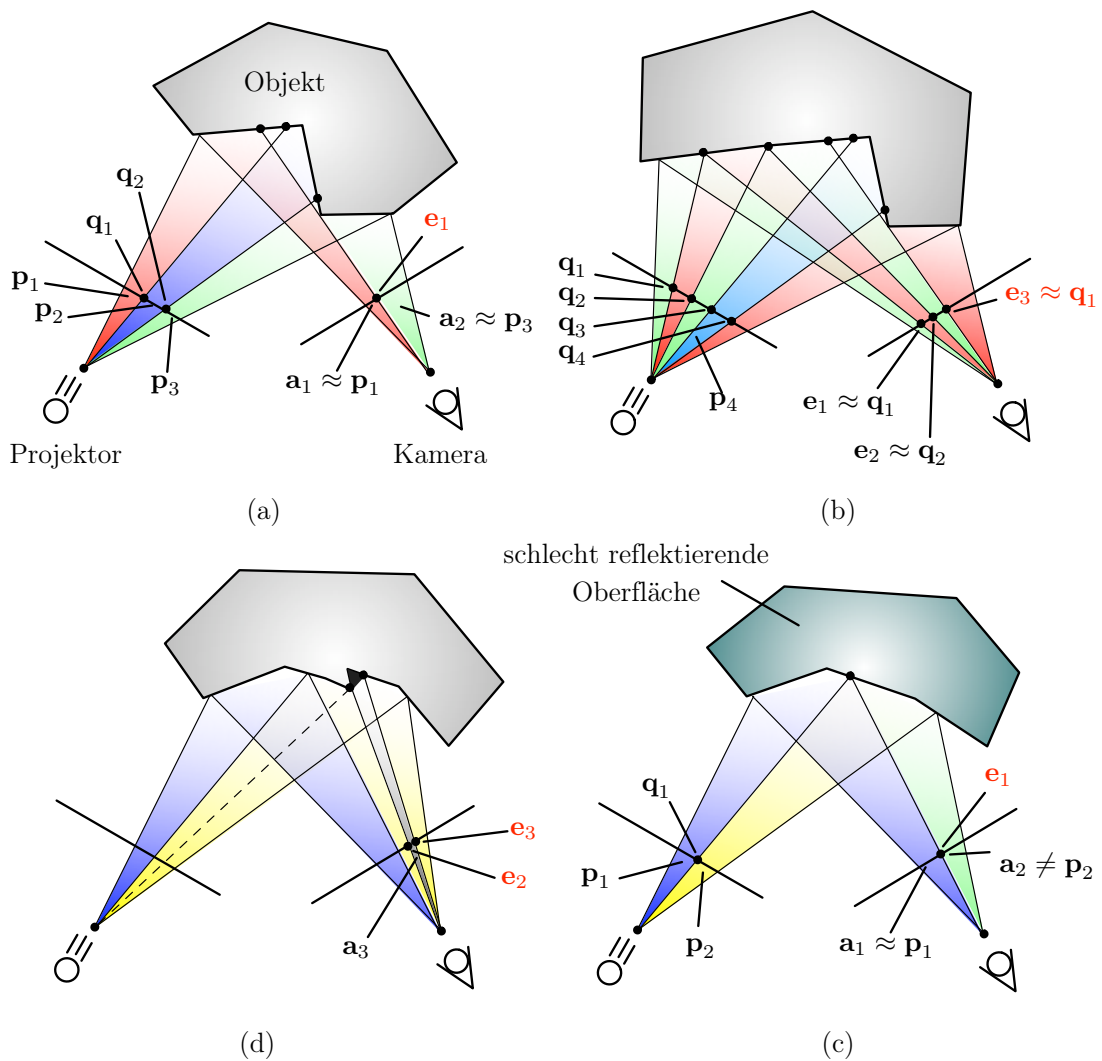


Abbildung 3.6: Inkorrekte Identifikation der Streifenübergänge in der Erfassung. (a) Nicht detektierte Streifen, wobei ein fehlerhafter Streifenübergang erkannt werden kann. (b) Nicht detektierte Streifen, wobei ein fehlerhafter Streifenübergang von einem korrekten Streifenübergang nicht zu unterscheiden ist. (c) Zusätzlich detektierte Streifen. (d) Streifen, deren Farbwerte verfälscht sind.

Um eine Streifenfolge mit den erwünschten Eigenschaften zu entwerfen, definieren wir den Begriff einer *Streifenumgebung* $\mathcal{U}_r(\mathbf{p}_i)$ mit Zentrum \mathbf{p}_i und Radius r wie folgt:

$$\mathcal{U}_r(\mathbf{p}_i) = \{ \mathbf{p}_{i'} \mid |i' - i| \leq r \} \quad .$$

Indem wir sicherstellen, dass jeder Streifen der Streifenfolge $\mathbf{P} = [\mathbf{p}_1 \dots \mathbf{p}_N]$ in seiner Streifenumgebung nicht mehrfach vorkommen kann, können wir die lokalen Mehrdeutigkeiten aus Abb. 3.6b, die zu nicht identifizierbaren Fehlern führen,

vermeiden. Dadurch, dass keine zwei Streifenfarben sehr nahe beieinander liegen dürfen, erhöhen wir dabei auch den Triangulationsfehler, der bei einer Falschzuordnung entstehen würde. Dadurch lässt sich eine Ausreißereliminierung leichter durchführen, da größere Ausreißer nicht mit Rauschen verwechselt werden können. Insbesondere muss für die erzeugte Streifensequenz \mathbf{P} in der Menge \mathcal{M}_l der Streifensequenzen liegen, die in der lokalen Umgebung eindeutige Streifen haben:

$$\mathcal{M}_l = \{[\mathbf{p}_1 \dots \mathbf{p}_N] \mid \forall i \in \{1, \dots, N\}, \forall \mathbf{p}_{i_1} \in \mathcal{U}_r(\mathbf{p}_i), \forall \mathbf{p}_{i_2} \in \mathcal{U}_r(\mathbf{p}_i) : \\ (i_1 = i_2) \vee (\mathbf{p}_{i_1} \neq \mathbf{p}_{i_2})\} \quad .$$

Da wir die Streifenübergänge mittels ihrer benachbarten Streifen in der Erfassung identifizieren, kann es vorkommen, dass ein falsch erfasster Streifenübergang sich nicht von einem korrekten Übergang unterscheiden lässt. Dies tritt vor allem bei Objekten mit hochfrequenten Unstetigkeiten und kleine Verdeckungen in der Oberfläche auf (siehe Abb. 3.6a). Man kann beobachten, dass ein neuer Streifenübergang aus Streifen entsteht, die oft sehr nahe beieinander liegen. Es liegt nun nahe zu verhindern, dass diese, aus den lokal umgebenden Streifen neu entstandenen, Streifenübergänge die Eindeutigkeit der projizierten Streifenübergänge in der gesamten Sequenz verletzen. Dies ist eine globale Eigenschaft. Alle Streifenpaare aus einer Umgebung dürfen also nicht gleichzeitig in weiteren, nicht überlappenden Umgebungen liegen. Die Menge der Streifensequenzen mit dieser Eigenschaft bezeichnen wir mit

$$\mathcal{M}_g = \{[\mathbf{p}_1 \dots \mathbf{p}_N] \mid |i_1 - i_2| \geq 2r \Rightarrow |\mathcal{U}_r(\mathbf{p}_{i_1}) \cap \mathcal{U}_r(\mathbf{p}_{i_2})| \leq 1\} \quad .$$

Im Gegensatz zu den Verfahren, die nur eine einzige Streifenprojektion benötigen [Zhang et al. '02, Li et al. '04], können wir, vergleichbar mit dem Verfahren aus [Hall-Holt & Rusinkiewicz '01], in der Zeit variierende Streifenübergänge verwenden, um zwischen projizierten und nicht projizierten (zusätzlich detektierten) Streifenübergängen unterscheiden zu können (siehe Abb. 3.6c). Da wir eine statische Szene erfassen, können wir bei der Erfassung ausnutzen, dass es sich nur dann um einen projizierten Streifenübergang handeln kann, wenn es unterschiedliche Zeiten t_1 und t_2 gibt, wo sich der Streifenübergang zwischen diese Zeiten verändert. Die Menge der Streifensequenzen mit dieser Eigenschaft nennen wir:

$$\mathcal{M}_t = \{[\mathbf{p}_1 \dots \mathbf{p}_N] \mid \forall i \in \{1, \dots, N\} \exists t_1, t_2 : t_1 \neq t_2, [\mathbf{c}_i^{t_1} \mathbf{c}_{i+1}^{t_1}] \neq [\mathbf{c}_i^{t_2} \mathbf{c}_{i+1}^{t_2}]\} \quad .$$

Das zu erzeugende Streifenmuster \mathbf{P} soll alle vorgestellten Eigenschaften besitzen:

$$\mathbf{P} \in (\mathcal{M}_l \cap \mathcal{M}_g \cap \mathcal{M}_t) \quad . \quad (3.5)$$

Für eine noch zuverlässigere Erfassung ist es möglich, weitere Bedingungen vorzusetzen, wie z. B. nur Streifenübergänge mit hohen Kontrasten zu erlauben

oder gewisse Farbübergänge, die durch das Scansystem schlecht erfasst werden, zu verbieten, wie in [Zhang et al. '02] festgestellt worden ist. Zum Zeitpunkt dieser Arbeit wurde in [Koninckx et al. '05] ein Verfahren präsentiert, dass die lokalen Intensitäten der projizierten Streifen, auf Kosten mehrerer zusätzlicher Aufnahmen, an die Geometrie und die nicht-linearen Reflexionseigenschaften der Szene automatisch anpasst. Dieser Ansatz könnte zweifelsfrei auch auf unser Scanverfahren übertragen werden.

3.3 Algorithmus zur Mustererzeugung

Nachdem wir, für den Entwurf des Streifenmusters, die Eigenschaften formuliert haben, die eine robustere Zuordnung ermöglichen, stellt sich die Frage, wie eine solche Streifensequenz zu erzeugen ist. Wie auch bei einigen bereits genannten Arbeiten, ist die Erzeugung von Streifensequenzen, die globale Eigenschaften besitzen, meistens nicht selbstverständlich. In [Zhang et al. '02, Li et al. '04] werden beispielsweise De-Bruijn-Sequenzen mittels einer rekursiven Funktion erzeugt. In [Hall-Holt & Rusinkiewicz '01] wird das Problem der Erzeugung von Streifensequenzen auf eine Graphensuche transformiert.

In diesem Abschnitt stellen wir einen Algorithmus vor, der Streifensequenzen mit den oben präsentierten Eigenschaften erzeugt. Wir erklären als Erstes die Hauptschritte der iterativ erzeugten Sequenz und welche Tests durchgeführt werden müssen, damit die erwünschten Bedingungen erfüllt werden. Anschließend zeigen wir, wie wir den Gültigkeitstest der globalen Eigenschaft mittels einer Nachschlagetafel (engl. look-up table) optimieren können und führen das Verfahren an einem Beispiel vor. Zum Schluss, zeigen wir, wie das Verfahren drastisch beschleunigt werden kann.

Iterative Erzeugung

Die Aufgabe besteht nun darin, einen Algorithmus zu finden, der die Sequenz $\mathbf{P} = [\mathbf{p}_1 \dots \mathbf{p}_N]$ aus N Streifen erzeugt, so dass die Bedingung aus Gleichung 3.5 erfüllt ist. Die Länge N , die Anzahl der Projektionsbilder K und der Streifenumgebungsradius r werden als Eingabe erwartet. Der i -te Streifen \mathbf{p}_i wird aus einer Menge von 2^{3K} Farbkombinationen \mathbf{p}^k gewählt und mit K Streifenbildern projiziert. Da \mathbf{p}^k die k -te Farbkombination ist, können wir die zu erzeugende Sequenz durch eine Folge $k(n)$ darstellen. $k(i) \in \mathbb{N}$ ist die i -te Farbkombination, die in \mathbf{P} vorkommt. Der Ansatz besteht nun darin, ausgehend von einem beliebigen Anfangswert $k(1)$, die restliche Sequenz bis zur Länge N iterativ zu bestimmen.

Es liegt nun nahe, bei jeder Iteration, den neu hinzugefügten Streifen zu testen, ob er gewisse Kriterien erfüllt. Die Bedingung, dass $\mathbf{P} \in (\mathcal{M}_l \cap \mathcal{M}_t)$, stellt keine Schwierigkeit dar, da sie eine lokale Eigenschaft beschreibt. Für $\mathbf{P} \in \mathcal{M}_g$ dagegen,

müssen alle Umgebungen der zuvor erzeugten Farbkombinationen in Betracht gezogen werden. Da wir nur wenige Streifenbilder projizieren wollen, haben wir nur eine begrenzte Anzahl von Streifenkombinationen zur Hand. Demzufolge müssen wir bei der Erzeugung von längeren Streifensequenzen eine Vielzahl von Farbkombinationen wiederverwenden (die Streifenübergänge bleiben trotzdem eindeutig). Durch die Mehrdeutigkeit der verwendeten Streifenfarben und die Suche nach einer längeren Sequenz N , ist es umso schwieriger zu garantieren, dass die gesetzte globale Bedingung erfüllt ist.

Für die Bedingung $\mathbf{P} \in (\mathcal{M}_l \cap \mathcal{M}_g)$, verwendet unser Algorithmus in jeder Iteration drei Tests. Das zugehörige Flussdiagramm wird in Abb. 3.7 gezeigt.

Zu Beginn der i -ten Iteration wird die neue hinzugefügte Farbkombination mit $k(i) := 1$ initialisiert. Anschließend wird getestet, ob sich $k(i)$ bis zu dem r -ten vorherigen Streifen wiederholt (siehe Abb. 3.7a). Insbesondere entsteht ein Streifenübergang, da $k(i) \neq k(i-1)$ gelten muss. Dieser Test erfüllt noch nicht die Bedingung, dass $\mathbf{P} \in \mathcal{M}_l$, da nur ein Radius r in einer einzigen Richtung betrachtet wird. Falls es sich um eine neue Streifenkombination handelt, so wird sie genommen und es kann mit $i := i + 1$ zum nächsten Iterationsschritt gegangen werden.

Falls sich $k(i)$ bereits in der Streifensequenz befindet, müssen wir in jeder Umgebung der bisher erzeugten Sequenz überprüfen, ob alle Paare von Farbkombinationen dieser Umgebung sich nicht in einer nicht überlappenden Umgebung der gesamten Sequenz finden lassen (siehe Abb. 3.7b). Je höher der Radius r dieser Umgebung gewählt wird, desto weiter entfernt werden Paare von Farbkombinationen in der gesamten Sequenz unter sich sein. Die Robustheit des projizierten Streifenbilds steigt somit mit einem höheren Radius r . Da es sich um einen recht aufwändigen Prozess handelt, zeigen wir weiter unten, wie wir diesen Gültigkeitstest mit einer Nachschlagetabelle beschleunigen können. Wenn der Test erfolgreich ist, wird $k(i)$ zur Streifensequenz aufgenommen.

Bei einem Fehlschlag des vorherigen Tests, müssen wir mit einer anderen Farbkombination neu anfangen. Da wir am Anfang der Iteration $k(i) := 1$ gesetzt haben, können wir inkrementell mit $k(i) := k(i) + 1$ die nächste Farbkombination nehmen. Wenn alle Farbkombinationen ausprobiert sind und keine ein gültiges Nachfolgeelement der Sequenz darstellt (siehe Abb. 3.7c), deutet es darauf hin, dass die Wahl der Farbkombinationen in der bisherigen Sequenz unangemessen ist. Insbesondere müssen diese geändert werden, indem wir mit $i := i - 1$ die Sequenz nach einem Rückverfolgungsschema (engl. back-tracking) zurücklaufen, um mit einer anderen Farbkombination zu testen.

Um zu garantieren, dass alle erzeugten Streifenübergänge sich über die Aufnahmen mindestens einmal ändern ($\mathbf{p} \in \mathcal{M}_l$), müssen wir einen weiteren Testschritt hinzufügen. Dieser befindet sich in unserer Implementierung zwischen dem ersten und zweiten Test. Er ist nicht in Abb. 3.7 abgebildet.

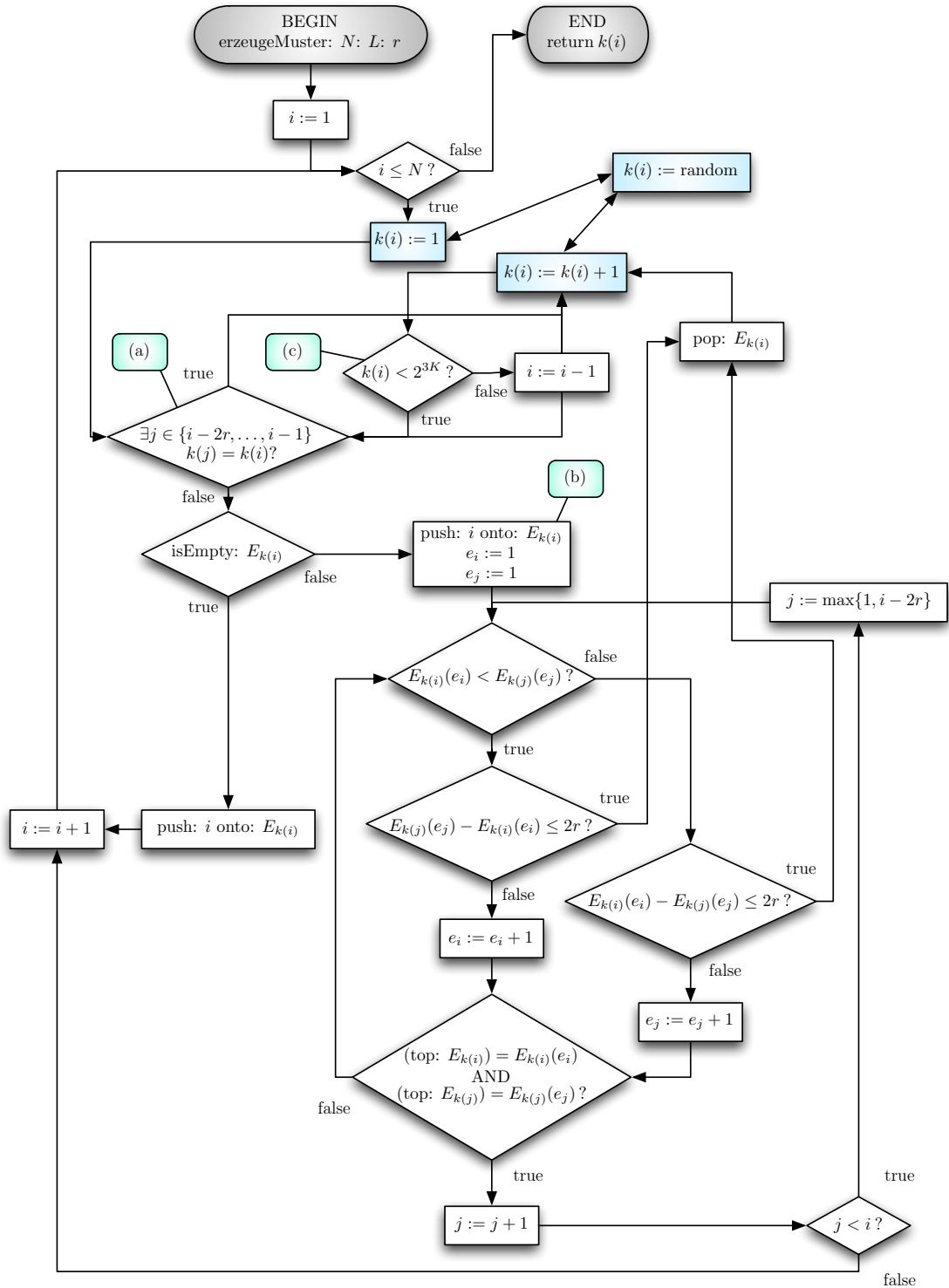


Abbildung 3.7: Flussdiagramm des Algorithmus für die Erzeugung von $\mathbf{P} \in (\mathcal{M}_l \cap \mathcal{M}_g)$. Die Erzeugung einer Farbkombination $k(i)$ des i -ten Streifens kann zur Beschleunigung durch einen Zufallszahlengenerator ersetzt werden (blau markiert).

Weil dieser Algorithmus alle Möglichkeiten iterativ durchsucht, ist er bei einer großen Länge N nicht praktikabel. Es ist es notwendig die Tests, durch eine passende Datenstruktur, zu optimieren und die inkrementelle Auswahl von neuen Farbkombinationen zu vermeiden.

Gültigkeitstest mit Nachschlagetabelle

Um die Gültigkeit der globalen Bedingung aus Gleichung 3.5 einer iterativ hinzugenommenen Farbkombination zu überprüfen, ist es wichtig, die Informationen aus zuvor bereits durchsuchten Umgebungen auszunutzen. Insbesondere ist es hilfreich zu wissen, an welchen Positionen sich bereits eine bestimmte Farbkombination in der bisher erzeugten Sequenz befindet. Dadurch können wir sofort feststellen, ob, bei der Hinzunahme einer neuen Farbkombination, eine Struktur entsteht, die bereits vorhanden und somit ungültig ist.

Die Position bereits verwendeter Farbkombinationen speichern wir in einer Nachschlagetabelle, die aus einer Liste von 2^{3K} Kellern (engl. stacks oder push-downs) besteht. Der k -te Keller wird mit E_k bezeichnet und stellt die entsprechende Farbkombination dar. Bei jeder push-Operation wird die Position i des aktuellen Streifens aufgenommen. Um auch auf Positionen, die unterhalb der zuletzt hinzugefügten Position liegen, zugreifen zu können, erlauben wir mit $E_k(e)$ den Zugriff auf das e -te hinzugefügte Element. Wie im Flussdiagramm in Abb. 3.7 skizziert wird, legen wir zunächst bei der Hinzunahme einer noch nicht verwendeten Farbkombination, die zugehörige Position i , deren Farbkombination $k = k(i)$ ist, auf E_k . Weiterhin wird in Abb. 3.7b vorübergehend die Position i im Keller abgespeichert. Daraufhin werden die Positionen der sich im Radius r von i befindenden, Farbkombinationen j überprüft. Insbesondere werden paarweise die Kellereinträge dieser Positionen mit den Kellereinträgen der neu hinzugefügten Farbkombination verglichen und getestet, ob deren Abstände $|i - j|$ innerhalb von r liegen. Wenn dies bei keiner paarweisen Farbkombination der Fall ist, ist die hinzugefügte Farbkombination gültig.

Abbildung 3.8 veranschaulicht den Ablauf dieses Algorithmus anhand eines einfachen Beispiels. Hier werden die ersten 9 Farbkombinationen erzeugt. Wir verwenden dazu den Radius $r = 1$ und projizieren $K = 2$ Streifenmuster. Auf der linken Seite wird eine Ablaufverfolgung der getesteten Farbkombinationen an jeder Stelle i gegeben. Beispielsweise wird bei der zweiten Iteration zunächst $k(2) = 1$ getestet. Da sich $k(1) = 1$ innerhalb des Radius befindet, scheitert diese Farbkombination schon beim ersten Test. Dies wird mit 1_a gekennzeichnet. Da die Kombination $k(2) = 2$ noch nicht verwendet wurde, wird sie akzeptiert und wir tragen sie entsprechend im Keller E_2 ein. Bei der vierten Iteration, stellen wir fest, dass keine Farbkombination innerhalb des Radius r gleich $k(4) = 1$ ist. Folglich müssen wir den zweiten Test durchführen. Hierbei vergleichen wir die Farbkombinationen $k(2) = 2$ mit $k(4) = 1$ in den zugehörigen Kellern E_2 bzw. E_1 . Da bereits

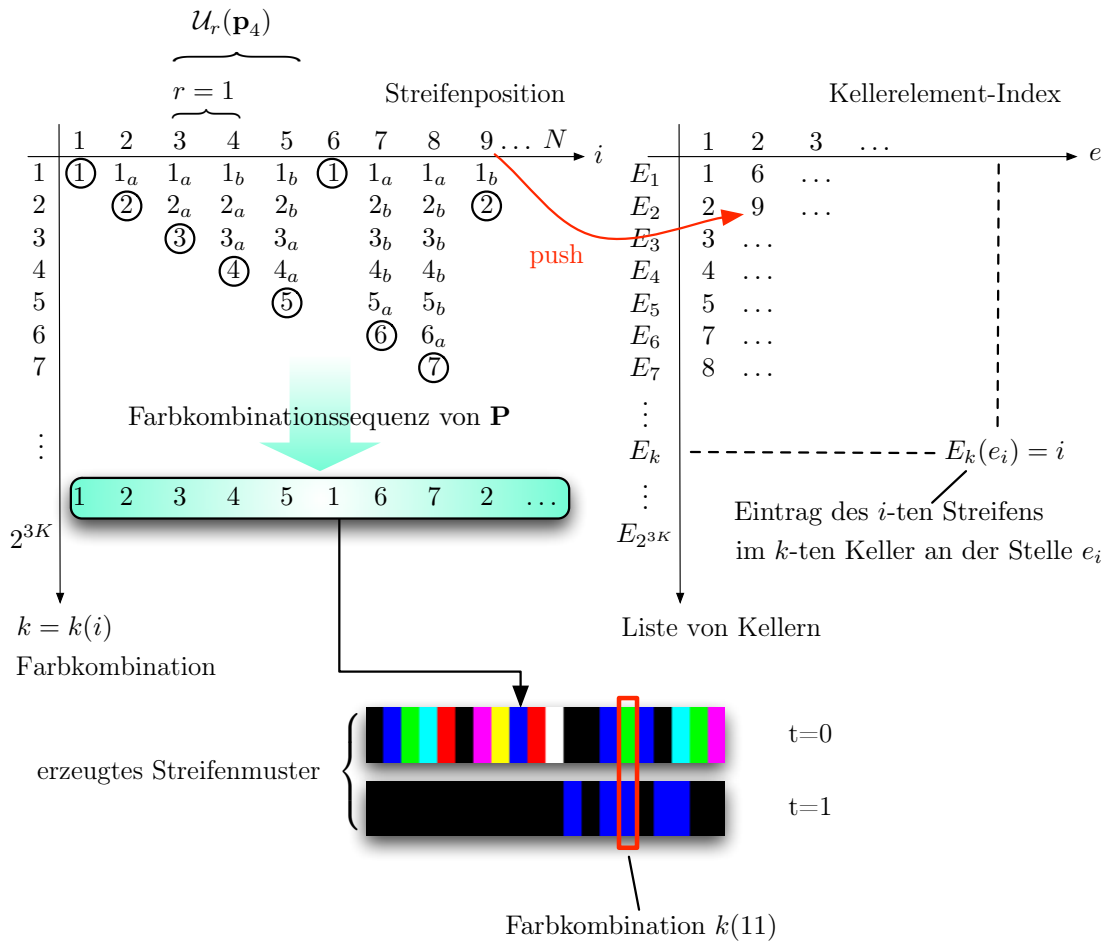


Abbildung 3.8: Ein Beispiel des Ablaufs des vorgestellten Gültigkeitstests für die Bedingung aus Gleichung 3.5. Hier werden die ersten 9 Farben erzeugt und $r = 1$ genommen. Links: Die Ablaufverfolgung der getesteten Farbkombinationen an jeder Position i . Die eingekreisten Zahlen sind die akzeptierten Farbkombinationen, die in die Sequenz aufgenommen werden. Rechts: Die Liste aus den 2^{3K} Kellern, die als Nachschlagetabelle verwendet wird.

$E_2(1) - E_1(1) = 1 \leq r = 2$ können wir schließen, dass $k(4) = 1$ ungültig ist und bezeichnen es mit 1_b .

Wir können an diesem Beispiel feststellen, dass dieser Algorithmus die Tendenz hat, sehr schnell alle möglichen Farbkombinationen auszuschöpfen. Der Grund dafür ist, dass die erwünschten Eigenschaften der Streifensequenz sehr strenge Bedingungen voraussetzen, die global über die gesamte Sequenz erfüllt werden müssen. Diese Eigenschaft bewirkt, dass neue Farbkombinationen ab einer bestimmten Position nur noch sehr langsam zu finden sind. Es ist daher besser, bei jeder Iteration, die Farbkombinationen nach einem anderen Schema zu testen.

Beschleunigung mit Zufallszahlen

Aus obiger Beobachtung lässt sich schließen, dass die Initialisierung von $k(i) := 1$ und die inkrementelle Zuweisung einer neuen Farbkombination zu einer wiederholenden Struktur in der Sequenz führt, obwohl die Eigenschaften aus Gleichung 3.5 berücksichtigt werden. Dies führt zu einer raschen Verlangsamung des Findens neuer Farbkombinationen aufgrund des Rückverfolgungsprozesses. Eine asymptotische Abschätzung der Laufzeit würde nämlich einen exponentiellen Aufwand $O(2^{3KN})$ geben. Obwohl die Generierung des Streifenmusters ein Off-Line-Prozess darstellt und nur einmal durchgeführt werden muss, ist dieser Aufwand nicht handhabbar.

Um die Regelmäßigkeit, bei der Auswahl der Farbkombination in jedem Iterationsschritt, zu brechen, liegt es nahe, den vorgestellten Algorithmus durch einen probabilistischen Ansatz zu modifizieren. Indem wir bei der Initialisierung $k(i) := 1$ durch $k(i) := \text{random}$ ersetzen, verhindern wir, dass die Farbkombination $k = 1$ bevorzugt wird (siehe Abb. 3.7). Zusätzlich nehmen wir bei jedem Fehlschlag eines der Tests, ebenfalls einen zufälligen Wert anstelle einer Inkrementierung. Trotz einer probabilistischen Farbauswahl, muss die Erzeugung des Streifenmusters iterativ geschehen, da die Wahrscheinlichkeit, dass eine direkt erzeugte Zufallssequenz die erwünschten Bedingungen erfüllen, sehr gering ist. In der Praxis wird ein kurzes Zeitlimit gesetzt (wenige Sekunden). Wenn innerhalb dieser Zeitbegrenzung der Algorithmus kein Ergebnis liefert, startet das Programm neu. Da keine falschen Ausgaben produziert werden, handelt es sich hier um einen *Las-Vegas*-Algorithmus. In Tabelle 3.1 geben wir einen Überblick der erzeugten Streifenmuster mit den zugehörigen Eigenschaften.

K	r	N	probabilistisch
1	2	25	Nein
2	2	600	Nein
2	7	100	Ja
2	4	200	Ja
2	3	400	Ja
2	2	800	Ja
2	1	1625	Ja
3	2	3000	Ja

Tabelle 3.1: Eine Übersicht einiger erzeugter Streifenmuster mit den entsprechenden Eigenschaften. Bei steigendem r , wird es schwierig, längere Streifensequenzen zu erzeugen.

In [Zhang et al. '02] werden Streifensequenzen aus 125 Streifen mit nur einem Streifenbild projiziert, damit die De-Bruijn-Sequenzen innerhalb der 3. Ordnung bleiben. Um noch mehr Streifen erzeugen zu können, müsste auch hier die Ord-

nung erhöht werden, wobei der Grad an Mehrdeutigkeit ebenfalls steigen würde. Rusinkiewicz verwendete in [Rusinkiewicz '01] eindeutig kodierte Schwarz-Weiß-Streifenbilder aus 111 Streifen, die aus einem Graphen hergeleitet werden. Obwohl unsere Streifensequenzen gleichzeitig lokale und globale Eigenschaften besitzen, die Falschzuordnungen vermeiden können, haben wir in dieser Arbeit einen Algorithmus vorgestellt, der Streifenmuster erzeugt, die sowohl aus wenigen Aufnahmen K bestehen, als auch eine mehr als ausreichende Länge N besitzen. Außerdem ist die Robustheit durch die Eingabe von r beliebig skalierbar.

Während die Erzeugung von zufälligen Streifenmustern eine beliebte Methode ist [Zhang et al. '04, Morano et al. '98, Chen et al. '97], wurde noch unserem Wissen kein Verfahren vorgestellt, das einen probabilistischen Ansatz verwendet, um die Erzeugung von Streifenmustern mit globalen Einschränkungen zu beschleunigen. Wir werden im Folgenden mit einer Reihe von komplexen Objekten demonstrieren, dass die Zuverlässigkeit der Streifenzuordnung, sowie der Identifikation falsch erfasster Streifenübergänge, wesentlich höher ist als bei den Verfahren, die ausschließlich auf der lokalen räumlichen Kohärenzannahme basieren.

Kapitel 4

Streifenzuordnung

In diesem Kapitel werden die notwendigen Vorverarbeitungsschritte, Kantendetektionsmethoden, sowie ein neuartiges Zuordnungsverfahren behandelt.

4.1 Vorverarbeitung der Aufnahmen

Ein wichtiger Schritt, der noch vor der Streifensegmentierung, vorgenommen werden muss, ist die Vorverarbeitung der Aufnahmen. Aufgrund der zuvor präsentierten Unzulänglichkeiten (siehe Kap. 2 und Kap. 3), sind die reflektierten Streifen besonders schwer zu erkennen. Insbesondere stellen hochfrequente Schattierungen bei komplexen Objekten, sowie schlechte Reflexionseigenschaften und texturierte Objektoberflächen die größten Hindernisse dar. Ohne eine entsprechende Behandlung können nicht nur eine Reihe von falsch detektierten Streifen (bzw. Streifenübergängen) entstehen, sondern eine korrekte Zuordnung wäre erst gar nicht möglich. Dies kommt daher, dass es sich hier um eine Kamera-Projektor-Zuordnung (aktives Verfahren), im Gegensatz zu einer Kamera-Kamera-Zuordnung, handelt. Man kann sich vorstellen, dass der Projektor ebenfalls eine Kamera darstellt, die ideale Streifenfarben aufnimmt. Eine korrekte Zuordnung kann jedoch nur erfolgen, wenn die Farben von Projektor und Kamera stark miteinander korrelieren.

Ziel dieses Abschnitts ist es, die vorausgesetzten Scanbedingungen auszunutzen, um mittels Bildverarbeitungsverfahren brauchbare Aufnahmen zu erhalten, die für eine konsistente Kantendetektion und Zuordnung unabdingbar sind. Die Vorverarbeitungspipeline wird, wie wir es bereits in [Li '04b] vorgestellt haben, in drei Schritten durchgeführt. Zunächst wird das eigentliche Objekt vom Hintergrund extrahiert, um nur auf einem *verwendeten Bereich* (engl. *region of interest*) zu operieren. Anschließend wird eine Farbkorrektur durchgeführt, um die projizierten Farben wiederherzustellen, die in der Aufnahme stark abweichen können. Zum Schluss wird das Rauschen im Bildsignal reduziert, wobei zu beachten ist, dass da-

durch die reflektierten Streifenübergänge nur wenig beeinflusst werden sollen. Aufgrund der höher auflösenden Bildaufnahmen, die wir in dieser Arbeit verwenden, bleibt zusätzliches hochfrequenten Rauschen erhalten, das nicht unbedingt einem *weißen Gaußschen Rauschprozess* (engl. *white gaussian noise*) entstammt, wie etwa die bereits vorgestellten Hühnerdraht-Effekte. Vollständigkeitshalber werden wir einige bereits bekannte Bildverarbeitungsmethoden wiederholen, die auch in dieser Arbeit implementiert wurden. Insbesondere werden Vor- und Nachteile, sowie experimentelle Erkenntnisse diskutiert.

Vordergrundsegmentierung

Der erste Schritt besteht darin, eine gegebene Bildaufnahme eines Objekts, auf das die Streifen projiziert werden, von seinem Hintergrund und von durch Schatten verdeckten Bereichen zu trennen. Daraus entsteht ein verwendeter und ein nicht verwendeter Bereich. Durch diese Segmentierung können wir nicht nur die weiteren Verarbeitungsschritte beschleunigen (indem wir nur noch einen Teilbereich der Aufnahme betrachten), sondern zusätzlich verhindern, dass Streifenkanten in ungünstigen Bereichen detektiert werden. Wir werden allerdings sehen, dass der große Vorteil darin liegt, dass wir dadurch in der Lage sind, schwarze Streifen zu verwenden.



Abbildung 4.1: Vordergrundsegmentierung mit Hilfe einer weiß beleuchteten Aufnahme (links). Wir erhalten den verwendeten Bereich (Mitte), wobei der nicht verwendete Bereich in den weiteren Schritten ausmaskiert wird (rechts).

Die Vordergrund- und Hintergrundsegmentierung ist schon auf dem ersten Blick keine triviale Aufgabe. Sie spielt für viele Anwendungen im Bereich des Maschinensehens eine wichtige Rolle. Die Schwierigkeit besteht darin, durch wenige manuelle Eingaben, Objekte mit ähnlichen Konnektivitäts- und Farbeigenschaften vor einem komplexen Hintergrund zu extrahieren. Neulich wurden eine Reihe von effektiven Verfahren präsentiert, die auf energieminimierenden *Graphenschnitten*

basieren (siehe dazu [Rother et al. '04, Ishikawa & Geiger '98]). Diese sind jedoch vor allem für komplexere Vordergrundsegmentierungen gedacht, wobei entsprechende manuelle Eingriffe notwendig sind.

Stark vereinfacht wird, bei den aktiv-optischen Triangulierungsverfahren mit Projektor, oft eine zusätzliche Aufnahme gemacht, wobei man den Hintergrund bei einer der Aufnahmen extra beleuchtet. Der Hintergrund ergibt sich aus den hohen Intensitäten der Differenz beider Bilder.

In dieser Arbeit reduzieren wir das Problem, indem wir ganz einfach die Erfassungen in einem abgedunkelten Raum durchführen. Zusätzlich zu der Tatsache, dass dadurch keine anderen Lichtquellen als die des Projektors vorhanden sind, können wir sogar annehmen, dass alle dunklen Bereiche zu den nicht verwendeten Bereichen gehören. Schattenbereiche würden auf diese Weise ebenfalls heraussegmentiert.

Dunkle Bereiche werden durch niedrige Intensitäten repräsentiert. Um diese Segmentierung unabhängig von der Helligkeit der strukturiert beleuchteten Aufnahmen zu machen, projizieren wir ein zusätzliches uniform weißes Bild.

Durch Mittelung der Farbkanäle erhalten wir den Helligkeitswert $i(\mathbf{u}) \in [0, 1]$ eines Farbpixels an der Stelle $\mathbf{u} = [u v]^t$ des weißbeleuchteten Bilds. Anschließend filtern wir das Bild mit Hilfe eines skalaren Medianfilters, um vor allem die impulsartigen Artefakte zu beseitigen, die im Vergleich zur Mehrheit der anderen Pixel, einen hohen Intensitätswert besitzen. In der Praxis nehmen wir meistens ein etwas größeres Operatorfenster (7×7), da in diesem Schritt nur die Helligkeit von Interesse ist und nicht die Ausgabequalität der Medianfilterung. Durch die Verwendung der originalen Bildauflösung in den Aufnahmen, ist dieser Schritt tatsächlich notwendig, um die hohen Intensitäten der „hot pixel“ zu eliminieren (siehe Abb. 2.19 auf Seite 52). Wir erhalten einen zuverlässigen maximalen Intensitätswert $i_{\max}(\mathbf{u})$ und minimalen Intensitätswert $i_{\min}(\mathbf{u})$ des gesamten Bildes, wodurch wir ein kontrastnormalisiertes Bild

$$i_n(\mathbf{u}) = \frac{i(\mathbf{u}) - i_{\min}(\mathbf{u})}{i_{\max}(\mathbf{u}) - i_{\min}(\mathbf{u})} \quad .$$

berechnen können. Da kein Umgebungslicht vorhanden ist und wir eine begrenzte Belichtungszeit verwenden, setzen wir in der Praxis $i_{\min}(\mathbf{u}) = 0$. Danach muss nur aus diesem kontrastnormalisierten Helligkeitsbild, durch einen Schwellwert $\sigma \in [0, 1]$, der verwendete Bereich \mathcal{U}_{roi} herausextrahiert werden:

$$\mathbf{u} \in \mathcal{U}_{\text{roi}} \quad \Leftrightarrow \quad i_n(\mathbf{u}) > \sigma \quad .$$

In der Praxis verwenden wir $\sigma = 0,25$. Aufgrund der Tatsache, dass \mathcal{U}_{roi} unabhängig von den strukturiert beleuchteten Aufnahmen erzeugt wurde, können wir insbesondere auch schwarze Streifen projizieren, die, aufgrund ihres hohen Kontrasts zu den anderen Farbstreifen, für die Kantendetektion von Vorteil sind. In den

folgenden Abschnitten wird ohne besonderen Hinweis nur noch auf \mathcal{U}_{roi} operiert (siehe Abb. 4.1).

Farbfehlerkorrektur

In den Aufnahmen der strukturiert beleuchteten Ansichten, sind die projizierten Streifen meistens sehr schwer zu erkennen und weder für eine Kantendetektion noch für eine Zuordnung angemessen. Eine Kantendetektion würde z. B. Kanten finden, die nicht zu den projizierten Streifenübergängen gehören. Unbekannte Reflexionseigenschaften des Objektmaterials und farbige Texturen, sowie ein colorimetrisch unkalibriertes Scansystem haben zur Folge, dass die erfassten Farben nicht denen der projizierten Streifen entsprechen. Außerdem entstehen durch *Farbübersprechen* (engl. *color crosstalk*) zusätzliche Texturfarben und Schattierungen der Objektoberfläche. Es liegt nahe, eine entsprechende Korrektur, wie in Abb. 4.2, durchzuführen.

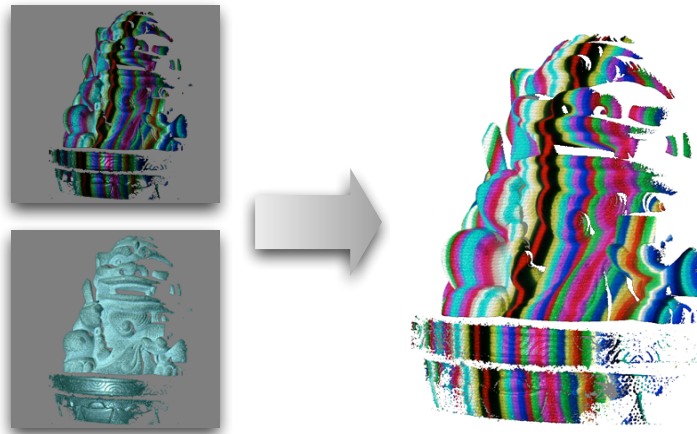


Abbildung 4.2: Mit Hilfe der weiß beleuchteten Aufnahme können wir eine Farbkorrektur durchführen.

Von den Reflexionsannahmen, die in Kap. 3 besprochen wurden, ausgehend, zeigen wir, wie ein linearer Zusammenhang zwischen der Lichtintensität des projizierten und des erfassten Lichts zustande kommt. Insbesondere zeigen wir, dass es möglich ist, durch die Wiederverwendung der mit weißem Licht projizierten Aufnahme, die Farben der projizierten Streifen zu rekonstruieren. Die Annahme, dass die Oberfläche des Objekts nicht farbig ist und keine Texturen enthält, ist ebenfalls nicht notwendig.

Dazu gehen wir zurück auf das Projektor-Kamera-Modell von [Caspi et al. '98]. Während Caspi et al. von einem traditionellen LCD-Projektor ausgehen, verwenden wir, einen DLP-Projektor, der in Kap. 2 ausführlich behandelt wurde. Dadurch, dass unser Ein-Chip-DLP-Projektor zur Farbdarstellung eine Modulation der Impulsbreite verwendet, werden die reflektierten RGB-Farbsignale über die

Zeit auf dem CCD der Kamera aufsummiert. Durch diese Beobachtung können wir, ohne zusätzliche Betrachtung der Zeitkomponente, die Modulation der Wellenlänge auf die Modulation der Impulsbreite übertragen. Dazu sei auf der Seite des Projektors $\mathbf{c}^{(p)} = [c_r^{(p)} c_g^{(p)} c_b^{(p)}]^t \in \{0, 1\}^3$ die Farbe eines Referenzpunkts des projizierten Streifenmusters.

Die Beleuchtungsintensität des roten Farbkanals in diesem Referenzpunkt ist gegeben durch $I_r^t(\lambda) \cdot P(c_r^{(p)})$, wobei $I_r^t(\lambda)$ die normalisierte Spektralverteilung eines projizierten roten Lichtstrahls über die Wellenlänge λ ist und $P(c_r^{(p)})$ ein monotoner nicht-linearer Faktor, der durch internes Farbübersprechen des Projektors entsteht. Die Spektralverteilung, die die Form eines Zapfens annimmt, wird normalisiert, um beispielsweise für den roten Lichtstrahl $I_r^t(\lambda_r) = 1$ zu erhalten, wobei $\lambda_r = 590$ nm. Entsprechend seien die grünen und blauen Spektralverteilungen definiert durch $I_g^t(\lambda)$ und $I_b^t(\lambda)$, sowie die zugehörigen Faktoren durch $P(c_g^{(p)})$ und $P(c_b^{(p)})$. Durch Modulation der Impulsbreite, werden die Beleuchtungsintensitäten aller Farbkanäle über die Zeit aufsummiert, wodurch sich folgende Beleuchtungsfunktion ergibt:

$$\begin{aligned} I^t(\lambda) &= I_r^t(\lambda) \cdot P(c_r^{(p)}) + I_g^t(\lambda) \cdot P(c_g^{(p)}) + I_b^t(\lambda) \cdot P(c_b^{(p)}) \\ &= \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}^t \begin{bmatrix} I_r^t(\lambda) \cdot P(c_r^{(p)}) \\ I_g^t(\lambda) \cdot P(c_g^{(p)}) \\ I_b^t(\lambda) \cdot P(c_b^{(p)}) \end{bmatrix} . \end{aligned}$$

Mit der Annahme, dass es sich bei der Reflexion auf dem Objekt um eine Lambertsche Reflexion handelt und dass sich die Oberfläche nicht gegenseitig beleuchtet, entsteht ein kanalunabhängiger linearer Zusammenhang zwischen der Beleuchtungsintensität $I^t(\lambda)$ und der reflektierten Lichtintensität $I^r(\lambda)$.

$$I^r(\lambda) = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}^t K \begin{bmatrix} I_r^t(\lambda) \cdot P(c_r^{(p)}) \\ I_g^t(\lambda) \cdot P(c_g^{(p)}) \\ I_b^t(\lambda) \cdot P(c_b^{(p)}) \end{bmatrix} ,$$

wobei

$$K = \begin{bmatrix} k_r & 0 & 0 \\ 0 & k_g & 0 \\ 0 & 0 & k_b \end{bmatrix} ,$$

auch *Albedo-Matrix* genannt wird. K beschreibt die Reflektivität der Oberfläche.

Nachdem das reflektierte Licht durch den Kamera-CCD erfasst wird, erhalten wir die entsprechende Antwort $\mathbf{c}^{(c)} = [c_r^{(c)} c_g^{(c)} c_b^{(c)}]^t$ des Kamerapixels:

$$\mathbf{c}^{(c)} = \begin{bmatrix} \int_0^\infty f_r(\lambda) I^r(\lambda) d\lambda \\ \int_0^\infty f_g(\lambda) I^g(\lambda) d\lambda \\ \int_0^\infty f_b(\lambda) I^b(\lambda) d\lambda \end{bmatrix},$$

wobei $f_r(\lambda)$, $f_g(\lambda)$ und $f_b(\lambda)$ die spektralen Antworten des Farbfilterungsprozesses der Kamera modellieren.

Indem wir zusätzlich eine ambiente Lichtintensität $\mathbf{c}^{\min} = [c_r^{\min} c_g^{\min} c_b^{\min}]^t$ in der Kameraaufnahme berücksichtigen, können wir die folgende Gleichung aufstellen:

$$\mathbf{c}^{(c)} = AK \begin{bmatrix} P(c_r^{(p)}) \\ P(c_g^{(p)}) \\ P(c_b^{(p)}) \end{bmatrix} + \mathbf{c}^{\min}, \quad (4.1)$$

wobei

$$A = \begin{bmatrix} \int f_r(\lambda) I_r^t(\lambda) d\lambda & \int f_r(\lambda) I_g^t(\lambda) d\lambda & \int f_r(\lambda) I_b^t(\lambda) d\lambda \\ \int f_b(\lambda) I_r^t(\lambda) d\lambda & \int f_b(\lambda) I_g^t(\lambda) d\lambda & \int f_b(\lambda) I_b^t(\lambda) d\lambda \\ \int f_g(\lambda) I_r^t(\lambda) d\lambda & \int f_g(\lambda) I_g^t(\lambda) d\lambda & \int f_g(\lambda) I_b^t(\lambda) d\lambda \end{bmatrix}$$

die *Kamera-Projektor-Kopplungsmatrix* (engl. *projector-camera coupling matrix*) darstellt. Wir haben somit durch die obige Gleichung eine direkte Beziehung zwischen einer projizierten Farbe $\mathbf{c}^{(p)}$ und einer erfassten Farbe $\mathbf{c}^{(c)}$ hergestellt.

Die Aufgabe besteht nun darin, aus dem gestörten $\mathbf{c}^{(c)}$ das zugehörige projizierte $\mathbf{c}^{(p)}$ zu gewinnen. Insbesondere können wir die ambiente Lichtintensität \mathbf{c}^{\min} durch eine uniforme Schwarzbild-Beleuchtung des Projektors erhalten. Die Faktoren des internen Farbübersprechens des Projektors P sind, wie bereits angesprochen, nicht linear und daher nicht invertierbar. Diese werden üblicherweise durch einen Off-Line-Farbkalibrierungsprozess vorberechnet und mittels einer Nachschlagetabelle zur Laufzeit ermittelt.

In dieser Arbeit haben wir das Problem durch die Annahme, dass der Projektor eine ideale Farbtreue besitzt, d. h. $P = \text{id}$, vereinfacht.

Es ist nun möglich, ohne Kenntnis von A und K , die Farbe $\mathbf{c}^{(p)}$ zu bekommen, indem wir die bereits mit uniformem weißem Licht projizierte Aufnahme verwenden. Für eine weiße Beleuchtung gilt nämlich bei der Projektorausgabe:

$$\begin{bmatrix} P(c_r^{(p)}) \\ P(c_g^{(p)}) \\ P(c_b^{(p)}) \end{bmatrix} = \begin{bmatrix} c_r^{(p)} \\ c_g^{(p)} \\ c_b^{(p)} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} =: \mathbf{c}^{(w)}.$$

Daraus ergibt sich die folgende Antwort des Kamerapixels:

$$\mathbf{c}^{\max} = \begin{bmatrix} c_r^{(\max)} \\ c_g^{(\max)} \\ c_b^{(\max)} \end{bmatrix} = AK\mathbf{c}^{(w)} + \mathbf{c}^{\min} \quad . \quad (4.2)$$

Letztendlich können wir durch Substitution von AK aus Gleichung 4.1 und 4.2 die gewünschte projizierte Farbe gewinnen:

$$\mathbf{c}^{(p)} = \left[\frac{c_r^{(c)} - c_r^{\min}}{c_r^{\max} - c_r^{\min}}, \frac{c_g^{(c)} - c_g^{\min}}{c_g^{\max} - c_g^{\min}}, \frac{c_b^{(c)} - c_b^{\min}}{c_b^{\max} - c_b^{\min}} \right]^t \quad .$$

Es stellt sich nun heraus, dass es sich hier nur noch um eine einfache Normalisierung der projizierten Farbe bezüglich einer Beleuchtung mit weißem und schwarzem Licht handelt. Bei unseren Experimenten konnten wir zusätzlich feststellen, dass bei der Projektion des schwarzen Lichts, kein ambientes Licht vorhanden ist, d. h. $\mathbf{c}^{\min} = \mathbf{o}$ für jedes Kamerapixel. Insgesamt ist es also nur noch notwendig, eine einzige zusätzliche Aufnahme zu den strukturiert beleuchteten Aufnahmen zu machen, nämlich eine mit weißer Beleuchtung (siehe Abb. 4.2).

Im Gegensatz zu [Zhang et al. '02], handelt es sich hierbei um eine On-Line-Farbkorrektur, da wir für jede Erfassung diese zusätzliche Aufnahme benötigen. Dadurch, dass wir in jedem Kamerapixel unabhängig von der Geometrie bzw. des Einfallswinkels des Projektorlichtstrahls auf die Objektoberfläche, die Matrix AK exakt bestimmen können, ist die Farbkorrektur besonders robust gegenüber Farbverfälschungen aufgrund von Schattierungen. Außerdem wird die weiße Beleuchtung ohnehin für die Vordergrundsegmentierung und für die Texturrekonstruktion verwendet, weshalb die Erfassungsgeschwindigkeit durch diesen Schritt nicht beeinträchtigt wird.

Rauschunterdrückungsmethoden

Obwohl wir auf unkomprimierten Bildern operieren und die oben behandelte Farbnormalisierung durchgeführt haben, sind immer noch hochfrequente Störungen vorhanden. Diese Störungen sind insbesondere ein Hindernis bei der Kantendetektion, da wir nur die Kanten finden wollen, die auch zu den projizierten Farbstreifen gehören.

Ursache dafür sind vor allem Objektoberflächen, deren nichtlineare Reflexion zu stark vom Lambertschen Reflexionsmodell abweichen. Wenn diese Oberflächen zusätzlich texturiert sind, kann auch die Farbnormalisierung keine vollständige Unterdrückung der Farbeigenschaften der Oberfläche erzielen. Das Hauptproblem liegt allerdings bei den, im Kap. 2 eingeführten, Hühnerdraht-Effekten (siehe Abb. 4.3

und Abb. 4.4a). Diese gitterartigen Strukturen werden von den Abständen zwischen den DMD-Spiegeln des DLP-Projektors hervorgerufen. Dadurch, dass wir die volle Bildauflösung der Kamera verwenden und keine interpolierten verkleinerten Bilder, sind diese auch in den Aufnahmen vorhanden.

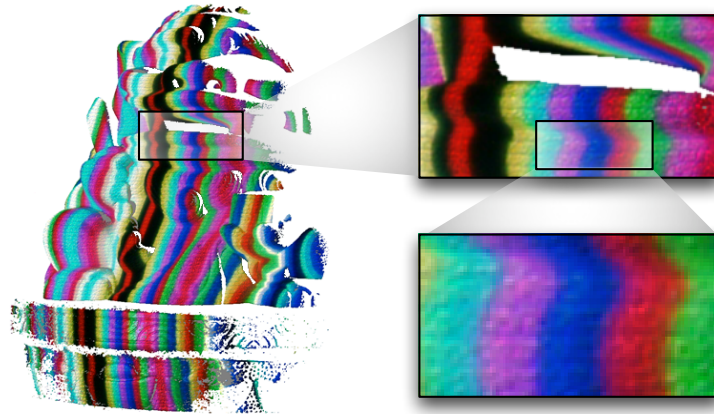


Abbildung 4.3: Eine Vergrößerung zeigt, dass hochfrequentes Rauschen im normalisierten Bild vorhanden ist. Die gitterartige Struktur wird vom Hühnerdraht-Effekt des Projektors verursacht.

Wir zeigen zunächst, dass lineare Glättungsoperatoren für unsere Anwendung unangemessen sind, bevor wir einige nicht lineare Glättungsverfahren präsentieren. Insbesondere geben wir einen kurzen Überblick über diese Verfahren und präsentieren deren Vor- und Nachteile, die für uns eine Rolle spielen. Für einen ausführlichen Einblick in die diskrete Signalverarbeitung verweisen wir auf [Oppenheim et al. '99].

Der naive Ansatz, Rauschen in einem diskreten Bildsignal zu unterdrücken, besteht zunächst darin, vorauszusetzen, dass das Rauschen in jedem Pixel statistisch unabhängig voneinander ist (stationärer Rauschprozess). Weiterhin wird angenommen, dass es sich um einen weißen Gaußschen Rauschprozess handelt. Ebenso ist es klar, dass die Spektraldichtefunktion eines verrauschten diskreten Grauwertbildsignals $f(\mathbf{u}) \in [0, 1]$ Werte ungleich Null in hochfrequenten Bereichen besitzt. Sei nun $\hat{f}(\boldsymbol{\nu}) = \text{DFT}\{f(\mathbf{u})\}$ die zweidimensionale diskrete Fouriertransformation von $f(\mathbf{u})$. Ein Tiefpassfilter mit einer zweidimensionalen Gaußfunktion $\hat{h}(\boldsymbol{\nu})$ würde dann die Amplituden der hohen Frequenzen mit $\hat{g}(\boldsymbol{\nu}) = \hat{h}(\boldsymbol{\nu})\hat{f}(\boldsymbol{\nu})$ reduzieren. Daraus ergibt sich im Ortsbereich, durch eine diskrete Faltung, das geglättete Bild $g(\mathbf{u}) = h(\mathbf{u}) * f(\mathbf{u})$, wobei $h(\mathbf{u})$ die Filterfunktion ist, die durch eine diskrete approximierte Gaußfunktion gegeben wird. Insbesondere stellt $h(\mathbf{u})$ bei unserer Implementierung ein *Filter mit endlicher Impulsantwort* (engl. *finite impulse response filter*) dar, da wir bei einer beschränkten Eingabe auch eine beschränkte Ausgabe erhalten. Daraus ergibt sich ein linearer Glättungsoperator mit einer Filterfunktion

$h(\mathbf{u})$, die üblicherweise durch die normalisierte Impulsantwortmatrix (Filtermaske) H beschrieben wird. Eine 3×3 -Maske kann z. B. durch $H = \frac{1}{16}[1\ 2\ 1]^t[1\ 2\ 1]$ gegeben werden.

Diese linearen Glättungsverfahren führen in unserem Fall zu zwei Problemen. Das erste Problem liegt in der Linearität des Filters. Sei $f_i(\mathbf{u})$ das ideale störungsfreie Bild von $f(\mathbf{u})$. Dann können wir das Rauschen $r(\mathbf{u})$, das vor allem durch die Hühnerdraht-Effekte entsteht, durch einen additiven Prozess modellieren, so dass $f(\mathbf{u}) = f_i(\mathbf{u}) + r(\mathbf{u})$ gilt. Im Frequenzbereich würde somit aus $\hat{f}(\boldsymbol{\nu}) = \hat{f}_i(\boldsymbol{\nu}) + \hat{r}(\boldsymbol{\nu})$ die tiefpassgefilterte Fouriertransformation $\hat{g}(\boldsymbol{\nu}) = \hat{h}(\boldsymbol{\nu})\hat{f}_i(\boldsymbol{\nu}) + \hat{h}(\boldsymbol{\nu})\hat{r}(\boldsymbol{\nu})$ entstehen. Es werden somit nicht nur die Störungen geglättet sondern, auch das ideale Bild, das die strukturierten Streifen enthält (siehe Abb. 4.4b). Das zweite Problem entsteht durch die Betrachtung der Farbkanäle. Dadurch, dass eine Faltung des Bildes mit der Impulsantwortmatrix eine gewichtete Mittelung einer lokalen Pixelnachbarschaft ist, werden bei Streifenübergängen die Farben vermischt. Es entstehen neue Farben, wodurch eine korrekte Zuordnung der Streifenfarben nicht mehr möglich ist. Obwohl wir später sehen werden, dass wir eigentlich gar nicht die lokalen Streifenübergangsbereiche für die Zuordnung verwenden, ist die Korrektheit der Farben trotzdem wichtig für einen späteren Ballungsprozess.

Eine Möglichkeit, die strukturiert beleuchteten Aufnahmen so zu glätten, dass die Kanten erhalten bleiben, wäre mit Hilfe eines so genannten *Bilateralfilters* [Tomasi & Manduchi '98]. Bei diesen Filtern wird zusätzlich zu der geometrischen Nachbarschaft, auch die photometrische Nachbarschaft betrachtet. Dies bedeutet, dass die gewichtete Mittelung sich nicht nur auf die räumliche Lokalität bezieht, sondern auch auf deren Intensitätswerte, wodurch eine schwächere Glättung an Streifenübergängen stattfinden würde, da hier ein starker Farbkontrast vorhanden ist. Das geglättete Bild wäre in diesem Fall

$$g(\mathbf{u}) = (h(\mathbf{u}) \cdot s(f(\mathbf{u}))) * f(\mathbf{u}) \quad ,$$

wobei $s(f(\mathbf{u}))$ die Filterfunktion ist, die die Ähnlichkeit zweier Pixelwerte über eine Nachbarschaft beschreibt. Dadurch, dass die photometrische Lokalität während der Laufzeit nicht direkt gegeben ist, wird sie üblicherweise durch eine Nachschlagetabelle implementiert. Während der Bilateralfilter auf Schwarz-Weiß-Bildern gute Ergebnisse liefert, entstehen bei farbigen Streifenübergängen dieselben Probleme wie bei der linearen Glättung, da bei einem Farbübergang unterschiedlich stark korrelierende Intensitäten in den Farbkanälen auftauchen können.

Eine andere Art von nichtlinearen Glättungsoperatoren sind die *Medianfilter*, die ebenfalls eine kantenerhaltende Eigenschaft besitzen. Bei einem Grauwertbild $f(\mathbf{u})$, sei $\mathcal{H}(\mathbf{u})$ das $N \times N$ -Operatorfenster des skalaren Medianfilters, welches die Menge aller Intensitäten einer lokalen Nachbarschaft von \mathbf{u} angibt. Es gilt für ungerade N :

$$f(\mathbf{u}') \in \mathcal{H}(\mathbf{u}) \quad \Leftrightarrow \quad |u' - u| \leq \frac{N-1}{2} \quad \text{und} \quad |v' - v| \leq \frac{N-1}{2} \quad ,$$

wobei $\mathbf{u} = [u, v]$ und $\mathbf{u}' = [u', v']$. Die Ausgabe des skalaren Medianfilters an der Stelle \mathbf{u} ist somit der Medianwert $\bar{f}(\mathbf{u})$ des Operatorfensters $\mathcal{H}(\mathbf{u})$. Insbesondere entstehen bei einem skalaren Medianfilter keine neuen Intensitätswerte, da $\bar{f}(\mathbf{u}) \in \mathcal{H}(\mathbf{u})$.

Die Erweiterung des Medianfilters auf Farbbilder, stellt eine zusätzliche Schwierigkeit dar, da eine gewisse Ordnung für Farbvektoren definiert werden muss. Dazu wird der Vektormedianfilter folgendermaßen definiert. Sei

$$\mathbf{f}(\mathbf{u}) = [f_r(\mathbf{u}) \ f_g(\mathbf{u}) \ f_b(\mathbf{u})]^t \in [0, 1]^3$$

ein Farbwert und $\mathcal{H}(\mathbf{u}) = \{\mathbf{f}_1, \dots, \mathbf{f}_{N^2}\}$ ein Operatorfenster, das die Menge der benachbarten Farbvektoren darstellt. Die Ausgabe des Vektormedianfilters wäre in diesem Fall:

$$\bar{\mathbf{f}}(\mathbf{u}) = \mathbf{f}_{j_{\min}} \quad , \quad j_{\min} = \arg \min_j \left\{ \sum_{i=1}^{N^2} \|\mathbf{f}_j - \mathbf{f}_i\| \right\} \quad .$$

Zusätzlich können Gewichte w_i zugewiesen werden durch

$$j_{\min} = \arg \min_j \left\{ \sum_{i=1}^{N^2} w_i \|\mathbf{f}_j - \mathbf{f}_i\| \right\} \quad .$$

Obwohl bei der Ausgabe keine neue Farben hinzukommen, haben wir bei unseren Tests festgestellt, dass schon bei kleinen Operatorfenstern und bei unterschiedlichen Gewichtungen, eine Vergrößerung des Bildes entsteht (siehe Abb. 4.4c). Dadurch weichen die Streifenübergänge stark von ihren tatsächlichen Positionen ab, wodurch eine verrauschte Punktwolke nach der optischen Triangulierung entstehen würde.

Deshalb verzichten wir auf die Voraussetzung, dass keine neue Farben hinzukommen dürfen, und wenden einfach den skalaren Medianfilter auf jeden Farbkanal unabhängig voneinander an. Daraus ergibt sich

$$\bar{\mathbf{f}}(\mathbf{u}) = \begin{bmatrix} \bar{f}_r(\mathbf{u}) \\ \bar{f}_g(\mathbf{u}) \\ \bar{f}_b(\mathbf{u}) \end{bmatrix} \quad .$$

Überraschenderweise war dieser einfache Medianfilter bereits in der Lage mit kleinen Operatorfenstern ($N = 3$) alle gitterartigen Strukturen, die aufgrund des Hühnerdraht-Effekts auftreten, zu eliminieren, ohne die strukturierten Streifenbilder zu verzerren (siehe Abb. 4.4d). Auch die Farbkontraste zwischen den Streifenübergängen konnten erhalten bleiben. Wir bemerken, dass es sich hierbei um denselben Medianfilter handelt, der auch in Adobe Photoshop 8.0 [Adobe '03] implementiert wird. Obwohl in der Literatur eine Reihe von weiteren Medianfiltern auf Vektoren vorgeschlagen werden [Koschan & Abidi '01], haben wir in dieser Arbeit keine weiteren Filter untersucht, da Letzterer alle unsere Ansprüche befriedigen konnte.

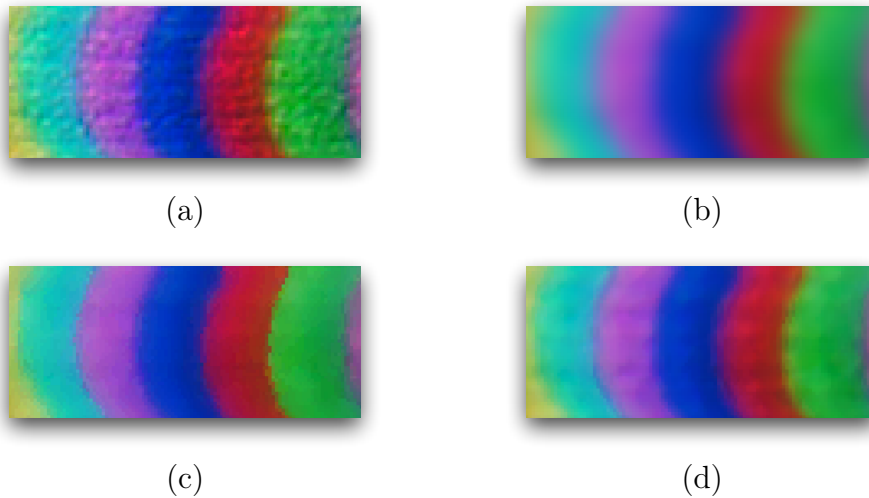


Abbildung 4.4: (a) Originalaufnahme. (b) Gaußfilterung. (c) Vektormedianfilterung. (d) Skalärer Medianfilter angewendet auf jedem Farbkanal.

4.2 Kantendetektion

Wir haben im vorherigen Abschnitt eine Vielzahl von Vorverarbeitungsmethoden vorgestellt. Diese sind notwendig, um möglichst frühzeitig in der Rekonstruktionspipeline, mit Hilfe von Bildverarbeitungsverfahren, so viele Störungen wie möglich zu beseitigen. Insbesondere haben wir gezeigt, dass es wichtig ist, die zusätzliche Aufnahme mit weißer Beleuchtung intensiv auszunutzen, sowie Annahmen voranzusetzen, die auch genau unserer Aufgabenstellung entsprechen.

Da die Identifikation unserer Streifen über die Streifenübergänge geschieht (siehe Kap. 3), liegt es nun nahe, mit Hilfe von Kantendetektionsverfahren, die Streifenübergänge in den Kameraaufnahmen zu lokalisieren, bevor wir mit der Zuordnung beginnen.

Aufgrund der Epipolareinschränkung und der Annahme, dass die projizierten Streifen einigermassen vertikal orientiert sind, können wir uns auf eine eindimensionale Kantendetektion beschränken, die nicht nur die Aufgabe vereinfacht, sondern in der Regel auch zu einer robusteren Segmentierung führt. Der Grund dafür ist, dass die Kontraste der vertikalen Streifenübergänge in horizontaler Richtung verlaufen und eine zweidimensionale Kantendetektion zusätzliche Merkmale finden würde, die hohe Gradientennormen in nicht horizontaler Richtung aufweisen.

Wir werden in diesem Abschnitt zunächst den Kantendetektor auf Farbbildern beschreiben, die die eindimensionale Variante von [Cumani '91] darstellt. Anschließend wird ein Schritt vorgestellt, der die Position der Streifenübergänge mit Subpixel-Genauigkeit approximiert. Weiterhin zeigen wir, wie wir mittels einer selektiven Kantensegmentierung, unter gewissen Voraussetzungen, genau die Kanten

extrahieren, die auch zu den projizierten Streifenübergängen gehören. Dieses Verfahren haben wir in [Li et al. '04] bereits präsentiert und signifikante Verbesserungen demonstriert. Dadurch, dass wir mehrere strukturiert beleuchtete Aufnahmen haben (i. d. R. zwei), befinden sich die Kanten desselben Streifenübergangs bei jeder Aufnahme nicht immer exakt an derselben Stelle. Wir zeigen zum Schluss, wie wir die detektierten Kanten aus mehreren Aufnahmen integrieren.

Multispektraler Kantendetektor

Es wurde bereits eine Reihe von verschiedenen Kantendetektoren entwickelt [Faugeras '93], welche meistens auf Grauwertbilder operieren. Die Umwandlung unserer Farbaufnahmen in ein Grauwertbild wäre nicht sinnvoll, da bei zwei benachbarten Streifen mit unterschiedlichen Farben und gleicher Helligkeit (z. B. Blau und Rot), in einem Grauwertbild, kein Streifenübergang mehr vorhanden ist. Eine Studie von Novak und Shafer [Novak & Shafer '87], hat gezeigt, dass etwa 10 % der Kanten dadurch verfehlt werden. Dies ist bei der gezeigten Verwendung von farbkorrigierten Bildern umso schlimmer, da im Idealfall alle projizierten Farben gleicher Helligkeit auch zu erfassten Farben gleicher Helligkeit führen.

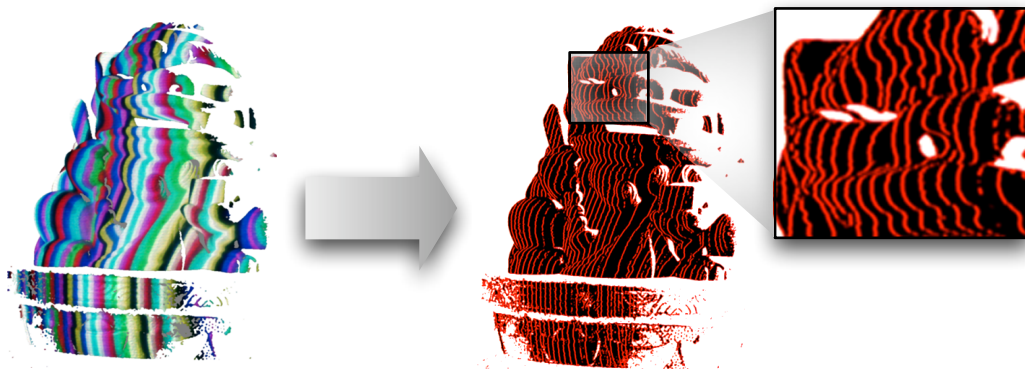


Abbildung 4.5: Durch die Kantendetektion werden die Streifenübergänge lokalisiert.

Eine Möglichkeit dies zu umgehen besteht darin, wie in [Koschan '95] erwähnt wurde, in allen Farbkanälen unabhängig voneinander die Kanten zu detektieren, um sie anschließend mit einer ODER-Operation zu kombinieren. Ein Problem hat dieses Verfahren allerdings bei Kanten, die in mehreren Farbkanälen detektiert werden, aber sich nicht exakt an derselben Stelle befinden.

Mit der zusätzlichen Betrachtung von Farbvariationen, definiert Cumani in [Cumani '91], eine multispektrale Kante als ein lokales Maximum der *Summe der quadrierten Kontraste* (engl. *square local contrast*) über alle Farbkanäle, wobei dies nichts anderes ist, als eine Betrachtung des quadrierten Abstandes zwischen zwei Farbwerten im dreidimensionalen RGB-Farbraum. Wie auch in [Zhang et al. '02],

verwenden wir diesen Ansatz für eine Kantendetektion auf den strukturiert beleuchteten Aufnahmen. Dazu sei $\mathbf{f}(u) = [f_r(u) f_g(u) f_b(u)]^t \in [0, 1]^3$ der Farbvektor einer Zeile an der Stelle $u \in \mathbb{N}$. Wir haben in der Praxis festgestellt, dass eine eindimensionale Prewitt-Faltungsmaske zur Bestimmung der diskreten Ableitung (bzw. des Kontrasts) eine besonders gute Wahl ist. Sie ist gegeben durch $\frac{1}{2}[1 \ 0 \ -1] = \frac{1}{2}[1 \ 1] * [1 \ -1]$, was eine Komposition aus einem Mittelungsfiler und einem Vorwärtsdifferenz-Operator ist. Die Mittelung ist wichtig, da die diskreten Signale nicht glatt sind und wegen der Tatsache, dass der Vorwärtsdifferenz-Operator nur eine approximierte Ableitung darstellt. Daraus ergibt sich die gemittelte diskrete Ableitung von $\mathbf{f}(u)$:

$$\bar{\Delta}\mathbf{f}(u) := \frac{1}{2} \begin{bmatrix} f_r(u) * [1 \ 0 \ -1] \\ f_g(u) * [1 \ 0 \ -1] \\ f_b(u) * [1 \ 0 \ -1] \end{bmatrix} = \frac{1}{2} (\mathbf{f}(u+1) - \mathbf{f}(u-1)) \quad .$$

Hieraus entsteht sofort die Summe der quadrierten Kontraste $s(u)$:

$$s(u) = \bar{\Delta}\mathbf{f}(u)^t \bar{\Delta}\mathbf{f}(u) = \frac{1}{4} ((\bar{\Delta}f_r(u))^2 + (\bar{\Delta}f_g(u))^2 + (\bar{\Delta}f_b(u))^2) \quad .$$

Wir bemerken, dass diese Kontrastbewertung auch für Grauwertbilder gültig ist. Insbesondere würde $s(u) = (\bar{\Delta}f(u))^2$ gelten, was nach der Festlegung eines Schwellwerts ebenfalls zu einer eindimensionalen Kantendetektion benutzt werden kann.

Um keine Kanten zu detektieren, die zwar nach der ersten Ableitung einen hohen Kontrast besitzen, aber kein lokales Maximum aufweisen, müssen wir ähnlich wie beim *Marr-Hildreth*-Operator (Laplace-of-Gaussian-Verfahren) feststellen, ob in der zweiten Ableitung ein Nulldurchgang existiert. Es wird trotzdem ein Schwellwert σ gesetzt, um zu überprüfen, ob eine potentielle Kante vorhanden ist oder nicht. Falls $s(u) > \sigma$ und ein Nulldurchgang in der Ableitung der Farbwerte vorhanden ist, dann wird eine Kante an der Stelle u detektiert.

Im Unterschied zur obigen Definition der diskreten Ableitung, verwenden wir für die zweite Ableitung eine abgewandelte Form (siehe Abb. 4.6). Grund dafür ist, dass, für die Bestimmung von $s(u)$, $u-1$ bis $u+1$ benötigt wird, wodurch für eine Ableitung von s an der Stelle u , $u-2$ bis $u+2$ benötigt würde. Entsprechend bräuchte man für die Ableitung von s an der Stelle $u+1$, $u-1$ bis $u+3$, also insgesamt $u-2$ bis $u+3$. Während eine Überprüfung, ob es sich um einen Nulldurchgang handelt, unproblematisch ist, haben wir keinen exakten Wert zwischen $s(u+1)$ und $s(u)$ für einen Vergleich mit einem Schwellwert. Deswegen vergleichen wir folgendermaßen die Vorwärtsdifferenzen $\Delta s(u-1) = s(u) - s(u-1)$ und $\Delta s(u) = s(u+1) - s(u)$, ob sie einen Nulldurchgang in der zweiten Ableitung darstellen:

$$|\Delta s(u-1) + \Delta s(u)| = |\Delta s(u-1)| + |\Delta s(u)| \Leftrightarrow \text{sign}(\Delta s(u-1)) = \text{sign}(\Delta s(u)) \quad .$$

Falls $\text{sign}(\Delta s(u-1)) \neq \text{sign}(\Delta s(u))$ und $s(u) > \sigma$, dann wird eine Kante an der Stelle u detektiert. Außerdem brauchen wir in diesem Fall insgesamt nur die Farbwerte an den Stellen $u-2$ bis $u+2$, was zusätzlich einen kleinen Vorteil für dicht beieinanderliegende Streifenübergänge bietet.

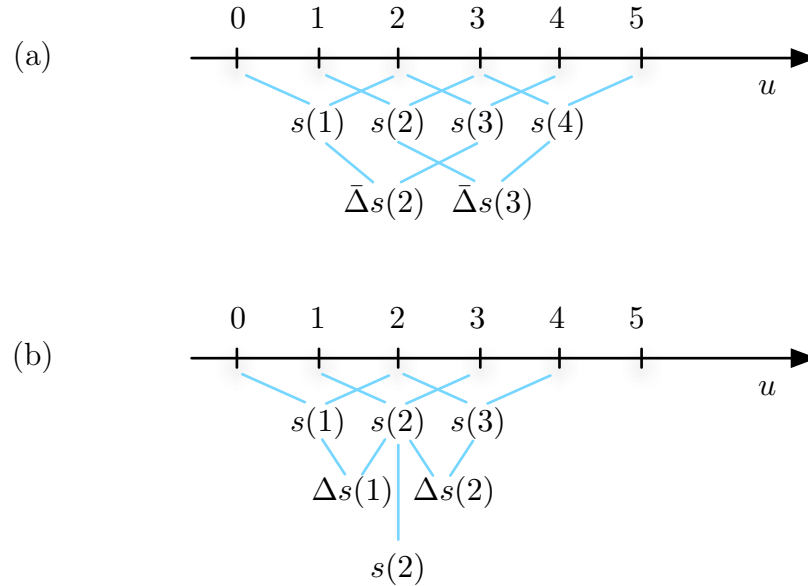


Abbildung 4.6: Auswertungsschema der multispektralen Kantendetektion. (a) Auswertung der zweiten Ableitung durch die zuvor definierte diskrete Ableitung $\bar{\Delta}$: Hier ist kein exakter Wert zwischen $s(1)$ und $s(4)$ für den Schwellwertvergleich vorhanden. (b) Abgewandelte Auswertung: Hier wird die Vorwärtsdifferenz Δ bei der zweiten Ableitung verwendet. $s(2)$ ist in diesem Beispiel der exakte Wert zwischen $s(1)$ und $s(3)$.

Subpixelgenaue Lokalisierung

Die Lokalisierung der Streifenübergänge auf dem Kamerabild, dient nach dem Zuordnungsschritt der optischen Triangulierung, wodurch aus jedem zugeordneten Streifenübergang ein Punkt im Raum entsteht. Jedes Kamerapixel wird durch die Position seines Zentrums dargestellt. Daraus ergibt sich $x = u + \frac{1}{2}$ als die Position des u -ten Pixels auf einer Zeile der Kamerabildebene. Weil u in \mathbb{N} liegt, kann folglich nur eine pixelgenaue Triangulierung stattfinden. Dies äußert sich in einem treppenförmigen Verlauf entlang der zu einem Streifenübergang gehörenden Punkte, wenn die erzeugte Punktwolke entsprechend vergrößert wird (siehe Abb. 4.7b).

Für hochgenaue Erfassungen verwenden Curless [Curless '97] und Zhang et al. [Zhang et al. '02] ein Raum-Zeit-Analyse-Verfahren, auf Kosten von mehreren zusätzlichen Aufnahmen. Da wir in dieser Arbeit von nur zwei strukturiert beleuchteten Aufnahmen ausgehen, approximieren wir die subpixelgenaue Lokalisierung

eines Streifenübergangs, in jeder Aufnahme unabhängig voneinander, anhand der zuvor ermittelten Summe der quadrierten Kontraste.

Inspiziert von der linearen Interpolationsmethode von [McIvor & Valkenburg '97], bestimmen wir die neue subpixelgenaue Position einer detektierten Kante mit

$$x = \frac{(u + 1) \cdot (s(u) - s(u - 1)) - u \cdot (s(u + 1) - s(u))}{2s(u) - s(u + 1) - s(u - 1)} .$$

Wir erzielen somit, wie in Abb. 4.7a dargestellt wird, eine höhere Genauigkeit.

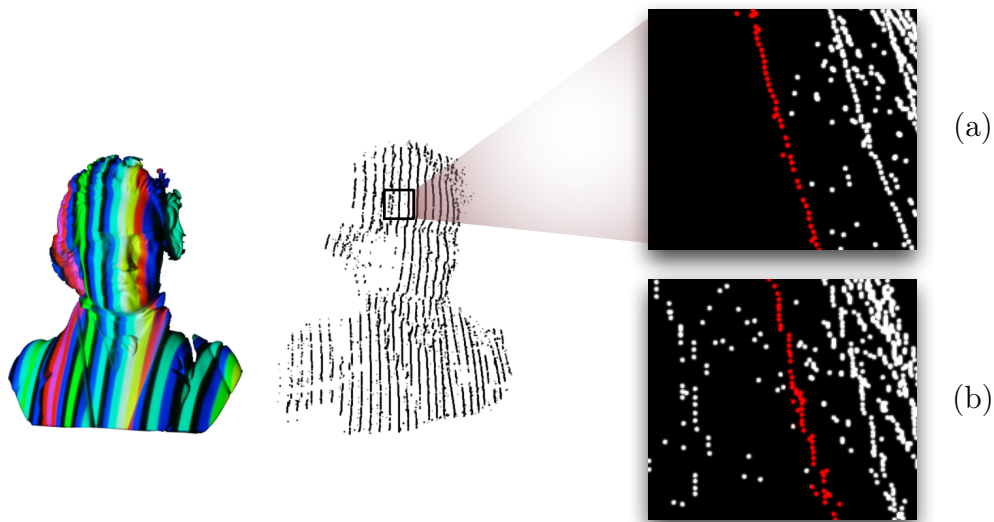


Abbildung 4.7: (a) Subpixelgenaue Kantendetektion. (b) Pixelgenaue Kantendetektion.

Selektive Kantendetektion und -integration

Die oben präsentierte Kantendetektion ist somit in der Lage, auch solche farbigen Streifenübergänge als Kanten zu detektieren, die dieselbe Helligkeit haben. Dennoch konnten wir feststellen, dass eine Menge an zusätzlichen Kanten detektiert werden, die sich eng an den Stellen anhäufen, die in der weiß beleuchteten Aufnahme besonders dunkel sind (z. B. an stark un stetigen Stellen, wo Schatten entstehen, die nicht durch die Vordergrundsegmentierung entfernt wurden). Ursache dafür ist der begrenzte Dynamikumfang der Kamera, wodurch diese Artefakte zu einer schwierigeren Zuordnung und Ausreißern in der Triangulierung führen. Wir haben dazu in [Li et al. '04] ein Verfahren vorgestellt, das annimmt, dass die Streifen bei der Erfassung eine Mindestbreite haben müssen (Es entspricht der höheren Abtastrate in vertikaler Richtung gegenüber der horizontale Richtung wie in Kap. 2 erklärt wird).

Das Prinzip besteht darin, nur solche Kanten zuzulassen, die keine andere Kanten innerhalb der nächsten w Pixel mit einer höheren Summe der quadrierten Kontraste aufweist. Es gilt $w \geq 2$. Kanten, bei denen wir noch nicht wissen, ob dieses Kriterium erfüllt ist, die aber zum Zeitpunkt der Überprüfung den höchsten Kontrast aufweisen, bezeichnen wir als Kandidatkanten, die sich an der Stelle u_c befinden. Das Verfahren fassen wir im selbsterklärenden Algorithmus 4.1 zusammen.

```

procedure selektiveKantendetektion:  $w$ 
  Setze keineKanteGefunden
  for  $u = 1, \dots, N$ 
    if Kante wird detektiert:  $s(u) > \sigma$  and  $\text{sign}(\Delta s(u-1)) \neq \text{sign}(\Delta s(u))$ 
      if keineKanteGefunden or  $u < u_c + w$ 
        if keineKanteGefunden or  $s(u) > s(u_c)$ 
          Aktualisiere Kandidatkante mit  $u_c = u$ 
          Setze kanteGefunden
        else
          Akzeptiere Kandidatkante in  $u_c$ 
          Setze wieder keineKanteGefunden
      end if
    end for

```

Algorithmus 4.1: Selektive Kantendetektion in einer Zeile mit N Pixel. Ohne Einschränkung der Allgemeinheit nehmen wir an, dass auch an den Rändern Kanten detektierbar sind.

Hierbei werden zwar auch Kanten entfernt, die zu den korrekten Streifenübergängen gehören, doch die Anzahl der eliminierten falschen Kanten, wie in Abb. 4.8 veranschaulicht wird, übertreffen diese deutlich. In der Praxis sollte w ein wenig kleiner sein, als die durchschnittliche Breite der Streifen in den Aufnahmen. In unseren Versuchen wählten wir meistens $w = 4$ oder $w = 5$. Für eine automatische Bestimmung von w wäre eine zeilenweise Autokorrelation über $s(u)$ durchaus denkbar. Dadurch würde man die Periodizität der Streifenübergänge gewinnen und in der Lage sein, einen sich in jeder Zeile adaptierenden w -Wert zu bestimmen.

Die Kantendetektion bezieht sich bislang nur auf einzelne Aufnahmen. Da wir für die Zuordnung von längeren Streifensequenzen zwei oder mehr Aufnahmen benötigen, müssen dieselben Streifenübergänge in allen Aufnahmen exakt an derselben Stelle gefunden werden. Doch wie wir in Kap. 3 gesehen haben, können Streifenübergänge in einer Aufnahme fehlen, da diese über mehrere Aufnahmen identifiziert werden. Es müssen daher die Kanten in allen Aufnahmen unabhängig voneinander detektiert werden, um sie im Nachhinein zu integrieren. Hier taucht das Problem auf, dass, wenn zwei gleiche Streifenübergänge über mehrere Aufnahmen vorhanden sind, in den meisten Fällen eine leichte Abweichung ihrer Positionen zu beobachten ist (siehe Abb. 4.9). Dies wird vor allem aufgrund der chromatischen Aberration, die durch die begrenzte Tiefenschärfe verstärkt wird, hervorgerufen (siehe Kap. 2). In der Praxis konnten wir Abweichungen bis zu 2 Pixel beobachten.

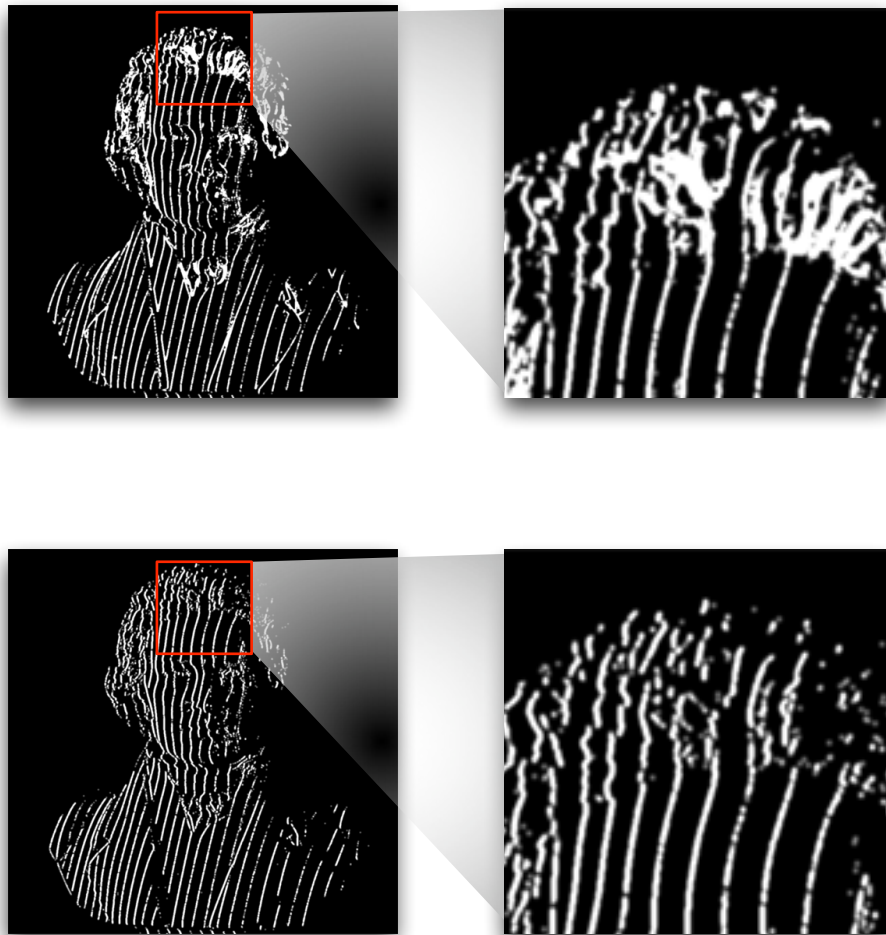


Abbildung 4.8: Durch die selektive Kantendetektion werden die meisten Kanten, die nicht zu den projizierten Streifenübergängen gehören, eliminiert. Oben: Ohne selektive Kantendetektion. Unten: Mit selektiver Kantendetektion.

Dieses Problem haben wir dadurch gelöst, dass wir über alle Aufnahmen die lokale Umgebung eines Streifenübergangs (bzw. detektierte Kante) betrachten und diese zusammenfassen. Die Streifenübergangsposition, die alle zusammengefassten Streifenübergänge repräsentiert, ist der Schwerpunkt aller Positionen. Auch dieser Schritt wird zeilenweise durchgeführt. Insbesondere können wir garantieren, dass innerhalb einer Aufnahme sich nicht mehrere Streifenübergänge innerhalb dieser lokalen Umgebung befinden, indem wir die Größe dieser Umgebung kleiner wählen als der w -Wert, den wir bei der selektiven Kantendetektion verwendet haben.

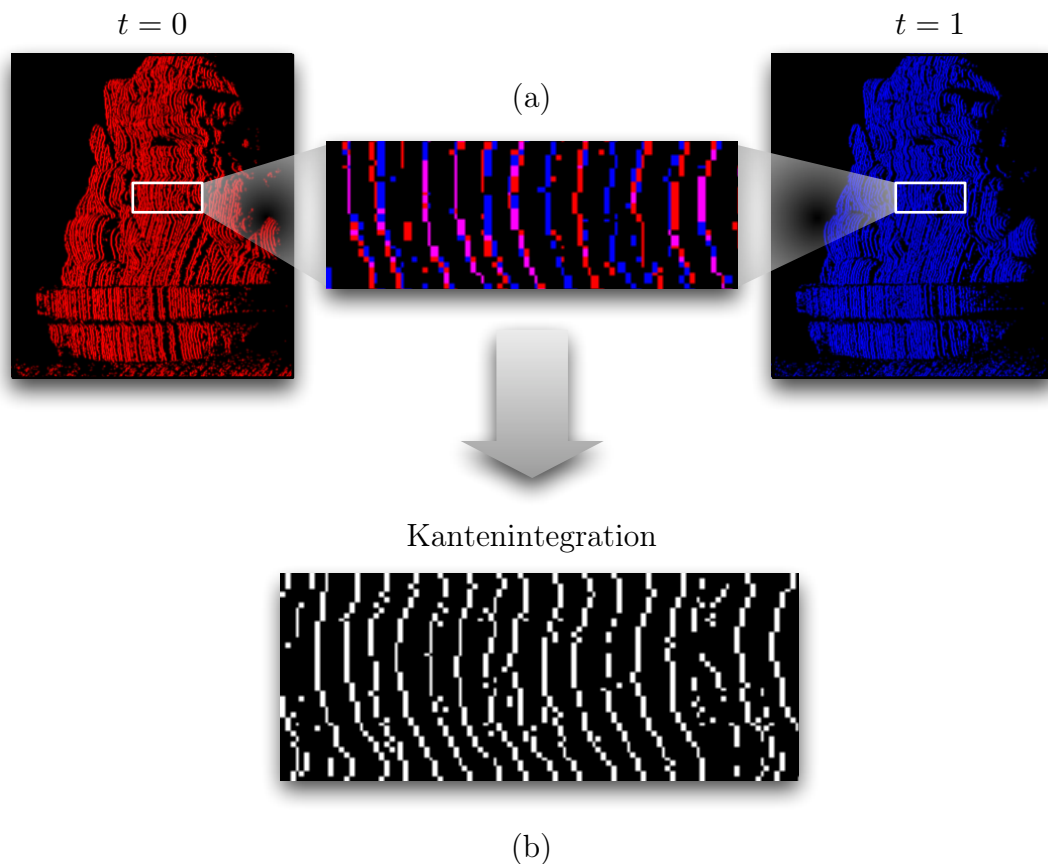


Abbildung 4.9: (a) Abweichende Positionen derselben Kanten über zwei Aufnahmen. (b) Durch die Kantenintegration werden nahe liegende Kanten zusammengefasst.

4.3 Zuordnungsverfahren

Nachdem wir alle Streifenübergänge mittels Kantendetektion über alle Aufnahmen lokalisiert haben, können wir mit der Zuordnung zwischen den projizierten und erfassten Streifenübergängen beginnen. Dieser Schritt ist stark mit dem Entwurf der projizierten Streifenmuster verknüpft und stellt einer der Hauptschwierigkeiten, bei optischen Triangulierungsverfahren mit strukturiertem Licht, die nur wenige Streifenbilder projizieren, dar. Zur Beschleunigung der Erfassung werden viele Streifen gleichzeitig projiziert, die bei der Erfassung wieder identifiziert werden müssen. Aufgrund der Unzulänglichkeiten, die in Kap.3 behandelt worden sind, können Mehrdeutigkeiten in der Praxis nicht vollständig vermieden werden. Insbesondere führen diese Mehrdeutigkeiten zu Verwechslungen während der Zuordnung, wodurch nach der optischen Triangulierung Ausreißer entstehen. Deswegen liegt es nahe, Methoden zu entwickeln, die die vorausgesetzten Kohärenzannahmen ausnutzen, um eine robustere Identifikation zu ermöglichen.

Da die Qualität einer Rekonstruktion stark von einem zuverlässigen Zuordnungsschritt abhängt, sind bisher vor allem Verfahren, die mittels rekursiver Unterteilung des projizierten Streifenmusters arbeiten [Rocchini et al. '01, Sato & Inokuchi '85, Posdamer & Altschuler '82], beliebt. Dadurch, dass in jeder Rekursion, die betrachteten Bereiche in zwei Teilbereiche aufgeteilt werden, können die Streifenübergänge in jedem Teilbereich unabhängig voneinander lokalisiert werden. Durch diese schrittweise Lokalisierung der Streifenübergänge, ist die Zuordnung implizit gegeben, da keine Mehrdeutigkeiten vorkommen. Insbesondere muss jedesmal nur eine einzige Streifenkante detektiert werden, was auch bedeutet, dass ein Streifenübergang nur mit einem Bit kodiert ist. Bei binärkodierten Streifenmustern benötigen diese Verfahren jedoch mindestens $\log n$ Projektionen, um eine Streifensequenz aus n Streifen zu erzeugen. In [Rocchini et al. '01] werden z. B. 9 Streifenbilder pro Ansicht verwendet.

Komplexe Zuordnungsverfahren sind besonders bei Verfahren, die von Mehrdeutigkeiten ausgehen, sinnvoll. Beispielsweise werden in [Zhang et al. '02, Li et al. '04] mehrdeutige Streifenübergänge projiziert, die anhand der lokalen Nachbarstreifen und mittels dynamischen Programmierens, die Zuordnung löst. Obwohl diese Verfahren mit nur einer einzigen Projektion auskommen, ist deren Robustheit, wie bereits angemerkt, unzufriedenstellend.

Die in dieser Arbeit entwickelten Projektionsmuster verwenden eine direkte Kodierung der Streifenübergänge über die Intensitätswerte ihrer Nachbarstreifen, um mit 2 Projektionen sehr lange eindeutige Streifensequenzen zu erzeugen. Dadurch dass unsere Streifenübergänge wie in [Hall-Holt & Rusinkiewicz '01] in jedem Farbkanal mit 2 Bit kodiert sind, ist eine triviale Zuordnung, wie bei den rekursiv unterteilten Streifenmustern, nicht möglich. Durch den zusätzlichen Einsatz der Farbkodierung, erschwert sich dieses Zuordnungsproblem aufgrund von nichtlinear reflektierenden Oberflächen und Farbübersprechen.

In dieser Arbeit wird ein neues Zuordnungsverfahren vorgestellt, das auf Basis der vorausgesetzten lokalen Raumkohärenz, die Empfindlichkeit von 2-Bit-kodierten Streifenübergängen gegenüber Falschzuordnungen reduziert. Insbesondere werden zum ersten Mal, nicht nur die Streifenübergänge innerhalb einer Zeile betrachtet, sondern durch ein Ballungsverfahren auch diejenigen über mehrere Zeilen. Außerdem wird ein effektiver *gieriger Algorithmus* (engl. *greedy algorithm*) vorgestellt, der eine besonders gute Annäherung der optimalen Zuordnung, die in Kap. 3 vorgestellt worden ist, darstellt. Unter der Verwendung von eindeutig kodierten Farbmustern hat dieses Zuordnungsverfahren erhebliche Vorteile gegenüber den Ansätzen mit dynamischem Programmieren.

Wir beschreiben in diesem Abschnitt zunächst, wie wir Streifenübergänge mittels ihrer benachbarten Streifenmitten identifizieren und welche Probleme daraus entstehen. Im Anschluss zeigen wir, wie ein Ballungsverfahren über den lokalen Kontrast der Streifenübergänge diese Nachteile beseitigt. Insbesondere ermöglicht

das Ballungsverfahren auch eine Zuordnung unter Berücksichtigung der lokalen Raumkohärenz in vertikaler Richtung, wodurch die Robustheit wesentlich erhöht wird. Zum Schluss wird der angesprochene Zuordnungsalgorithmus vorgestellt.

Streifenmitten zur Identifikation der Streifenübergänge

Dadurch, dass ein Streifenübergang zeilenweise über das lokale Maximum der ersten Ableitung lokalisiert wird, ist die Stärke des lokalen Kontrasts ab einem bestimmten Schwellwert nicht mehr von Bedeutung. Dies ist insbesondere der Fall bei multispektralen Kantendetektionen, da die Summe der Quadrate der Ableitungen in jedem Kanal aufsummiert werden. Bei der Zuordnung von Streifenübergängen über die Werte ihrer Nachbarstreifen, sind dagegen zuverlässige Farbtintensitäten von hoher Wichtigkeit.

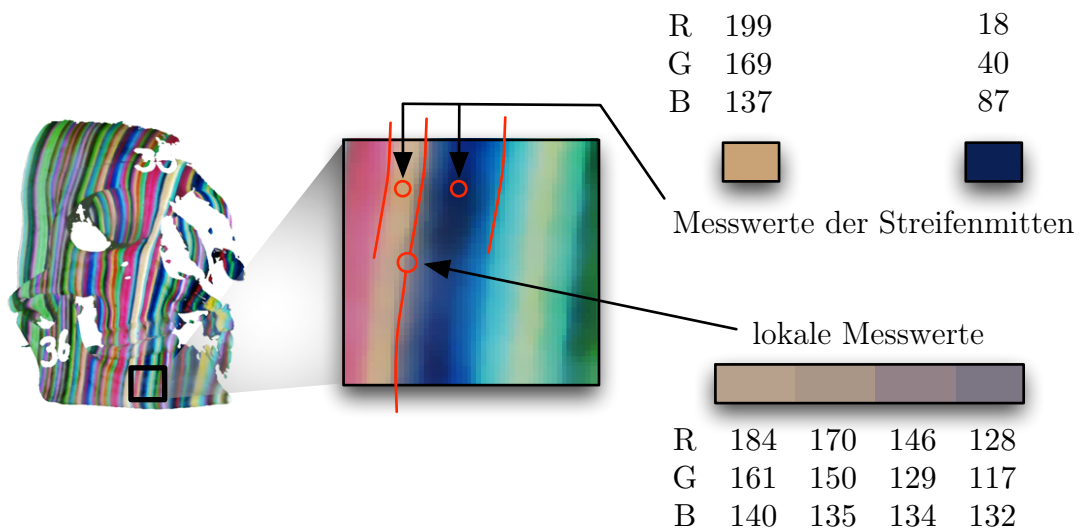


Abbildung 4.10: Eine vergrößerte Ansicht des Unterkiefers eines Schädels. Die Vergrößerung wurde absichtlich mit der Nächster-Nachbar-Abtastung durchgeführt, damit die originalen Pixelstrukturen erhalten bleiben. Die angegebenen lokalen 24-Bit-Messwerte liegen weit von den Sollwerten entfernt (Gelb-Blau-Übergang) und können nicht für die Identifikation der Streifenübergänge verwendet werden. Dagegen sind die 24-Bit-Messwerte der Streifenmitten deutlich näher an den Idealwerten.

Aufgrund der Unzulänglichkeiten des Scansystems, wie etwa die begrenzte Tiefenschärfe des Projektors (siehe Kap. 2), und der schlechten Reflexionseigenschaften von Objekten, sind die Kontraste der lokalen Farbwerte an den Streifenübergängen meistens sehr schwach, obwohl eine Normalisierung der Farbwerte bei den Aufnahmen durchgeführt worden ist (siehe Abb. 4.10). Dies ist insbesondere der Fall, wenn zusätzlich hochauflösende vergrößerte Aufnahmen gemacht werden, die für die Genauigkeit notwendig sind. Während wir direkt mit einer Bildauflösung

von 3024×2016 Pixel arbeiten, wird in verwandten Arbeiten oft mit einer geringeren Auflösung abgetastet, was wohl der Grund ist, warum dieses Problem unseres Wissens bisher noch nicht angesprochen wurde. In [Zhang et al. '02] werden z. B. auf 864×576 Pixel verkleinerte Aufnahmen verwendet und in [Rusinkiewicz '01] werden sogar durch die Zwischenzeilenabtastung nur 640×240 Pixel verarbeitet.

Mit dem Argument, dass bei einer ebenen Oberfläche, der höchste Farbkontrast, bei benachbarten Streifen eines Streifenübergangs, sich genau zwischen den Streifenmitten befindet, verwenden wir diesen bei der Zuordnung. Insbesondere hat jeder Streifenübergang in jedem Farbkanal nicht mehr die zwei benachbarten Pixelwerte, sondern jeweils die Pixelwerte der linken Streifenmitte und die der rechten Streifenmitte. Die Positionen der Streifenmitten werden durch den Mittelwert der Position von zwei aufeinander folgenden Streifenübergängen gebildet. Dadurch liegen die Messwerte viel näher bei den zugehörigen Sollwerten und die Zuverlässigkeit der Farben ist unabhängig von der Kamerabildauflösung und vom Vergrößerungsfaktor der Aufnahmen.

Die Streifenmitten müssen insbesondere nach der Integration der kantendetektierten Aufnahmen ermittelt werden, da unsere Streifenmuster nicht unbedingt, an jedem Streifenübergang zu einem bestimmten Zeitpunkt t , einen Farbübergang voraussetzt (siehe Abb. 3.1). Während die Messungen der Streifenmitten zuverlässigere Farben liefern, ist die Tatsache, dass diese Farben auch tatsächlich diejenigen der benachbarten Streifen eines Streifenübergangs sind, nicht unbedingt garantiert. Wie in Abb. 4.11 skizziert wird, entstehen falsche Farbzuweisungen aufgrund von folgenden Unzulänglichkeiten:

- **nicht verwendete Bereiche innerhalb des Streifens:** Durch die Segmentierung der betrachteten Bereiche entstehen Lücken beim Zeilendurchlauf für die Bestimmung der Streifenmitten. In Abb. 4.11a sehen wir, dass sich eine Streifenmitte genau an einer Lücke befindet. Dieses Problem wird dadurch behoben, dass wir als Streifenmitte den nächsten, sich im betrachteten Bereich befindenden, Farbwert verwenden.
- **zusätzliche Farbwerte innerhalb des Streifens:** Aufgrund von Texturen und nichtlinear reflektierenden Oberflächen, ist eine perfekte Normalisierung der Aufnahmen nicht möglich. Dadurch entstehen falsche Farbwerte, die sich innerhalb eines Streifens befinden (siehe Abb. 4.11b). Wenn ausschließlich die zugehörige Kamerazeile betrachtet wird, kann dieser Fehler nicht behoben werden.
- **nicht detektierte Streifenübergänge:** Ein weiteres Problem ergibt sich, wenn Streifenübergänge (z. B. aufgrund einer schlechten Fokussierung des Projektors oder zu dicht beieinanderliegenden Streifenübergängen), nicht detektiert werden. In diesem Fall würde, zwischen den benachbarten Streifenübergängen des nicht erkannten Streifenübergangs, eine falsche Streifenmitte

ermittelt. Da der falsche Farbwert höchstens die Intensitäten beider angrenzenden Streifen besitzt, ist diese Abweichung meistens nicht sehr hoch (siehe Abb. 4.11c).

- **zusätzlich detektierte Streifenübergänge:** Obwohl bei der Kantendetektion berücksichtigt wurde, dass es sich nur um Streifenübergänge handeln kann, wenn sich die Streifenübergänge an einer bestimmten Stelle mindestens einmal verändern, können trotzdem zusätzlich detektierte Streifen entstehen. Dies ist insbesondere der Fall bei Texturen, die sich innerhalb von Streifen befinden, deren Farben sich über die Zeit ändern. Daraus werden, wie in Abb. 4.11d skizziert wird, zwei neue Streifenübergänge detektiert, die eine korrekte Zuordnung erschweren.

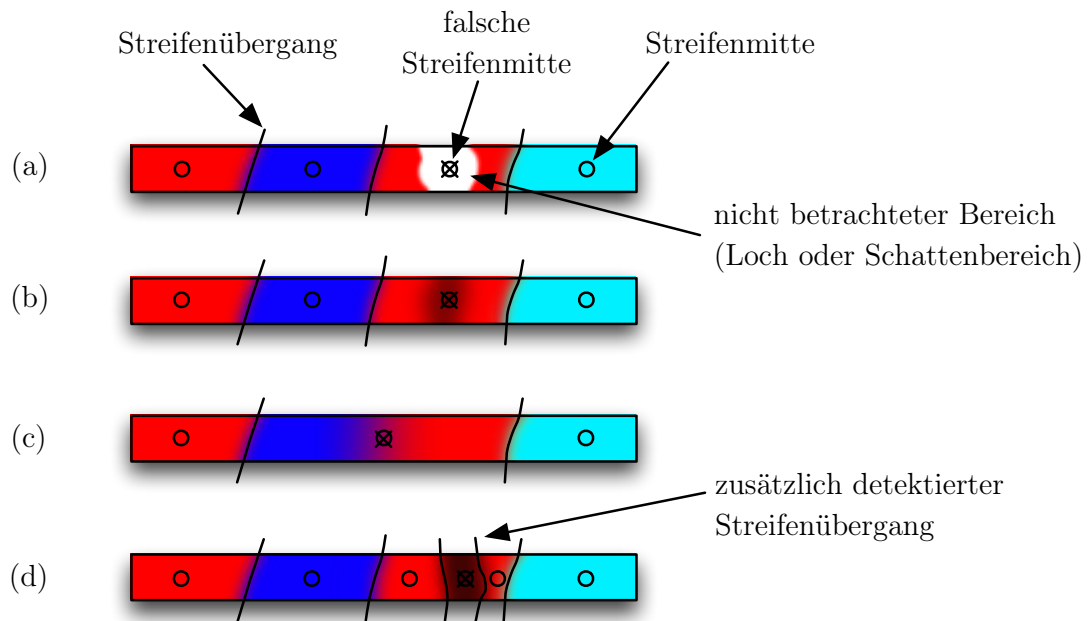


Abbildung 4.11: Problemfälle bei der Bestimmung der Farbe der Streifenmitte (angekreuzte Kreise). Entsprechend werden den Streifenübergängen die falschen Intensitätswerte von benachbarten Streifen zugewiesen.

Ballung über lokalem Kontrast

Wir haben gezeigt, dass eine zuverlässige Zuordnung unabhängig von der Kameraauflösung nur möglich ist, indem wir die Streifenmitten betrachten. Dadurch, dass der Einsatz von Streifenmitten durch eine Reihe von Unzulänglichkeiten stark eingeschränkt ist, liegt es nahe eine Methode zu finden, die Intensitätswerte dieser Streifenmitten durch zusätzliche Informationen korrigiert.

Es wird zunächst beobachtet, dass diese Unzulänglichkeiten vor allem durch Störungen aufgrund von unregelmäßigen hochfrequenten Oberflächeneigenschaften entstehen, die sowohl geometrischer als auch photometrischer Natur sind. Insbesondere haben diese Störungen eine, im Vergleich zu den projizierten Streifen, unterschiedliche Struktur. Wie bereits in Kap. 2 gesehen, entsteht durch die hohe Kameraauflösung eine höhere Abtastfrequenz der Streifenübergänge in vertikaler Richtung (d. h. über mehrere Zeilen). Zusammen mit der Annahme der vertikalen lokalen Raumkohärenz der Oberfläche aus Kap. 3, können wir davon ausgehen, dass die Streifenübergänge stückweise stetig in vertikaler Richtung verlaufen.

Aus diesen beiden Beobachtungen wird klar, dass wir die regelmäßige Struktur der Streifenübergänge in vertikaler Richtung ausnutzen können, um die Artefakte zu beseitigen. In gewisser Weise können wir, durch die Betrachtung über mehrere Zeilen, diese Artefakte als impulsartige Ausreißer identifizieren. Dies werden wir mit einer Medianwertbestimmung erreichen, da diese zur Beseitigung von impulsartigen Störungen sehr gut geeignet ist.

Um zunächst die in vertikaler Richtung stückweise stetigen Streifenübergänge identifizieren zu können, müssen wir dementsprechend ein geeignetes Merkmal finden, das einen Streifenübergang beschreibt. Die einzige zuverlässige Information, die uns zu Verfügung steht, um einen Streifenübergang über mehrere Zeilen identifizieren zu können, ist die der lokalen Farbwerte der Nachbarstreifen. Obwohl diese lokalen Farbwerte für die Zuordnung der Streifen unangemessen sind (die Messwerte weichen zu stark von den Sollwerten der Projektion ab), können diese, aufgrund der Ähnlichkeit zu den Farbwerten desselben Streifenübergangs in den nächsten Zeilen, zugeordnet werden. In Analogie zu Stereokorrespondenzen ist eine solche Zuordnung sinnvoll, da zwei oder mehrere Messwerte und nicht Sollwert und Messwert einander zugeordnet werden.

Die Aufgabe besteht nun darin, über alle Zeilen ähnliche Streifenübergänge auf Basis ihrer lokalen Kontraste zu gruppieren. Es handelt sich dabei um ein typisches *Klassifikationsproblem*, wobei jede Gruppierung eine Klasse darstellt. Da die Messwerte der lokalen Intensitäten der Streifenübergänge zu weit von den Sollwerten abweichen und die Zuordnung noch ungelöst ist, können wir keine Aussage über die Zugehörigkeit einer Klasse zu einem der Sollwerte (projizierter Streifenübergang) machen. Außerdem ist die Anzahl der Klassen bei der Erfassung unbekannt, da wir nicht wissen, wieviele projizierte Streifenübergänge von der Kamera erfasst werden. Die Anzahl der Klassen müssen also ebenfalls durch das Klassifikationsverfahren ermittelt werden. Man spricht von einem *unüberwachten Lernprozess*.

Zur Lösung dieses Klassifikationsproblems benutzen wir das Konzept von Ballungsverfahren welche von [Kroschel '04] inspiriert wurden.

Zusammen mit der Notation der Farbkodierung, die wir in Kap. 3 eingeführt haben, bezeichnen wir den *Merkmalsvektor* $\mathbf{r}_{j,k}$ des j -ten erfassten Streifenübergangs in der Zeile k mit

$$\mathbf{r}_{j,k} = \begin{bmatrix} \mathbf{a}_{j,k} \\ \mathbf{a}_{j+1,k} \end{bmatrix} \in [0, 1]^{6K} \quad k = 1, \dots, M_k \quad ,$$

wobei $\mathbf{a}_{j,k}$ und $\mathbf{a}_{j+1,k}$ die Intensitätsvektoren über K Aufnahmen der benachbarten Pixel des j -ten Streifenübergangs an der k -ten Zeile repräsentieren. Entsprechend ist M_k die Anzahl der erfassten Streifenübergänge der k -ten Zeile. Insbesondere befindet sich der Merkmalsvektor in einem $6K$ -dimensionalen Vektorraum.

Dadurch, dass die nicht homogen verteilten Merkmalsvektoren jeweils einen Streifenübergang darstellen, sind sie klassentrennend. Sie konzentrieren sich somit jeweils in *Ballungen* (engl. *cluster*).

Ziel ist es, den Merkmalsvektorraum, angepasst auf die Ballungen, in Q Partitionen \mathcal{P}_i , $i = 1, \dots, Q$ zu zerlegen. Damit wir die noch unbekannt Merkmalsvektoren mit einer Partition, die in der Regel aus mehreren Merkmalsvektoren bestehen, vergleichen können, müssen wir die \mathcal{P}_i jeweils durch einen Repräsentanten vertreten. Hierbei bieten sich die zugehörigen Schwerpunkte der sich in einer Partition befindenden Merkmalsvektoren an. Daraus ergibt sich die Definition eines *Ballungsschwerpunkts* \mathbf{b}_i der Partition \mathcal{P}_i :

$$\mathbf{b}_i = \frac{1}{|\mathcal{P}_i|} \sum_{\mathbf{r} \in \mathcal{P}_i} \mathbf{r} \in [0, 1]^{6K} \quad . \quad (4.3)$$

Entsprechend verwenden wir, für den Vergleich zwischen einem unbekannt Merkmalsvektor $\mathbf{r}_{j,k}$ und einem Ballungsschwerpunkt \mathbf{b}_i , den euklidischen Abstand (L_2 -Norm) d :

$$d(\mathbf{r}_{j,k}, \mathbf{b}_i) = \|\mathbf{r}_{j,k} - \mathbf{b}_i\|^{\frac{1}{2}} \quad .$$

Insbesondere wird hier nicht der quadratische Abstand genommen, da wir später sehen werden, dass d mit einem vorgegebenen Schwellwert verglichen werden muss.

Mit der zusätzlichen Beobachtung, dass wir eindeutig kodierte Farbübergänge projizieren, können wir schließen, dass innerhalb einer Zeile k maximal nur ein Merkmalsvektor $\mathbf{r}_{j,k}$ zu einer bestimmten Partition \mathcal{P}_i zugewiesen werden darf. Es gilt die *Eindeutigkeitsbeschränkung*. Dadurch können wir nach jeder Zuweisung den Suchraum verkleinern, indem wir eine weitere Zuweisung mit \mathcal{P}_i verbieten. Es ist deswegen sinnvoll, die Klassifikation zeilenweise durchzuführen und nach jedem Zeilendurchgang alle \mathcal{P}_i wieder zu erlauben.

Den Ballungsalgorithmus, der alle Streifenübergänge über alle Zeilen, anhand ihrer lokalen Intensitäten, klassifiziert, können wir als einen iterativen Prozess beschreiben. Insbesondere besteht jede Iteration aus zwei Schritten: ein Klassifikationsschritt und ein Lernprozess. Da es sich um einen unüberwachten Lernprozess

handelt, definieren wir einen Schwellwert σ , der dazu dient, eine neue Partition zu erzeugen, falls der Abstand eines Merkmalsvektors zu allen Ballungsschwerpunkten zu groß, also größer als σ ist. Die Initialisierung des Algorithmus geschieht, indem alle Merkmalsvektoren (bzw. Streifenübergänge) $\mathbf{r}_{j,1}$ der ersten Zeile jeweils die ersten Partitionen sind, also $\mathbf{b}_j := \mathbf{r}_{j,1}$ für und $\mathcal{P}_i := \{\mathbf{r}_{j,1}\}$, $j = 1, \dots, M_1$. Anschließend durchläuft der Algorithmus für $k = 2, \dots, R$ die restlichen $R - 1$ Zeilen der Aufnahme und führt bei jeder k -ten Iteration die oben genannten zwei Schritte aus:

1. **Klassifikation:** Für alle Merkmalsvektoren $j = 1, \dots, M_k$:

(a) Bestimme den nächstliegenden Ballungsschwerpunkt

$$i_b := \arg \min_{i \in \{1, \dots, Q\}} \{d(\mathbf{r}_{j,k}, \mathbf{b}_i)\}$$

und dessen Abstand

$$d_b := \min_{i \in \{1, \dots, Q\}} d(\mathbf{r}_{j,k}, \mathbf{b}_i) \quad .$$

(b) Falls $d_b < \sigma$, so wird der Merkmalsvektor der entsprechenden Partition zugewiesen:

$$\mathcal{P}_{i_b} := \mathcal{P}_{i_b} \cup \{\mathbf{r}_{j,k}\} \quad .$$

Anschließend wird \mathcal{P}_{i_b} für alle weitere Iterationen j gesperrt.

(c) Falls $d_b \geq \sigma$, so wird der Merkmalsvektor $\mathbf{r}_{j,k}$ für die Erzeugung einer neuen Partition vorgemerkt.

2. **Lernprozess:**

(a) Entsperre alle Partitionen \mathcal{P}_i , $i = 1, \dots, Q$ und aktualisiere die zugehörigen Ballungsschwerpunkte \mathbf{b}_i gemäß Gleichung 4.3.

(b) Erzeuge neue Partitionen für alle nicht zugeordneten, aber vorgemerkten Merkmalsvektoren. Inkrementiere dementsprechend Q nach jeder Erzeugung.

Das Ballungsverfahren fasst somit alle, durch Merkmalsvektoren charakterisierten, Streifenübergänge über mehrere Zeilen in jeweils dasjenige \mathcal{P}_i zusammen, das denselben projizierten Streifenübergang darstellt. In der Regel werden mehr Partitionen gefunden als projizierte Streifenübergänge vorhanden sind. Außerdem müssen die in den Partitionen liegenden Streifenübergänge nicht unbedingt in aufeinander folgenden Zeilen liegen. Der vorgestellte Algorithmus liefert kein optimales Ergebnis und ist abhängig von der Reihenfolge der Zeilendurchläufe und vor allem

der Vergleiche zwischen den Merkmalsvektoren und den Ballungsschwerpunkten. Nichtsdestoweniger ist er trotz seines kubischen Laufzeitaufwands besonders effizient, da sich die Anzahl der neu erzeugten Partitionen in der Praxis in Grenzen hält (siehe Tabelle 4.1 auf Seite 113). Eine optimale Lösung ist auch nicht notwendig, da es sich nicht, um die Zuordnung der projizierten zu den erfassten Streifenübergängen handelt, sondern nur um eine approximative Abschätzung der Zusammenhänge zwischen den Streifenübergängen über mehrere Zeilen. Zusätzlich konnten wir feststellen, dass dieses Verfahren bei stark unterschiedlichen σ -Werten ungefähr die gleichen Ballungen liefert, wodurch sich die Robustheit des Verfahrens, gegenüber der Wahl von σ , zeigt.

Dadurch, dass wir in vertikaler Richtung mittels der lokalen Kontraste zusammengehörige Streifenübergänge identifiziert haben, erhalten wir ebenfalls die zugehörigen Streifenmitten dieser Übergänge. Wie erwartet, können wir innerhalb einer Partition aus den zugehörigen Streifenmitten, die Ausreißer dadurch identifizieren, dass ihre Farbwerte stark von der Mehrzahl der anderen Farbwerte abweichen. Wie bereits angedeutet, können solche Ausreißer als eine Form von Impulsrauschen gesehen werden, welches durch die Bestimmung der Medianwerte beseitigt werden kann.

Sei nun $\mathbf{s}_{j,k,l}^{\text{links}} \in [0, 1]^{3K}$ der linke und $\mathbf{s}_{j,k,l}^{\text{rechts}} \in [0, 1]^{3K}$ der rechte zu $\mathbf{r}_{j,k}^{(l)} \in [0, 1]^{6K}$ gehörende l -te Streifenmittenvektor einer Partition \mathcal{P}_i . Daraus ergibt sich der linke Medianvektor $\mathbf{m}_i^{\text{links}} \in [0, 1]^{3K}$, der aus den komponentenweisen skalaren Medianwerten einer Partition \mathcal{P}_i von $\mathbf{s}_{j,k,l}^{\text{links}} \in [0, 1]^{3K}$ für $l = 1, \dots, |\mathcal{P}_i|$ besteht. Entsprechend wird der rechte Medianvektor $\mathbf{m}_i^{\text{rechts}} \in [0, 1]^{3K}$ mit dem rechten Streifenmittenvektor definiert.

Die linken und rechten Medianvektoren der jeweiligen Partitionen repräsentieren auf diese Weise die korrigierten Streifenmittenwerte aller Streifenübergänge bzw. Merkmalsvektoren, die sich in der Partition befinden. Eine robustere Zuordnung wird hergestellt, indem wir die projizierten Streifenübergänge mit den zugehörigen linken und rechten Medianvektoren korrespondieren.

Gierige Zuordnung in einer sortierten Datenstruktur

Ausgehend von den oben präsentierten korrigierten Streifenmitten, müssen wir die in Kap. 3 definierte Metrik (Gleichung 3.2 auf Seite 71), für die optimale Zuordnung der projizierten Streifen, modifizieren.

Sei nun $\mathbf{m}_j^{\text{links}}$ der linke und $\mathbf{m}_j^{\text{rechts}}$ der rechte korrigierte Streifenmittenvektor eines Streifenübergangs \mathbf{e}_j , der jeweils durch den Medianvektor der zugehörigen Partition dargestellt wird. Da wir uns wieder, aufgrund der Epipolareinschränkung, in die eindimensionale Zuordnung begeben, und die Eindeutigkeitsbeschränkung gilt, können wir davon ausgehen, dass alle korrigierten Streifenmittenvektoren zu ei-

ner unterschiedlichen Partition gehören und somit einen anderen Streifenübergang repräsentieren. Die Indizierung mit j ist somit gerechtfertigt.

Daraus ergibt sich der folgende quadratische Abstand d_ϕ zwischen einem projizierten \mathbf{q}_i und einem erfassten Streifenübergang $\mathbf{m}_j = [\mathbf{m}_j^{\text{links}} \mathbf{m}_j^{\text{rechts}}]^t$:

$$d_\phi(\mathbf{q}_i, \mathbf{m}_j) := \left\| \begin{bmatrix} \mathbf{p}_i \\ \mathbf{p}_{i+1} \end{bmatrix} - \begin{bmatrix} \mathbf{m}_j^{\text{links}} \\ \mathbf{m}_j^{\text{rechts}} \end{bmatrix} \right\|^2 ,$$

wodurch dann die selbe Definition für die global optimale Zuordnung aus Gleichung 3.4 auf Seite 71 übernommen werden kann. Wir erinnern uns, dass in einer Zeile $N - 1$ Streifenübergänge projiziert und $M - 1$ Streifenübergänge erfasst werden und eine gesamte Zuordnung Φ aus L Paaren von projizierten und erfassten Streifenübergängen besteht. Da die Aufgabe darin besteht, eine optimale Zuordnung $\Phi^* = \arg \min_\Phi \{\sigma(\Phi)\}$ zu finden, wobei $\sigma(\Phi) = \sum_{l=1}^L d_\phi(\mathbf{q}_{i_l}, \mathbf{m}_{j_l})$, würde der vollständige Suchraum $O(N^M)$ Elemente enthalten. Diese Nicht-Handhabbarkeit führt zu einer Reihe von möglichen Suchstrategien, die gewisse Annahmen voraussetzen, um den Suchraum einzuschränken. Typische Beschränkungen, die aus dem Gebiet der Stereoverfahren geläufig sind, können wir unter den bereits eingeführten Begriffen Eindeutigkeitsbeschränkung, Epipolareinschränkung und Monotonizitätsannahme zusammenfassen.

Während das mehrpässige dynamisch programmierte Zuordnungsverfahren aus [Zhang et al. '02] erlaubt auf die Monotonizitätsannahme zu verzichten, konnten wir in [Li et al. '04, Li '04b] feststellen, dass, durch den Rückverfolgungsschritt, zusammenhängende Zuordnungen in einer Richtung stark bevorzugt werden. Dadurch entstehen vor allem an den Konturen starke Ausreißer, aufgrund von mehrdeutigen Zuordnungsmöglichkeiten, sowie zusätzlich gefundene Zuordnungen.

In dieser Arbeit wird ein unterschiedlicher Ansatz verfolgt. Ausgehend von unseren korrigierten Messwerten und der Tatsache, dass wir eindeutig kodierte Farbstreifen projizieren, können wir davon ausgehen, dass es sich genau dann um einen projizierten Streifenübergang handelt, wenn eine hohe Korrelation mit dem entsprechenden erfassten Streifenübergang besteht. Erfasste Kanten, die einen schwachen Farbkontrast haben oder keinen gültigen Streifenübergang darstellen, sollten nicht genommen werden. Diese Annahmen führen in natürlicher Weise zu einem gierigen Algorithmus.

Zum Verständnis präsentieren wir zunächst eine naive Implementierung, bevor wir auf eine optimierte Version eingehen.

Die Menge \mathcal{Q} aller projizierten Streifenübergänge sei wie folgt definiert:

$$\mathcal{Q} := \{\mathbf{q}_1, \dots, \mathbf{q}_{N-1}\} \quad .$$

Entsprechend sei die Menge \mathcal{E} aller erfassten Streifenübergänge gegeben durch:

$$\mathcal{E} := \{\mathbf{m}_1, \dots, \mathbf{m}_{M-1}\} \quad .$$

Die Zuordnung ermittelt zeilenweise aus der Eingabe \mathcal{Q} und \mathcal{E} die zugehörige Gesamtzuordnung Φ , die mit $\Phi := \emptyset$ initialisiert wird, wie folgt:

Für $l = 1, \dots, \min\{N - 1, M - 1\}$:

1. Finde Zuordnung ϕ_l mit minimalem Abstand zwischen \mathbf{m}_j und \mathbf{q}_i für $i = 1, \dots, N - 1$ und $j = 1, \dots, M - 1$:

$$\phi_l := \begin{bmatrix} i_{\min} \\ j_{\min} \end{bmatrix} = \arg \min_{\begin{bmatrix} i \\ j \end{bmatrix}} \{d_\phi(\mathbf{q}_i, \mathbf{m}_j) \mid \mathbf{q}_i \in \mathcal{Q}, \mathbf{m}_j \in \mathcal{E}\} \quad .$$

2. Nimm ϕ_l in die Gesamtzuordnung auf:

$$\Phi := \Phi \cup \{\phi_l\} \quad .$$

3. Entferne die bereits zugeordneten Streifenübergänge wegen der Eindeutigkeitseinschränkung:

$$\mathcal{Q} := \mathcal{Q} \setminus \{\mathbf{q}_{i_{\min}}\} \quad , \quad (4.4)$$

$$\mathcal{E} := \mathcal{E} \setminus \{\mathbf{m}_{j_{\min}}\} \quad . \quad (4.5)$$

4. Wenn $\mathcal{Q} = \emptyset$ oder $\mathcal{E} = \emptyset$, breche ab.

Der Algorithmus durchläuft somit alle projizierten und erfassten Streifenübergänge, um jedes Mal die Zuordnung ϕ_l zu finden, die den geringsten quadratischen Abstand hat. Dies wird solange wiederholt, bis keine Zuordnung mehr möglich ist.

Diese Lösung Φ ist insbesondere keine optimale Lösung, da es durchaus vorkommen kann, dass die Wahl eines geringsten quadratischen Abstands verhindert, eine andere Zuordnungskombination zu finden, die möglicherweise eine global bessere Lösung ist. Nichtsdestotrotz haben wir experimentell herausgefunden, dass diese Variante bessere Ergebnisse als die dynamisch programmierten Ansätze liefert, da wir von besonders zuverlässigen Streifenübergangswerten ausgehen und diese eindeutig kodiert sind (siehe Abb. 4.12).

Obwohl dieser Algorithmus die gewünschte Zuordnung liefert, ist er nicht besonders effektiv, da er eine Laufzeit von $O(\min\{M, N\}MNR)$ hat, wobei R die Anzahl der Zeilen ist (siehe Tabelle 4.1).

Da sich in jeder Iteration von l , die Berechnungen der Minima wiederholen, können wir eine effektivere Lösung erreichen, indem wir alle Abstände zwischen den \mathbf{q}_i und den \mathbf{e}_j in sortierten Listen abspeichern, um jedesmal nur noch die Minima der entsprechenden Listen zu betrachten. Dieser Algorithmus sieht, mit derselben Initialisierung wie im obigen Verfahren, für alle Zeilen folgendermaßen aus:

Alle Indizes und Abstände der erfassten Streifenübergänge zum j -ten projizierten Streifenübergang \mathbf{q}_j werden durch die Menge \mathcal{D}_j dargestellt:

$$\begin{aligned} \mathcal{D}_j &:= \left\{ \left[\begin{array}{c} d_\phi(\mathbf{q}_1, \mathbf{m}_j) \\ 1 \end{array} \right], \dots, \left[\begin{array}{c} d_\phi(\mathbf{q}_{N-1}, \mathbf{m}_j) \\ N-1 \end{array} \right] \right\} \\ &= \{\mathbf{d}_{j,1}, \dots, \mathbf{d}_{j,N-1}\} \quad . \end{aligned}$$

Für $j = 1, \dots, M-1$:

1. Für $i = 1, \dots, N-1$, bestimme alle \mathcal{D}_j :

$$\mathcal{D}_j := \mathcal{D}_j \cup \left\{ \left[\begin{array}{c} d_\phi(\mathbf{q}_i, \mathbf{e}_j) \\ i \end{array} \right] \right\} \quad .$$

Für $j = 1, \dots, M-1$:

1. Sortiere alle \mathcal{D}_j aufsteigend nach $[1\ 0] \mathbf{d}_{j,i}$. Daraus ergibt sich:

$$\mathcal{D}_j = \{\mathbf{d}_j^{(1)}, \dots, \mathbf{d}_j^{(N-1)}\} \quad ,$$

$$\text{mit } [1\ 0] \mathbf{d}_j^{(1)} \leq \dots \leq [1\ 0] \mathbf{d}_j^{(N-1)}.$$

Für $l = 1, \dots, \min\{N-1, M-1\}$:

1. Finde in den sortierten Mengen den Index j_{\min} des erfassten Streifenübergangs, der den minimalen Zuordnungsabstand besitzt:

$$j_{\min} := \arg \min_j \left\{ [1\ 0] \mathbf{d}_j^{(1)} \mid j = 1, \dots, M-1 \right\} \quad .$$

2. Daraus folgt für den zum minimalen Zuordnungsabstand gehörende Index des projizierten Streifenübergangs:

$$i_{\min} := [0\ 1] \mathbf{d}_{j_{\min}}^{(1)} \quad .$$

3. Nimm die gefundene Zuordnung in die Gesamtzuordnung auf:

$$\Phi := \Phi \cup \{\phi_l\} = \Phi \cup \left\{ \left[\begin{array}{c} i_{\min} \\ j_{\min} \end{array} \right] \right\} \quad .$$

4. Entferne aufgrund der Eindeutigkeitsbeschränkung für alle $j = 1, \dots, M - 1$ die Abstände und Indizes der gefundenen Zuordnung aus \mathcal{D}_j :

$$\mathcal{D}_j := \mathcal{D}_j \setminus \{\mathbf{d}_{j_{\min}}^{(1)}\} \quad .$$

Dieser Schnitt kann durch geschickte Implementierung mit einer Indextabelle in $O(M)$ ausgeführt werden.

5. Entferne entsprechend die bereits zugeordneten Streifenübergänge gemäß Gleichung 4.4 und 4.5.
6. Obwohl die \mathcal{D}_j immer noch sortiert sind, entstehen Lücken bei den betrachteten Indizes. Diese müssen wie folgt für alle $j = 1, \dots, M - 1$ aktualisiert werden:

$$\mathcal{D}_j = \{\mathbf{d}_j^{(1)}, \dots, \mathbf{d}_j^{(N-2)}\} \quad ,$$

$$\text{mit } [1\ 0] \mathbf{d}_j^{(1)} \leq \dots \leq [1\ 0] \mathbf{d}_j^{(N-2)}.$$

7. Wenn $\mathcal{Q} = \emptyset$ oder $\mathcal{E} = \emptyset$, breche ab.

Durch diesen effektiven Algorithmus werden alle zeilenweisen Zuordnungen über alle R Zeilen nur noch mit einem Aufwand von $O(\min\{N, M\}NR \log(N))$ erzeugt. Da die Kamera und der Projektor nur eine begrenzte Bildauflösung haben, ist die Anzahl der detektierten Streifenübergänge begrenzt. Die Laufzeiten dieses Algorithmus sind deshalb in der Praxis, wie in Tabelle 4.1 demonstriert wird, praktikabel. Zusätzlich wird betont, dass es sich lediglich um einen Off-Line-Prozess handelt, wo die Qualität der Ausgabe eine größere Rolle spielt, als der Rechenaufwand.

Zum Schluss erweitern wir die Zuordnung noch in einem Nachverarbeitungsschritt, der wiederum die lokale Raumkohärenz der projizierten Streifen ausnutzt. Wie bereits in Kap. 3 angesprochen, nehmen wir an, dass eine Zuordnung ϕ nur genau dann gültig ist, wenn der zugehörige erfasste Streifenübergang einen zugeordneten Nachbarn besitzt, der auch Nachbar des zu ϕ gehörigen projizierten Streifenübergangs ist. Insbesondere muss für ein $\phi = [\mathbf{q}_i \ \mathbf{m}_j]^t$ folgendes gelten:

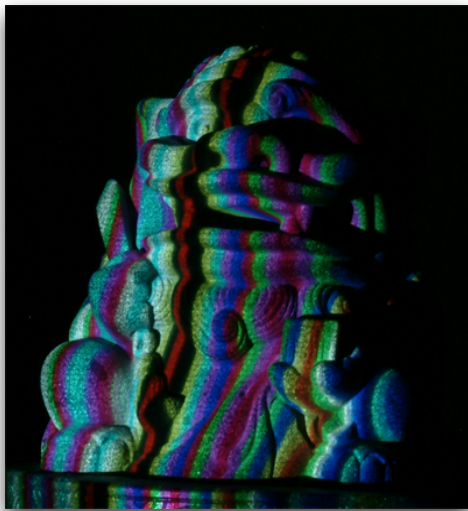
$$\phi \in \Phi \quad \Leftrightarrow \quad [\mathbf{q}_{i+1} \ \mathbf{m}_{j+1}]^t \in \Phi \quad \text{oder} \quad [\mathbf{q}_{i-1} \ \mathbf{m}_{j-1}]^t \in \Phi \quad .$$

Dadurch, dass der vorgestellte Zuordnungsalgorithmus erst dann abbricht, wenn keine Zuordnungen mehr möglich sind, kommt es trotz seiner Robustheit zu zusätzlich zugeordneten Streifenübergängen, die aufgrund von falsch detektierten Kanten entstehen. Mit dem Argument, dass es wichtiger ist, weniger und korrektere Zuordnungen zu finden, können wir mit diesem Vorgang zugeordnete Streifenübergänge

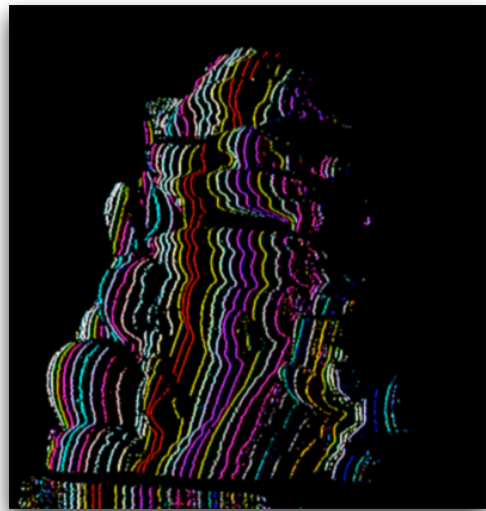
beseitigen, die die lokale Stetigkeitsannahme der Oberfläche verletzen. Da eine Zuordnung einem triangulierten Punkt im Raum entspricht, würde es bedeuten, dass ein zugeordneter erfasster Streifenübergang, der das obige Kriterium nicht erfüllt, ein Ausreißer oder Rauschen darstellt. Auch hier haben wir experimentell nachgewiesen, dass eine Nachverarbeitung mit der räumlichen Kohärenzannahme wesentlich korrektere Ergebnisse liefert.

Testobjekt	Anzahl der erfassten Streifenübergänge	Anzahl der Partitionen	Ballung	naive Zuordnung	optimierte Zuordnung
wachender Löwe	47270	818	23 s	19 min 6 s	2 min 56 s
Gesichtsschädel	52409	549	43 s	9 min 42 s	1 min 25 s
Freiformfläche	45687	630	43 s	12 min 30 s	1 min 15 s

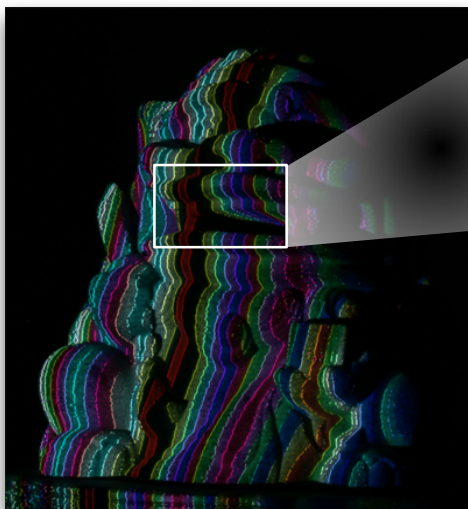
Tabelle 4.1: Statistiken von typischen Zuordnungsprozessen mit drei Testobjekten (siehe Abb. 4.13). Auf alle Objekte werden Streifensequenzen mit 200 Streifen projiziert. Die Komplexität der Objektaufnahmen wird durch die Anzahl der erfassten Streifenübergänge charakterisiert. Die Anzahl der Partitionen beschreibt, wieviele Streifenübergänge durch das Ballungsverfahren zusammengefasst werden. Die letzten drei Spalten geben jeweils die Laufzeiten der entsprechende Prozesse an. Der Flaschenhals im gesamten Prozess liegt vor allem bei der Zuordnung. Insbesondere ist das vorgestellte optimierte Zuordnungsverfahren wesentlich schneller als die naive Implementierung. Die Laufzeiten wurden auf einem Rechner mit einem 1,8 GHz-PowerPC-970-FX-Prozessor gemessen.



(a)



(b)



(c)

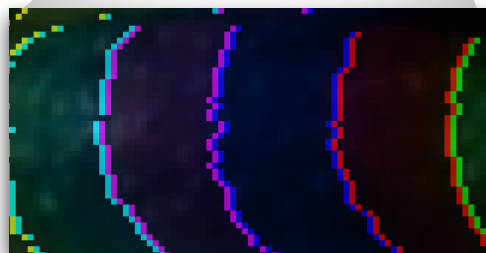
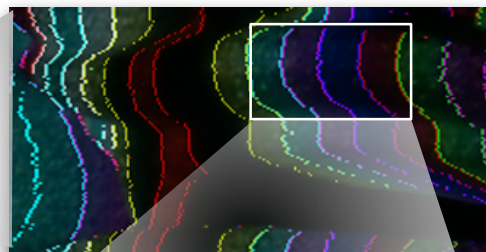


Abbildung 4.12: (a): Originalaufnahme einer Projektion mit strukturier-tem Licht. (b) Visualisierung der Zuordnung. (c) Eine Überlagerung der Zuordnungsvisualisierung und der Originalaufnahme: Eine Vergrößerung zeigt die Korrektheit der Zuordnung. Insbesondere wird jeder Streifenübergang durch zwei Farbwerte repräsentiert, die die entsprechende Zuordnung darstellen.

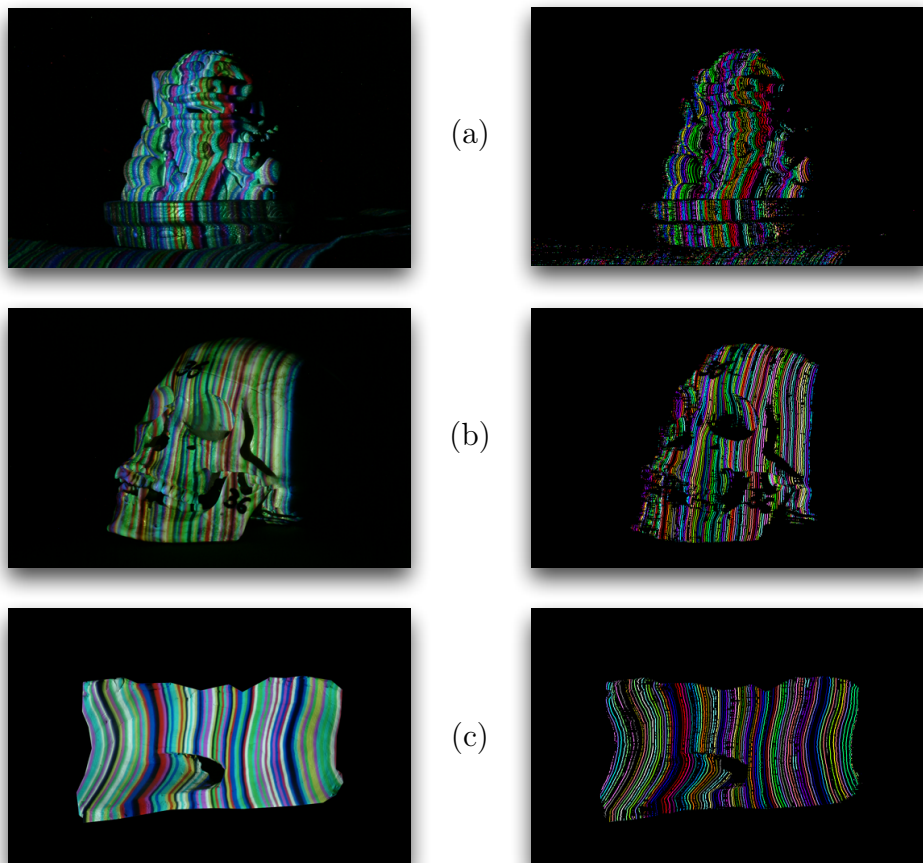


Abbildung 4.13: Zuordnungsvisualisierungen von einem wachenden Löwen (a), einem Gesichtsschädel (b) und einer Freiformfläche aus Ton (c). Links sind jeweils die farbkorrigierten Bilder und rechts die Zuordnungsvisualisierung für $t = 0$.

Kapitel 5

Flächenrekonstruktion

Dieses Kapitel beginnt mit der optischen Triangulierung, die aus zugeordneten Daten eine Punktwolke rekonstruiert. Anschließend zeigen wir, dass wir unter Ausnutzung von zusätzlichen Informationen aus vorhergehenden Schritten, ein topologisch konsistentes Dreiecksnetz aus dieser Punktwolke erzeugen können. Zum Schluss werden Ausreißer beseitigt und Rauschen mittels Glättungsverfahren unterdrückt.

5.1 Optische Triangulierung

Wir haben im vorherigen Kapitel beschrieben, wie wir die Punkte der erfassten Streifenübergänge auf der Kamerabildebene zu den projizierten Streifenübergängen zuordnen. Daraufhin können wir, wie in Kap. 1 beschrieben wird, für jede Zuordnung, einen Punkt im Raum finden, indem wir die Sichtgeraden, der erfassten Punkte mit den projizierten Ebenen der zugeordneten Streifenübergänge schneiden. Aus dieser optischen Triangulierung entsteht somit für alle Zuordnungen, die aus zwei Aufnahmen berechnet worden sind, eine Punktwolke.

In diesem Abschnitt werden Konzepte der projektiven Geometrie ohne besondere Einführung eingesetzt. Die Betrachtung im *projektiven Raum* zeichnet sich vor allem durch ihre einfache Struktur aus. Aufgrund ihrer stark eingeschränkten Invarianzeigenschaften ist sie besonders geeignet für eine formale geometrische Beschreibung von Abbildungen, bei der die Invarianzen im euklidischen oder affinen Sinne verletzt wären, wie etwa bei einer perspektivischen Projektion.

Wie verweisen für einen detaillierten Einblick in die angewandte projektive Geometrie in den Bereichen des Maschinensehens und der Computergeometrie auf [Faugeras '93, Boehm & Prautzsch '94, Pollefeys '99].

Kamera- und Projektor-Modell

Die Kamera und der Projektor werden beide durch das ideale Lochkamera-Modell, das in Abb. 5.1 skizziert wird, beschrieben. Dadurch entsteht ein geometrischer Zusammenhang zwischen einem Punkt im Raum und seiner zugehörigen Projektion auf der Kamerabildebene. Mit der Annahme, dass dieses Modell keine Linsenverzerrung hervorruft, wird es vollständig durch dessen interne und externe Parameter beschrieben.

Sei $u' = [u' \ v' \ 1]^t$ ein Bildebenenpunkt in der projektiven Ebene \mathcal{P}^2 und $x = [x \ y \ z \ 1]^t$ der zugehörige Raumpunkt im projektiven Raum \mathcal{P}^3 . Beide Punkte werden in projektiven homogenen Koordinaten dargestellt und sind bis auf einem gemeinsamen *Homogenitätsfaktor* $\rho \neq 0$ eindeutig bestimmt. Insbesondere gilt die Äquivalenzrelation $u' \sim u \Leftrightarrow u' = \rho u$.

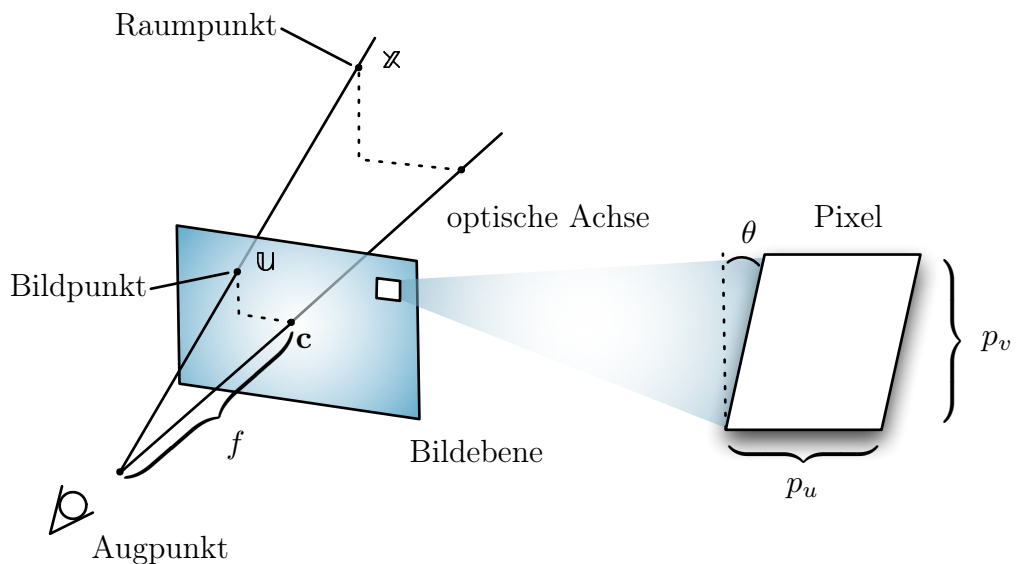


Abbildung 5.1: Das ideale Lochkamera-Modell.

Wir betrachten zunächst eine Kamera, deren Augpunkt sich im Ursprung des Weltkoordinatensystems befindet und deren Bildebene $z = 1$ ist. Insbesondere hat diese Kamera die Brennweite $f = 1$. Die perspektivische Projektion von x auf u' auf der Kamerabildebene ist dann durch folgende Gleichung gegeben:

$$\underbrace{\rho}_{u'} \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\mathbb{A}} \underbrace{\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}}_x, \quad (5.1)$$

wobei \mathbb{A} die *Abbildungsmatrix* (engl. *mapping matrix*) genannt wird. Insbesondere gilt $u' = \frac{x}{z}$ und $v' = \frac{y}{z}$, was auch mit dem Strahlensatz hergeleitet werden kann.

Um das obige Modell auf die tatsächliche Kamera zu übertragen, müssen die *internen Parameter* bestimmt werden, da beispielsweise eine Brennweite ungleich 1 vorliegt (siehe Abb. 5.1). Es liegt nun nahe, eine lineare Abbildung \mathbb{K} zwischen einem Punkt $\mathfrak{u}' = [u' \ v' \ 1]^t$ im physikalischen Koordinatensystem und dem zugehörigen Punkt \mathfrak{u} im Bildkoordinatensystem zu finden. Die *Kalibrierungsmatrix* (engl. *calibration matrix*) \mathbb{K} ist explizit durch eine Reihe von verknüpften zweidimensionalen affinen Transformationen auf der Kamerabildebene gegeben. Dadurch, dass der Hauptpunkt $\mathfrak{c} = [c_u \ c_v]$ (Schnittpunkt der optischen Achse mit der Kamerabildebene) i. Allg. nicht mit dem Ursprung des Bildkoordinatensystems übereinstimmt, ist eine Verschiebung auf der Ebene mit einer Translationsmatrix \mathbb{M}_p notwendig. Weiterhin muss eine Kamera mit Brennweite f (Abstand zwischen Hauptpunkt und Kameraaugpunkt), durch eine zusätzliche Skalierungsmatrix \mathbb{M}_f mit Skalierungsfaktor f in u - und v -Richtung modelliert werden. Auch die CCD-Pixelgrößen müssen in Betracht gezogen werden. Diese werden durch eine Skalierungsmatrix \mathbb{M}_s mit den Skalierungsfaktoren $\frac{1}{p_u}$ und $\frac{1}{p_v}$ in den entsprechenden Richtungen dargestellt. Zum Schluss wird noch eine Scherungsmatrix \mathbb{M}_h mit Scherungsparameter $\tan \theta$ gegeben, wobei θ der Scherungswinkel ist. Insgesamt ergibt sich daraus:

$$\begin{aligned} \mathfrak{u} &= \mathbb{M}_p \mathbb{M}_f \mathbb{M}_s \mathbb{M}_h \mathfrak{u}' \\ &= \underbrace{\begin{bmatrix} \frac{f}{p_u} & \frac{f}{p_v} \tan \theta & c_u \\ 0 & \frac{f}{p_v} & c_v \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbb{K}} \mathfrak{u}' \quad , \end{aligned} \quad (5.2)$$

wobei wir in der Regel $\theta = 0$ annehmen.

Auch die Annahme, dass der Kameraaugpunkt mit dem Ursprung des Weltkoordinatensystems übereinstimmt, kann mit einer zusätzlichen euklidischen Bewegung fallen gelassen werden. Diese Bewegung bestimmt die Lage und Orientierung der Kamera im Raum, welche auch als externe Parameter der Kamera bezeichnet werden. Wir beobachten, dass eine Bewegung \mathbb{M} der Szenenobjekte (aus der Kameraperspektive), genau der umgekehrten Bewegung \mathbb{M}^{-1} der Kamera im Weltkoordinatensystem entspricht. Daraus ergibt sich für eine Kamerabewegung \mathbb{M}^{-1} mit Rotationsmatrix $R \in SO(3)$ (Orientierung) und Translationsvektor $\mathfrak{t} \in \mathbb{R}^3$ (Lage):

$$\mathbb{M}^{-1} = \begin{bmatrix} R & \mathfrak{t} \\ \mathbf{0}^t & 1 \end{bmatrix} \quad ,$$

die *Bewegungsmatrix* (engl. *motion matrix*) \mathbb{M} :

$$\mathbb{M} = \begin{bmatrix} R^t & -R^t \mathbf{t} \\ \mathbf{o}^t & 1 \end{bmatrix} . \quad (5.3)$$

Wir können somit aus den Gleichungen 5.1, 5.2 und 5.3 das Kamera-Modell folgendermaßen zusammenfassen:

$$\rho \mathbf{u} = \mathbb{K} \mathbb{A} \mathbb{M} \mathbf{x} \quad (5.4)$$

$$= \mathbb{P} \mathbf{x} , \quad (5.5)$$

wobei die 3×4 -Matrix \mathbb{P} auch *Kamera-Projektionsmatrix* (engl. *camera projection matrix*) genannt wird. Insbesondere sind alle internen und externen Parameter implizit durch \mathbb{P} gegeben.

Es ist leicht, die Lage des Augpunkts bei gegebener Kamera-Projektionsmatrix \mathbb{P} zu bestimmen. Die Lage ist nämlich nur durch den Translationsvektor \mathbf{t} der Matrix \mathbb{M} gegeben. Es gilt:

$$\begin{aligned} \mathbb{P} &= \mathbb{K} \mathbb{A} \mathbb{M} \\ &= [\mathbb{K} \mathbf{o}] \begin{bmatrix} R^t & -R^t \mathbf{t} \\ \mathbf{o}^t & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbb{K} R^t & -\mathbb{K} R^t \mathbf{t} \end{bmatrix} . \end{aligned}$$

Wir können somit $\mathbb{K} R^t$ und $-\mathbb{K} R^t \mathbf{t}$ aus \mathbb{P} ablesen. Daraus ergibt sich:

$$\mathbf{t} = -(\mathbb{K} R^t)^{-1} (-\mathbb{K} R^t \mathbf{t}) .$$

Während sich die anderen Parameter ebenfalls extrahieren lassen (z. B. mittels *Kruppa-* oder *RQ-Zerlegung*), ist für diese Arbeit nur die Lage des Augpunkts der Kamera sowie des Projektors von Interesse. Wir werden später sehen, dass sie zur Ausreißereliminierung verwendet wird.

In dieser Arbeit gehen wir, wie in Kap. 1 bereits erwähnt, von einem kalibrierten Scansystem aus, d. h. dass \mathbb{P} sowohl für die Kamera als auch für den Projektor bekannt ist.

Schnitt der Streifenebenen mit den Sichtgeraden

Ausgehend vom obigen Kamera-Modell können wir nun die Gleichung der projizierten Streifenübergänge angeben, um mittels eines Schnitts einer Ebene in impliziter Darstellung mit einer Geraden einen triangulierten Punkt zu erzeugen.

Aus der Sicht des Projektors wird ein Punkt u_p , der auf der Projektorbildebene liegt, auf einen Punkt x_p auf der Objektoberfläche projiziert. Es gilt:

$$\rho u_p = \mathbb{P}_p x_p \quad , \quad (5.6)$$

wobei \mathbb{P}_p die Projektionsmatrix des Projektors ist.

Zusätzlich ist explizit eine vertikale Gerade $u_p(\eta)$ eines Streifenübergangs in der Projektorbildebene folgendermaßen gegeben:

$$u_p(\eta) = \begin{bmatrix} u_s \\ \eta \\ 1 \end{bmatrix} \quad ,$$

wobei u_s die horizontale Koordinate des Streifenübergangs in der Projektorbildebene darstellt und $\eta \in \mathbb{R}$. Daraus ergibt sich folgende implizite Darstellung dieses Streifenübergangs:

$$n^t u_p = 0 \quad , \quad (5.7)$$

wobei $n^t = [1 \ 0 \ -u_s]$. Wir können somit aus der impliziten Darstellung der Streifenübergangsgerade die projizierte Streifenübergangsebene erzeugen, indem wir Gleichung 5.6 in Gleichung 5.7 einsetzen:

$$n^t \mathbb{P}_p x_p = 0 \quad . \quad (5.8)$$

Sei nun x_c ein Raumpunkt, der auf u_c in der Kamerabildebene projiziert wird, dann gilt:

$$\rho u_c = \mathbb{P}_c x_c \quad , \quad (5.9)$$

wobei \mathbb{P}_c die Projektionsmatrix der Kamera ist.

Der Gerade-Ebene-Schnitt ergibt sich somit aus den Gleichungen 5.9 und 5.8, indem wir $x_c = x_p = x$ setzen:

$$\left. \begin{array}{l} n^t \mathbb{P}_p x = 0 \\ \mathbb{P}_c x = \rho u_c \end{array} \right\} \Leftrightarrow \underbrace{\begin{bmatrix} n^t \mathbb{P}_p \\ \mathbb{P}_c \end{bmatrix}}_{\mathbb{A}} x = \rho \underbrace{\begin{bmatrix} 0 \\ u_c \end{bmatrix}}_{\mathbb{b}} \quad .$$

Der triangulierte Raumpunkt x ist somit die Lösung von $\mathbb{A}x = \mathbb{b}$. Wir erhalten entsprechend die affinen Koordinaten dieses Punkts, indem wir x durch ihre letzte Komponente teilen, so dass $x = [x \ y \ z \ 1]^t$.

Durch die optische Triangulierung erhalten wir somit eine ausreißerbehaftete Punktwolke, wie die Visualisierung in Abb. 5.2 zeigt. Diese Ausreißer werden wir erst nach der Erzeugung des Dreiecksnetzes beseitigen, da noch keine topologische Informationen vorliegen.

Zusätzlich ist zu bemerken, dass die Verteilung der Ungenauigkeit eines Punkts x , wie in [Hebert et al. '93] gezeigt wurde, ellipsenförmig ist (siehe Abb. 5.3). Insbesondere liegt die Hauptachse dieser Ellipse in der projizierten Streifenebene zu x , da die Hauptfehlerquellen in der Kamerabildebene entstehen.

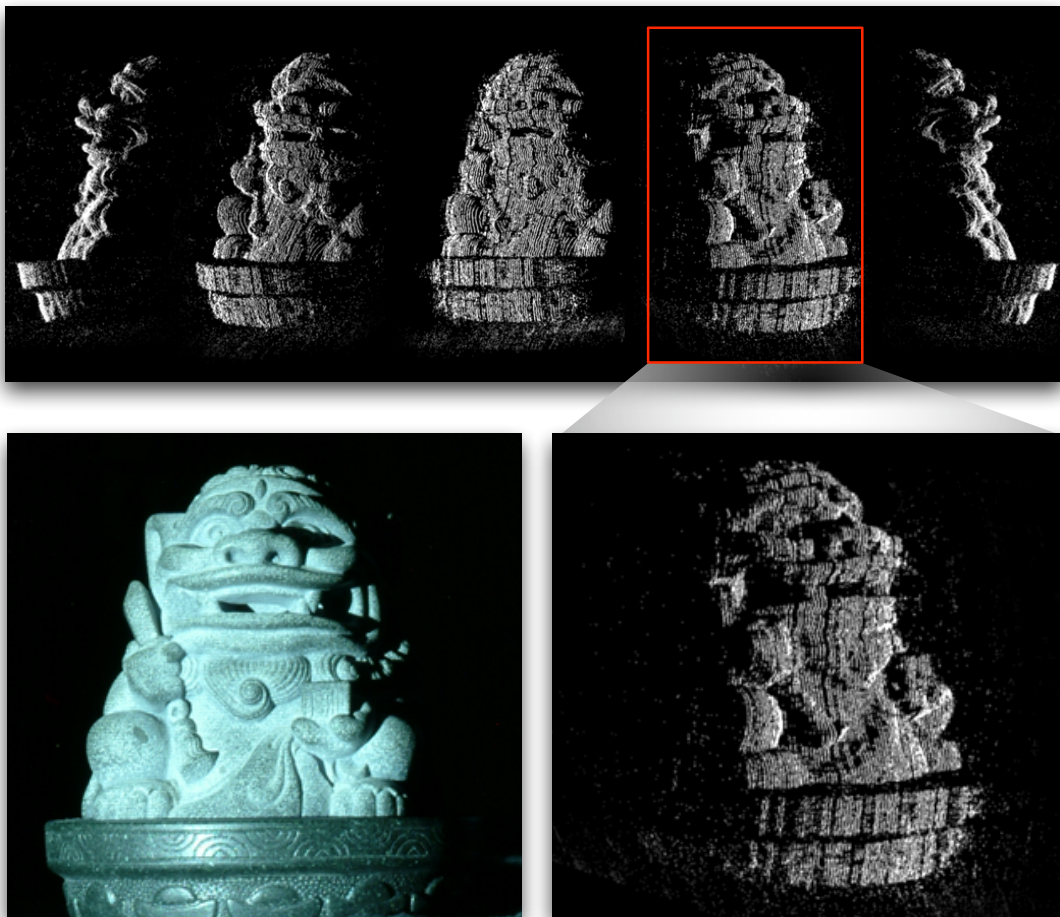


Abbildung 5.2: Unterschiedliche Ansichten einer optisch triangulierten Punktwolke. Hier können wir eine hohe Anzahl von Ausreißern, die aufgrund von Falschzuordnungen entstehen, beobachten.

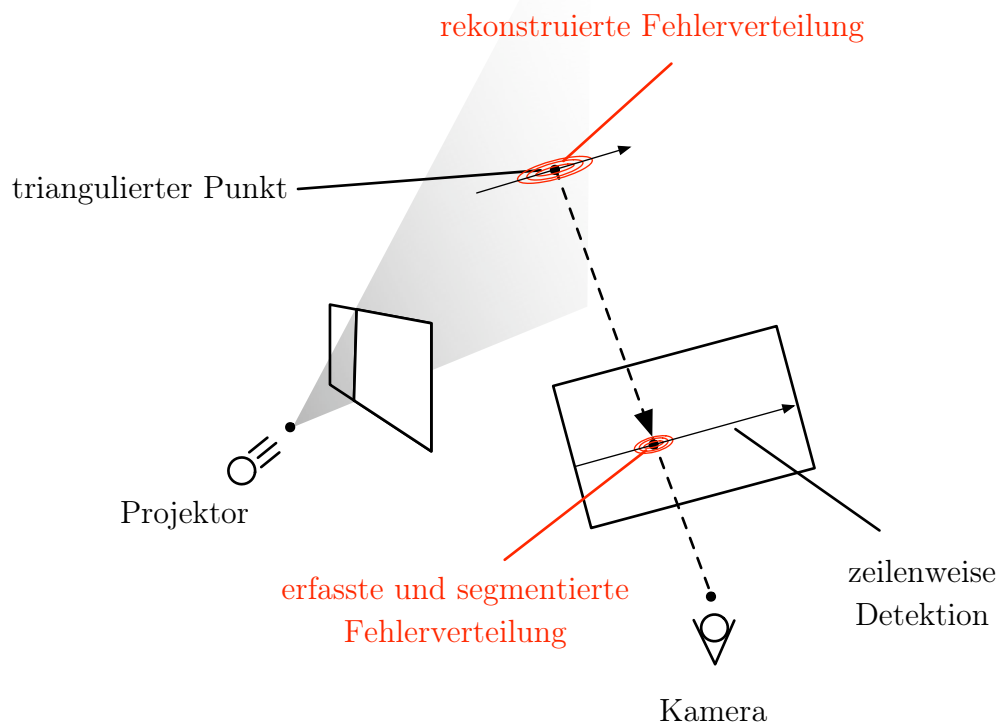


Abbildung 5.3: Ellipsenförmige Verteilung der Ungenauigkeit. Die Hauptachse dieser Ellipse liegt in der projizierten Streifenebene.

5.2 Dreiecksnetzerzeugung

Die Punktwolke, die wir im letzten Abschnitt durch optische Triangulierung erzeugt haben, liefert uns die abgetasteten geometrischen Daten der Objektoberfläche aus einer Kamera- bzw. Projektoransicht. Insbesondere wollen wir diese Punktwolke, die nur eine begrenzte Punktdichte besitzt, durch ein Dreiecksnetz approximieren, das letztendlich eine orientierbare, kompakte und zusammenhängende Fläche $\mathcal{S} \subset \mathbb{R}^3$ repräsentiert. Dadurch, dass ein Dreiecksnetz nur eine stückweise lineare 2-Mannigfaltigkeit darstellt (C^0 -stetig), besteht die Aufgabe nun darin, aus dieser Punktmenge (C^{-1} -stetig) eine konsistente Dreiecksnetztopologie (Konnektivität) zu finden, die zu einer guten Annäherung von S durch das Dreiecksnetz führt und vor allem keine Degenerierungen, wie etwa Selbstdurchdringungen, besitzt. Dieser Prozess wird im Unterschied zur optischen Triangulierung, die bisher behandelt worden ist, ebenfalls Triangulierung genannt. In dieser Arbeit wird zu Vermeidung der Begriffsverwirrung immer auf die optische Triangulierung bezogen.

Die Hauptschwierigkeiten, die bei Punktwolkenvernetzungen auftauchen, können wir folgendermaßen zusammenfassen:

- **Rauschen:** Durch die ungenaue Bilderfassung und der vielen Bildverarbeitungsschritte verlieren die lokalisierten Kanten an Genauigkeit. Diese führen zu einer verrauschten triangulierten Punktwolke. Insbesondere verlaufen dabei die größten Abweichungen, wie wir im vorherigen Abschnitt gesehen haben, entlang der Sichtgeraden des Projektors. Aufgrund der hohen Kameraauflösung, die wir verwenden, haben wir bei der Erfassung eine besonders hohe Abtastrate. Dies führt dazu, dass der Signal-zu-Rauschverhältnis schon bei kleinen Ungenauigkeiten stark sinkt. Es ist somit schwierig die Nachbarschaft eines Punkts korrekt abzuschätzen, um eine konsistente lokale Konnektivität zu finden. Außerdem können Details, die kleiner als das Rauschen sind, nicht mehr rekonstruierbar sein. Rauschen und Ungenauigkeiten sind in allen Scansystemen vorhanden.
- **Ausreißer:** Im Gegensatz zum Rauschen, entstehen Ausreißer durch Falschzuordnungen der Punkte. Sie werden dadurch charakterisiert, dass ihre Abweichungen in der Regel besonders groß sind. Dies liegt daran, dass die Sichtgerade eines erfassten Streifenübergangspunkts mit einer falschen projizierten Ebene geschnitten wird. In diesem Fall verläuft der Fehler entlang der Kamerasichtgerade. Wie wir in Abb. 5.2 sehen können, ist unsere erzeugte Punktwolke bei Objekten mit schlechten Reflexionseigenschaften stark von Ausreißern betroffen, da eine erhöhte Unsicherheit bei der Zuordnung entsteht. Ausreißer müssen daher identifiziert und beseitigt werden, ohne dass Löcher in der Rekonstruktion entstehen. Ausreißer treten vor allem bei Scanverfahren auf, die wie hier, auf eine schnelle Erfassung setzen, wodurch Mehrdeutigkeiten vorkommen.
- **Unregelmäßige Punktdichte:** Nicht homogen abgetastete Punkte stellen ein weiteres Hindernis für eine topologisch konsistente Vernetzung dar. An Stellen, wo die Abtastdichte nicht ausreichend ist, wird es schwierig zu erkennen, ob es sich tatsächlich um ein Loch im Objekt handelt oder nicht. Außerdem ist es schwierig, eine konsistente Nachbarschaft zu finden, so dass das daraus entstehende Dreiecksnetz keine Selbstdurchdringungen enthält. Wir haben in Kap. 2 bereits die Anisotropie unserer Abtastrate in vertikaler und horizontaler Richtung festgestellt. Diese Unzulänglichkeit kommt vor allem bei Scanverfahren vor, die mit einem Projektor strukturierte Streifen projizieren.
- **unbekannte Orientierung:** Eine globale konsistente Orientierung eines Dreiecksnetzes spielt eine wichtige Rolle bei der Nachverarbeitung von Dreiecksnetzen, vor allem wenn es darum geht Ausreißer durch ihre Orientierung

zu entfernen. Auch bei vielen Glättungsverfahren, ist die Orientierung wichtig, da dazu die Oberflächenkrümmung bestimmt werden muss. Während in den Punktwolken alleine keine explizite Angabe über eine Orientierung vorliegt, haben wir den Vorteil, dass wir aufgrund der Scankonfiguration eine besonders robuste Abschätzung über die Orientierung machen können.

Zur Erzeugung von Dreiecksnetzen aus Punktwolken, sind bereits viele Arbeiten entstanden. Bei den meisten Verfahren müssen strenge Voraussetzungen erfüllt werden, da nur die geometrischen Informationen in den Punktwolken vorliegen. Diese Methoden sind besonders empfindlich gegenüber den oben genannten Unzulänglichkeiten und haben meistens eine relativ hohe Laufzeit. Bei Delaunay-basierten Verfahren [Boissonnat '84, Edelsbrunner & Mücke '94] muss beispielsweise vorausgesetzt werden, dass die Punktdichte einigermaßen regelmäßig ist. Auch bei den gegenüber unregelmäßigen Punktwolken robusteren Verfahren von Amenta et al. [Amenta et al. '01] muss das Signal-zu-Rauschverhältnis hoch genug sein und es dürfen keine Ausreißer existieren, da eine Interpolation der Punktdaten vorgenommen wird. Ausreißer müssen daher durch einen Vorverarbeitungsschritt entfernt werden. Verfahren die eine lokale Flächenparametrisierung über die Nachbarschaft von Punkten annehmen [Bernardini et al. '99, Gopi et al. '00, Linsen '01] sind ebenfalls von verrauschten Daten betroffen, da die Abschätzung einer lokalen Topologie dadurch erschwert wird (siehe Abb. 5.6). Durch die implizite Darstellung der gesuchten Fläche mittels einer vorzeichenbehafteten Distanzfunktion, zeichnen sich volumetrische Methoden [Hoppe '95] vor allem durch ihre Robustheit gegenüber Ausreißern und Rauschen aus. Bei diesen Verfahren wird ein Dreiecksnetz mit Hilfe des Marching-Cubes-Algorithmus [Lorensen & Cline '87] aus einer Voxel-Struktur extrahiert. Dieser Ansatz führt zu starken Abweichungen von der Objekt Oberfläche, wenn eine zu grobe Voxel-Struktur gewählt wird.

Wie auch in [Curless '97], wird das Problem der unbekanntenen Topologie in dieser Arbeit stark vereinfacht. Dadurch, dass die Entstehung der Punktwolke und die Scankonfiguration vollständig bekannt sind, sind wir in der Lage, wichtige A-Priori-Informationen auszunutzen, die wichtige Aussagen bezüglich der Topologie und der Orientierung, sowie der räumlichen Kohärenz implizit enthalten. Wir werden uns nun diese Aussagen näher anschauen:

- **Topologieinformation:** Aufgrund der optischen Triangulierung können wir einen zugeordneten Punkt im Raum zurück auf die Kamera- und Projektorbildebene projizieren und unabhängig voneinander betrachten. Auf der Kamerabildebene können wir, aufgrund der zeilenweisen Kantendetektion, für jeden zugeordneten Punkt sagen, in welcher Zeile sich dieser Punkt befindet. Somit wäre die Topologie für alle Punkte der Kamerabildebene, die demselben Streifenübergang zugeordnet sind, in vertikaler Kamerarichtung gegeben. Auf der Projektorbildebene liegt ein zugeordneter Punkt genau auf einem der

projizierten Streifenübergänge. Da die projizierte Streifensequenz bekannt ist, können wir somit, von der Projektorbildebene aus, die Nachbarschaft von zugeordneten Punkten bestimmen. Aufgrund der Epipolarbeschränkung, die wir in Kap. 3 behandelt haben, können wir diese horizontale Topologieinformation direkt auf die Kamerabildebene übertragen. Insgesamt gibt uns die Information der Zuordnung und der Scankonfiguration eine konsistente Topologie in horizontaler und vertikaler Richtung von der Kamerabildebene aus. Insbesondere reduziert sich dadurch die Vernetzung in ein zweidimensionales Problem, da nur noch auf der Kamerabildebene ein Dreiecksnetz gefunden werden muss, das die gegebene Konnektivität nicht verletzt. Außerdem entstehen keine Selbstdurchdringungen bei der Netzerzeugung, da unter Berücksichtigung der gegebenen Topologie keine Selbstdurchdringungen in der Bildebene vorkommen können.

- **Orientierungsinformation:** Wie bereits angedeutet, können wir aufgrund der bekannten Lage der Kamera oder des Projektors eine global konsistente Orientierung eines Dreiecksnetzes bestimmen, indem wir alle Dreiecke passend zur Normale der Kamera- oder Projektorbildebene orientieren. Dies ist klar, da nur eine Seite der rekonstruierten Fläche von Kamera und Projektor aus sichtbar ist.
- **Räumliche Kohärenz:** Mit lokaler Raumkohärenz der Objektoberfläche ist insbesondere eine anisotrope lokale Stetigkeit gemeint, d. h. dass auf der Kamerabildebene weder in horizontaler noch in vertikaler Richtung kleine Löcher entstehen dürfen. Dadurch, dass wir eine höhere Abtastung der Oberfläche in vertikaler Richtung haben, ist auch die Wahrscheinlichkeit höher, dass aufgrund von nicht detektierten Streifenübergängen in einer Zeile, kleine Löcher in vertikaler Richtung entstehen. Deswegen müssen wir auch bei mehreren fehlenden Streifenübergängen derselben Zuordnung in aufeinander folgenden Zeilen annehmen, dass diese miteinander verbunden sind. In horizontaler Richtung der Kamerabildebene ist diese Annahme, wegen der Streifenbreite, schwächer. Insbesondere wenn in einer Zeile der Kamerabildebene zwei benachbarte Streifenübergänge zwei projizierten Streifenübergänge zugeordnet sind, die in der projizierten Sequenz aber nicht nahe genug beieinander liegen, können wir annehmen, dass es sich um einen Loch handelt. Bei der Dreiecksnetzerzeugung müssen diese Kriterien berücksichtigt werden, um zwischen Löchern in der Objektoberfläche und einer verfehlten Abtastung zu unterscheiden.

In [Curless '97] wird z. B. nur die räumliche Kohärenz berücksichtigt, indem die triangulierte Punktwolke auf der Kamerabildebene durch die Bestimmung der nächsten Nachbarn zweidimensional vernetzt und die Konnektivität auf die Punktwolke übertragen wird. Dieses Verfahren ist nur möglich wenn eine regelmäßige Abtastung in horizontaler und vertikaler Richtung der Kamerabildebene vorliegt. In

dieser Arbeit wird ein Netzerzeugungsverfahren vorgestellt, das alle oben vorgestellten A-Priori-Informationen gezielt ausnutzt, um ein konsistentes Netz in linearer Zeit zu erzeugen. Insbesondere können Ausreißer durch die Netzerzeugung identifiziert und beseitigt werden. Im Gegensatz zu [Curless '97] verwenden wir nicht die räumliche Kohärenz, um die Topologie zu extrahieren, sondern um Löcher während der Netzerzeugung zu überbrücken. Die Topologie ist ausschließlich durch unsere bekannte Zuordnung und die Scankonfiguration gegeben.

Vernetzungsalgorithmus

Der Vernetzungsalgorithmus wird in zwei Schritte aufgeteilt, die sich jeweils in einer anderen Datenstruktur abspielen. Zunächst wird ein Vorverarbeitungsschritt durchgeführt bevor das eigentliche Dreiecksnetz durch einen inkrementellen Prozess erzeugt wird.

Im ersten Schritt wird die horizontale lokale Raumkohärenz ausgenutzt um festzustellen, ob es sich bei einer unvollständigen Zuordnung um einen fehlenden Streifenübergang handelt oder tatsächlich um ein Loch in der Objektoberfläche. Falls es kein Loch im Objekt ist, werden redundante Punkte durch Interpolationen eingefügt, da wir später sehen werden, dass eine Dreiecksnetzerzeugung nur zwischen benachbarten projizierten Streifenübergängen stattfindet. Wir gehen somit von einer Datenstruktur aus, die vom Zuordnungsalgorithmus ausgegeben wird. Diese Datenstruktur wird durch folgende Mengen

$$\mathcal{H}_r = \{i_1^{(r)}, \dots, i_{|\mathcal{H}_r|}^{(r)}\} \quad , \quad r = 1, \dots, R$$

beschrieben, wobei R die Anzahl der Zeilen des Kamerabildes darstellt. $i_l^{(r)} \in \{1, \dots, N - 1\}$ ist der Index der projizierten Streifensequenz aus $N - 1$ Streifenübergängen der l -ten Zuordnung in der Zeile r . Beispielsweise stellt $i_2^{(4)} = 3$ den dritten projizierten Streifenübergang in der Streifensequenz, der in der 4. Zeile als zweites zugeordnet wurde. Insbesondere gilt i. Allg. nicht, dass $i_l^{(r)} < i_{l+1}^{(r)}$, da keine Monotonizitätsannahme vorausgesetzt wird. Zusätzlich sei für jedes $i_l^{(r)}$, $\mathbf{x}_{l,r}$ der zugehörige Punkt im Raum. Diese Datenstruktur wird in der Praxis beispielsweise durch eine Liste von verketteten Listen implementiert.

Weiterhin wird ein Schwellwert $\sigma_h \in \mathbb{N}$ festgelegt. Wenn in einer Zeile der Kamerabildebene zwei benachbarte Streifenübergänge zwei projizierten Streifenübergängen zugeordnet sind, die durch weniger als σ_h projizierte Streifenübergänge getrennt werden, handelt es sich nicht um ein Loch im Objekt. Diesen Prozess können wir folgendermaßen formal beschreiben:

Für alle $r = 1, \dots, R$:

1. Für alle $l = 1, \dots, |\mathcal{H}_r| - 1$
 - (a) Berechne $i_{\text{diff}} = i_{l+1}^{(r)} - i_l^{(r)} - 1$.

- (b) Falls $i_{\text{diff}} \leq \sigma_h$, dann handelt es sich nicht um einen Loch im Objekt und es werden i_{diff} Punkte linear interpoliert zwischen $\mathbf{x}_{l,r}$ und $\mathbf{x}_{l+1,r}$ eingefügt. Der k -te eingefügte Punkt hat somit die Koordinate:

$$\mathbf{x}_k = \frac{(i_{\text{diff}} + 1 - k)}{i_{\text{diff}} + 1} \mathbf{x}_{l,r} + \frac{k}{i_{\text{diff}} + 1} \mathbf{x}_{l+1,r} \quad .$$

2. Aktualisiere \mathcal{H}_r entsprechend der neu eingefügten Punkte.

Dieser Schritt stellt somit sicher, dass in jeder Zeile eine lokale Stetigkeit vorhanden ist. Wir bemerken, dass das Einfügen von linear interpolierten Punkten nicht zu Punkten führen kann, die von der Kamera aus nicht sichtbar sind.

Der zweite Schritt ist der Netzerzeugungsschritt. Hier wird zunächst die vorherige, gegebenenfalls aktualisierte Datenstruktur transponiert. Die neue Datenstruktur wird somit durch folgende Mengen beschrieben:

$$\mathcal{V}_i = \{r_1^{(i)}, \dots, r_{|\mathcal{V}_i|}^{(i)}\} \quad , \quad i = 1, \dots, N - 1 \quad ,$$

wobei hier insbesondere $r_h^{(i)} < r_{h+1}^{(i)}$ für $h = 1, \dots, |\mathcal{V}_i|$ gilt. $r_h^{(i)}$ gibt die Zeile der h -ten Zuordnung mit dem i -ten projizierten Streifenübergang an. Beispielsweise bedeutet hier $r_2^{(3)} = 4$, dass sich die zweitkleinste Zeile einer Zuordnung mit dem dritten projizierten Streifenübergang in der vierten Zeile befindet. \mathcal{V}_i wird ebenfalls durch eine Liste von verkettete Listen implementiert. Man erhält sie durch eine aufsteigende Sortierung der Elemente von \mathcal{H}_r für $r = 1, \dots, R$. Hier wird der zu $r_h^{(i)}$ zugeordnete Punkt mit $\mathbf{x}_{h,i}$ bezeichnet.

Weiterhin wird ein zweiter Schwellwert $\sigma_v \in \mathbb{N}$ festgelegt, der die maximale Anzahl angibt, bei der aufeinander folgende fehlende Zeilen eines selben projizierten Streifenübergangs überbrückt werden sollen. Insbesondere setzt die Überbrückung eine vertikale lokale Stetigkeit voraus. In der Regel ist aufgrund der Streifenbreite $\sigma_v \geq \sigma_h$.

Der Algorithmus sieht somit folgendermaßen aus:

Seien h_l und h_r die betrachteten linken und rechten Zeilen von zwei benachbarten projizierten Streifenübergängen (siehe Abb. 5.4). Sie repräsentieren jeweils die unteren beiden Ecken eines betrachteten Dreiecks. Weiterhin sei d_l die Höhe eines Dreiecks, dessen obere Ecke auf dem linken projizierten Streifenübergang liegt und d_r entsprechend diejenige auf dem rechten. Diese zwei Dreiecke bilden somit zwei unterschiedliche Konfigurationen, die wir mit linker und rechter Konfiguration bezeichnen.

Für alle $i = 1, \dots, N - 2$:

Setze $h_l := 0$ und $h_r := 0$

1. Solange es auf beiden benachbarten projizierten Streifenübergängen noch zugeordnete Punkte gibt, d. h. $h_l \leq |\mathcal{V}_i|$ oder $h_r \leq |\mathcal{V}_{i+1}|$:

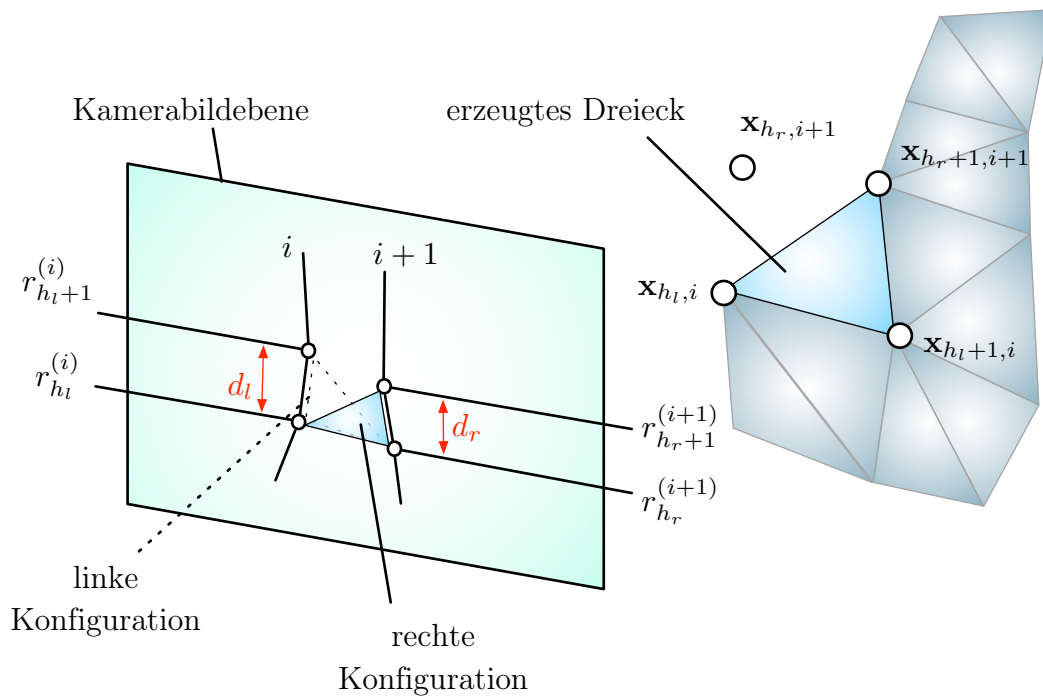


Abbildung 5.4: Die Dreiecksnetzerzeugung.

- (a) Berechne die Höhe der Dreiecke beider Konfigurationen:

$$d_l = \max\{r_{h_l+1}^{(i)} - r_{h_l}^{(i)}, r_{h_l+1}^{(i)} - r_{h_r}^{(i+1)}\}$$

$$d_r = \max\{r_{h_r+1}^{(i+1)} - r_{h_r}^{(i+1)}, r_{h_r+1}^{(i+1)} - r_{h_l}^{(i)}\} \quad .$$

- (b) Falls die Höhe von einer der beiden Konfigurationen klein genug ist, d. h. $d_l \leq \sigma_v$ oder $d_r \leq \sigma_v$ dann kann ein Dreieck erzeugt werden:
- i. Falls die linke Konfiguration kleiner ist, d. h. $d_l < d_r$, erzeuge Dreieck $\mathbf{x}_{h_l+1,i}\mathbf{x}_{h_l,i}\mathbf{x}_{h_r,i+1}$ und setze $h_l := h_l + 1$.
 - ii. Sonst erzeuge Dreieck $\mathbf{x}_{h_r+1,i+1}\mathbf{x}_{h_r,i+1}\mathbf{x}_{h_l,i}$ und setze $h_r := h_r + 1$.
- (c) Sonst handelt es sich um ein Loch im Objekt und wir müssen es überspringen:
- i. Falls die untere linke Ecke beider Dreiecke tiefer liegt, als die untere rechte, d. h. $r_{h_l}^{(i)} < r_{h_r}^{(i+1)}$ dann erhöhe die untere linke Ecke mit $h_l := h_l + 1$.
 - ii. Sonst erhöhe die untere rechte Ecke $h_r := h_r + 1$.

Der Algorithmus erzeugt somit inkrementell Dreiecke, die sich jeweils zwischen zwei benachbarten projizierten Streifenübergängen befinden. Mit den Schwellwerten $\sigma_h = 2$ und $\sigma_v = 40$ konnten wir bei unseren Experimenten gute Ergebnisse

erzielen. Es ist zusätzlich zu bemerken, dass bei der Erzeugung eines Dreiecks, die Reihenfolge der Punkte die Orientierung der Fläche bestimmen kann.

Im zweiten Schritt können tatsächlich Flächen (bzw. Dreiecke) entstehen, die aufgrund von Überdeckungen von der Kamera aus nicht sichtbar sind. Die interpolierten redundanten Punkte befinden sich auch i. Allg. nicht auf den projizierten Streifenübergängen. Um dies sicherzustellen, müsste man die interpolierten Punkte als Schnittpunkte der Strecke zwischen den beiden erfassten Punkten mit den entsprechenden projizierten Streifenübergangsebenen wählen. In der Praxis konnten wir allerdings keine großen Unterschiede feststellen. Dadurch, dass wir später eine Glättung durchführen, bei der die Punkte verschoben werden, ist diese Genauigkeit sowieso nicht von hoher Bedeutung.

Wir können feststellen, dass das entstandene Dreiecksnetz zwar eine konsistente Topologie besitzt, aber dennoch stark verrauscht und ausreißerbehaftet ist. Dies liegt daran, dass die Netzerzeugung eine stückweise lineare Interpolation der verrauschten und mit Ausreißer behafteten Punkte konstruiert. Es müssen daher zunächst die Ausreißer entfernt werden, um anschließend das Rauschen mittels eines Glättungsverfahrens zu unterdrücken.

Ausreißerbeseitigung

Ausgehend vom erzeugten Dreiecksnetz, können wir, mit Hilfe zweier Standardmethoden, die Ausreißer eliminieren.

Die erste Methode besteht darin, alle Dreiecke zu entfernen, deren nach außen zeigende Seite vom Projektor und von der Kamera aus nicht sichtbar sind. Es werden somit alle von Kamera und Projektor abgewandten Dreiecke entfernt. Dies sind genau diejenigen Dreiecke, die zwischen ihrer nach außen zeigenden Normalen und der Richtung ihrer Sichtgeraden in Richtung des Augpunkts, einen Winkel größer als 90° haben. Um den Vektor in Richtung der Sichtgeraden zu berechnen, brauchen wir allerdings die Koordinaten des Augpunkts der Kamera und des Projektors. Diese können aus der entsprechenden Projektionsmatrix, wie in Abschnitt 5.1 gezeigt wurde, einfach extrahiert werden.

Mit der zweiten Methode werden alle Dreiecke entfernt, deren längste Kante größer als ein vorgegebener Schwellwert ist. Diese entarteten Dreiecke entstehen häufig durch Ausreißer in der Punktwolke.

In der Praxis besteht die gesamte Netzerzeugungspipeline aus vier Schritten. Es wird zunächst ein Dreiecksnetz erzeugt, um dann die abgewandten Dreiecke zu entfernen. Insbesondere entstehen dadurch neue Löcher bzw. lokale Unstetigkeiten. Im Anschluss wird wieder ein Dreiecksnetz erzeugt, so dass die lokale Stetigkeit wiederhergestellt wird. Zum Schluss werden Dreiecke mit langen Kanten, die größere Unstetigkeiten überbrücken, eliminiert. Wir sehen in Abb. 5.5, dass fast alle Ausreißer dadurch beseitigt werden.

Ein direkter Vergleich mit der Fächerwolkenrepräsentation aus [Linsen '01] zeigt, dass die Verwendung von A-Priori-Informationen, bei Ausreißern und unregelmäßigen Punktdichten, unabdingbar ist.

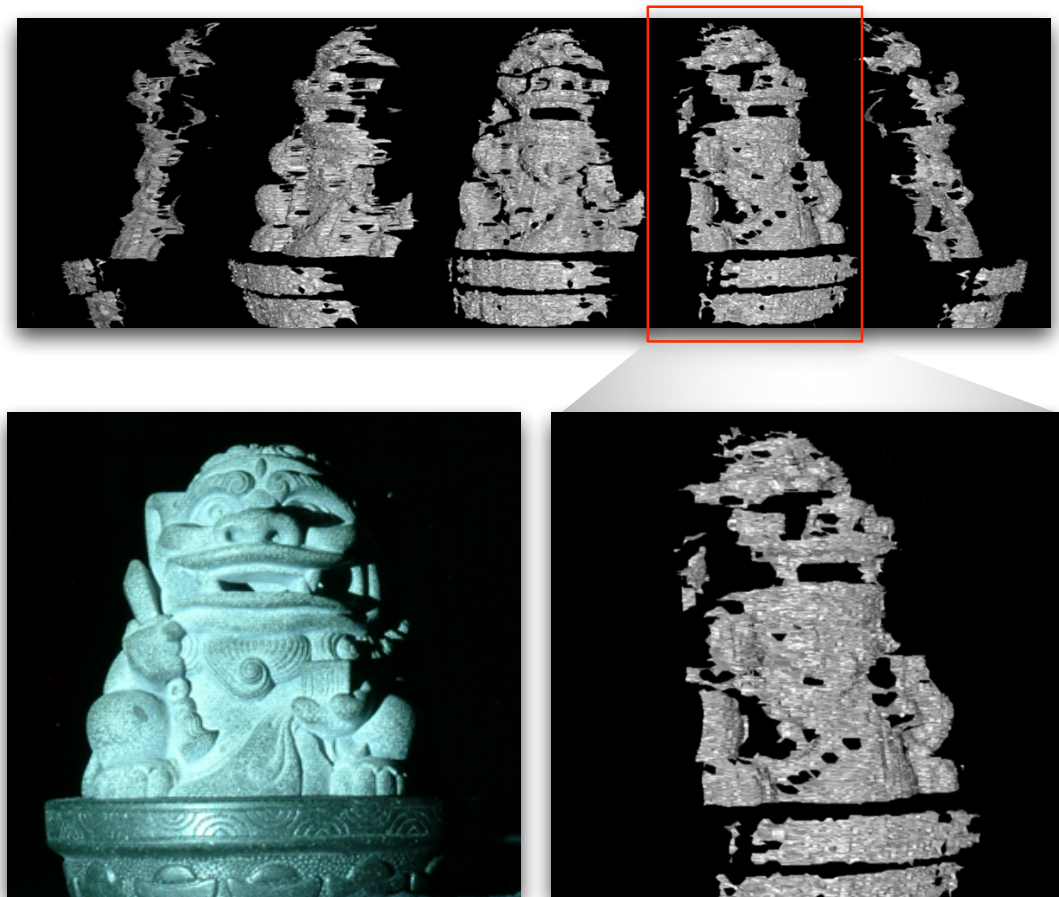


Abbildung 5.5: Unterschiedliche Ansichten eines erzeugten ausreißerbereinigten Dreiecksnetzes. Trotz der topologischen Konsistenz, ist weiterhin Rauschen zu beobachten.

5.3 Glättung

Wir haben im vorherigen Abschnitt gezeigt, wie ein Dreiecksnetz mit konsistenter Topologie aus einer Punktwolke in Streifenkonstellation erzeugt werden kann. Obwohl wir durch Entfernen von Dreiecken die Ausreißer beseitigt haben, ist das Dreiecksnetz weiterhin stark verrauscht (siehe Abb. 5.7a), da bislang keine Änderungen an der Geometrie der Punktwolke bzw. des Dreiecksnetzes vorgenommen wurden.

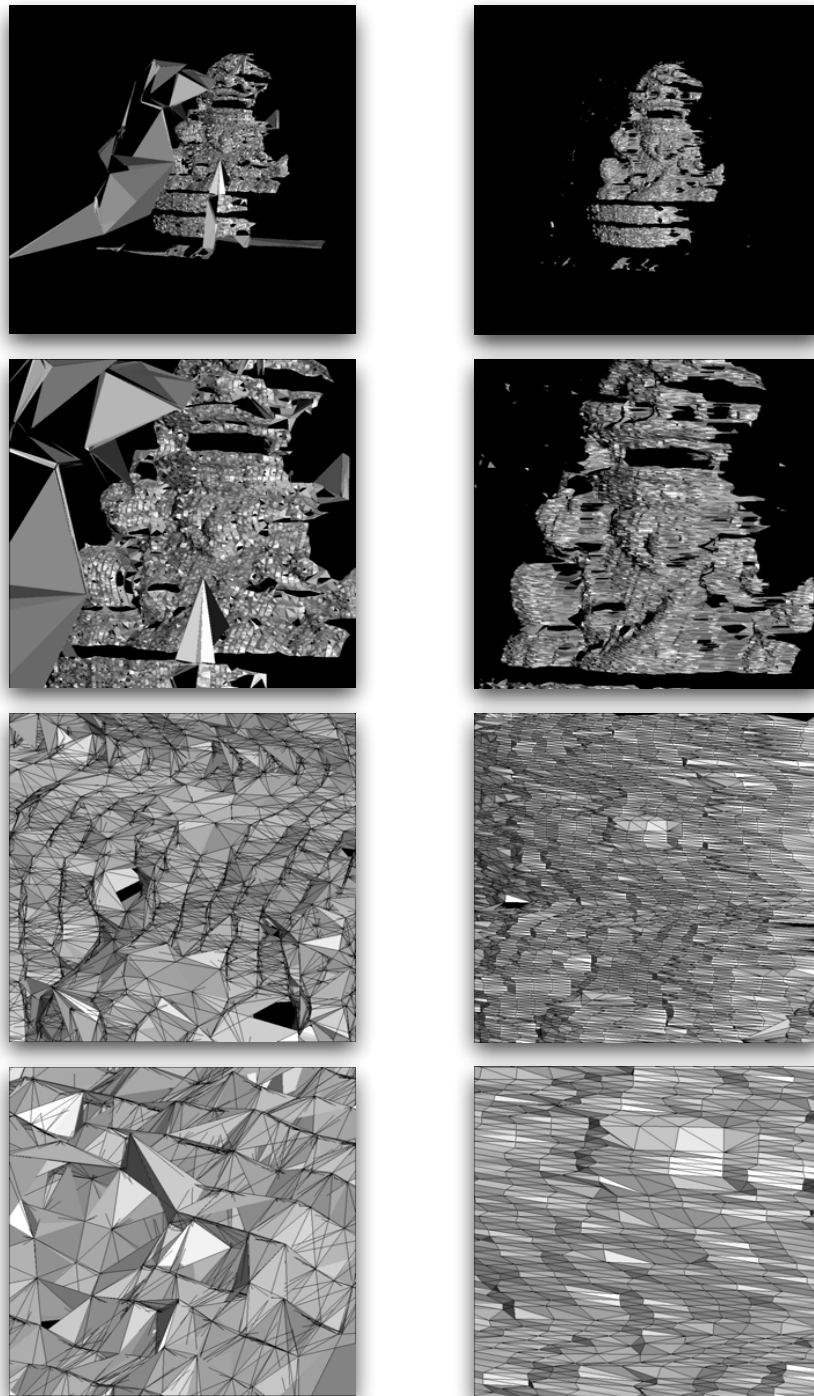


Abbildung 5.6: Links: Punktwolkendarstellung durch die Fächerwolken von [Linsen '01]. Rechts: Unsere Dreiecksnetzerzeugung unter Ausnutzung von A-Priori-Kenntnissen.

Polygonale Netze werden, in Analogie zu Bildern, üblicherweise als Signale betrachtet, wobei die Rauschanteile gerade den hochfrequenten Amplituden einer Spektralzerlegung entsprechen. Um das Rauschen zu unterdrücken, werden Glättungsverfahren eingesetzt. Aufgrund der Tatsache, dass unsere erzeugten Dreiecksnetze aus bis zu mehreren hunderttausend Punkten bestehen können, sind für uns nur die Algorithmen von Interesse, die einen linearen Zeit- und Speicheraufwand haben. Bei den Glättungsverfahren werden typischerweise nur die Punkte eines Dreiecksnetzes, jedoch nicht dessen Konnektivität, verändert. Die Herausforderung besteht nun darin, das Rauschen eines Dreiecksnetzes so zu eliminieren, dass geometrische Merkmale erhalten bleiben.

Wir beginnen diesen Abschnitt zunächst mit der Einführung von signaltheoretischen Konzepten für Dreiecksnetze, die von [Taubin '95, Taubin '00] inspiriert wurden. Anschließend werden einige bekannte Glättungsverfahren, die auf dem diskreten Laplace-Operator basieren, untersucht. Insbesondere werden wir die Vor- und Nachteile dieser Algorithmen und vor allem deren Auswirkungen auf unsere unregelmäßig abgetastete Punktwolke präsentieren. Dadurch, dass diese Glättungsverfahren, durch mehrmaligen Anwenden des Laplace-Operators, eine Konvergenz der Punkte in Richtung des Schwerpunkts hervorrufen, wird ein Schrumpfen des Netzes beobachtet. Um dies zu vermeiden, werden wir zum Schluss zwei Volumenerhaltungsstrategien diskutieren.

Signalanalyse von Dreiecksnetzen

Mit der Beobachtung, dass die traditionelle Fouriertransformation eines Signals als eine Linearkombination von Eigenvektoren des Laplace-Operators betrachtet werden kann, überträgt Taubin durch die Definition eines diskreten Laplace-Operators die Fourier-Spektralanalyse auf polygonale Netze.

Das diskrete Signal eines Dreiecksnetzes wird folgendermaßen durch seine Punkte dargestellt:

$$X = [\mathbf{x}_1, \dots, \mathbf{x}_n]^t \quad ,$$

wobei $\mathbf{x}_i \in \mathbb{R}^3$ der i -te Punkt darstellt und n die Anzahl der Punkte ist.

Die *Nachbarschaft 1. Ordnung* von \mathbf{x}_i in einem Dreiecksnetz ist die Menge

$$\mathcal{N}_i = \{j : (\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{E}\} \quad ,$$

wobei \mathcal{E} die Menge aller Kanten des Dreiecksnetzes ist. Insbesondere ist \mathbf{x}_i ein Nachbar von \mathbf{x}_j wenn $j \in \mathcal{N}_i$.

Weiterhin sei der diskrete Laplace-Operator folgendermaßen definiert:

$$\Delta \mathbf{x}_i := \sum_{j \in \mathcal{N}_i} w_{ij} (\mathbf{x}_j - \mathbf{x}_i) \quad ,$$

wobei

$$\sum_{j \in \mathcal{N}_i} w_{ij} = 1 \quad \text{und} \quad w_{ij} \geq 0 \quad . \quad (5.10)$$

Dadurch dass der Laplace-Operator eine lineare Transformation darstellt, können wir ihn auf das Signal X anwenden. Wir erhalten dadurch folgende Matrix:

$$\Delta X = (W - I) X =: -K \quad ,$$

wobei $W = (w_{ij})$ die Gewichtsmatrix und I die Einheitsmatrix ist. Insbesondere ist K symmetrisch. Da sich jede reelle symmetrische Matrix über \mathbb{R} diagonalisieren lässt, gilt

$$K = EDE^{-1} \quad ,$$

wobei D eine Diagonalmatrix ist, deren Diagonalelemente reelle Eigenwerte von K sind und die Komponenten von E die zugehörigen n -dimensionalen Eigenvektoren sind. Aufgrund der Diagonalisierbarkeit von K , bilden diese Eigenvektoren eine Basis des n -dimensionalen Raums, wodurch wir das Dreiecksnetzsignal X folgendermaßen zerlegen können:

$$X = EC \quad .$$

Intuitiv können wir beobachten, dass ein Laplace-Operator eine lokale Variation des Signals beschreibt, wodurch in natürlicher Weise die Frequenzen dieses Signals durch seine Eigenwerte gegeben werden. Es ergibt sich daraus eine Spektralzerlegung und E stellt die Fouriermatrix einer diskreten Fouriertransformation dar.

Der ideale Ansatz, Rauschen zu unterdrücken, besteht nun darin, im Frequenzraum die hochfrequenten Koeffizienten dieser Spektralzerlegung zu entfernen. Dadurch dass wir eine sehr hohe Anzahl von Punkten haben, ist selbst eine *schnelle Fouriertransformation* (engl. *fast fourier transform*) mit einer Laufzeit von $O(n \log n)$ nicht praktikabel.

Tiefpassfilterung

In [Taubin '95] werden die Punkte \mathbf{x}_i , analog zu einer Gauß-Filterung für parametrisierte Kurven, über eine lineare Interpolation ihrer lokalen Nachbarschaft \mathcal{N}_i repositioniert. Dieser Prozess wird solange iteriert, bis die gewünschte Glättung erreicht wird. Daraus entsteht der folgende Iterationsschritt für \mathbf{x}_i :

$$\mathbf{x}'_i = \mathbf{x}_i + \lambda \Delta \mathbf{x}_i \quad ,$$

wobei \mathbf{x}'_i die neuen Koordinaten des i -ten Punkts sind. $\lambda \in (0, 1)$ stellt eine Schrittweite dar, die verhindert, dass bei einer iterierten Anwendung Oszillationen im

Signal entstehen [Linsen '01]. Ein Iterationsschritt auf das gesamte Signal X kann somit durch folgende Faltung implementiert werden:

$$X' = (I - \lambda K)X = f(K)X \quad ,$$

wobei $f(K)$ die Transferfunktion eines linearen Tiefpassfilters repräsentiert. Insbesondere werden bei steigender Anzahl von Iterationsschritten, die Signalanteile zu Frequenzen ungleich Null unterdrückt und es entsteht ein Schrumpfungsprozess in Richtung des Schwerpunkts aller Punkte. Dieser Schwerpunkt stellt gerade den Signalanteil mit einer Frequenz gleich Null dar.

Die unterschiedlichen Varianten von Laplace-Glättungsverfahren unterscheiden sich vor allem durch die Wahl der Gewichte w_{ij} des Laplace-Operators. Die erste Variante ist die einer *uniformen Gewichtung* (Abb. 5.7b), wobei das Gewicht nur von der Anzahl der Nachbarn abhängt:

$$w_{ij} = |\mathcal{N}_i|^{-1} \quad .$$

Daraus entsteht der wohl bekannte *Umbrella-Operator*, der die Tendenz hat, alle Kanten des Dreiecksnetzes auf gleiche Länge zu bringen, wodurch ein Dreiecksnetz mit einer regelmäßigeren Punktdichte entsteht. Dies liegt daran, dass die Gewichte nur von der Dreiecksnetztopologie abhängen und die Interpolation über eine Mittelung der Nachbarschaft geschieht. Der Umbrella-Operator ist somit in unserem Fall ungeeignet, da wir, von der Kamera aus gesehen, längere horizontale Kantenlängen haben als vertikale Kantenlängen. Bereits nach wenigen Iterationsschritten konnten wir feststellen, dass viele Merkmale des Dreiecksnetzes verschwinden.

Fujiwara schlägt in [Fujiwara '95] vor, die unregelmäßigen Kantenlängen zu kompensieren, indem Gewichte über die inversen Kantenlängen definiert werden (Abb. 5.7c):

$$w_{ij} = \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^{-1}}{\sum_{j \in \mathcal{N}_i} \|\mathbf{x}_i - \mathbf{x}_j\|^{-1}} \quad .$$

Insbesondere werden hier die geometrischen Eigenschaften des Dreiecksnetzes ausgenutzt. Der daraus entstehende Laplace-Operator ist zwar unabhängig von der Kantenlänge der Dreiecke, berücksichtigt jedoch nicht unregelmäßig große Winkel der Dreiecke. Während wir bei unseren Dreiecksnetzen weniger ein Problem mit unregelmäßig großen Winkeln haben, konnten wir nach einer gewissen Anzahl von Iterationsschritten feststellen, dass viele Falten entstehen. Dies liegt daran, dass an vielen Stellen die vertikale Abtastdichte so hoch ist, dass sich die Punkte, aufgrund der hohen w_{ij} -Werte, zusammenziehen. Es ist zu bemerken, dass dieses Zusammenziehen vor allem eine Bewegung in Tangentialrichtung ist.

In [Desbrun et al. '99] werden Gewichte eingeführt, die sowohl die Kantenlängen als auch die Winkelgrößen berücksichtigen (Abb. 5.7d). Durch eine Approximie-

rung der mittleren Krümmung werden die Punkte dem Krümmungsfluss entsprechend bewegt. Daraus ergibt sich folgende Gewichtung:

$$w_{ij} = \frac{\cot \theta_j^l + \cot \theta_j^r}{\sum_{j \in \mathcal{N}_i} \cot \theta_j^l + \cot \theta_j^r} \quad ,$$

wobei θ_j^l und θ_j^r die beiden der Kante $\mathbf{x}_i \mathbf{x}_j$ gegenüberliegenden Winkel sind. Dadurch, dass hier nur noch die innere Geometrie der Fläche verwendet wird, haben die Bewegungen der Punkte insbesondere keine Tangentialkomponente. Somit sind auch in unseren Experimenten keine Falten mehr entstanden. Außerdem werden durch die kanten- und winkelgrößenerhaltenden Eigenschaften auch die Merkmale nach mehreren Iterationen besser erhalten, als bei den anderen beiden Verfahren.

Da bei den letzten beiden Verfahren nach jedem Iterationsschritt die Gewichte neu berechnet werden müssen, sind diese langsamer als der Umbrella-Operator. Trotz großer Datenmenge, terminierten beide jedoch in allen unseren Versuchen nach weniger als einer Minute. Wir haben in dieser Arbeit keine weiteren Glättungsverfahren mehr untersucht, da das letzte vorgestellte Verfahren unsere Ansprüche befriedigen konnte. Abbildung 5.8 zeigt mehrere Ansichten eines mit dem Krümmungsfluss geglätteten Dreiecksnetzes.

Volumenerhaltung

Wir haben gesehen, dass Laplace-Operatoren die unangenehme Eigenschaft haben, Dreiecksnetze nach jeder Iteration zu schrumpfen. Dies ist für nachfolgende Rekonstruktionsschritte, wie etwa die Registrierung oder die Netzintegration, besonders kritisch. Um entgegen zu wirken, verwendet Taubin eine Transferfunktion 2. Grades:

$$f(K) = (I - \lambda K)(I - \mu K) \quad ,$$

wobei $\mu < -\lambda$ gesetzt wird. Die erste Schrittweite λ dient somit weiterhin zur Glättung, während die zweite die verkleinerte Fläche wieder zurück vergrößert. Diese Methode wird auch $\lambda|\mu$ -Algorithmus genannt. In der Implementierung werden einfach zwei Laplace-Operatoren mit jeweils λ und μ als Schrittweite iterativ nacheinander ausgeführt, bis die maximale Anzahl von Iterationen erreicht wird. Da diese Entschumpfungsschritte wieder die hohen Frequenzanteile des Signals verstärken, glättet dieses Verfahren meistens sehr langsam, d. h. dass eine hohe Anzahl von Iterationen notwendig ist. Zusätzlich ist eine manuelle Feinjustierung der Schrittweiten notwendig, um eine optimale Lösung zu erzielen und es kann i. Allg. nicht das exakte Ausgangsvolumen erhalten werden.

Eine exakte Volumenerhaltungsmethode wurde in [Desbrun et al. '99] vorgestellt. Hierbei wird das Ausgangsvolumen V und das Volumen nach der Glättung V' berechnet, um hinterher das Dreiecksnetz wieder zurück auf das Ausgangsvolumen

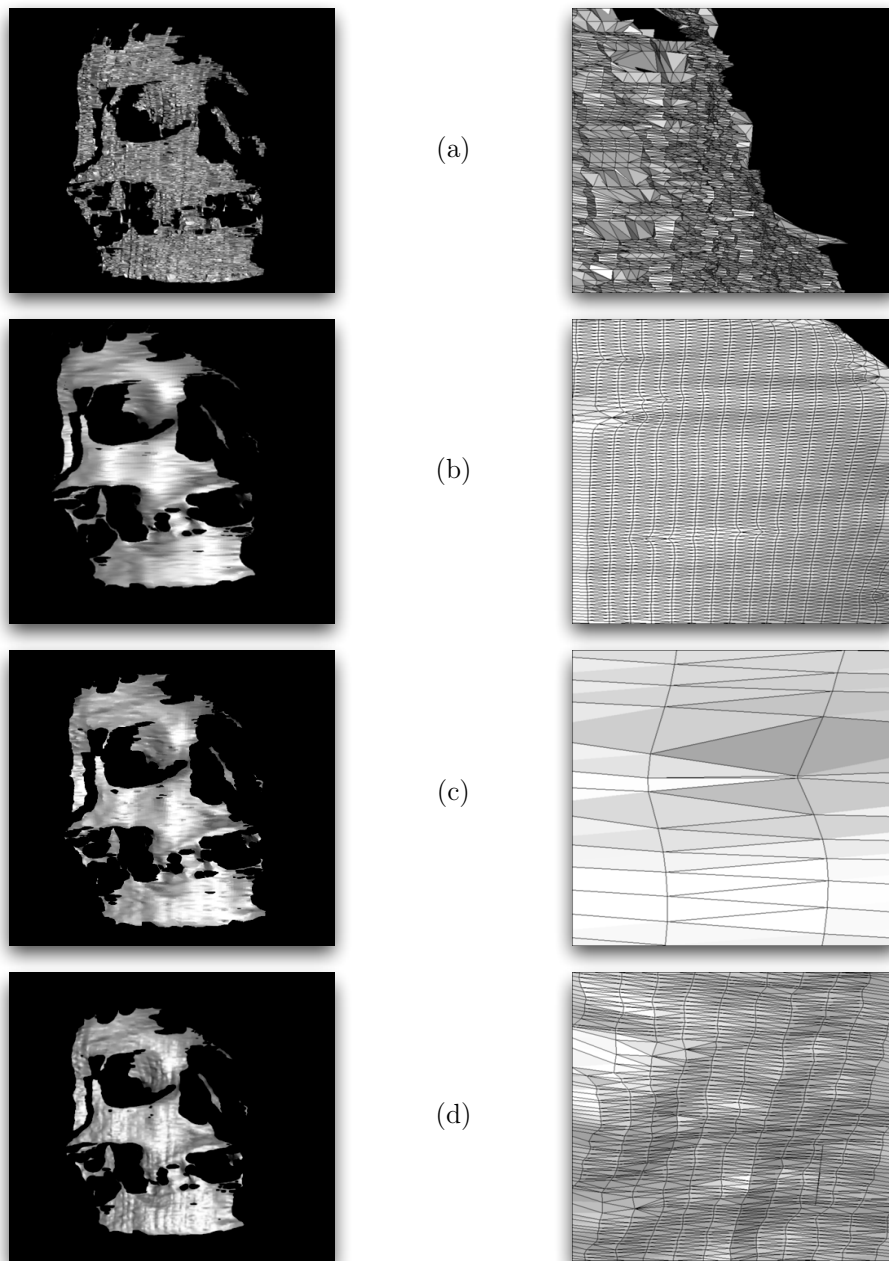


Abbildung 5.7: Laplace-Glättungen mit jeweils 20 Iterationen und $\lambda = 0,5$.
(a) Verrauschtes Dreiecksnetz. (b) Glättung mit uniformer Gewichtung. (c) Glättung mit inverser Kantenlänge. (d) Glättung durch Krümmungsfluss.

zu skalieren. Während Desbrun et al. das innere Volumen eines Dreiecksnetzes berechnen, betrachten wir das „äußere“ Volumen, das eine Aufsummierung aller

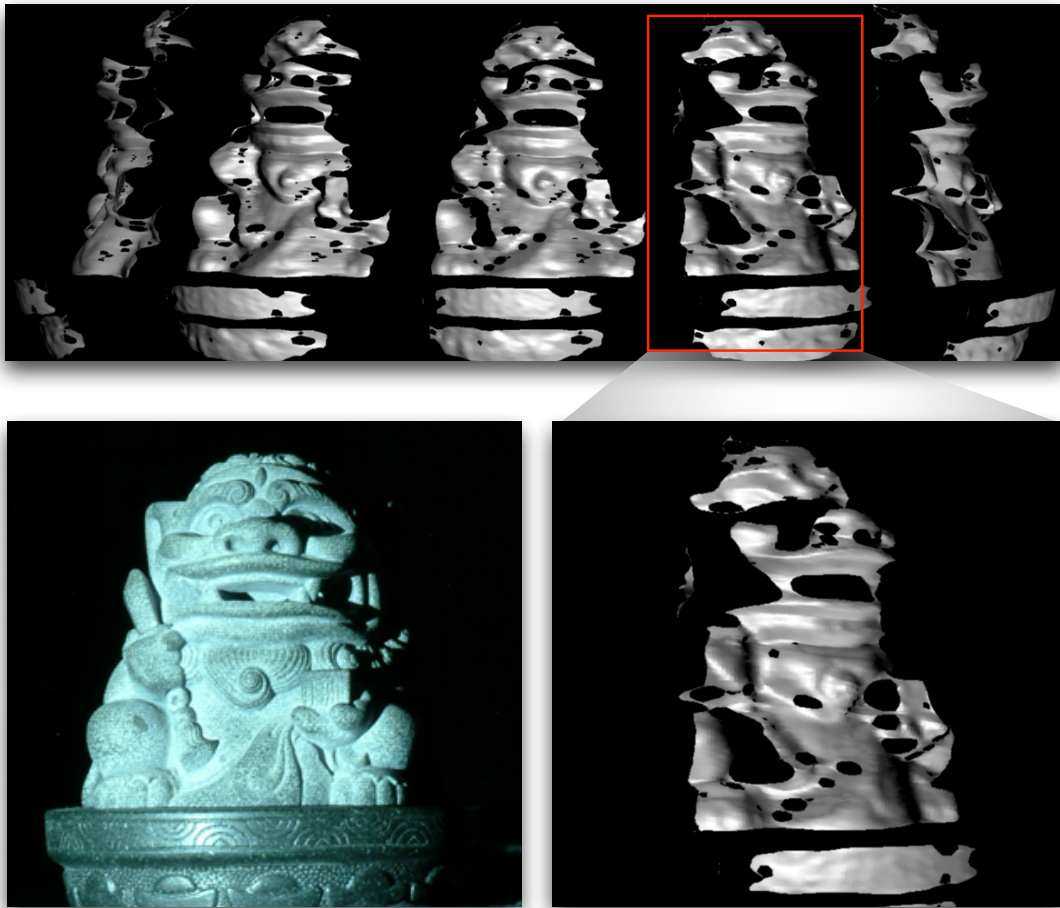


Abbildung 5.8: Unterschiedliche Ansichten eines mit dem Krümmungsfluss geglätteten Dreiecksnetzes. Die entstandenen Löcher werden durch einen nachfolgenden Integrationsschritt beseitigt, der in dieser Arbeit nicht behandelt wird.

Tetraeder darstellt, die jeweils aus den Eckpunkten der Dreiecke zusammen mit dem Projektoraugpunkt bestehen. Seien $\mathbf{x}_k^{(1)}, \mathbf{x}_k^{(2)}$ und $\mathbf{x}_k^{(3)}$ die Eckpunkte des Dreiecks des k -ten Tetraeders. Somit ist das gesamte Volumen V aller n Tetraeder wie folgt gegeben:

$$V = \frac{1}{6} \sum_{i=1}^n \bar{\mathbf{x}}_k^t \mathbf{n}_k \quad ,$$

wobei $\bar{\mathbf{x}}_k = \frac{1}{3}(\mathbf{x}_k^{(1)} + \mathbf{x}_k^{(2)} + \mathbf{x}_k^{(3)})$ und $\mathbf{n}_k = (\mathbf{x}_k^{(2)} - \mathbf{x}_k^{(1)}) \times (\mathbf{x}_k^{(3)} - \mathbf{x}_k^{(1)})$. Die Rückskalierung nach der Glättung, ergibt sich aus der Multiplikation aller Punktkoordinaten, bezüglich des Augpunkts, mit dem Skalierungsfaktor $\beta = \left(\frac{V}{V'}\right)^{\frac{1}{3}}$. Insbesondere kann dadurch sichergestellt werden, dass das exakte Ausgangsvolumen erhalten wird und dass keine Beeinträchtigung der asymptotischen Laufzeit hervorgerufen wird, da

das Volumen in linearer Zeit berechnet wird. Zusätzlich kommt diese Methode, im Gegensatz zum $\lambda|\mu$ -Algorithmus, ohne die Eingabe von zusätzlichen Parametern aus.

Wir bemerken zum Schluss, dass in der Praxis vermieden werden sollte, die Volumina der Tetraeder direkt aufzusummieren. Denn bei einer sehr großen Anzahl von Tetraedern (bzw. Dreiecken) werden ab einem gewissen Punkt sehr kleine Gleitkommazahlen auf eine sehr große Gleitkommazahl addiert. Aufgrund der begrenzten Anzahl von Bits in der *Mantisse*¹ entsteht dadurch ein numerisches Problem. Eine einfache Lösung, die wir implementiert haben, besteht darin, die Tetraedervolumina jeweils paarweise zu addieren, so dass immer Volumina betrachtet werden, die zur gleichen Größenordnung gehören. Diese numerisch stabile Aufsummierung der Tetraedervolumina fassen wir in Algorithmus 5.1 zusammen, wobei wir o. B. d. A. annehmen, dass n eine Zweierpotenz ist.

```

SummiereTetraederVolumina:  $(V_1, \dots, V_n)$ 
for  $l = 1, \dots, \log_2 n$ 
  for  $k = 1, \dots, \frac{n}{2}$ 
     $V_k = V_{2k-1} + V_{2k}$ 
  end for
end for
return:  $V_1$ 

```

Algorithmus 5.1: Numerisch stabile Aufsummierung von Tetraedervolumina.

¹Im Rechner wird bei der Darstellung von betragsmäßig sehr großen oder sehr kleinen Zahlen die Gleitkommazahldarstellung verwendet. Eine Zahl x entspricht einer halblogarithmischen Form $x = \pm m \cdot b^e$, wobei m die Mantisse, b die Basis und e der Exponent ist. Insbesondere gilt, dass nur eine begrenzte Anzahl von Bits für die Darstellung der Mantisse zur Verfügung steht.

Kapitel 6

Ergebnisse

In diesem letzten Kapitel, beschreiben wir zunächst unsere Implementierung. Im Anschluss präsentieren wir einige Ergebnisse unserer Rekonstruktionsverfahren.

6.1 Implementierung

Im Rahmen der Entwicklung eines flexiblen 3D-Scansystems spielt es eine besondere Rolle, auf spezialisierte Hardware zu verzichten und möglichst eine robuste Softwarelösung zu implementieren.

Ziel dieser Scansoftware ist es zunächst, die in Kap. 3 vorgestellten Streifenmuster zu erzeugen. Danach werden diese Muster auf das Objekt projiziert und es werden wenige Aufnahmen (i. d. R. drei) mit einer Digitalkamera gemacht. Ausgehend von den drei Aufnahmen, darunter zwei strukturiert beleuchtete und eine mit weißer Beleuchtung, erzeugt die Software ein geglättetes dreidimensionales Dreiecksnetz aus einer Kameraansicht. Die Ausgabe liegt in einem offenen Dateiformat vor und kann für weitere Schritte oder zur Visualisierung verwendet werden.

Während sich diese Arbeit überwiegend auf das Scanverfahren fokussiert, ist eine organisierte Entwicklungsumgebung unabdingbar für korrekte Ergebnisse und eine effiziente Implementierung. Wir haben in dieser Arbeit mehrere Softwaremodule von Grund auf programmiert, die alle, in den vorherigen Kapiteln vorgestellten, Verfahren implementieren. Insbesondere wurde im Gegensatz zu unserer vorhergehenden Software LIThIum [Li '04a] mehr Wert auf Modularität gelegt, um in Zukunft sehr einfach auch andere Datenstrukturen verwenden zu können, die für ein bestimmtes Verfahren besser geeignet wären. Die Wahl der Hardwareausstattung, sowie einen kurzen Überblick über den Entwurf der Software, werden in diesem Abschnitt zusammengefasst.

Hardwareumgebung

Die Hardwareausstattung und -eigenschaften wurden bereits in den Kapiteln 1 und 2 ausführlich behandelt. Vollständigkeitshalber fassen wir hier noch einmal die eingesetzten Geräten zusammen.

Wie auch bei den meisten optischen Triangulierungsverfahren mittels strukturiertem Licht, verwenden wir hauptsächlich einen Videoprojektor, eine Digitalkamera und ein Rechner für die Verarbeitung der Daten und für die Visualisierung. Der Videoprojektor ist ein portabler ASK M2 DLP-Projektor mit DVI- und VGA-Eingang und einer maximalen Bildauflösung von 1024×768 Pixeln (XGA). Die eingesetzte Kamera ist eine professionelle Fuji Finepix S2 Pro Spiegelreflex-Digitalkamera mit Super CCD-Technologie der 3. Generation und IEEE 1394a Firewire Ein-/Ausgang und einer maximalen Bildauflösung von 4256×2848 Pixeln (interpoliert) und 3024×2016 Pixeln (nicht interpoliert). Die Bilddaten können insbesondere im unkomprimiertem TIFF-RGB- oder RAW-Format vorliegen. Der Rechner, auf dem die Software implementiert wurde und das Dreiecksnetz erzeugt wird, ist mit einer 1.8 GHz PowerPC-970-FX-CPU und 768 MB RAM Hauptspeicher ausgestattet. Für die Visualisierung der rekonstruierten Fläche wurde eine nVidia-GeForce-FX-5200-Graphikkarte mit 64 MB VRAM eingesetzt.

Für einen automatisierteren Prozess bei längerem Scanbetrieb verwendeten wir einen zusätzlichen Server für die Ansteuerung der Kamera und des Projektors. Beim Server handelt es sich um einen 2.4 GHz-Pentium-4-Rechner mit 512 MB-RAM Hauptspeicher. Dabei werden die auf dem Client (der Rechner auf dem die Daten verarbeitet werden) erzeugten Streifenmuster im PNG-Format an den Server übertragen, der diese dann nacheinander projiziert und durch Ansteuerung der Kamera automatisch fotografiert. Auch die Blendeneinstellungen, sowie die Belichtungszeiten der Kamera können durch diesen Server angesprochen werden. Nach einem Scanvorgang werden die Aufnahmen an den Client zurückgeschickt und können dort für die Rekonstruktion des Objekts verwendet werden.

Softwareentwurf

Dadurch, dass wir uns in dieser Arbeit auf den Entwurf neuer Verfahren konzentrierten, war es weniger sinnvoll, eine vollständige Softwarelösung anzustreben, wie bei einer unserer vorhergehende Arbeiten [Li '04a, Li '04b]. Dennoch wurde eine geeignete Entwicklungsumgebung verwendet und eine sorgfältige Softwarearchitektur entworfen, so dass im Falle einer Umstrukturierung des Verfahrens nur einen Teilbereich des Programms betroffen wäre. Insbesondere werden bei unseren Spezifikationen oft redundante Daten ausgegeben, da die Eingaben von noch nicht entworfenen Teilverfahren unklar sind. Im Hinblick auf eine fertige Softwarelösung des Scanverfahrens, wurden die einzelnen Programmmodule so implementiert, dass sie leicht wiederverwendbar und erweiterbar sind. Auch auf eine Abhängigkeit von weiteren quelltextoffenen Projekten, wurde in dieser Arbeit ausdrücklich verzich-

tet, da auch bei bekannten Verfahren oft Modifikationen durchgeführt werden müssen und der Anpassungsaufwand dazu nicht immer leicht abzuschätzen ist.

Die Software wurde unter Mac OS X 10.3 entwickelt und es wurde Xcode 1.5 als integrierte Entwicklungsumgebung eingesetzt. Außerdem handelt es sich bei unserer Implementierung um reinen C-/Objective-C-Code [Duncan '02], der in *Cocoa*-Applikationen [Hillegass '02] übersetzt wurde. Insbesondere wurden komplexere Datenstrukturen und Verwaltungsmechanismen der einzelnen Module objektorientiert in Objective-C programmiert. Routinen, bei denen eine effektive Datenverarbeitung besonders wichtig ist, wie etwa Bildverarbeitungsprozeduren oder primitive geometrische Operationen, wurden prozedural in C geschrieben. Zusätzlich wurde, je nach Einsatzmöglichkeit, die zur Verfügung stehende AltiVec-Vektorrecheneinheit der CPU verwendet, um die Datenverarbeitung von hochauflösenden Bildern durch Parallelverarbeitung zu beschleunigen, wie z. B. die Faltsoperationen auf Bildern.

Die einzigen verwendeten Bibliotheken, auch *Frameworks* genannt, waren die ApplicationKit-, Foundation-, vImage- und vecLib-Frameworks. Insbesondere wurden für typische lineare Algebra Operationen meistens die optimierten BLAS-[Netlib '02] und LAPACK-C-Routinen [Anderson et al. '99], die sich im vecLib-Framework befinden, verwendet.

Aus softwaretechnischer Sicht, wurde die Architektur weniger nach einem *Von-oben-nach-unten*-Schema (engl. *top-down*) entworfen, wie es in [Li '04a] der Fall war. Grund dafür waren die fehlenden Kenntnisse einer festen Spezifikationen unserer neuen Verfahren zum Zeitpunkt der Implementierung. Hierbei war die schwache Typisierung der Programmiersprache Objective-C von besonderem Vorteil, da eine Umstrukturierung der Datenstrukturen weniger zeitaufwändig war.

Dadurch, dass wir noch keine graphische Oberfläche implementiert haben, war auch eine schichtenbasierte Struktur auf einer höheren Architekturebene nicht notwendig. Die Daten durchlaufen somit eine stark modularisierte Pipeline, wobei in jedem Zwischenschritt die entsprechenden Ergebnisse für unsere Analysen ausgegeben werden. Datenstrukturen und Ein- und Ausgabeklassen, die mehrfach eingesetzt werden, wurden alle separat in wiederverwendbaren Modulen implementiert.

Die einzelnen Module bestehen jeweils aus einer Vielzahl von Klassen und Funktionen und werden in Reihenfolge der Pipeline kurz vorgestellt:

- **Calibrate:** Dieses Modul beinhaltet das Rahmenwerk für die Kalibrierung des Scansystems und die Segmentierungsverfahren für das automatische Erkennen der Kalibrierebenen. Insbesondere beinhaltet sie die Klassen für das Lochkamera-Modell, das für die optische Triangulierung verwendet wird.
- **GeneratePattern:** Hier wird das in dieser Arbeit entwickelte Streifenmuster erzeugt und als Folge von Bildern im PNG-Format ausgegeben. Als Eingabe werden die Anzahl der Streifenmuster, der lokale Umgebungsradius der

Streifensequenz (siehe Kap. 3), die Ausgabebildauflösung und die Streifensequenzlänge spezifiziert. Zusätzlich ermöglicht sie die Anzahl der verwendeten Farben einzustellen, falls das zu scannende Objekt bestimmte Farben schlecht reflektiert.

- **ReidentifyStripeBoundary:** In diesem Modul, werden die in Kap. 4 vorgestellten Verfahren implementiert, d. h. die Bildvorverarbeitungsklassen und die Kantendetektionsklassen, sowie die Zuordnungsklassen. Für jeden Schritt können die Schwellwertparameter einzeln bestimmt werden, obwohl die voreingestellten Parameter in den meisten Fälle bereits gute Ergebnisse liefern. Hier wird oft auf die effizient implementierten C-Routinen aus dem *Utilities*-Modul zugegriffen, wie etwa die Kantendetektions- oder Medianfilterungsprozeduren. Alle Aufnahmen werden in einem *SessionInputPack*-Objekt geladen, worin alle nachfolgenden Bildverarbeitungsschritte ihre Ausgaben abspeichern und wiederverwenden können. Das *ImageRenderer*-Objekt aus *Utilities* gibt die Teilergebnisse in einem beliebigen Bildformat aus.
- **Triangulate:** Hier wird die optische Triangulierung durchgeführt und es werden entsprechend die Datenstrukturen für die Punktwolke definiert. Die Ausgabemethoden der Punktwolke im OFF-Format (Object File Format) werden direkt im Punktwolkenobjekt implementiert.
- **Tessellate:** Dieses Modul ist zuständig für die Dreiecksnetzerzeugung, die Ausreißerbeseitigung und das Glätten des Dreiecksnetzes. Hier werden insbesondere die Dreiecksnetzklassen definiert, die die Punktwolkenobjekte des vorherigen Moduls wiederverwenden. Die Ergebnisse werden ebenfalls im OFF-Format ausgegeben.
- **Utilities:** Hier werden alle Datenstrukturen und Funktionen definiert, die über mehrere Module wiederverwendet werden können. Dieses Modul wird insbesondere in Geometrie- und Bildverarbeitungsroutinen aufgeteilt.

6.2 Experimente

Wir stellen in diesem Abschnitt Rekonstruktionsergebnisse von zwei besonders herausfordernden Objekten vor. Auf beide Objekte wurden 200 Streifen projiziert und mit einer Auflösung von 3024×2016 Pixel erfasst. Die vollständigen Rekonstruktionen dieser Objekte dauern jeweils weniger als 5 Minuten.

Die Schwierigkeiten bei der Rekonstruktion des wachenden Löwens (Abb. 6.1b), lagen vor allem bei den schlechten Reflexionseigenschaften und bei der besonders kleinen Größe dieser Statue. In unserer vorhergehenden Arbeit [Li '04b] war die Zuordnung dieses Objekts, aufgrund der schwachen Kontraste an den Streifenübergängen, nicht möglich.

Der Schädel (Abb. 6.1a), stellt ein besonders interessantes Scanobjekt dar. Aufgrund der hohen Anzahl von Löchern und Selbstverdeckungen, konnten wir feststellen, ob sich unser Scanverfahren auch für geometrisch komplexe Objekte eignet.



(a)



(b)

Abbildung 6.1: (a) Der Schädel: Die Schwierigkeiten liegen vor allem bei der hohen Anzahl von Löchern und Selbstverdeckungen (b) Der wachende Löwe: Die schlechten Reflexionseigenschaften der Oberfläche erschweren die Zuordnung.

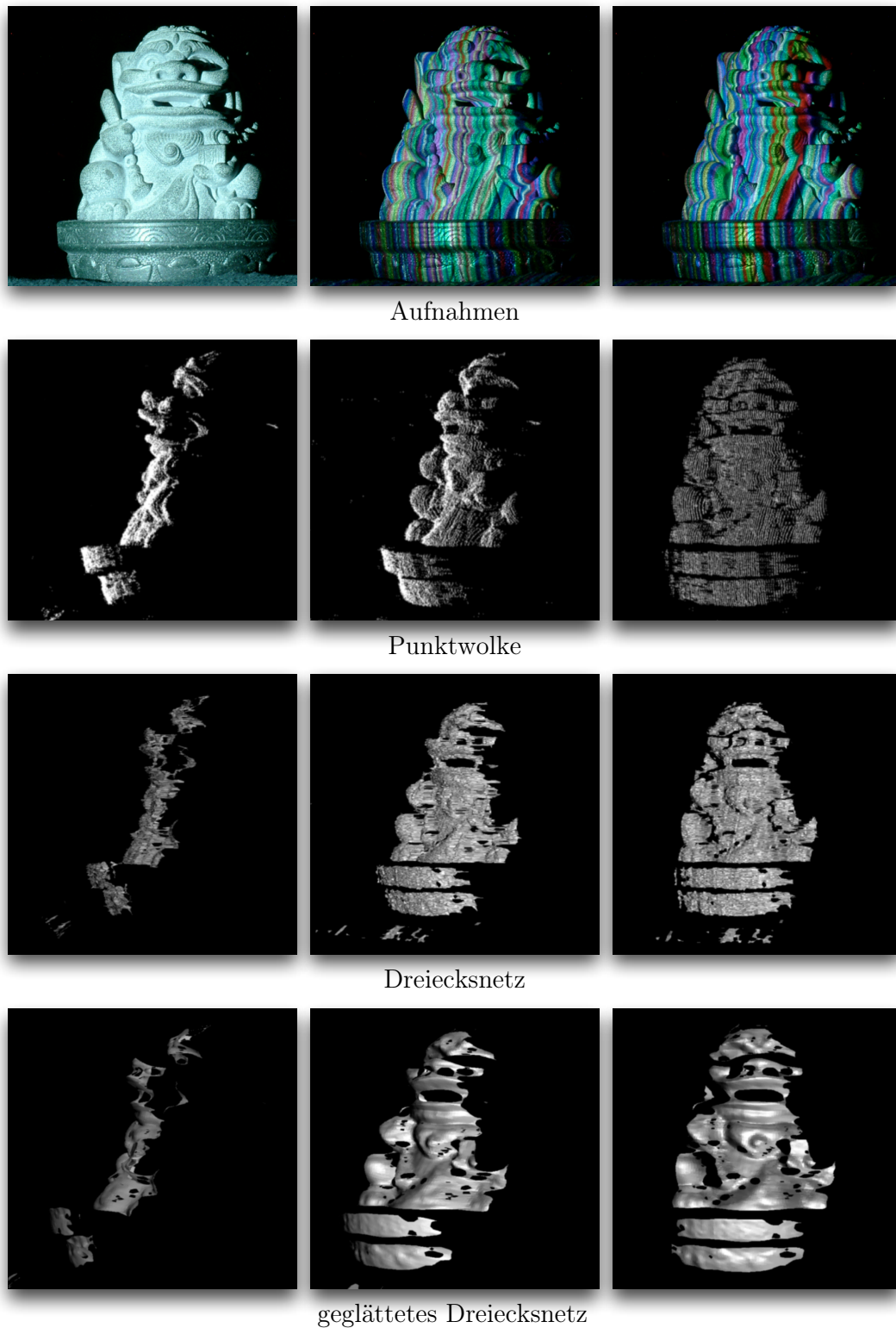


Abbildung 6.2: Rekonstruktionen des wachenden Löwens (Vorderseite). Er besteht aus 27897 Punkten und 52862 Dreiecken.

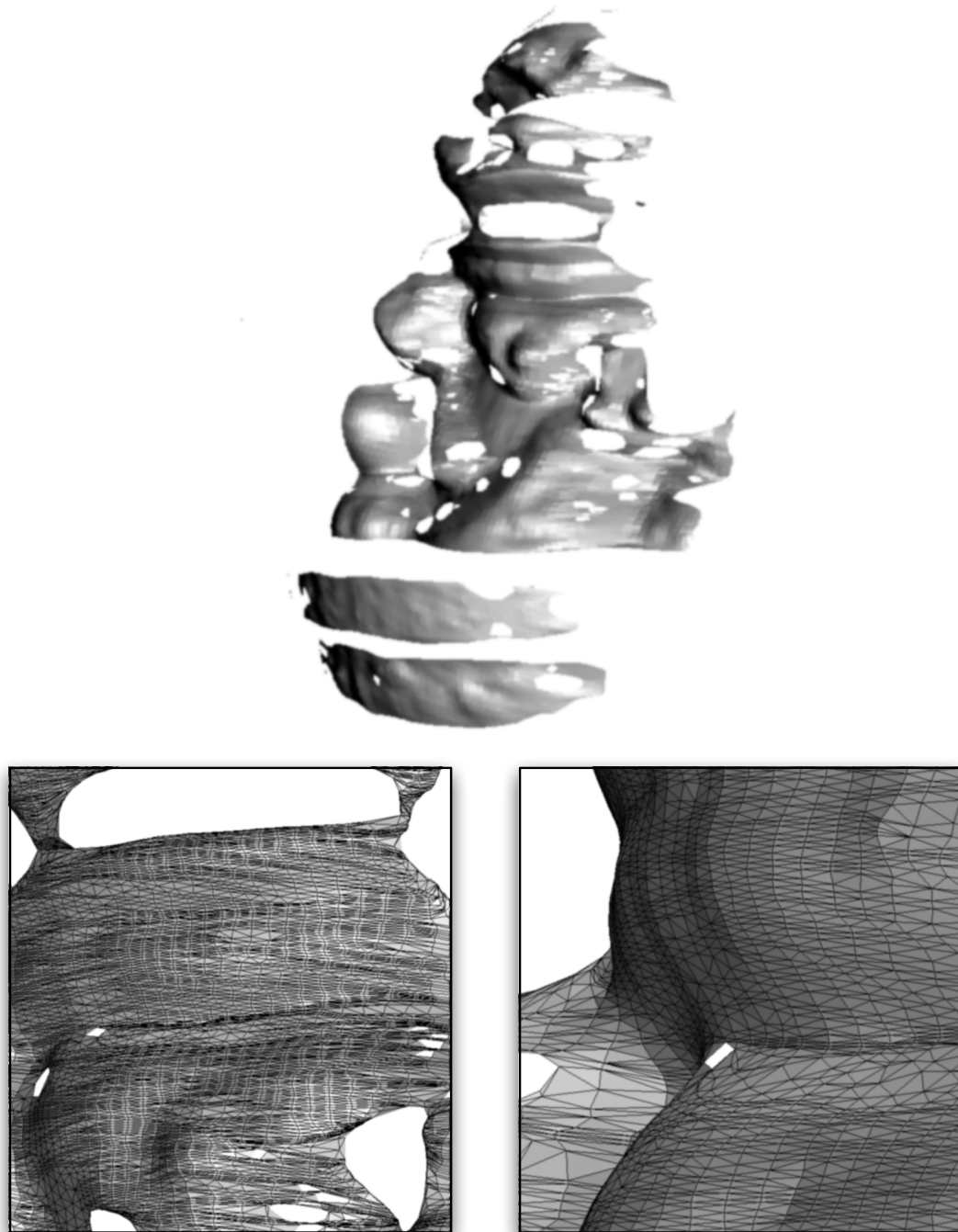


Abbildung 6.3: Vergrößerte Ansichten des geglätteten Dreiecksnetzes.

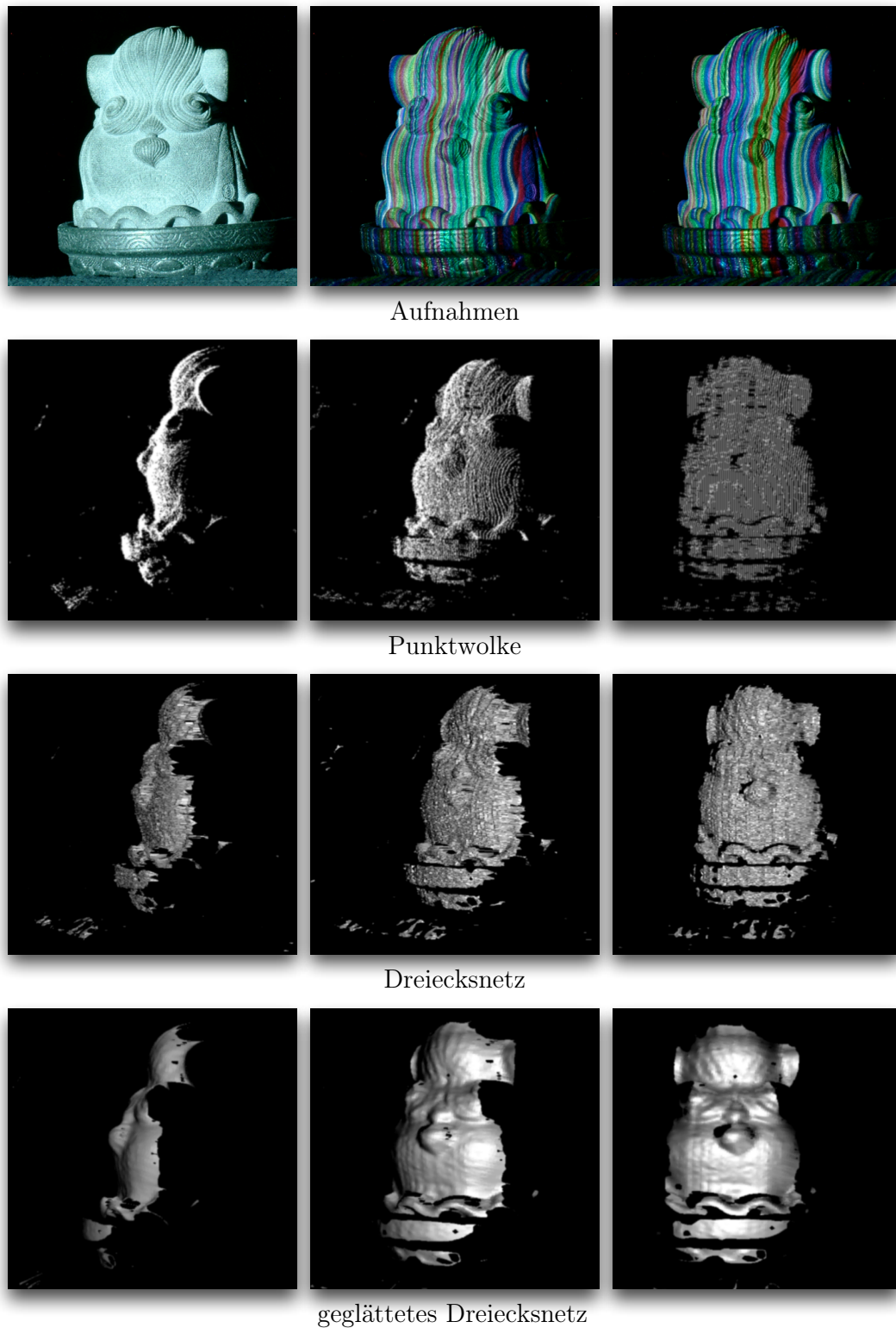


Abbildung 6.4: Rekonstruktionen des wachenden Löwens (Rücken). Er besteht aus 31363 Punkten und 60226 Dreiecken.

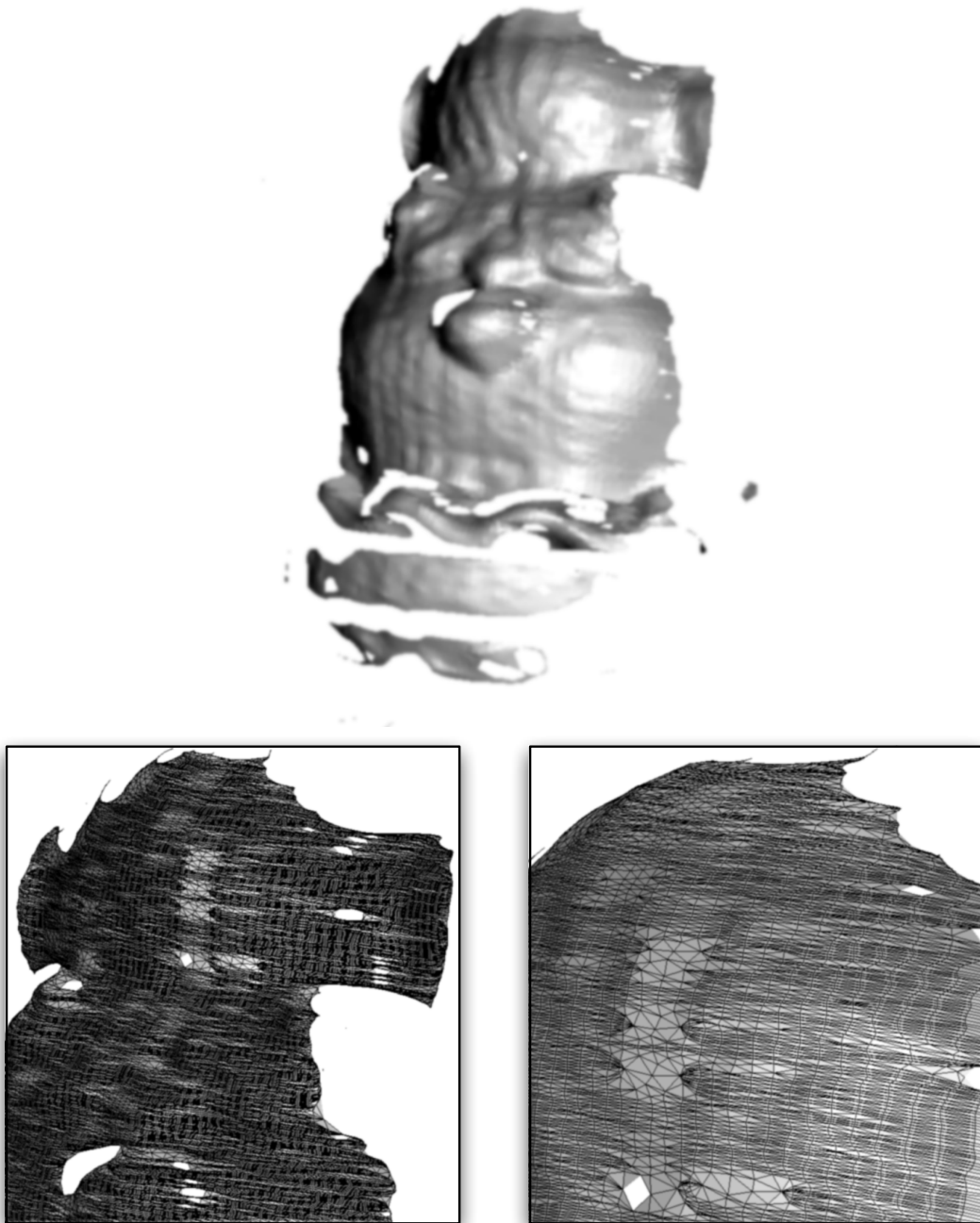


Abbildung 6.5: Vergrößerte Ansichten des geglätteten Dreiecksnetzes.

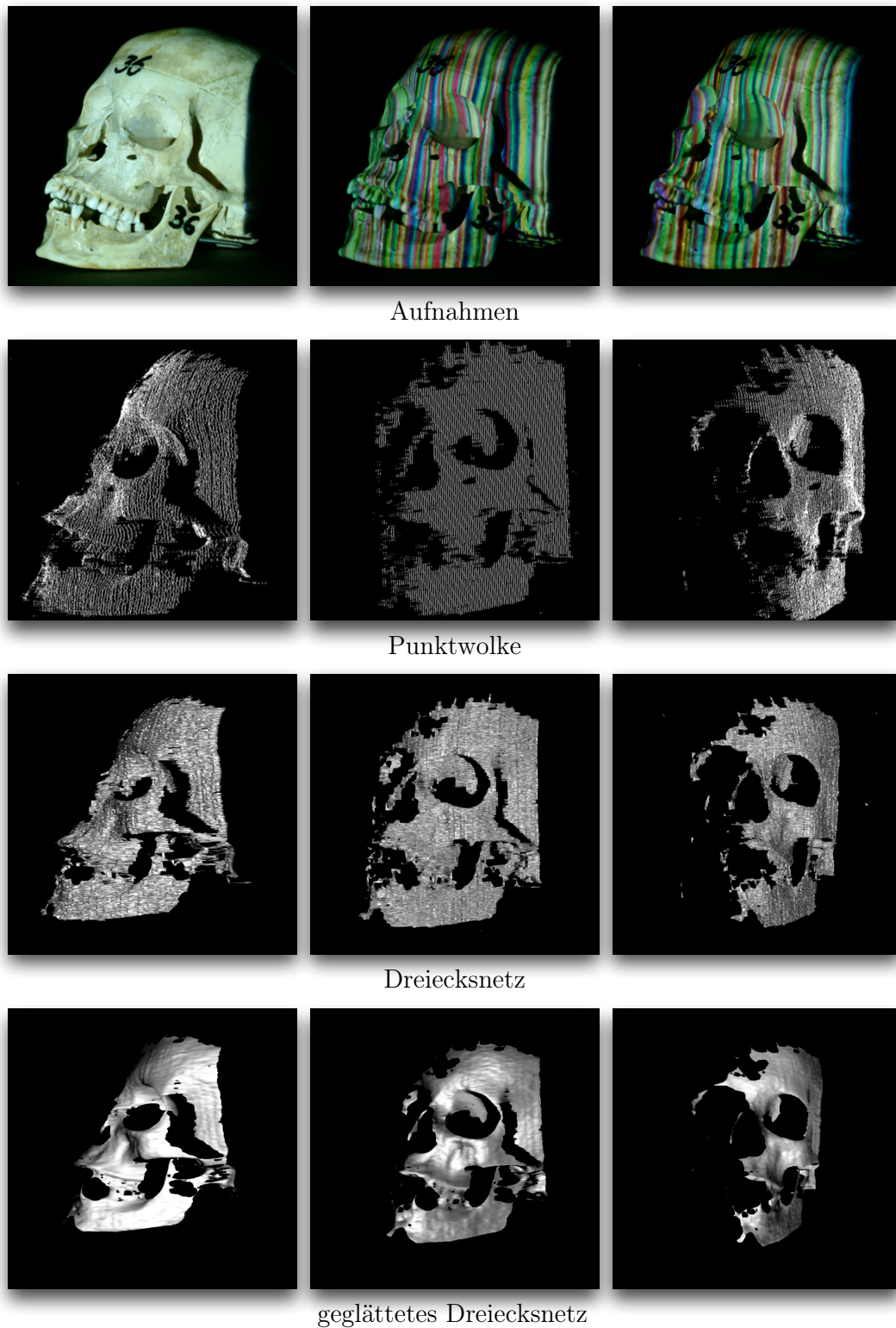


Abbildung 6.6: Rekonstruktionen des Gesichtsschädels. Er besteht aus 41468 Punkten und 79317 Dreiecken.

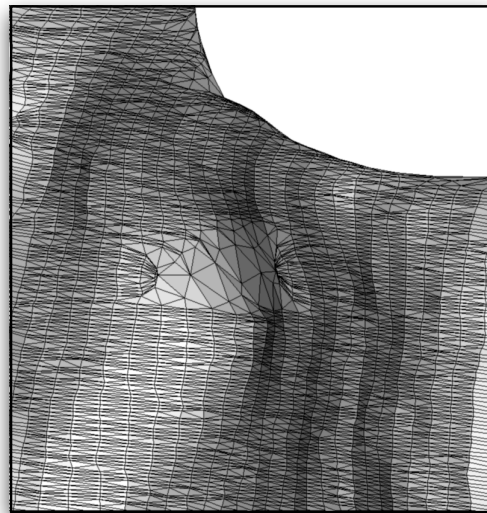
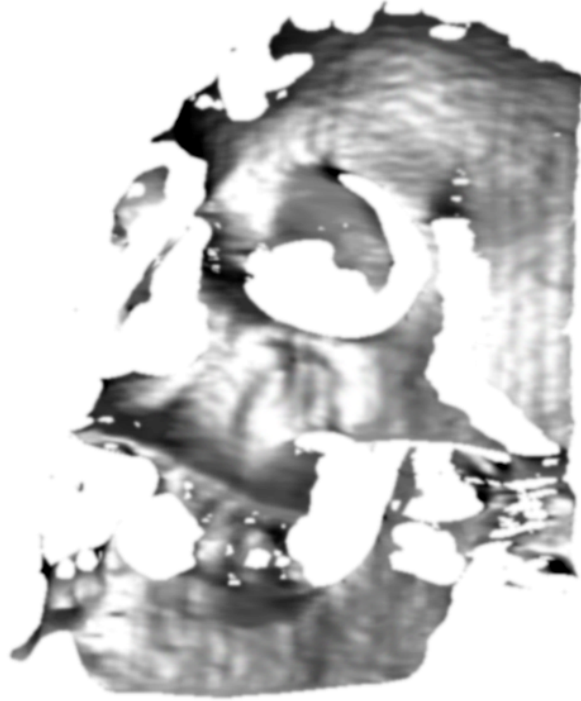


Abbildung 6.7: Vergrößerte Ansichten des geglätteten Dreiecksnetzes.

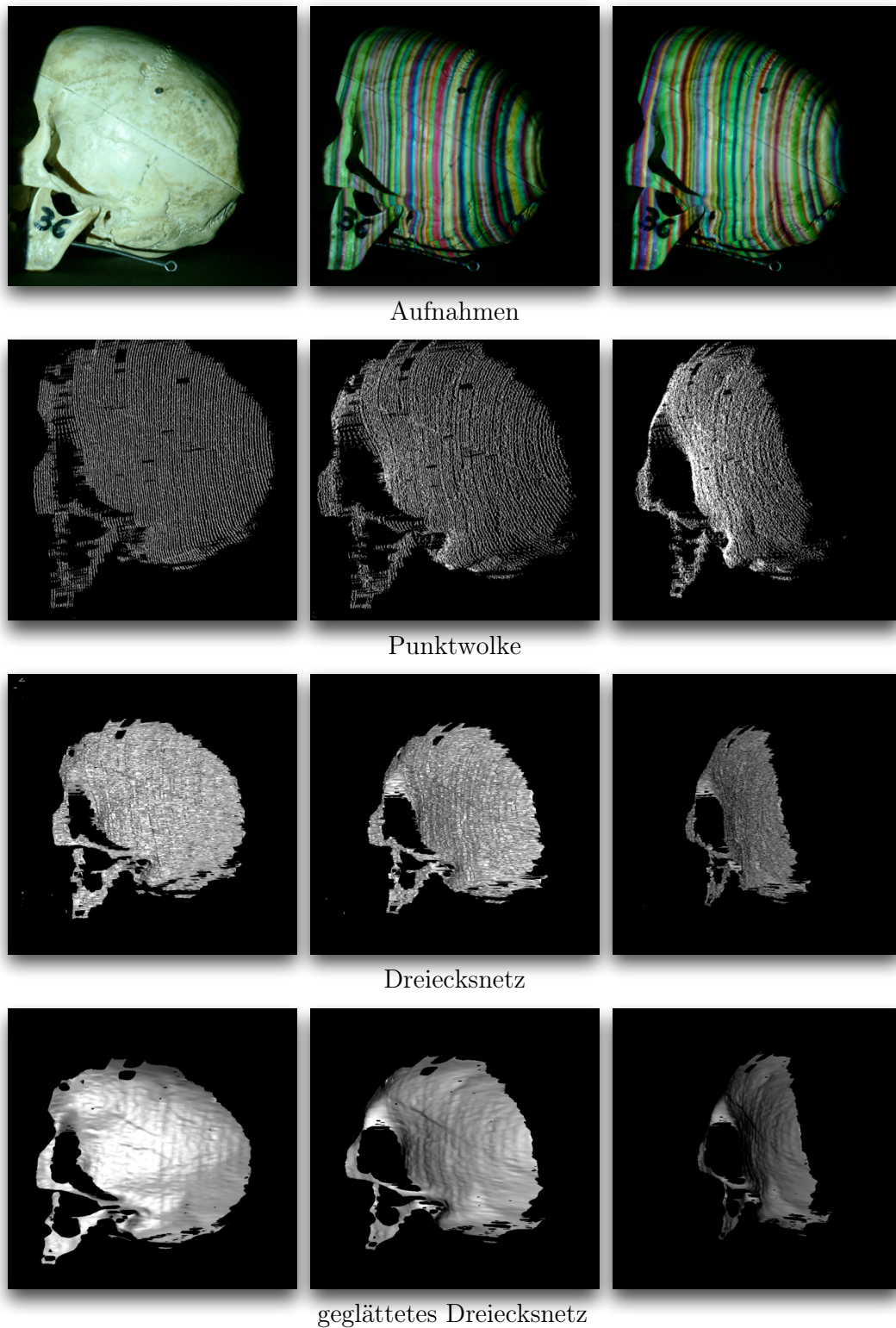


Abbildung 6.8: Rekonstruktionen des Hirnschädels. Er besteht aus 44804 Punkten und 87370 Dreiecken.

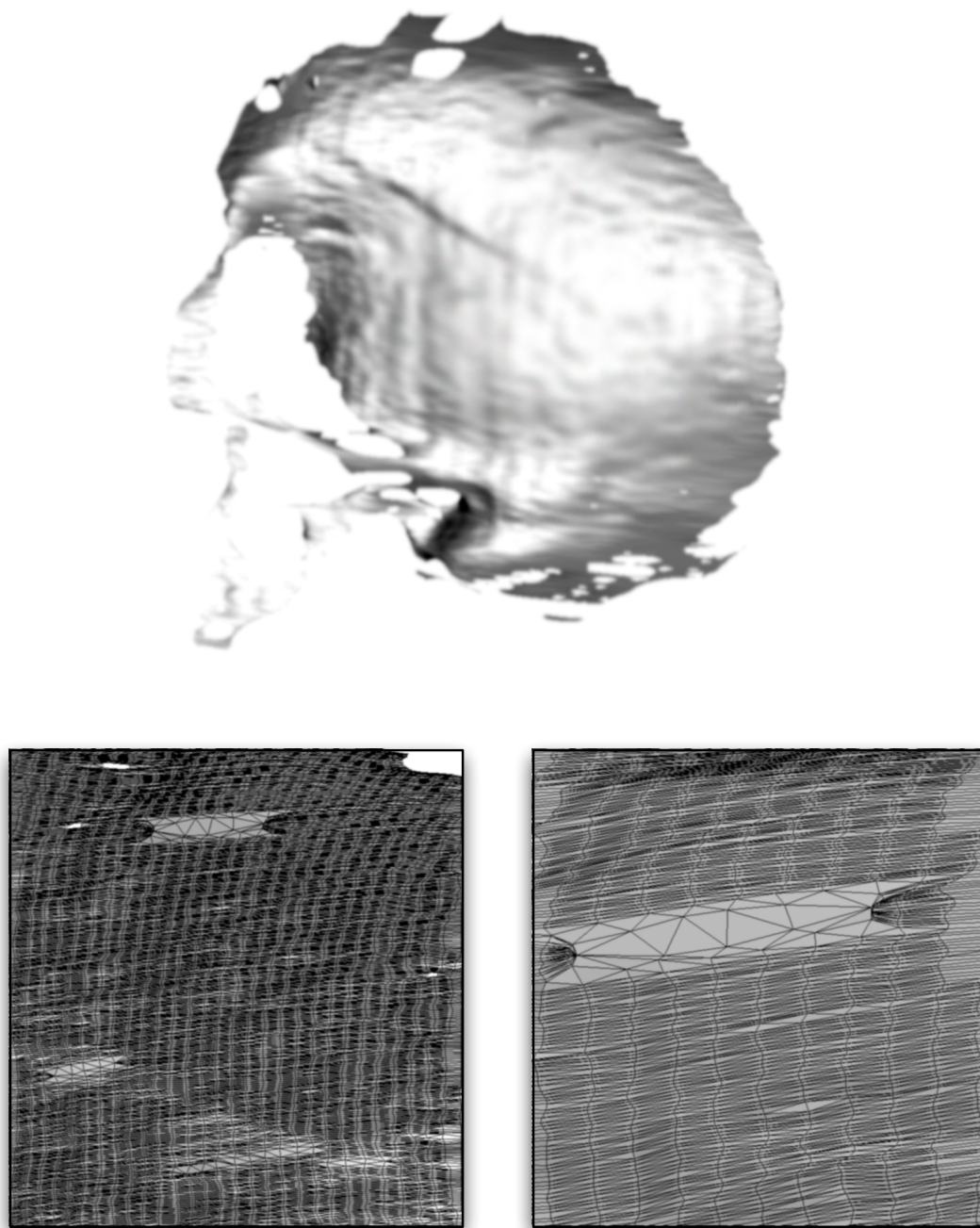


Abbildung 6.9: Vergrößerte Ansichten des geglätteten Dreiecksnetzes.

Kapitel 7

Schlusswort

7.1 Zusammenfassung

In dieser Arbeit wurde ein neuartiges 3D-Scanverfahren mit strukturiertem Licht entworfen, das aus drei Aufnahmen ein hochauflösendes geglättetes Dreiecksnetz rekonstruiert. Insbesondere verwenden wir für die Rekonstruktion nur eine Digitalkamera und einen Videoprojektor. Wegen der bei Scanverfahren, die nur sehr wenige Aufnahmen benötigen, mangelnden Robustheit gegenüber Ausreißern, wurde eine Vielzahl von Methoden entwickelt, die gezielt diese Nachteile beseitigen.

Unter streng vorgegebenen Kohärenzannahmen und Beschränkungen des Scansystems, werden eindeutig kodierte farbige Streifenmuster erzeugt, die auf das Objekt projiziert werden. Diese Streifensequenzen haben zusätzlich zu der Eigenschaft, dass alle Streifen innerhalb einer lokalen Umgebung eindeutig sind, auch die globale Eigenschaft, dass alle Streifenpaare aus einer Umgebung sich nicht gleichzeitig in weiteren nicht überlappenden Umgebungen befinden dürfen. Dadurch werden Falschzuordnungen aufgrund von hochfrequenten Selbstverdeckungen und schlecht reflektierenden Oberflächen drastisch reduziert.

Ausgehend vom entwickelten Streifenmuster bieten wir ein neues Paradigma für das Zuordnungsproblem. Aufgrund des schwachen Kontrasts an Streifenübergängen, verwenden wir für die Zuordnung die Medianwerte der Streifenmitten aus Partitionen, die aus durch Ballung über mehrere Zeilen zusammengefasste, lokale Streifenübergänge bestimmt wurden. Die Zuordnung selbst wurde durch einen optimierten gierigen Algorithmus gelöst. Obwohl dieses Verfahren nur eine Annäherung an eine global optimale Lösung ist, haben wir gezeigt, dass es, aufgrund der Verwendung der Medianwerte, dennoch besonders zuverlässig ist. Insbesondere haben wir ein Verfahren vorgestellt, das nicht nur innerhalb einer Zeile zuordnet, sondern alle Zeilen der Aufnahme berücksichtigt. Die Robustheit dieses Verfahrens wurde in dieser Arbeit durch eine Vielzahl von Experimenten demonstriert und mit traditionellen, dynamisch programmierten Zuordnungsverfahren verglichen.

Im Gegensatz zu den meisten anderen Scanverfahren, integrieren wir in unserer Rekonstruktionspipeline zusätzlich die Flächenrekonstruktion. Dadurch sind wir in der Lage, wichtige A-Priori-Kenntnisse vorheriger Schritte auszunutzen, um eine konsistente Topologie des zu erzeugenden Dreiecksnetzes abzuschätzen. Insbesondere werden lokale räumliche Kohärenzannahmen wiederverwendet, um das Problem von unregelmäßigen Punktdichten zu lösen. Auch hier haben wir gezeigt, dass typische Verfahren, die nur anhand einer Abschätzung der lokalen Flächenparametrisierung von Punktwolken eine Fläche rekonstruieren, fehlschlagen, wenn die Abtastungen der Oberfläche hoch anisotropisch sind.

Aufgrund der hohen Robustheit unseres Verfahrens, können auch nicht optimal bestimmte Eingabeparameter der einzelnen Schritte, zu guten Ergebnissen führen.

7.2 Ausblick

Dadurch, dass in dieser Arbeit ein vollständiges Scansystem, vorgestellt wurde, war es nicht möglich eine Vielzahl von bekannten Verfahren zu untersuchen, die möglicherweise bessere Lösungen zu einigen Schritten in der Rekonstruktionspipeline bieten. Dennoch ist es uns gelungen, ein Rahmenwerk zu präsentieren, dessen Methoden robust genug sind, um alle unsere gesetzten Ziele zu erreichen. Durch den modularen Aufbau unseres Verfahrens, können somit weitere Untersuchungen erfolgen, um entweder die vorgestellten Algorithmen durch noch robustere und theoretisch fundiertere zu ersetzen oder um die Rekonstruktionspipeline zu erweitern. Zukünftige Verbesserungs- und Erweiterungsvorschlägen können wir folgendermaßen zusammenfassen:

- **Robustere Streifen:** Die präsentierten Streifensequenzen wurden unabhängig vom Objekt erzeugt. Wie bereits angesprochen, könnte eine robustere Erfassung dadurch erreicht werden, dass die Oberflächenreflexion eines Objekts in einem Vorverarbeitungsschritt berücksichtigt wird, um entsprechend die projizierten Farben an die Reflektivität anzupassen. Weiterhin ist es möglich, unseren Streifenerzeugungsalgorithmus so zu modifizieren, dass sich nur hohe Kontraste zwischen den Streifenübergängen befinden dürfen. Dadurch könnte durchaus eine höhere Zuverlässigkeit, sowohl bei der Kantendetektion als auch bei der Zuordnung, erzielt werden.
- **Optimale Zuordnung zwischen den Zeilen:** Ein weiterer zentraler Punkt stellt das Zuordnungsproblem während der Ballung über dem lokalen Kontrast dar. Dadurch, dass bei unserem Verfahren das Ergebnis von der Reihenfolge der betrachteten Zeilen abhängt, haben wir insbesondere keine optimale Lösung. Auch eine effizientere Lösung könnte von Vorteil sein. Energiemini-misierende Graphenschnitte können als Ansatzpunkt für weitere Untersuchungen in Betracht gezogen werden.

- **Effizientere Zuordnung der Streifenmitten:** Dadurch, dass unser gieriger Zuordnungsalgorithmus, eine vollständige Suche ausführt, ist die Laufzeitkomplexität, trotz der Optimierung, sehr hoch. Geläufige, wie etwa auf *mehrstufige* (engl. *multi-resolution*) Analysen basierende Techniken, können eingesetzt werden, um den Suchraum zu beschränken.
- **Erweiterung der Rekonstruktionspipeline:** Im Hinblick auf die Rekonstruktion eines vollständigen Objekts aus mehreren Ansichten, muss die Rekonstruktionspipeline erweitert werden. Die wichtigsten Schritte, die in Kap. 1 bereits vorgestellt wurden, sind die Registrierung, die Flächenintegration und die Texturrekonstruktion. Zusätzlich wäre ein Sichtplanungsverfahren, das vorgibt aus welcher Perspektive als nächstes erfasst werden soll, hilfreich.
- **Scansoftware:** Schon zu Beginn der Implementierung wurde darauf geachtet, dass ein hoher Grad an Wiederverwendbarkeit vorhanden ist. Es ist geplant, eine vollständige Scansoftware zu entwickeln, die auch die zu ergänzenden Schritte der Rekonstruktionspipeline enthält. Insbesondere muss ein Kalibrierungsverfahren unmittelbar integriert werden, sowie eine graphische Oberfläche, für eine leichtere Handhabung und Auswertung der Zwischenergebnisse.

Literaturverzeichnis

- [Adobe '03] Adobe. *ADOBE SYSTEMS INCORP., Adobe Photoshop User Guide*, 2003.
- [Amenta et al. '01] Nina Amenta, Sunghee Choi und Ravi Krishna Kolluri. The power crust. In *SMA '01: Proceedings of the sixth ACM symposium on Solid modeling and applications*, S. 249–266, New York, NY, USA, 2001. ACM Press.
- [Anderson et al. '99] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney und D. Sorensen. *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, third. Auflage, 1999.
- [Bayer '76] Bryce E. Bayer. Color imaging array, 1976.
- [Bernardini & Rushmeier '02] Fausto Bernardini und Holly E. Rushmeier. The 3D Model Acquisition Pipeline. *Comput. Graph. Forum*, 21, Nr. 2, S. 149–172, 2002.
- [Bernardini et al. '99] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva und G. Taubin. The Ball-Pivoting Algorithm for Surface Reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 5, Nr. 4, S. 349–359, 1999.
- [Bernardini et al. '01] Fausto Bernardini, Ioana M. Martin und Holly Rushmeier. High-Quality Texture Reconstruction from Multiple Scans. *IEEE Transactions on Visualization and Computer Graphics*, 7, Nr. 4, 2001.
- [Bernardini et al. '02] Fausto Bernardini, Holly Rushmeier, Ioana M. Martin, Joshua Mittleman und Gabriel Taubin. Building a Digital Model of Michelangelo's Florentine Pietà. *IEEE Computer Graphics and Applications*, 22, Nr. 1, S. 59–67, 2002.
- [Besl & McKay '92] P. J. Besl und N. D. McKay. A method for registration of 3D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14, Nr. 2, S. 239–256, February 1992.

- [Besl '88] Paul J. Besl. Active, optical range imaging sensors. *Mach. Vision Appl.*, 1, Nr. 2, S. 127–152, 1988.
- [Billings '98] Richard Billings. Color separation in an active pixel cell imaging array using a triple-well structure, 1998.
- [Blinn '98] James F. Blinn. Ten More Unsolved Problems in Computer Graphics. *IEEE Comput. Graph. Appl.*, 18, Nr. 5, S. 86–89, 1998.
- [Boehm & Prautzsch '94] Wolfgang Boehm und Hartmut Prautzsch. *Geometric concepts for geometric design*. A.K. Peters, Wellesley, MA, USA, 1994. 424 p., ISBN 1-56881-004-0.
- [Boissonnat '84] Jean-Daniel Boissonnat. Geometric structures for three-dimensional shape representation. *ACM Trans. Graph.*, 3, Nr. 4, S. 266–286, 1984.
- [Boyer & Kak '87] K. L. Boyer und A. C. Kak. Color-encoded structured light for rapid active ranging. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9, Nr. 1, S. 14–28, 1987.
- [Buehler et al. '01] Chris Buehler, Michael Bosse, Leonard McMillan, Steven J. Gortler und Michael F. Cohen. Unstructured Lumigraph Rendering. In Eugene Fiume (Hrsg.), *SIGGRAPH 2001, Computer Graphics Proceedings*, S. 425–432. ACM Press / ACM SIGGRAPH, 2001.
- [Carrhill & Hummel '85] Brian Carrhill und Robert Hummel. Experiments with the intensity ratio depth sensor. *Computer Vision, Graphics, and Image Processing*, 32, S. 337–358, 1985.
- [Caspi et al. '98] Dalit Caspi, Nahum Kiryati und Joseph Shamir. Range Imaging With Adaptive Color Structured Light. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20, Nr. 5, S. 470–480, 1998.
- [Chazan & Kiryati '95] G. Chazan und N. Kiryati. Pyramidal intensity-ratio depth sensor. Technical Report Technical report 121, Center for Communication and Information Technologies, Department of Electrical Engineering, Technion, Haifa, Israel, October 1995.
- [Chen & Medioni '92] Yang Chen und Gerard Medioni. Object modelling by registration of multiple range images. *Image Vision Comput.*, 10, Nr. 3, S. 145–155, 1992.
- [Chen et al. '97] Chu-Song Chen, Yi-Ping Hung, Chiann-Chu Chiang und Ja-Ling Wu. Range data acquisition using color structured lighting and stereo vision. *Image Vision Comput.*, 15, Nr. 6, S. 445–456, 1997.

- [Chen '95] Shenchang Eric Chen. QuickTime VR — An Image-Based Approach to Virtual Environment Navigation. *Computer Graphics*, 29, Nr. Annual Conference Series, S. 29–38, 1995.
- [Cumani '91] A. Cumani. Edge detection in multispectral images. *Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing*, 53, Nr. 1, S. 40–51, January 1991.
- [Curless & Levoy '95] Brian Curless und Marc Levoy. Better Optical Triangulation Through Spacetime Analysis. In *ICCV*, S. 987–994, 1995.
- [Curless & Levoy '96] Brian Curless und Marc Levoy. A volumetric method for building complex models from range images. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, S. 303–312, New York, NY, USA, 1996. ACM Press.
- [Curless et al. '00] Brian Curless, Steve Seitz, Jean-Yves Bouguet, Paul Debevec, Marc Levoy und Shree Nayar. 3D Photography. In *ACM Siggraph 2000 Course Notes, Computer Graphics Proceedings, Course No. 19*, August 24th 2000.
- [Curless '97] Brian Curless. New Methods for Surface Reconstruction from Range Images. Technical Report CSL-TR-97-733, 1997.
- [Curless '01] Brian Curless. *VripPack User's Guide*, 2001. <http://www-graphics.stanford.edu/software/vrip> (gesehen 2005).
- [Cyberware '99] Cyberware. Cyberware Inc., 1999. <http://www.cyberware.com/> (gesehen 2005).
- [Davies & Nixon '98] C. J. Davies und M. S. Nixon. A Hough transform for detecting the location and orientation of three-dimensional surfaces via color encoded spots. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 28, Nr. 1, S. 90–95, 1998.
- [DDWG '99] Digital Display Working Group DDWG. *Digital Visual Interface DVI Specification Revision 1.0*, 1999.
- [Debevec et al. '00] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duijker, Westley Sarokin und Mark Sagar. Acquiring the Reflectance Field of a Human Face. In Kurt Akeley (Hrsg.), *Siggraph 2000, Computer Graphics Proceedings*, S. 145–156. ACM Press / ACM SIGGRAPH / Addison Wesley Longman, 2000.
- [Desbrun et al. '99] Mathieu Desbrun, Mark Meyer, Peter Schröder und Alan H. Barr. Implicit fairing of irregular meshes using diffusion and curvature flow.

- In *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, S. 317–324, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.
- [Dhond & Aggarwal '89] U. Dhond und J. Aggarwal. Structure from Stereo – A Review. *IEEE Transactions on Systems, Man and Cybernetics*, 19, Nr. 6, S. 1489–1510, November/December 1989.
- [Duncan '02] Andrew M. Duncan. *Objective-C Pocket Reference*. O'Reilly Verlag, 1.. Auflage, 2002.
- [Edelsbrunner & Mücke '94] Herbert Edelsbrunner und Ernst P. Mücke. Three-Dimensional Alpha Shapes. *ACM Transactions on Graphics*, 13, Nr. 1, S. 43–72, 1994.
- [Fasano et al. '03] Andrea Fasano, Marco Callieri, Paolo Cignoni und Roberto Scopigno. Exploiting Mirrors for Laser Stripe 3D Scanning. In *3DIM*, S. 243–250, 2003.
- [Faugeras '93] Olivier Faugeras. *Three-dimensional computer vision: a geometric viewpoint*. MIT Press, Cambridge, MA, USA, 1993.
- [Fautz '03] Michael Fautz. *Objekt- und Texturrekonstruktion mit einer robotergeführten Kamera*. Dissertation, Universität Karlsruhe (TH), 2003.
- [Fujiwara '95] K. Fujiwara. Eigenvalues of laplacians on a closed riemannian manifold and its nets. In *Proceedings of the AMS*, 1995.
- [Goesele '04] Michael Goesele. *New Acquisition Techniques for Real Objects and Light Sources in Computer Graphics*. Dissertation, Universität des Saarlandes, July 2004.
- [Gopi et al. '00] M. Gopi, S. Krishnan und C. T. Silva. Surface Reconstruction Based on Lower Dimensional Localized Delaunay Triangulation. In M. Gross und F. R. A. Hopgood (Hrsg.), *Computer Graphics Forum (Eurographics 2000)*, Bd. 19(3), 2000.
- [Gortler et al. '96] Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski und Michael F. Cohen. The Lumigraph. *Computer Graphics*, 30, Nr. Annual Conference Series, S. 43–54, 1996.
- [Hall-Holt & Rusinkiewicz '01] Olaf Hall-Holt und Szymon Rusinkiewicz. Stripe Boundary Codes for Real-Time Structured-Light Range Scanning of Objects. In *Eighth International Conference on Computer Vision (ICCV)*, july 2001.

- [Hebert et al. '93] R. Hebert, D. Laurendeau und D. Poussart. Scene reconstruction and description: geometric primitive extraction from multiple viewed scattered data. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, S. 286–292, June 1993.
- [Hecht '01] Eugene Hecht. *Optics*. Addison-Wesley, Reading, Massachusetts, 4. Auflage, 2001.
- [Heikkila & Silven '97] Janne Heikkila und Olli Silven. A Four-step Camera Calibration Procedure with Implicit Image Correction. In *CVPR '97: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, S. 1106, Washington, DC, USA, 1997. IEEE Computer Society.
- [Hillegass '02] Aaron Hillegass. *Cocoa Programming for Mac OS X*. Addison-Wesley, 2002.
- [Hoppe '95] Hugues Hoppe. *Surface reconstruction from unorganized points*. Dissertation, Seattle, WA, USA, 1995.
- [Hornbeck & Nelson '88] L. J. Hornbeck und W. E. Nelson. Bistable Deformable Mirror Device. *Spatial Light Modulators and Applications*, 8, 1988.
- [Hornbeck '95] L. J. Hornbeck. *Digital Light Processing for High- Brightness High-Resolution Applications*, 1995.
- [Hornbeck '96] L. J. Hornbeck. A New MEMS-Based Display Technology (Keynote Address). *Technical Digest of the IEEEJ 14th Sensor Symposium*, S. 297–304, 1996.
- [Ishikawa & Geiger '98] H. Ishikawa und D. Geiger. Segmentation by Grouping Junctions. In *CVPR '98: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, S. 125, Washington, DC, USA, 1998. IEEE Computer Society.
- [Kanade et al. '91] Takeo Kanade, A. Gruss und L.R. Carley. A Very Fast VLSI Rangefinder. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation (ICRA '91)*, Bd. 2, S. 1322–1329, April 1991.
- [Konica-Minolta '03] Konica-Minolta. Konica Minolta Inc., 2003. <http://www.konicaminolta-3d.com/> (gesehen 2005).
- [Koninckx et al. '05] Thomas Koninckx, Pieter Peers, Philip Dutre und Luc Van Gool. Scene-adapted Structured Light. *IEEE Computer Society, International Conference on Computer Vision and Pattern Recognition, CVPR 2005*, June 2005.

- [Koschan & Abidi '01] Andreas Koschan und Mongi Abidi. A Comparison of Median Filter Techniques For Noise Removal in Color Images. In *Proc. of the 7th German Workshop on Color Image Processing, Erlangen, Germany, S.* 69–79, October 2001.
- [Koschan '95] A. Koschan. A comparative study on color edge detection, 1995.
- [Kroschel '04] Kristian Kroschel. *Statistische Informationstechnik*. Springer, 4. Auflage, 2004.
- [Laurentini '94] A. Laurentini. The Visual Hull Concept for Silhouette-Based Image Understanding. *IEEE Trans. Pattern Anal. Mach. Intell.*, 16, Nr. 2, S. 150–162, 1994.
- [Levoy & Hanrahan '96] Marc Levoy und Pat Hanrahan. Light Field Rendering. *Computer Graphics*, 30, Nr. Annual Conference Series, S. 31–42, 1996.
- [Levoy et al. '00] Marc Levoy, Kari Pulli, Brian Curless, Szymon Rusinkiewicz, David Koller, Lucas Pereira, Matt Ginzton, Sean Anderson, James Davis, Jeremy Ginsberg, Jonathan Shade und Duane Fulk. The Digital Michelangelo Project: 3D Scanning of Large Statues. In Kurt Akeley (Hrsg.), *Siggraph 2000, Computer Graphics Proceedings*, S. 131–144. ACM Press / ACM SIGGRAPH / Addison Wesley Longman, 2000.
- [Levoy '05] Marc Levoy. *The Stanford 3D Scanning Repository*, 2005. <http://graphics.stanford.edu/data/3Dscanrep/> (gesehen 2005).
- [Li et al. '02] Ming Li, Hartmut Schirmacher, Marcus Magnor und Hans-Peter Seidel. Combining Stereo and Visual Hull Information for On-line Reconstruction and Rendering of Dynamic Scenes, 2002.
- [Li et al. '04] Hao Li, Raphael Straub und Hartmut Prautzsch. Fast Subpixel Accurate Reconstruction using Color Structured Light. In *Proceedings of the Fourth IASTED International Conference on Visualization, Imaging, and Image Processing*, S. 396–401. The International Association of Science and Technology for Development (IASTED), ACTA Press, September 2004.
- [Li '04a] Hao Li. LIThIum Release 0.3, A 3D scanning and visualization tool using color structured light (GPL), 2004. <http://i33www.ira.uka.de/~hao/Research/software.html> (gesehen 2005).
- [Li '04b] Hao Li. Reconstruction Using Structured Light. Technical report, Universität Karlsruhe (TH), 2004.
- [Linsen '01] Lars Linsen. *Oberflächenrepräsentation durch Punktwolken*. Dissertation, Universität Karlsruhe (TH), 2001.

- [Lorensen & Cline '87] William E. Lorensen und Harvey E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, S. 163–169, New York, NY, USA, 1987. ACM Press.
- [Ma et al. '04] Yi Ma, Stefano Soatto, Jana Kosecká und S. Shankar Sastry. *An Invitation to 3-D Vision*. Springer-Verlag, NY, USA, 2004.
- [Marschner et al. '99] Stephen R. Marschner, Stephen H. Westin, Eric P. Lafortune, Kenneth E. Torrance und Donald P. Greenberg. Image-Based BRDF Measurement Including Human Skin. In *Rendering Techniques*, S. 131–144, 1999.
- [McIvor & Valkenburg '97] Alan M. McIvor und Robert J. Valkenburg. Substripe Localisation for Improved Structured Light System Performance, 1997.
- [Miyazaki et al. '01] Daisuke Miyazaki, Takeshi Oishi, Taku Nishikawa, Ryusuke Sagawa, Ko Nishino, Takashi Tomomatsu, Yutaka Takase und Katsushi Ikeuchi. The great Buddha project: modeling cultural heritage through observation. S. 181–193, 2001.
- [Moore '65] Gordon E. Moore. Cramming More Components onto Integrated Circuits. *Electronics*, 38(8), April 1965.
- [Morano et al. '98] Raymond A. Morano, Cengizhan Ozturk, Robert Conn, Stephen Dubin, Stanley Zietz und Jonathan Nissanov. Structured Light Using Pseudorandom Codes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20, Nr. 3, S. 322–327, 1998.
- [Netlib '02] Netlib. *Basic Linear Algebra Subprograms (BLAS)*, 2002. <http://www.netlib.org/blas/faq.html> (gesehen 2005).
- [Nicodemus et al. '92] F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg und T. Limperis. Geometrical considerations and nomenclature for reflectance. S. 94–145, 1992.
- [Novak & Shafer '87] C. L. Novak und S. A. Shafer. Color Edge Detection. In *Proc. Of DARPA Image Understanding Workshop*, S. 35–37, Los Angeles (USA). Los Altos, CA, 1987. Morgan Kaufmann Publishers, Inc.
- [Ohta & Kanade '85] Y. Ohta und Takeo Kanade. Stereo by Intra- and Inter-Scanline Search Using Dynamic Programming. *IEEE Trans. Pattern Analysis and Machine Intelligence*, PAMI-7, Nr. 2, S. 139–154, March 1985.
- [Oppenheim et al. '99] Alan V. Oppenheim, Ronald W. Schafer und John R. Buck. *Discrete-time signal processing (2nd ed.)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1999.

- [Pollefeys '99] Marc Pollefeys. *Self-Calibration and Metric 3D Reconstruction from Uncalibrated Image Sequences*. Dissertation, Katholieke Universiteit Leuven, Kardinaal Mercierlaan 94-3001 Heverlee, May 1999.
- [Posdamer & Altschuler '82] J. L. Posdamer und M. D. Altschuler. Surface Measurement by Space-Encoded Projected Beam Systems. *j-CGIP*, 18, Nr. 1, S. 1–17, January 1982.
- [Proesmans et al. '96] M. Proesmans, L. Van Gool und A. Oosterlinck. One-Shot Active 3D Shape Acquisition. In *ICPR '96: Proceedings of the International Conference on Pattern Recognition (ICPR '96) Volume III-Volume 7276*, S. 336, Washington, DC, USA, 1996. IEEE Computer Society.
- [Pulli et al. '97] K. Pulli, T. Duchamp, H. Hoppe, J. McDonald, L. Shapiro und W. Stuetzle. Robust meshes from multiple range maps. In *NRC '97: Proceedings of the International Conference on Recent Advances in 3-D Digital Imaging and Modeling*, S. 205, Washington, DC, USA, 1997. IEEE Computer Society.
- [Ramanath et al. '02] R. Ramanath, W. E. Snyder, G. L. Bilbro und W. A. Sander. Demosaicking methods for Bayer color arrays. *Journal of Electronic Imaging*, 11, S. 306–315, Juli 2002.
- [Reichenbach et al. '91] Stephen E. Reichenbach, Stephen K. Park und Ramkumar Narayanswamy. Characterizing Digital Image Acquisition Devices. *Optical Engineering*, 30, Nr. 2, S. 170–171, February 1991.
- [Rocchini et al. '01] C. Rocchini, Paulo Cignoni, C. Montani, P. Pingi und Roberto Scopigno. A low cost 3D scanner based on structured light. In A. Chalmers und T.-M. Rhyne (Hrsg.), *EG 2001 Proceedings*, Bd. 20(3), S. 299–308. Blackwell Publishing, 2001.
- [Rother et al. '04] Carsten Rother, Vladimir Kolmogorov und Andrew Blake. GrabCut: interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23, Nr. 3, S. 309–314, 2004.
- [Rushmeier et al. '98] Holly Rushmeier, Fausto Bernardini, Joshua Mittleman und Gabriel Taubin. Acquiring Input for Rendering At Appropriate Levels Of Detail: Digitizing a Pietià. S. 81–92, 1998.
- [Rusinkiewicz & Levoy '01] Szymon Rusinkiewicz und Marc Levoy. Efficient Variants of the ICP Algorithm. In *Third International Conference on 3D Digital Imaging and Modeling (3DIM)*, Juni 2001.
- [Rusinkiewicz et al. '02] Szymon Rusinkiewicz, Olaf Hall-Holt und Marc Levoy. Real-time 3D model acquisition. In *SIGGRAPH '02: Proceedings of the 29th*

- annual conference on Computer graphics and interactive techniques*, S. 438–446, New York, NY, USA, 2002. ACM Press.
- [Rusinkiewicz '01] Szymon Rusinkiewicz. *Real-time acquisition and rendering of large three-dimensional models*. Dissertation, 2001.
- [Salvi et al. '04] J. Salvi, J. Pagés und J. Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37, Nr. 4, S. 827–849, 2004.
- [Sato & Inokuchi '85] K. Sato und S. Inokuchi. Three-dimensional surface measurement by space encoding range imaging. *Journal of Robotic Systems*, 2, S. 27–39, 1985.
- [Szeliski & Shum '97] Richard Szeliski und Heung-Yeung Shum. Creating full view panoramic image mosaics and environment maps. *Computer Graphics*, 31, Nr. Annual Conference Series, S. 251–258, 1997.
- [Szeliski & Zabih '00] Richard Szeliski und Ramin Zabih. An Experimental Comparison of Stereo Algorithms. In *ICCV '99: Proceedings of the International Workshop on Vision Algorithms*, S. 1–19, London, UK, 2000. Springer-Verlag.
- [Taubin '95] Gabriel Taubin. A Signal Processing Approach to Fair Surface Design. *Computer Graphics*, 29, Nr. Annual Conference Series, S. 351–358, 1995.
- [Taubin '00] Gabriel Taubin. Geometric signal processing on polygonal meshes. *EUROGRAPHICS 2000: State of the Art Report (STAR)*, 2000.
- [Tomasi & Manduchi '98] C. Tomasi und R. Manduchi. Bilateral Filtering for Gray and Color Images. In *ICCV '98: Proceedings of the Sixth International Conference on Computer Vision*, S. 839, Washington, DC, USA, 1998. IEEE Computer Society.
- [Tsai '87] Roger Y. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. S. 221–244, 1987.
- [Turk & Levoy '94] Greg Turk und Marc Levoy. Zippered polygon meshes from range images. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, New York, NY, USA, 1994. ACM Press.
- [Várady et al. '97] Tamás Várady, Ralph R. Martin und Jordan Cox. Reverse engineering of geometric models - an introduction. *Computer-Aided Design*, 29, Nr. 4, S. 255–268, 1997.

- [Vedula et al. '98] S. Vedula, P. Rander, H. Saito und T. Kanade. Modeling, Combining, and Rendering Dynamic Real-World Events From Image Sequences, 1998.
- [Yamada et al. '00] Tetsuo Yamada, Katsumi Ikeda, Yong-Gwan Kim, Hideki Wakoh, Tetsuo Toma, Tomohiro Sakamoto, Kauaki Ogawa, Eiichi Okamoto, Kazuyuki Masukane, Kazuya Oda und Masafumi Inuiya. A Progressive Scan CCD Image Sensor for DSC Applications. *IEEE Journal of Solid-State Circuits*, 35, Nr. 12, 2000.
- [Yoder '97] Lars Yoder. *The Digital Display Technology of the Future*, 1997.
- [Zhang et al. '02] Li Zhang, Brian Curless und Steven M. Seitz. Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming. In *The 1st IEEE International Symposium on 3D Data Processing, Visualization, and Transmission*, S. 24–36, June 2002.
- [Zhang et al. '03] Li Zhang, Brian Curless und Steven M. Seitz. Spacetime Stereo: Shape Recovery for Dynamic Scenes. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, S. 367–374, June 2003.
- [Zhang et al. '04] Li Zhang, Noah Snavely, Brian Curless und Steven M. Seitz. Spacetime Faces: High-Resolution Capture for Modeling and Animation. In *ACM Annual Conference on Computer Graphics*, S. 548–558, August 2004.
- [Zhang '00] Zhengyou Zhang. A Flexible New Technique for Camera Calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22, Nr. 11, S. 1330–1334, 2000.

Lebenslauf

- 17.01.1981 in Saarbrücken geboren
- 15.09.1992 - 15.05.1999 Deutsch-Französisches Gymnasium Saarbrücken
15.05.1999 Erlangung des Baccalaureat Franco-Allemand
- 01.11.1999 - 01.08.2000 Wehrdienst beim 2. FallschirmPanzerAbwehr-
Bataillon 262 in Merzig
- Seit 01.10.2000 Studium im Fach Informatik an der Universität
Karlsruhe (TH)
- 08.04.2002 Vordiplom
- 01.10.2002 - 01.09.2003 Erasmus Auslandstudium an der École Nationale
Supérieure d'Informatique et des Mathématiques
Appliquées de Grenoble
- 01.05.2004 - 31.03.2005 Wissenschaftliche Hilfskraft am Institut für
Betriebs- und Dialogsysteme, Universität
Karlsruhe (TH)