

**Evaluation von  
Glanzlichtdetektionsverfahren  
in der Endoskopie**

**Studienarbeit im Fach Informatik**

vorgelegt  
von

Philipp Schaer

Geboren am 7. Mai 1980 in Troisdorf

Angefertigt am

Institut für Computervisualistik  
Arbeitsgruppe Aktives Sehen  
Universität Koblenz–Landau

Betreuer: Prof. Dr.-Ing. D. Paulus und Dipl.-Inform. F. Vogt (Universität Erlangen)

Beginn der Arbeit: 15.06.2004

Abgabe der Arbeit: 09.12.2004

Ich versichere, dass ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Die Richtlinien des Instituts für Studien- und Diplomarbeiten habe ich gelesen und anerkannt, insbesondere die Regelung des Nutzungsrechts.

Koblenz, den 09.12.2004

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung und Motivation</b>	<b>5</b>
<b>2</b>	<b>Bildgewinnung durch Endoskopie</b>	<b>9</b>
2.1	Aufbau eines Endoskops . . . . .	9
2.2	Probleme . . . . .	10
<b>3</b>	<b>Glanzlichtdetektion</b>	<b>13</b>
3.1	HSV-Schwellwerte . . . . .	13
3.2	Glanzlichtkanten . . . . .	14
3.2.1	Dichromatisches Reflexionsmodell . . . . .	14
3.2.2	Vereinfachung des Reflexionsmodells . . . . .	16
3.2.3	Invariante Farbräume . . . . .	17
3.2.4	Farbinvarianz-Taxonomie . . . . .	20
3.2.5	Farbinvariante Gradienten . . . . .	20
3.2.6	Klassifikation der Kanten . . . . .	24
<b>4</b>	<b>Experimente und Evaluation</b>	<b>25</b>
4.1	Versuchsaufbau . . . . .	25
4.1.1	Synthetische Bildsequenz . . . . .	26

4.1.2	Reale Operationsaufnahmen . . . . .	27
4.1.3	Reale Laboraufnahmen . . . . .	28
4.2	Zusammenfassung der Ergebnisse . . . . .	30
<b>5</b>	<b>Fazit</b>	<b>33</b>
5.1	Zusammenfassung . . . . .	33
5.2	Ausblick . . . . .	34
5.2.1	Hardwaregestützte Beschleunigung . . . . .	34
5.2.2	Glanzlichtsubstitution durch Lichtfelder . . . . .	36
	<b>Anhang A - Implementation</b>	<b>39</b>
	<b>Anhang B - Tabellen</b>	<b>51</b>
	<b>Literaturverzeichnis</b>	<b>54</b>

# Kapitel 1

## Einleitung und Motivation

In medizinischen Bereichen, wie z.B. der Darmkrebsvorsorge oder bei minimalinvasiven Operationstechniken, ist die Endoskopie nicht mehr wegzudenken. Von den ersten Beschreibungen eines Endoskops durch Hippocrates [RR98], hat sich die Endoskopie bis Heute zu einem hochtechnisierten Einsatzgebiet der Medizin weiterentwickelt.

Der Begriff der Endoskopie stammt aus dem Griechischen. Er setzt sich zusammen aus den Wörtern *endo*: innen, innerhalb und *skopein*: betrachten. Geschah dieses Betrachten des Inneren ehemals durch ein starres Rohr und eine Kerze, so arbeiten heutige Endoskope mit Glasfasern und CCD Chips zur Lichtübertragung und Bildaufzeichnung.

Der Einsatz von Bildverarbeitung in der Medizin bringt naheliegende Vorteile und Verbesserungen mit sich. Durch die Unterstützung des behandelnden Arztes oder des medizinischen Personals können die Operations- und Behandlungsbedingungen verbessert werden, was wiederum zu einer Verbesserung der Behandlungsergebnisse führt. Gerade in der Endoskopie kann durch die unterstützende Bildverarbeitung eine Reduzierung des Operationsrisikos und eine Verminderung der Komplikationen erreicht werden.

Diese Arbeit behandelt nun ein spezielles Problem bei Endoskopieaufnahmen, *Glanzlichter*. Als Glanzlichter bezeichnet man Stellen im Bild, die durch starke Reflexionen

vollständig überbelichtet sind, so daß das eigentliche aufgenommene Material nicht mehr zu erkennen ist. Bei Endoskopieaufnahmen im Inneren des Körpers kommt es häufig zu sehr ausgeprägten Glanzlichtern, da sich das Licht der Beleuchtungsquelle am feuchten Gewebe im Inneren des Körpers spiegelt. Diese Glanzlichter behindern den Arzt und verhindern, das Gewebe richtig zu erkennen. Das Ziel dieser Arbeit richtet sich nun auf die Detektion dieser Glanzlichtstellen.

Um diese Glanzlichter sicher zu detektieren, werden zwei unterschiedliche Ansätze vorgestellt, implementiert und gegeneinander evaluiert. Hierbei handelt es sich um ein Schwellwertverfahren im *HSV* Farbraum und um Farbgradienten, die auf dem dichromatischen Reflexionsmodell basieren. Die Detektion ist nur ein Vorverarbeitungsschritt für weitere Bildverarbeitungsschritte, um z.B. durch die Anwendung von Lichtfeldern die Glanzlichter korrigieren zu können. Hierzu ist allerdings eine robuste und performante Detektion der Glanzlichter unerlässlich.

In Kapitel 2 wird allgemein der Aufbau eines Endoskops beschrieben und auf die Probleme bei der Bildgewinnung mit diesen Mitteln eingegangen. Dabei wird speziell die Problematik der Glanzlichter vorgestellt.

Kapitel 3 behandelt die theoretischen Grundlagen für die Glanzlichtdetektion. Sowohl die Ansätze für die *HSV*-Schwellwerte, als auch für die Farbgradienten werden erläutert. Besonderes Augenmerk wird hierbei auf die Herleitung der neuen Farbräume  $c_1c_2c_3$  und  $l_1l_2l_3$  gelegt. Diese Farbräume werden genutzt, um unterschiedliche Kanten in Bildern zu klassifizieren. Mit dem vorgestellten Verfahren ist es möglich, Schatten-, Geometrie-, Material- und Glanzlichtkanten voneinander abzugrenzen. Hierzu werden die theoretischen Grundlagen des dichromatischen Reflexionsmodells vorgestellt.

Die Evaluation der beiden unterschiedlichen Glanzlichtdetektionsverfahren ist Inhalt von Kapitel 4. Hierzu werden die Verfahren auf synthetische Bildsequenzen, reale Operationsaufnahmen, die mit einem Endoskop aufgenommen wurden, und auf reale Aufnahmen, die künstlich arrangiert wurden, angewendet.

In Kapitel 5 werden die Ergebnisse und Erkenntnisse als Fazit zusammengefasst. Es werden weitere Arbeitsfelder und Verbesserungen vorgestellt, die allerdings nicht Teil dieser Arbeit waren. Diese sollen einen Ausblick auf weitere Arbeitsfelder geben.

Im Anhang A wird die Implementation zusammengefasst. Anschließend wird auf die Besonderheiten eingegangen, die für die Umsetzung der beiden Verfahren notwendig waren. Es wird kurz die Programmumgebung PUMA vorgestellt, in der die Algorithmen implementiert wurden. Die einzelnen Programmmodule werden vorgestellt und die interne Struktur erläutert.

Anhang B enthält einige Tabellen, die zur Unterstützung für den Evaluationsteil dienen.



# Kapitel 2

## Bildgewinnung durch Endoskopie

### 2.1 Aufbau eines Endoskops

Prinzipiell muss zwischen *starren* und *flexiblen* Endoskopen unterschieden werden. Starre Endoskope werden häufig in der minimalinvasiven Chirurgie eingesetzt, wogegen flexible Endoskope häufig in der Diagnostik eingesetzt werden. Allgemein kann man auch sagen, dass starre Endoskope für alle “unnatürlichen” und flexible Endoskope für alle “natürlichen” Körperöffnungen verwendet werden. Der traditionelle Aufbau eines Endoskops basiert auf Glasfasern sowohl für die Bild-, als auch für die Lichtübertragung.

Als Lichtquelle werden bis zu 400 Watt starke Xenon-Lampen eingesetzt. Durch die flexiblen Glasfasern und die Optik geht ein Teil der Lichtmenge verloren. Eine weitere Steigerung der Lichtleistung kann allerdings nicht durchgeführt werden, da sonst das empfindliche Gewebe im Innern des Körpers geschädigt werden könnte. Dies ist auch der Grund, warum die Lichtquelle nicht direkt am Endoskopkopf montiert wird. Die extreme Hitzeentwicklung würde das Gewebe verbrennen. Das Licht aus der Lichtquelle wird durch die lichtleitenden, flexiblen Glasfasern zum Endoskopkopf geleitet.

Die Bildübertragung geschah bisher ebenfalls über Glasfasern. Hierbei wird für die Übertragung ein kohärentes Glasfaserbündel von bis zu 50.000 Einzelfasern eingesetzt, dies

entspricht einer Bildauflösung von ca. 220 x 200 Bildpunkten. Am Endoskop befinden sich zwei Okulare, eines am Kopf und eines am Ende des Endoskops. Das erste Okular rastert das Bild auf. Dieses aufgerasterte Bild wird durch die Glasfaser bis zum zweiten Okular gespiegelt. Dort wird das Bild wieder zusammengesetzt. Dieses Bild kann nun direkt am Okular betrachtet werden oder auch auf einen externen Bildschirm mit Hilfe einer Videokamera übertragen werden. Diesen Vorgang bezeichnet man als *indirekte Videoendoskopie*.

## 2.2 Probleme

Das vorgestellte Verfahren der Bildübertragung mittels Glasfasern bringt Nachteile mit sich, da z.B.

- eine schlechte Bildauflösung durch eine zu geringe Anzahl an Glasfasern,
- falsch zugeordnete Glasfasern,
- ungleichmäßig dicke Glasfasern,
- zu dicke Fasern

die Bildqualität erheblich verschlechtern können.

Moderne Endoskope arbeiten inzwischen mit CCD-Technologie, so daß die oben beschriebenen Probleme nicht mehr auftreten. Die Mindestgröße von CCD-Chips erschwert bisher noch den Einsatz bei der Endoskopie von Blutgefäßen, Harn-, Gallen- oder Atemwegen.

Eine schlechte Bildqualität erschwert die Arbeit von Ärzten, sowohl bei der Diagnostik, als auch in der minimalinvasiven Chirurgie, da diese sich vollständig auf die ihnen gelieferten Bilder verlassen müssen. Auch für eine spätere Weiterverarbeitung der Bilder durch den Rechner ist eine schlechte Bildqualität nicht wünschenswert.

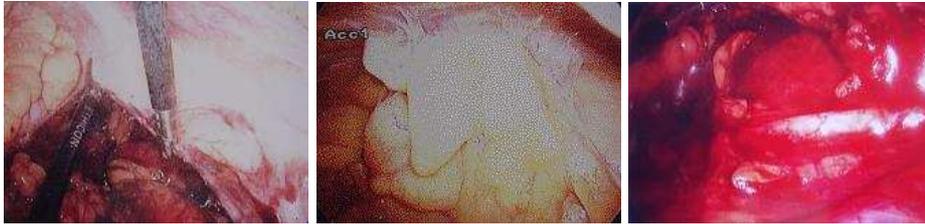


Bild 2.1: *Typische Bildstörungen bei Endoskopischen Aufnahmen: (a) Farbfehler, (b) Schwebepartikel, (c) Glanzlichter und Farbfehler*

Eine besondere Problematik in der Endoskopie stellen Glanzlichter dar. Die besonderen Gegebenheiten der Instrumente und des Körperinneren verstärken den Glanzlichteffekt. Durch den hohen Wasseranteil haben menschliche Organe eine glatte, reflektierende Oberfläche, spiegeln dadurch also besonders stark. Desweiteren können durch das Spülen der Wunde bei einer Operation Flüssigkeiten auf die Gewebeoberfläche gelangen, ebenso wie Blut, welches als Flüssigkeit einen hohen Reflexionsgrad aufweist. Die Lichtquelle beim Endoskop jedoch sitzt nahezu parallel zur Aufnahmeoptik, so daß das ausgesendete Licht auf dieser spiegelnden Oberfläche direkt zurückgeworfen werden kann. Durch diese Reflexion entstehen an der Oberfläche starke Kontraste.

Bild 2.1 zeigt einige weitere Probleme bei der Aufnahme von Endoskopiebildern, darunter Farbverfälschungen und Schwebepartikel. Diese Probleme stellen weitere Arbeitsfelder dar und werden z.B. in [VKP<sup>+</sup>01] behandelt.



# Kapitel 3

## Glanzlichtdetektion

Inhalt dieses Kapitels ist die Vorstellung zweier unterschiedlicher Verfahren zur Glanzlichtdetektion in endoskopischen Bildern, die Detektion mittels einfacher *HSV-Schwellwerte* und mit Hilfe von *Gradientenbilder*, die auf dem *dichromatischen Reflexionsmodells* basieren. Diese beiden Verfahren sind, neben anderen, eine kleine Auswahl von Glanzlichtdetektionsverfahren. Koschan [Kos97] stellt eine Übersicht über verschiedene weitere Glanzlichtdetektionsverfahren zusammen.

### 3.1 HSV-Schwellwerte

Ein intuitiver Ansatz zur Detektion von Glanzlichtern ist ein einfaches Schwellwertverfahren im *HSV* Farbraum. Im dreidimensionalen *HSV* Farbraum wird Farbe durch den Farbwinkel (engl. *Hue*), die Farbsättigung (engl. *Saturation*) und die Farbhelligkeit (engl. *Value*) beschrieben. Wenn man nun davon ausgeht, dass Glanzlichter überbelichtete Stellen im Bild sind, so kann man davon ausgehen, dass Glanzlichter durch eine hohe Helligkeit und eine niedrige Farbsättigung gekennzeichnet sind. Die Werte des Farbwinkels sind nicht mehr relevant, da durch die Überbelichtung diese Farbinformationen verloren gehen.

Definiere Schwellwerte für Sättigung SAT und Intensität VAL; Definiere Binärmaske H;	
FOR alle Bildzeilen $y$	
FOR alle Bildspalten $x$	
IF	Sättigung an Position $x, y \leq \text{SAT}$ und Intensität an Position $x, y \geq \text{VAL}$
THEN	Setze Binärmaske H an Position $x, y = 1$
ELSE	Setze Binärmaske H an Position $x, y = 0$

Bild 3.1: Detektion von Glanzlichtern durch Schwellwerte im HSV Farbraum

Konkret bedeutet dies, dass mit einfachen Schwellwerten das Bild binarisiert werden kann, um eine Maske mit den detektierten Glanzlichtern zu erstellen. In Bild 3.1 wird ein solches Binarisierungsverfahren skizziert.

Glanzlichter wurden mittels *HSV*-Schwellwerten von [VKP<sup>+</sup>01]<sup>1</sup> in der Endoskopie des Bauchraums detektiert, wobei  $H \in [0, 359]$ ,  $S \in [0, 255]$  und  $V \in [0, 255]$ . Für eine Sequenz von Gallenbildern wurden z.B. Schwellwerte von  $0 \leq H \leq 359$ ,  $0 \leq S \leq 20$  und  $V \geq 200$  verwendet. Für eine Aufnahme der Speiseröhre wurden folgende Werte verwendet:  $0 \leq H \leq 359$ ,  $0 \leq S \leq 40$  und  $V \geq 200$ .

## 3.2 Glanzlichtkanten

### 3.2.1 Dichromatisches Reflexionsmodell

Das *dichromatische Reflexionsmodell* basiert auf den Ausführungen von Shafer [Sha85]. Die spektrale Verteilung des reflektierten Lichtes setzt sich danach aus einer Linearkombination einer *spiegelnden* und einer *diffusen* Reflexion zusammen (vgl. Bild 3.2 und Formel 3.1). Beim spiegelnden Anteil spricht man auch von einem *Oberflächenreflexionsanteil*

<sup>1</sup>In diesem Paper werden für den Value Wert andere Schwellwerte definiert, die allerdings auf einem Schreibfehler beruhen. Die Schwellwerte werden in dieser Arbeit in der korrigierten Fassung dargestellt.

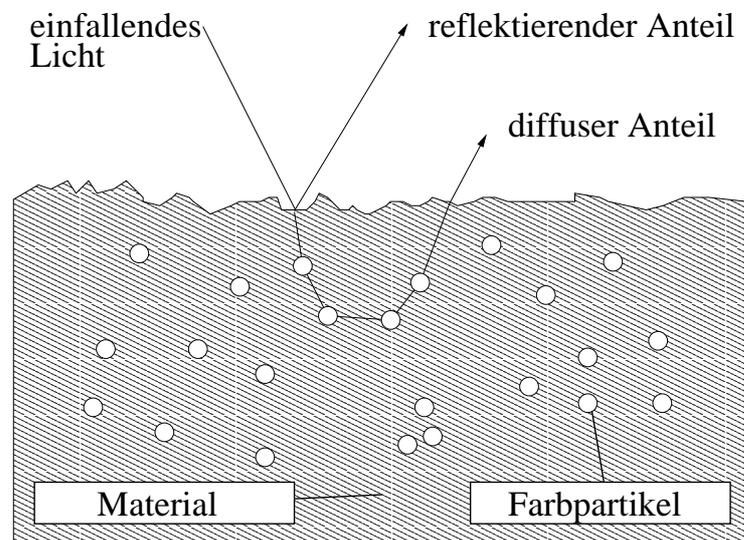


Bild 3.2: Schematische Darstellung des dichromatischen Reflexionsmodells mit den spiegelnden Oberflächen- und diffusen, matten Körperreflexionsanteilen

(engl. interface oder surface reflection) und beim diffusen von einem *Körperreflexionsanteil* (engl. body reflection).

Bei der Oberflächenreflexion trifft das Licht auf die Materialoberfläche und wird unverändert zurückgeworfen, bei der Körperreflexion hingegen tritt das einfallende Licht in das beleuchtete Medium ein und wird in den Farbpartikeln des Mediums gestreut. Dabei wird das einfallende Licht spektral verändert und im Gegensatz zur Oberflächenreflexion diffus zurückgeworfen.

In [GS00] wird ein Verfahren vorgestellt um Glanzlichter mit Hilfe von Farbgradienten zu detektieren. Hierzu wird das dichromatische Reflexionsmodell genutzt, Farbsprünge in Farbbildern zu klassifizieren. Mit diesem Klassifikationsverfahren ist es möglich, Gradienten in Bildern in folgende Klassen zu unterteilen:

- Kanten hervorgerufen durch Schatten oder Geometrieveränderung,
- Kanten hervorgerufen durch Materialänderung,
- Kanten hervorgerufen durch Glanzlichter

Hierzu werden drei unterschiedliche Merkmale zur Klassifikation verknüpft: Farbmerkmale, farbinvariante Merkmale für matte Objekte und farbinvariante Merkmale für glänzende Objekte.

Überlegungen zu Eigenschaften benachbarter Bildpunkte sind die Grundlage für die Klassifikation, wie sie von Gevers in [GS99a] vorgeschlagen werden. Diese Überlegungen basieren auf dem dichromatischen Reflexionsmodell mit einer konstanten Beleuchtungsquelle. Gevers betrachtet die Fragestellung, welche Farbwertveränderungen auftreten können. Für unterschiedliche Oberflächen sind verschiedene Merkmale invariant. Nach Gevers sind zwei Farbwerte für die reflektierte Farbe ausschlaggebend: die *Beleuchtungsfarbe* und die *Objektfarbe*.

### 3.2.2 Vereinfachung des Reflexionsmodells

Allgemeine Annahme ist ein Bild mit einem infinitesimal Oberflächenstück. Wir benutzen den Rot-, Grün- und Blaukanal mit einer spektralen Sensitivität, die durch  $f_R(\lambda)$ ,  $f_G(\lambda)$  und  $f_B(\lambda)$  angegeben wird. Dies wird genutzt, um ein Bild des Oberflächenstücks zu erhalten, welches, beleuchtet durch die SPD, von einem einfallenden Licht durch  $e(\lambda)$  beschrieben wird. Die gemessenen Sensorwerte werden in Anlehnung an Shafer [Sha85] beschrieben durch:

$$C = m_b(\mathbf{n}, \mathbf{s}) \int_{\lambda} f_C(\lambda) e(\lambda) c_b(\lambda) d\lambda + m_s(\mathbf{n}, \mathbf{s}, \mathbf{v}) \int_{\lambda} f_C(\lambda) e(\lambda) c_s(\lambda) d\lambda \quad (3.1)$$

wobei  $C = \{R, G, B\}$  jeweils das  $C$ te Sensorsignal bezeichnet.  $c_b$  und  $c_s$  bezeichnen jeweils für den Körper und die Oberfläche die Reflexions- (Fresnel) und Rückstrahleigenschaften (Albedo).  $\lambda$  bezeichnet die Wellenlänge des Lichtes,  $\mathbf{n}$  die Oberflächennormale,  $\mathbf{s}$  ist die Ausrichtung der Beleuchtungsquelle und  $\mathbf{v}$  ist die Ausrichtung des Betrachters. Die geometrischen Terme  $m_b$  und  $m_s$  bezeichnen die geometrische Abhängigkeit zwischen der Körper- und der Oberflächenreflexion. Wir sehen hier auch, dass die Körperreflexion nicht von der geometrischen Ausrichtung des Betrachters abhängig ist, da diese Reflexion, im Gegensatz zur Oberflächenreflexion, diffus ist.

Gehen wir von einem neutralen Interface Reflexion Modell (NIR) aus sowie einer “weißen” Beleuchtung, dann sind  $e(\lambda) = e$  und  $c_s(\lambda) = c_s$ , also konstant. Neutrale Interface Reflexion bedeutet in diesem Zusammenhang, dass angenommen wird, dass  $c_s(\lambda)$  einen konstanten Wert, unabhängig von der Wellenlänge des Lichts, besitzt. So ergibt sich ein gemessener Sensorwert, der beschrieben wird durch:

$$C_w = em_b(\mathbf{n}, \mathbf{s})k_C + em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_S \int_{\lambda} f_C(\lambda)d\lambda \quad (3.2)$$

wobei  $C_w \in \{R_w, G_w, B_w\}$  die roten, grünen und blauen Sensorrückmeldungen unter der Annahme einer weißen Beleuchtung bezeichnen.  $k_C = \int_{\lambda} f_c(\lambda)c_b(\lambda)d\lambda$  ist eine zusammengefasste Bezeichnung, basierend auf den Sensoren und den Reflexionseigenschaften.

Aufgrund der weißen Beleuchtungsbedingung gilt folgende Vereinfachung:  $\int_{\lambda} f_R(\lambda)d\lambda = \int_{\lambda} f_G(\lambda)d\lambda = \int_{\lambda} f_B(\lambda)d\lambda = f$ . So ergibt sich:

$$C_w = C_b + C_s = em_b(\mathbf{n}, \mathbf{s})k_c + em_s(\mathbf{n}, \mathbf{s}, \mathbf{v})c_S f \quad (3.3)$$

wobei  $C_b$  der Körperreflexion und  $C_s$  der Oberflächenreflexion entspricht.

### 3.2.3 Invariante Farbräume

Wie in Kapitel 3.2.1 gezeigt, setzt sich das gesehene Licht aus einer Linearkombination der spiegelnden und diffusen Teile zusammen. Die Folgerung daraus ist, dass für matte, stumpfe Oberflächen und glänzende Oberflächen unterschiedliche Annahmen gemacht werden können. Auf Grundlage dieser Überlegungen werden zwei Farbräume vorgeschlagen, die diese Unterschiedlichkeit berücksichtigen.

#### Invariante Farbmerkmale für matte Objektflächen

Basierend auf der Beschreibung der Körperreflexion in Formel 3.3,  $em_b(\mathbf{n}, \mathbf{s})k_c$ , kann eine gleichmäßig gefärbte Oberfläche trotzdem eine große Zahl von unterschiedlichen *RGB*-Werten aufweisen. Diese unterschiedlichen Werte resultieren z.B. aus einer Krümmungen

des Objektes. Der  $RGB$ -Farbraum ist also ungeeignet um farbinvariante Merkmale zu extrahieren. Gever schlägt daher einen neuen, normalisierten Farbraum  $c_1c_2c_3$  vor.

Dieser basiert auf den Ideen der Farbnormalisierung von  $RGB$  nach  $rgb$ , da diese nicht sensitiv auf Oberflächenausrichtung, Beleuchtungsausrichtung und Beleuchtungsintensität sind.

$$r(R, G, B) = \frac{em_b(\mathbf{n}, \mathbf{s})k_R}{em_b(\mathbf{n}, \mathbf{s})(k_R + k_G + k_B)} = \frac{k_R}{k_R + k_G + k_B} \quad (3.4)$$

$$g(R, G, B) = \frac{em_b(\mathbf{n}, \mathbf{s})k_G}{em_b(\mathbf{n}, \mathbf{s})(k_R + k_G + k_B)} = \frac{k_G}{k_R + k_G + k_B} \quad (3.5)$$

$$b(R, G, B) = \frac{em_b(\mathbf{n}, \mathbf{s})k_B}{em_b(\mathbf{n}, \mathbf{s})(k_R + k_G + k_B)} = \frac{k_B}{k_R + k_G + k_B} \quad (3.6)$$

Hier ist nur noch eine Abhängigkeit von den Sensoren und den Reflexionseigenschaften gegeben.

Desweiteren ist der Farbwinkel  $H$  invariant für matte Oberflächen. Auf Grundlage des  $HSV$ -Farbraumes ist der Farbwinkel nicht abhängig von Farbsättigung oder Farbtintensität.

$$H(R, G, B) = \arctan\left(\frac{\sqrt{3}em_b(\mathbf{n}, \mathbf{s})(k_G - k_B)}{em_b(\mathbf{n}, \mathbf{s})((k_R - k_G) + (k_R - k_B))}\right) \quad (3.7)$$

$$= \arctan\left(\frac{\sqrt{3}(k_G - k_B)}{((k_R - k_G) + (k_R - k_B))}\right) \quad (3.8)$$

Auf Grundlage dieser beiden Farbmodelle wird der Farbraum  $c_1c_2c_3$  wie folgt definiert:

$$c_1 = \arctan\left(\frac{R}{\max(G, B)}\right) \quad (3.9)$$

$$c_2 = \arctan\left(\frac{G}{\max R, B}\right) \quad (3.10)$$

$$c_3 = \arctan\left(\frac{B}{\max R, G}\right) \quad (3.11)$$

Es kann nun bewiesen werden, dass der  $c_1c_2c_3$ -Farbraum, der den Winkel der Körperreflexion angibt, invariant für matte Objetoberflächen ist (vgl. [GS99a]).

### Invariante Farbmerkmale für matte und glänzende Objetoberflächen

Wir nehmen an, dass unter den gegebenen Bedingungen (NIR-Modell), die Farbe der Glanzlichter nicht von der Farbe des Objektes abhängen, auf denen sie erscheinen, sondern ausschließlich von der Farbe der Beleuchtungsquelle. Folglich liegt bei einer weißen Beleuchtungsquelle die Oberflächenreflexion, aufgrund der unterschiedlichen Rückstrahleigenschaften (Albedoeigenschaften), in einem Farbcluster auf der Grauachse des  $RGB$ -Farbraums. Dies entspricht der Farbtintensität  $V$ . Für einen gegebenen Punkt auf einer glänzenden Oberfläche werden die Komponenten der Körperreflexion und der Oberflächenreflexion zusammenaddiert. Folglich muss die beobachtete Farbe innerhalb des Dreiecksfarbclusters im  $RGB$ -Farbraum liegen, das von den beiden Reflexionskomponenten aufgespannt wird. Es kann erneut der Farbton  $H$  herangezogen werden. Dieser drückt den Winkel für einen Farbvektor in Bezug zur Unbuntachse aus (vgl. Formel 3.7).

Gevers schlägt den Farbraum  $l_1l_2l_3$  vor, der invariant sowohl für matte als auch für glänzende Oberflächen ist. Diese entsprechen den normalisierten Farbdifferenzen:

$$l_1 = \frac{(R - G)^2}{(R - G)^2 + (R - B)^2 + (G - B)^2} \quad (3.12)$$

$$l_2 = \frac{(R - B)^2}{(R - G)^2 + (R - B)^2 + (G - B)^2} \quad (3.13)$$

$$l_3 = \frac{(G - B)^2}{(R - G)^2 + (R - B)^2 + (G - B)^2} \quad (3.14)$$

	Geometrie	Schatten	Glanzlichter	Material
$RGB$	+	+	+	+
$c_1 c_2 c_3$	-	-	+	+
$l_1 l_2 l_3$	-	-	-	+

Tabelle 3.1: *Taxonomie der Farbinvarianzen verschiedener Farbmodelle.* - bedeutet, dass der Farbraum invariant bzgl. dieser Merkmale ist, + bedeutet, dass der Farbraum sensitiv auf diese Merkmale reagiert.

Auch für diesen Farbraum kann bewiesen werden, dass er invariant für matte und glänzende Objekte ist (vgl. [GS99a]).

### 3.2.4 Farbinvarianz-Taxonomie

Im vorherigen Kapitel wurden die unterschiedlichen Farbmerkmale aufgrund von Materialien und Beleuchtung betrachtet. Hieraus lässt sich schlussfolgern, dass  $l_1 l_2 l_3$  variant bzgl. Materialänderungen ist.  $c_1 c_2 c_3$  ist variant bzgl. Materialänderungen und Glanzlichtern wobei  $RGB$  variant bzgl. allen Merkmalen ist, also Material- und Geometrieänderungen, bzw. Schatten und Glanzlichtern. Die Farbinvarianz-Taxonomie wird in Tabelle 3.1 dargestellt. Die Taxonomie basiert auf den Empfindlichkeiten der Farbmodelle bzgl. der Merkmale, die in Abschnitt 3.2.3 vorgestellt wurden.

Ausgehend von dieser Taxonomie wird eine Klassifizierung der Farbkanten möglich sein, wobei in unserem Fall nur die Glanzlichtkanten von Bedeutung sind.

### 3.2.5 Farbinvariante Gradienten

In diesem Abschnitt werden farbinvariante Kanten vorgestellt, die auf den Farbräumen aus Kapitel 3.2.3 basieren. Da das Verfahren nicht nur für dreidimensionale  $RGB$  Bilder angewendet werden kann, wird ein Ansatz für mehrdimensionale Farbgradienten vorgestellt.

### Farbgradienten in vektorwertigen Bildern

Der Ansatz für Farbgradienten in vektorwertigen Bildern geht auf Silvano di Zenzo [dZ86] zurück und wurde in [SR96] erweitert.

$\Theta(x_1, x_2) : \mathbb{R}^2 \rightarrow \mathbb{R}^m$  ist ein Bild mit den Komponenten  $\Theta_i(x_1, x_2) : \mathbb{R}^2 \rightarrow \mathbb{R}$ , wobei  $i = 1, 2, \dots, m$ . Für dreidimensionale Farbbilder (z.B. *RGB*) gilt  $m = 3$ . Folglich ist an einer gegebenen Bildposition der Bildwert ein Vektor in  $\mathbb{R}^m$ . Die Differenz zweier benachbarter Punkte  $P = (x_1^0, x_2^0)$  und  $Q = (x_1^1, x_2^1)$  ist definiert durch  $\Delta\Theta = \Theta(P) - \Theta(Q)$ . Unter der Annahme einer infinitesimalen Verschiebung wird diese Differenz zu einem Differential  $d\Theta = \sum_{i=1}^2 \frac{\partial\Theta}{\partial x_i} dx_i$ . Die quadrierte Form wird dann bezeichnet durch:

$$d\Theta^2 = \sum_{i=1}^2 \sum_{k=1}^2 \frac{\partial\Theta}{\partial x_i} \frac{\partial\Theta}{\partial x_k} dx_i dx_k \quad (3.15)$$

$$= \sum_{i=1}^2 \sum_{k=1}^2 g_{ik} dx_i dx_k \quad (3.16)$$

$$= \begin{pmatrix} dx_1 \\ dx_2 \end{pmatrix}^T \begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix} \begin{pmatrix} dx_1 \\ dx_2 \end{pmatrix} \quad (3.17)$$

wobei  $g_{ik} := \frac{\partial\Theta}{\partial x_i} \frac{\partial\Theta}{\partial x_k}$ . Die Extrema der quadratischen Form entsprechen den Richtungen der Eigenvektoren der Matrix  $[g_{ik}]$  und die Werte an diesen Stellen entsprechen den Eigenwerten, die wie folgt definiert sind:

$$\lambda_{\pm} = \frac{g_{11} + g_{22} \pm \sqrt{(g_{11} - g_{22})^2 + 4g_{12}^2}}{2} \quad (3.18)$$

Die zugehörigen Eigenvektoren sind  $\cos \theta_{\pm}$  und  $\sin \theta_{\pm}$ , wobei  $\theta_+ = \frac{1}{2} \arctan \frac{2g_{12}}{g_{11} - g_{22}}$  und  $\theta_- = \theta_+ + \frac{\pi}{2}$ . Folglich repräsentieren die Eigenwerte  $\theta_-$  und  $\theta_+$  die Richtung der minimalen und maximalen Änderungen an einer gegebenen Bildposition. Die entsprechende Stärke der Änderung ist durch die beiden Eigenwerte  $\lambda_-$  und  $\lambda_+$  gegeben. Die Stärke dieses vektorwertigen Gradienten ergibt sich aus einem Vergleich zwischen  $\lambda_-$  und  $\lambda_+$ . Eine Möglichkeit besteht darin, die beiden Werte voneinander zu subtrahieren (vgl. [SR96]). Eine allgemeine Form der Änderung kann wie folgt definiert werden:

$$\nabla C = \sqrt{\lambda_+ - \lambda_-} \quad (3.19)$$

Dies wird im nächsten Abschnitt dazu genutzt, Gradienten in farbinvarianten Bildern zu definieren.

### Gradienten in farbinvarianten vektorwertigen Bildern

Für die drei vorgestellten Farbräume ( $RGB$ ,  $l_1l_2l_3$  und  $c_1c_2c_3$ ) werden nun farbinvariante Gradienten vorgestellt, die auf den Überlegungen des vorherigen Abschnitts beruhen.

$$\nabla C_{RGB} = \sqrt{\lambda_+^{RGB} - \lambda_-^{RGB}} \quad (3.20)$$

$$\lambda_{\pm} = \frac{g_{11}^{RGB} + g_{22}^{RGB} \pm \sqrt{(g_{11}^{RGB} - g_{22}^{RGB})^2 + 4(g_{12}^{RGB})^2}}{2} \quad (3.21)$$

wobei

$$g_{11}^{RGB} = \left| \frac{\partial R}{\partial x} \right|^2 + \left| \frac{\partial G}{\partial x} \right|^2 + \left| \frac{\partial B}{\partial x} \right|^2 \quad (3.22)$$

$$g_{22}^{RGB} = \left| \frac{\partial R}{\partial y} \right|^2 + \left| \frac{\partial G}{\partial y} \right|^2 + \left| \frac{\partial B}{\partial y} \right|^2 \quad (3.23)$$

$$g_{12}^{RGB} = \frac{\partial R}{\partial x} \frac{\partial R}{\partial y} + \frac{\partial G}{\partial x} \frac{\partial G}{\partial y} + \frac{\partial B}{\partial x} \frac{\partial B}{\partial y} \quad (3.24)$$

Weiter können farbinvariante Gradienten für matte Objekte mit Hilfe des  $c_1c_2c_3$  Farbraums beschrieben werden:

$$\nabla C_{c_1c_2c_3} = \sqrt{\lambda_+^{c_1c_2c_3} - \lambda_-^{c_1c_2c_3}} \quad (3.25)$$

$$\lambda_{\pm} = \frac{g_{11}^{c_1c_2c_3} + g_{22}^{c_1c_2c_3} \pm \sqrt{(g_{11}^{c_1c_2c_3} - g_{22}^{c_1c_2c_3})^2 + 4(g_{12}^{c_1c_2c_3})^2}}{2} \quad (3.26)$$

wobei

$$g_{11}^{c_1c_2c_3} = \left| \frac{\partial c_1}{\partial x} \right|^2 + \left| \frac{\partial c_2}{\partial x} \right|^2 + \left| \frac{\partial c_3}{\partial x} \right|^2 \quad (3.27)$$

$$g_{22}^{c_1 c_2 c_3} = \left| \frac{\partial c_1}{\partial y} \right|^2 + \left| \frac{\partial c_2}{\partial y} \right|^2 + \left| \frac{\partial c_3}{\partial y} \right|^2 \quad (3.28)$$

$$g_{12}^{c_1 c_2 c_3} = \frac{\partial c_1}{\partial x} \frac{\partial c_1}{\partial y} + \frac{\partial c_2}{\partial x} \frac{\partial c_2}{\partial y} + \frac{\partial c_3}{\partial x} \frac{\partial c_3}{\partial y} \quad (3.29)$$

Auf gleiche Weise werden Gradienten für glänzende Objekte definiert. Diese basieren auf dem Farbraum  $l_1 l_2 l_3$ :

$$\nabla C_{l_1 l_2 l_3} = \sqrt{\lambda_+^{l_1 l_2 l_3} - \lambda_-^{l_1 l_2 l_3}} \quad (3.30)$$

$$\lambda_{\pm} = \frac{g_{11}^{l_1 l_2 l_3} + g_{22}^{l_1 l_2 l_3} \pm \sqrt{(g_{11}^{l_1 l_2 l_3} - g_{22}^{l_1 l_2 l_3})^2 + 4(g_{12}^{l_1 l_2 l_3})^2}}{2} \quad (3.31)$$

wobei

$$g_{11}^{l_1 l_2 l_3} = \left| \frac{\partial l_1}{\partial x} \right|^2 + \left| \frac{\partial l_2}{\partial x} \right|^2 + \left| \frac{\partial l_3}{\partial x} \right|^2 \quad (3.32)$$

$$g_{22}^{l_1 l_2 l_3} = \left| \frac{\partial l_1}{\partial y} \right|^2 + \left| \frac{\partial l_2}{\partial y} \right|^2 + \left| \frac{\partial l_3}{\partial y} \right|^2 \quad (3.33)$$

$$g_{12}^{l_1 l_2 l_3} = \frac{\partial l_1}{\partial x} \frac{\partial l_1}{\partial y} + \frac{\partial l_2}{\partial x} \frac{\partial l_2}{\partial y} + \frac{\partial l_3}{\partial x} \frac{\partial l_3}{\partial y} \quad (3.34)$$

	Geometriekante	Schattenkante	Glanzlichtkante	Materialkante
$\nabla C_{RGB}$	+	+	+	+
$\nabla C_{c_1c_2c_3}$	-	-	+	+
$\nabla C_{l_1l_2l_3}$	-	-	-	+

Tabelle 3.2: *Taxonomie der Farbkanten basierend auf der Sensitivität verschiedener Farbmodelle. - bedeutet, dass der Farbraum invariant bzgl. dieses Merkmal ist, + bedeutet, dass der Farbraum sensitiv auf dieses Merkmal reagiert.*

### 3.2.6 Klassifikation der Kanten

Auf Grund der Taxonomie bzgl. Farbinvarianz und der Herleitung der Gradienten kann davon ausgegangen werden, dass die Taxonomie in Bild 3.2 die Farbkanten klassifizieren kann. Es kann unterschieden werden zwischen (1) Schatten- oder Geometriekanten, (2) Materialkanten und (3) Glanzlichtkanten.

Der Klassifikator aus [GS00], der in Listing 3.1 dargestellt wird, kann die gefundenen Kanten in die drei genannten Kategorien eingruppiert.

```

1 IF ( $\nabla C_{RGB}$ ) >  $t_1$  AND ( $\nabla C_{c_1c_2c_3}$ ) ≤  $t_2$ 
2 THEN klassifiziere als Schatten- oder Geometriekante
3 ELSE
4 IF ( $\nabla C_{c_1c_2c_3}$ ) >  $t_2$  AND ( $\nabla C_{l_1l_2l_3}$ ) ≤  $t_3$ 
5 THEN klassifiziere als Glanzlichtkante
6 ELSE
7 IF ( $\nabla C_{l_1l_2l_3}$ ) >  $t_3$ 
8 THEN klassifiziere als Materialkante

```

Listing 3.1: *Klassifikator für die Unterscheidung in Schatten- oder Geometriekanten, Glanzlichtkante oder Materialkanten, basierend auf den vorgestellten Farbgradienten.*

# Kapitel 4

## Experimente und Evaluation

Die vorgestellten Verfahren wurden in die Programmumgebung *PUMA*<sup>1</sup> implementiert, die an den Universitäten Erlangen und Koblenz entwickelt wurde und immer noch weiterentwickelt wird. PUMA stellt eine komplette Programmierumgebung in C++ für die Bildverarbeitung zur Verfügung. Auf die Details der Umsetzung wird speziell im Anhang dieser Arbeit ab Seite 39 eingegangen.

### 4.1 Versuchsaufbau

Zur Evaluation der beiden Glanzlichtdetektionsverfahren wurden unterschiedliche Bildsequenzen und Einzelbilder verwendet. Es wurden sowohl endoskopische Bilder aus Operationen, synthetische und reale Laboraufnahmen zur Evaluation verwendet. Bild 4.1 zeigt für jeden Typus ein Beispielbild. Alle Messungen und Berechnungen wurden auf einem Athlon XP 2600+ durchgeführt.

Für eine quantitative Auswertung der Experimente wurden einige Ergebnisse in Tabellen festgehalten. Die in den Tabellen ab Seite 51 festgehaltenen Ergebnisse basieren auf einem Vergleich der durch die Algorithmen ermittelten Masken und einer manuell er-

---

<sup>1</sup>Programmierungsumgebung für die Musteranalyse  
<http://www.uni-koblenz.de/~puma>

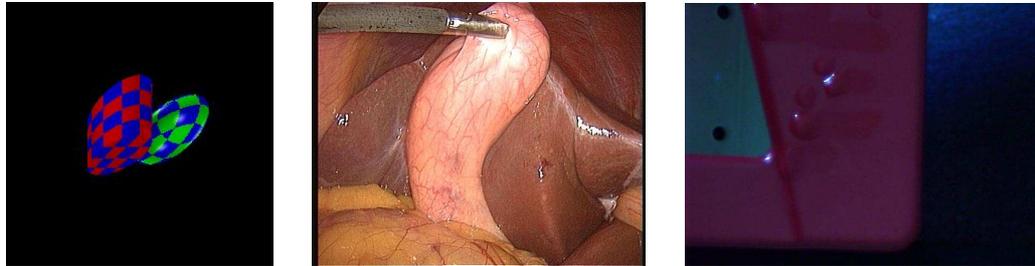


Bild 4.1: (a) Eine synthetisch gerenderte Bildsequenz, (b) eine reale Aufnahme aus einer Operation, (c) eine reale Aufnahme aus einem Laboraufbau

stellten Maske. Für jeden Versuchsszenario sind zwei Tabellen angelegt. Zur Bildung der ersten Tabelle (z.B. Tabelle 5.2) wurde dabei ein Vergleich auf Pixelebene durchgeführt. Für die zweite Tabelle (z.B. Tabelle 5.3) wurden dagegen die detektierten, zusammengefassten Regionen verglichen.

#### 4.1.1 Synthetische Bildsequenz

Die synthetisch generierte Bildsequenz bestand aus 1024 Einzelbildern und zeigt zwei dreidimensionale Objekte, die durch eine Punktlichtquelle beleuchtet werden. Die Oberfläche ist spiegelnd und so sind je nach Objektposition die Glanzlichter an unterschiedlichen Positionen oder nicht vorhanden. Die Bildsequenz wurde freundlicherweise von Florian Vogt<sup>2</sup> von der Universität Erlangen zur Verfügung gestellt.

Ergebnisse der Glanzlichtdetektion für beide Verfahren sind in Abbildung 4.2 zu sehen. Deutlich ist zu sehen, dass die Farbgradienten Methode mehr Pixel als Glanzlicht detektierte als die *HSV* Schwellwerte Methode. Beide Ergebnissbilder wurden zweimal dilatiert um geschlossene Regionen zu erhalten.

Die Ergebnisse des Vergleichs der beiden Verfahren sind in den Tabellen 5.2 und 5.3 zusammengefasst. Deutlich sind die Unterschiede in den falsch positiven und falsch negati-

---

<sup>2</sup><http://www5.informatik.uni-erlangen.de/~vogt/>

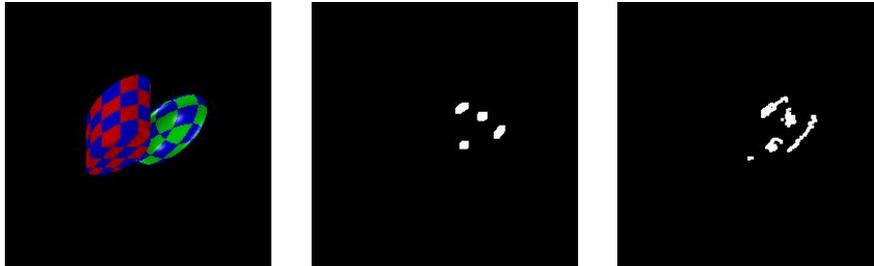


Bild 4.2: (a) Eine synthetisch gerenderte Bildsequenz, (b) detektierte Glanzlichter mit *HSV* Schwellwerten, (c) detektierte Glanzlichter mit Glanzlichtkanten

ven Werten zu erkennen. Das *HSV*-Verfahren zeichnet sich durch eine höhere Sensitivität und Spezifität auf Pixelebene aus. Bei einem Vergleich der Regionen ist das auf den Farbgradienten basierende Detektionsverfahren nur geringfügig schlechter.

#### 4.1.2 Reale Operationsaufnahmen

Die in Abbildung 4.3 dargestellten Ergebnisse wurden auf Bilder einer Laparoskopie angewendet. Eine Laparoskopie ist eine endoskopische minimalinvasive chirurgische Operation in der Bauchhöhle, meist z.B. an der Gallenblase. Eine solche Gallenblase, nebst Operationswerkzeug ist auch in der Abbildung zu sehen. Das feuchte menschliche Gewebe und das metallische Operationswerkzeug bewirken starke Spiegelungen, hervorgerufen durch die in der Einleitung schon beschriebenen sehr hellen Lichtquellen in medizinischen Endoskopen.

Die Eigenschaften menschlichen Gewebes entsprechen nicht dem dichromatischen Reflexionsmodell, da das Gewebe kein dielektrisches<sup>3</sup> inhomogenes Material ist.

Beide Verfahren detektieren unterschiedliche Glanzlichter, bzw. überbelichtete Stellen im Bild. Am offensichtlichsten sind die Unterschiede in der großen überbelichteten Stelle in der oberen Mitte des Originalbildes. Die Schwellwerte für dieses Verfahren sind Tabelle 4.2

<sup>3</sup>dielektrisch = nicht leitend, bzw. Isolator



Bild 4.3: (a) Eine reale Aufnahme eines medizinischen Endoskops bei der Operation einer Gallenblase (Laparoskopie), (b) detektierte Glanzlichter mit HSV Schwellwerten, (c) detektierte Glanzlichter mit Glanzlichtkanten

zu entnehmen. Ein Großteil der kleineren Glanzlichter mit scharfen Kontrastübergängen wurden von beiden Verfahren gut erkannt, doch sind bei den Farbgradienten eine größere Anzahl an Falschpixeln durch Rauschen im Bild zu beobachten. Diese lassen sich auch nicht durch angepasste Schwellwerte eliminieren. Beim HSV Schwellwert Verfahren wurde zweimal dilatiert, bei den Farbgradienten viermal.

Die Ergebnisse eines Vergleichs der beiden Verfahren sind in den Tabellen 5.4 und 5.5 zusammengefasst. Der Vergleich beruht auf einer manuellen Markierung der Glanzlichter und den algorithmisch bestimmten Markierungen. Auf Grund der erheblichen Anzahl an kleineren Glanzlichtregionen ist ein wirklicher Vergleich der einzelnen, detektierten Regionen nur schwer möglich. Es wurde sich dabei auf die größten zusammenhängenden Regionen mit mehr als 50 Glanzlichtpixeln beschränkt. Unterscheiden sich die beiden Verfahren durch eine erheblich voneinander abweichende Sensitivität auf Pixelebene, so ergibt ein Vergleich der als korrekt positiv klassifizierten Regionen, dass die beiden Verfahren die großen Glanzlichtregionen nahezu gleich zuverlässig detektieren.

### 4.1.3 Reale Laboraufnahmen

Die folgende Bildsequenz wurde im Labor der Arbeitsgruppe Aktives Sehen der Universität Koblenz aufgenommen. Es wurde eine Sony DFW\_X700 Digitalkamera benutzt, die

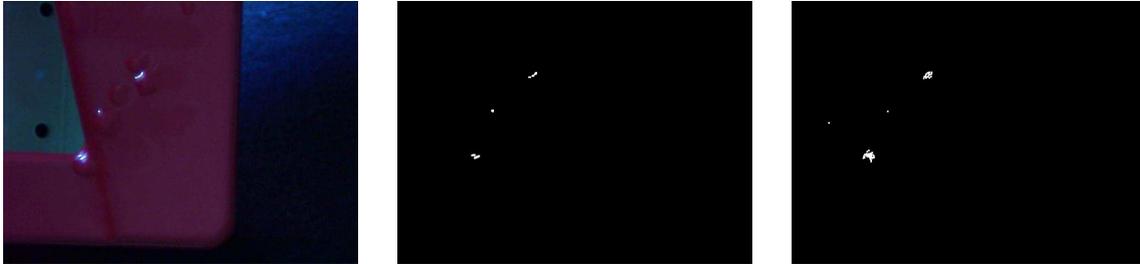


Bild 4.4: (a) Eine reale Aufnahme eines Tackers unter einem L-Arm, (b) detektierte Glanzlichter mit HSV-Schwellwerten, (c) detektierte Glanzlichter mit Glanzlichtkanten

an einem L-Arm montiert war. Als Aufnahmeobjekt wurde ein einfacher roter Tacker benutzt. Um Glanzlichter zu erzeugen, wurde der Tacker mit Wassertropfen beträufelt. Auf Bild 4.4 sind diese Glanzlichter sehr gut zu erkennen. Es wurde eine Sequenz mit 50 Bildern aufgenommen, bei der das Objekt und die Lichtquelle stillstanden und die Kamera sich um das Objekt bewegte.

Die Ergebnisse überraschten, da die Unterschiede zwischen HSV-Schwellwerten und Farbgradienten nur sehr marginal ausfielen. Bis auf einen einzigen Ausreisserpixel gibt es quasi keine Unterschiede zwischen den Ergebnissen. Allerdings sind die Bilder der Glanzlicht-Farbgradienten nicht auf dieser Abbildung zu sehen. Laut dem Klassifikator für Farbkanten von Gevers müssten die abgebildeten Kanten zu geometrisch hervorgerufenen Kanten gehören und nicht zu Glanzlichtern. Die Ergebnisse der Glanzlichtdetektion mit Hilfe der Farbgradienten ergab trotz unterschiedlichster Schwellwerte für alle beiden Parameter  $\lambda_t$  und  $c_t$  nur Rauschen und eine große Anzahl von Störpixeln.

Die quantitativen Ergebnisse sind in den Tabellen 5.6 und 5.7 zusammengefasst. Auch hier zeigt sich der geringe Unterschied in den Ergebnissen für die Sensitivität, als auch für die Spezifität. Die Ergebnisse für die detektierten Regionen sind sogar vollständig deckungsgleich.

## 4.2 Zusammenfassung der Ergebnisse

Die Detektion der Glanzlichter mittels *HSV* Schwellwerten war bei jedem Bildtyp der schnellste und auch sicherste Weg der Glanzlichtbestimmung. Wie in Tabelle 4.1 zu sehen, war jede Berechnung der *HSV* Schwellwerte ca. vier bis fünfmal schneller als die Farbgradientenbestimmung. Diese Ergebnisse sind nicht verwunderlich, da für die Anwendung der *HSV* Schwellwerte nur einmal über das gesamte Bild iteriert werden muss, bei der Berechnung der Farbgradienten hingegen muss mehr als viermal über alle Bildpunkte iteriert werden. Auch müssen für die Farbgradientenbestimmung mehr temporäre Bilder im Speicher angelegt, sowie mehr externe Funktionsaufrufe gestartet werden (z.B. Canny-Operator).

Vor dem Hintergrund der höheren Komplexität der Farbgradientenbestimmung sind die Ergebnisse enttäuschend. Die Qualität der Ergebnisse ist nicht besser, eher schlechter (es werden also mehr falsche Pixel als Glanzlichter erkannt) und dies bei einer weit längeren Berechnungszeit. Desweiteren ist die Bestimmung der Schwellwerte für die Farbgradientenbestimmung komplizierter als bei den *HSV* Schwellwerten. Im *HSV* Farbraum sind die Schwellwerte sehr intuitiv und leicht anschaulich. Bei den Farbgradienten hingegen muss mit den sehr abstrakten Farbräumen  $l_1l_2l_3$  und  $c_1c_2c_3$  gearbeitet werden.

Beide Verfahren haben den Nachteil, dass sie mindestens zwei von Hand gesetzte Schwellwerte benötigen, es also keine vollautomatischen Prozesse sind. Desweiteren müssen die Schwellwerte über "trial-and-error" Verfahren ermittelt werden. Wobei, wie schon erwähnt, die Schwellwertbestimmung im *HSV* Farbraum einfacher und intuitiver ist.

Beide Ansätze konnten auf allen drei Bildtypen die Glanzlichter erkennen, wobei allerdings bei den Farbgradienten die Fehlerquote höher lag als bei den *HSV* Schwellwerten. Überraschend waren die fälschlicherweise als Geometriekante klassifizierten Kanten bei den Aufnahmen des Tackers. Allerdings ist das Verfahren von Gevers universeller anwendbar, da es nicht nur Glanzlicht-, sondern auch Material-, Schatten- und Geometriekanten unterscheiden kann.

	<i>Synth. Bild</i> 256 x 256 Pixel	<i>Laboraufnahme</i> 640 x 480 Pixel	<i>OP-Aufnahme</i> 720 x 576 Pixel
<b>Glanzlichtkante</b>	0,23	1,36	1,79
<b>Alle Kanten</b>	0,24	1,42	1,85
<i>HSV</i> <b>Schwellwerte</b>	0,05	0,24	0,32

Tabelle 4.1: Messwerte für die Durchführung der einzelnen Detektionsverfahren für jeweils ein Bild. Alle Angaben in Sekunden.

	<i>Synth. Bild</i>	<i>Laboraufnahme</i>	<i>OP-Aufnahme</i>
<b>Gradienten Schwellwerte</b>	rgbt: - ct: - lt: -	rgbt: 20 ct: 80 lt: -	rgbt: 94 ct: 133 lt: 120
<i>HSV</i> <b>Schwellwerte</b>	st: 80 vt: 200	st: 20 vt: 200	st: 20 vt: 220

Tabelle 4.2: Verwendete Schwellwerte für die angewendeten Testbildsequenzen.



# Kapitel 5

## Fazit

### 5.1 Zusammenfassung

In dieser Arbeit wurde zwei unterschiedliche Verfahren zur Glanzlichtdetektion implementiert, Schwellwerte im *HSV* Farbraum und Farbgradienten auf Grundlage des dichromatischen Reflexionsmodells. Beide Verfahren wurden theoretisch begründet und an synthetischen und realen Aufnahmen getestet.

Der ursprüngliche Anwendungsfall für die Glanzlichtdetektion lag in der medizinischen Bildverarbeitung, genauer in der Anwendung von medizinischen Endoskopen. Hier stellen die Glanzlichter ein besonderes Problem dar, da sie die Behandlung des Arztes behindern und möglicherweise das Operations- und Behandlungsergebnis verschlechtern können. Um Glanzlichter aus Bildern zu eliminieren, werden Techniken wie die Glanzlichtsubstitution mit Hilfe von Lichtfeldern vorgestellt [VPN02], die allerdings eine robuste Detektion der Glanzlichter vor ihrer Substitution voraussetzen.

Mit Hinblick auf den medizinischen Anwendungsfall sind die *HSV*-Schwellwerte sicherlich die bessere Wahl, da sie schnell und robust berechnet werden können. Die Ausgangsbedingungen für die *HSV*-Schwellwerte sind im Körperinneren besser gegeben, da hier natürlicherweise keine weissen Stellen vorkommen, die fälschlicherweise als Glanzlichter

detektiert werden könnten. Die Farbgradienten nach dem Modell von Gevers leiden unter der Problematik, dass sie eine hohe Anzahl an Fehlerpixeln produzieren und desweiteren verhältnismäßig schwierig anzuwenden sind. Die schlechten Ergebnisse könnten darauf zurückgeführt werden, dass die Bedingungen für das dichromatische Reflexionsmodell, nämlich dielektrische inhomogene Materialien, nicht für menschliches Gewebe gelten. Aber auch die weiteren Testaufnahmen konnten keine besseren Ergebnisse erzielen, so daß als Folge dessen von einer Anwendung der Farbgradienten zur Glanzlichtdetektion bei medizinisch endoskopischen Aufnahmen abgeraten werden muss.

Die Bildverarbeitung im medizinischen, besonders im endoskopischen Bereich stellt allerdings noch ein großes Arbeitsfeld dar, das viel Spielraum für Verbesserungs- und Beschleunigungsverfahren bietet.

## 5.2 Ausblick

### 5.2.1 Hardwaregestützte Beschleunigung

Mit Hinblick auf die gravierenden Unterschiede in den Berechnungszeiten für die beiden Verfahren stellt sich die Frage, wie diese Verfahren beschleunigt werden können.

Moderne Grafikhardware hat einen Entwicklungsstand erreicht, an dem reine Leistungssteigerungen, häufig in traditionellen Massstäben wie Polygon Füllraten gemessen, immer weniger wichtig werden. Selbst "low-end" Grafikhardware aus dem Computerspielbereich erreichen inzwischen Geschwindigkeiten, die ausreichen um z.B. jeden einzelnen Pixel eines 640x480 Bildschirms mit einem transformierten, beleuchteten und texturierten Polygon mehr als 50 mal in einem  $\frac{1}{30}$  einer Sekunde zu überschreiben (vgl. [McC00] und [THO02]). Die Hardwarehersteller verlagern daher ihre Entwicklungen weg von festverdrahteten Pipelinefunktionen hin zu flexiblen und reprogrammierbaren Grafikpipelines. So entwickeln diese Chipsätze eine immer größere Funktionalität und Komplexität. So verfügte schon der NVIDIA GeForce3 Chip über mehr Transistoren als ein Pentium IV. Der Nachfolger GeForce4 konnte dann sogar mehr als 1,2 Billionen Operationen pro Se-

kunde berechnen. Die Entwicklung in diesem Bereich ist sogar so schnelllebig, dass von den Grafikkartenherstellern (in diesem Falle ATi und NVIDIA) das Mooresches Gesetz<sup>1</sup> schon seit Jahren übertroffen wird!

Bis heute werden medizinische Bildverarbeitungsalgorithmen häufig auf spezieller, meist sehr teurer Hardware berechnet. Diese Hardware ist für die Verarbeitungskette nötig, da z.B. während einer Operation die Bilder in Echtzeit<sup>2</sup> vorliegen müssen. Die CPU kann diesen Ansprüchen meist nicht entsprechen, da die Verfahren zu komplex sind. Wäre es möglich diese Algorithmen, statt mit speziell hergestellten Hardware, auf einer Standardgrafikkarte zu rechnen, würde dies Geld und Entwicklungsaufwand sparen.

Arbeiten von [Mac03] oder [MA03] haben gezeigt, dass Standardverfahren der Bildverarbeitung wie z.B. der NOMSIE Algorithmus oder die Fast Fourier Transformation auf handelsüblicher Grafikkarte berechnet werden können, teilweise sogar schon in Echtzeit. In Bild 5.1 sind Bildverarbeitungsoperatoren, die mit Hilfe der Fourier Transformation realisiert wurden, dargestellt.

Relativ einfache Algorithmen, wie das *HSV*-Schwellwert-Verfahren, lassen sich allerdings auch durch den Einsatz optimierter Bibliotheken (z.B. IPL<sup>3</sup>, IPP<sup>4</sup>, OpenCV) soweit beschleunigen, dass sie in Echtzeit durchführbar sind. In [VKNS] wird ein Echtzeitsystem für die Endoskopie des Bauchraums basierend auf diesen Bibliotheken vorgestellt.

---

<sup>1</sup>Gordon Moore, der Gründer des Chipherstellers Intel, bemerkte 1965, dass die Dichte der Transistoren auf einer integrierten Schaltung mit der Zeit exponentiell ansteigt. Ursprünglich wurde von Moore eine Verdopplung alle 12 Monate vorausgesagt. Diese Vorhersage musste später leicht auf alle 18 Monate korrigiert werden, welches aber nichts am exponentiellen Charakter änderte.

<sup>2</sup>Echtzeit bedeutet, dass das Ergebnis einer Berechnung innerhalb eines gewissen Zeitraumes garantiert vorliegt, das heißt bevor eine bestimmte Zeitschranke erreicht ist. Für die medizinische Anwendung sind hier mindestens 30 Bilder pro Sekunde gemeint.

<sup>3</sup>Intel Processing Library

<sup>4</sup>Intel Integrated Performance Primitives

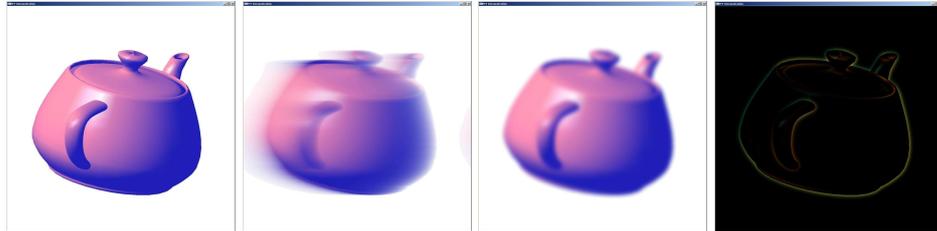


Bild 5.1: Beispiele für Bildverarbeitungsoperatoren auf Grundlage einer Diskreten Fourier Transformation auf der GPU: (a) Original Utah Teapot, (b) Teapot mit FFT Motion Blur, (c) Teapot mit FFT Lowpass Filter, (d) Teapot mit LaPlace Filter - Die Abbildungen wurden aus [McC00] entnommen.

### 5.2.2 Glanzlichtsubstitution durch Lichtfelder

Die Detektion, wie sie in dieser Arbeit beschrieben wird, ist nur einer der Schritte in der Verarbeitungskette zur Verbesserung von Endoskopieaufnahmen. Ein Verarbeitungsschritt, der auf die Detektion aufsetzt, ist die Glanzlichteliminierung, bzw die Glanzlichtsubstitution. Wie bereits in der Einleitung erwähnt, ist es bei medizinischen Aufnahmen dem Arzt nicht möglich, das unter den Glanzlichtern befindliche Gewebe eindeutig zu erkennen und ggf. krankhaftes Gewebe zu klassifizieren. Durch Vogt [VPN02] wurde eine neue Technik zur Glanzlichtkorrektur vorgeschlagen. Ein sogenanntes *Lichtfeld* (engl. light field) wird aus einer aufgenommenen Bildsequenz errechnet. Die binäre Glanzlicht-Maske aus dem Detektionsschritt wird für jedes einzelne Bild der Bildsequenz als *Vertrauenskarte* für die Lichtfeldpixel eingesetzt. Das Resultat ist ein Lichtfeld, in dem die Pixel an Glanzlichtposition mit den Pixeln aus Bildern der Sequenz interpoliert werden, die nicht überbelichtet sind (vgl. Bild 5.2).

Lichtfelder sind in den Bereichen Rechnersehen und Computergrafik ein relativ neues Verfahren. Sie beschreiben eine diskrete Version der *plenoptischen Funktion* (vgl. [LH96]), die es ermöglicht, neue, unbekannte Ansichten einer Szene zu generieren. Hierzu wird der hochdimensionale Raum der plenoptischen Funktion auf vier Dimensionen reduziert, indem nur konstante Radianz (drei Farbwerte) entlang von Lichtstrahlen gemessen werden.

Dieser Lichtstrahl wird definiert durch seine Schnittpunkte  $(s, t)$  und  $(u, v)$  zweier paralleler Ebenen (vgl. Bild 5.3).

Der Vorteil dieser Methode ist, dass die überbelichteten Pixel mit *realen* Werten ersetzt werden, die aus Bildern der Sequenz gewonnen wurden, die nicht an den entsprechenden Bildpositionen überbelichtet waren.



Bild 5.2: *Glatzlichtsubstitution mit Lichtfeldern: (1) Gerendertes Gallenblasenbild (ohne Vertrauenskarte), (2) Gerendertes Gallenblasenbild (mit Vertrauenskarte), (3) Differenzbild - Die Abbildungen wurden aus [VPN02] entnommen.*

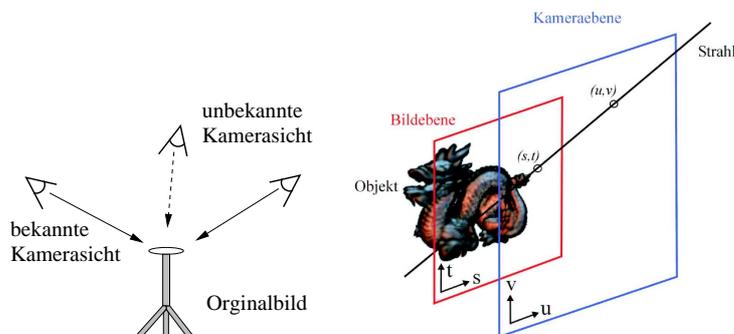


Bild 5.3: (a) *Schematische Darstellung der Generierung neuer Ansichten aus unbekannter Position, (b) Zwei-Ebenen Parametrisierung eines Lichtfelds. Jeder Strahl wird durch die Koordinaten, an denen er die Bild- und Kameraebene  $((s, t)$ - und  $(u, v)$ -Ebene) schneidet, beschrieben.*



# Anhang A - Implementation

## PUMA Module

PUMA ist in verschiedene einzelne Module aufgeteilt, die für verschiedene Nutzergruppen, wie Anwender, Entwickler, usw., verschiedene Schnittstellen bietet, um ihre Arbeit in PUMA einzubringen (vgl. Usecase Diagramm in Abbildung 5.4) Für die Implementation dieser Studienarbeit waren zwei dieser Module von Bedeutung: *Ipop* und *Animals*.

*Ipop* ist eine Zusammenstellung von Algorithmen und Verfahren, wie z.B. Medianfiltern, Kantendetektoren, etc.. Die Glanzlichtkantendetektion und das HSV-Schwellwertverfahren wurde hier in den beiden neuen *Ipop*-Modulen *Gevers* und *BinarizeSV* implementiert.

*Animals* sind lauffähige, eigenständige Programme, die die Funktionalitäten der anderen Module implementieren und anwenden. Das Parsen der Kommandozeile, Einlesen der Dateien, Anwenden der Algorithmen und schließlich das Schreiben der Ausgabebilder ist in diesem *Animal* implementiert. `HiLightDetAnimal`.

## **BinarizeSV.C und Gevers.C**

*Ipop* steht für *Image Processing Operations*, also für Bildverarbeitungsoperationen. Auch die beiden Erweiterungen, die in den Dateien `BinarizeSV.C` und `Gevers.C` enthalten sind, sind Operationen für die Bildverarbeitung.

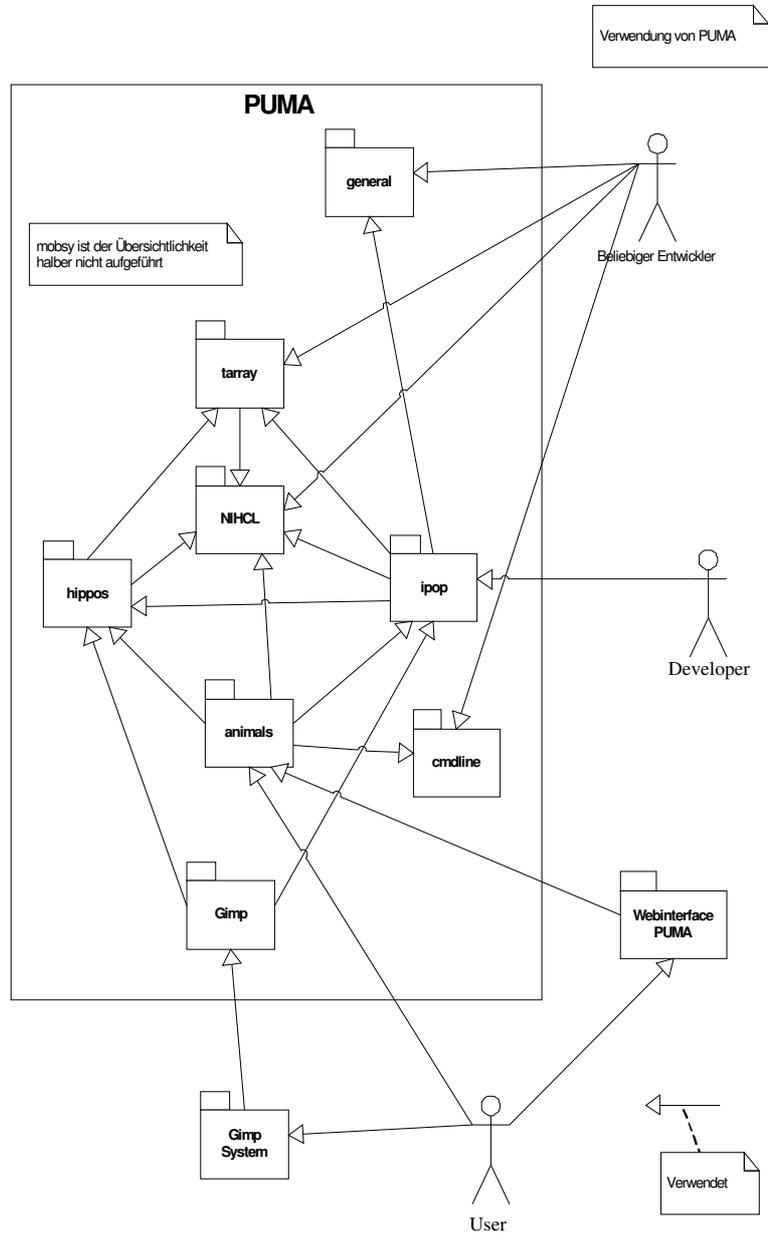


Bild 5.4: Verschiedene Module in PUMA für unterschiedliche Nutzergruppen. Die Abbildung wurde aus der PUMA Dokumentation [pum] entnommen

**BinarizeSV.C**

Ein Objekt der Klasse `BinarizeSV` kann durch den Standardkonstruktor mit festgelegten Default-Schwellwerten oder durch den Anwender selbstgewählte Schwellwerte angelegt werden. Diese Schwellwerte  $S$  und  $V$  beziehen sich auf die beiden Farbkomponenten des  $HSV$ -Farbraums, die für das Binarisierungsverfahren benötigt werden. In Formel 3.1 wird zwar auch der  $H$  Wert für das Schwellwertverfahren verwendet, da dieser aber jedesmal den vollen Wertebereich von  $H \in [0, 359]$  ausnutzt, ist eine Prüfung dieser Farbkomponente hinfällig. Das Ausgabebild dieser Operation ist ein Grauwertbild  $G$  mit den Werten 255 an den Stellen, an denen die Schwellwertbedingungen zutrafen und sonst 0. Ein einfaches Binärbild mit nur 0 und 1 als valide Bildwerte hätte ausgereicht, doch solch einen Bildtyp gibt es in der PUMA Umgebung nicht.

**Gevers.C**

Ebenso wie ein Objekt der Klasse `BinarizeSV` kann auch ein Objekt der Klasse `Gevers` mit festgelegten Default-Schwellwerten mittels des Standardkonstruktors oder mit selbstgewählten Schwellwerten erzeugt werden. Auch hier werden im Gegensatz zum Verfahren aus Bild 3.1 nur zwei Schwellwerte benötigt, da eine Klassifikation in die unterschiedlichen Kantentypen (Geometrie-, Schatten-, Material- und Glanzlichtkante) nicht notwendig ist. Eine Klassifikation als Glanzlichtkante ist ausreichend. Somit genügen zwei Schwellwerte, einen für die  $l_1l_2l_3$  und einen für die  $c_1c_2c_3$  Werte. Es wurde jedoch auch das komplette Gradientenklassifikationsverfahren implementiert, so dass beide Varianten zur Verfügung stehen.

In PUMA gibt es keinen eigenen Datentyp für `float`-Bilder, also Bilder, die ihre Werte als Fließkommazahlen halten. Es existieren nur `byte`-Bilder. Diese Bilder haben also nur einen ganzzahligen Wertebereich von  $[0, 255]$ . Da bei der Berechnung der Gradienten allerdings reelle Werte als Ergebnis vorliegen, mussten temporär `float`-Bilder angelegt werden. Intern repräsentiert PUMA Bilder als zweidimensionale Felder, auch als `Matrix` bezeichnet. Also wurden für die Gradienten zweidimensionale Fließkommafelder angelegt, um keine Rundungsfehler in die Berechnung einfließen zu lassen. Erst am Ende der Berechnung werden die Werte auf einen ganzzahligen Wertebereich von  $[0, 255]$  abgebil-

det, um sie als PUMA-Bildobjekte schreiben zu können. Der folgende Code erzeugt die Matrizen für die  $c_1c_2c_3$  Werte,  $l_1l_2l_3$  analog.

```
Matrix <float> c1mat(xsize,ysize);
Matrix <float> c2mat(xsize,ysize);
Matrix <float> c3mat(xsize,ysize);
```

Diese Matrizen werden mit den Werten für  $c_1c_2c_3$  und  $l_1l_2l_3$  gefüllt, die sich wie folgt berechnen:

```
static float l1 (float r, float g, float b){
    return (fabs(r-g)) / (fabs(r-g) + fabs(r-b) + fabs(g-b));
    // fabs(x) returns the absolute value of parameter x.
}
static float c1 (float r, float g, float b){
    if (max(g,b)==0.0)
        return 0.0;
    else
        return atan(r/max(g,b));
}
```

Anschließend wird ein Canny-Kantendetektor mit einer “non-maximum suppression” auf das  $\nabla C_{RGB}$  angewendet. Für die folgende Anwendung der Schwellwerte werden nur die Pixel des Bildes bearbeitet, die einen Bildwert  $\neq 0$  besitzen.

Letztendlich werden die beiden Schwellwerte angewendet:

```
// Used to map the pixel values to [0,255]
float mult_c = 255 / maxc;
float mult_l = 255 / maxl;

for (int i = 0; i < ysize; ++i){
    for (int j = 0; j < xsize; ++j){
```

```
if (temp[i][j] != 0){
    if (l1l2l3_matrix[i][j] * mult_l > thres_l
        && c1c2c3_matrix[i][j] * mult_c <= thres_c){
        out[i][j] = 255;
    }
}
}
```

Die Werte `mult_c` und `mult_l` werden genutzt, um die Pixelwerte auf den ganzzahligen Wertebereich `[0,255]` abzubilden.

## HiLightDetAnimal.C

Im Animal-Modul `HiLightDetAnimal` werden die Verfahren aus den Ipop-Modulen eingebunden und weiterverarbeitet. Desweiteren werden mehrere Hilfsfunktionen inkludiert. Durch Parameter können dem Programm auf der Kommandozeile mehrere Optionen mitgegeben werden. Eine Auflistung der möglichen Parameter befindet sich in Tabelle 5.2.2. Es ist sowohl möglich zwischen den unterschiedlichen Glanzlichtdetektionsverfahren zu wählen, als auch die verschiedenen Schwellwerte zu definieren.

Hauptaufgabe der Routine ist das Parsen der Eingabeparameter und das Einlesen der Dateinamen. Es ist notwendig mindestens zwei Dateinamen anzugeben, erstens mindestens ein Eingabebild und zweitens einen Dateinamen für das Ausgabebild. Alternativ kann auch mehr als ein Eingabebild angegeben werden. Angenommen es werden  $n$  Dateinamen angegeben, dann sind alle  $n - 1$  Bilder Eingabebilder und das  $n$ te Bild gibt den Prototypen für die Ausgabebilder an. Prototyp bedeutet, dass in diesem Falle dem Dateinamen eine Laufnummer angehängt wird um eine spätere Weiterverarbeitung durch ein Script zu erleichtern. Ein solches Script kann z.B. die Einzelbilder wieder zu einen Videostream zusammensetzen (vgl. Skript 5.3). Anstatt mehrere Dateinamen vom gleichen Namensaufbau manuell einzugeben, ist auch der Einsatz des `*` Operators auf der Kommandozeile erlaubt. Die Funktionalität der Kommandozeile ersetzt intern den Stern durch einen String aller

Option	Typ	Funktion	zulässige Werte und Default
-verbose	Bool	mehr Statusmeldungen	true oder false (default)
-hsv_threshold	Bool	HSV Schwellwert	true (default) oder false
-complete_gevers	Bool	alle Farbgradienten	true oder false (default)
-gevers	Bool	nur Glanzlichtkanten	true oder false (default)
-st	Integer	SAT Schwellwert	[0,255], 20 (default)
-vt	Integer	VAL Schwellwert	[0,255], 200 (default)
-rgbt	Integer	RGB Schwellwert	[0,255], 20 (default)
-ct	Integer	$c_1c_2c_3$ Schwellwert	[0,255], 20 (default)
-lt	Integer	$l_1l_2l_3$ Schwellwert	[0,255], 200 (default)
-d	Integer	Anzahl der Dilatationen	[0,255], 2 (default)

Tabelle 5.1: Parametertabelle zum Animal Modul HiLightDetAnimal

passenden Dateinamen. Ein typischer Aufruf des Programms könnte z.B. so aussehen:

```
$>HiLightDetAnimal -gevers input*.ppm output.pgm
```

Alternativ kann das Programm auch mit den Tool `xanimals` aus dem PUMA-Paket gestartet werden, so dass dem Benutzer eine grafische Benutzeroberfläche zur Verfügung steht.

```
$>xanimals HiLightDetAnimal
```

Angenommen es gibt 10 Dateien mit den Namen `input0` bis `input9`, dann sind die Ergebnissbilder `output0` bis `output9` benannt. Bei nur einem Eingabe- und Ausgabebild entfällt diese Regelung. Desweiteren können noch mehrere Dilatationschritte auf das Ergebnisbild angewendet werden. Per Defaultwert sind dies zwei Dilatationsdurchläufe.

## Füllalgorithmus für Kantenbilder

Anders als beim *HSV*-Schwellwert Verfahren entstehen bei den Glanzlichtkantenbildern keine zusammenhängenden Regionen. Gerade im Zusammenhang mit Kantenbildern, die

mit Hilfe des Canny-Algorithmus erzeugt wurden, wird häufig das Verfahren des *Constraint Thinning* eingesetzt.

Beim Constraint-Thinning-Verfahren werden Lücken in den Kanten geschlossen. Diese Lücken treten häufig als Fehler bei der Non-Maxima-Suppression-Berechnung auf. Dies geschieht besonders an den Stellen, an denen mehrere Kanten aneinander stoßen. Dabei wird auf das Gradientenbild der Canny-Operation zurückgegriffen, bei dem noch keine markierten Maxima gelöscht wurden. Diese Informationen können zur Vervollständigung der lückenhaften Kanten dienen (vgl. [Can86]).

Entsteht eine vollständig geschlossene Kante kann diese z.B. durch das *Boundary Filling*<sup>5</sup> Verfahren gefüllt werden. Listing 5.1 skizziert dieses rekursive Verfahren zum Füllen von Kantenbildern. Jeder Aufruf von `boundaryFill()` kann sich selbst noch viermal aufrufen. Die Logik hinter dieser Routine ist sehr simpel. Befinden wir uns nicht am Rand der Kante oder sollte der aktuelle Punkt schon gefüllt worden sein, so lassen wir unsere Nachbarpunkte sich selbst füllen. Dieses Verfahren benötigt nur einen Startpunkt innerhalb der geschlossenen Region, um diese vollständig zu füllen.

Dieses Verfahren wurde in dieser Arbeit allerdings nicht umgesetzt, da die meist sehr kleinen Kanten durch die Dilatation ohnehin zusammengefasst wurden. Ein weiteres Zusammenfassen der Kanten war also nicht notwendig, könnte aber ggf. in Fällen von großen Glanzlichtregionen notwendig sein. Desweiteren konnte die Voraussetzung für das Boundary-Filling nicht erreicht werden, also eine vollständig zusammenhängende Kante.

## Einlesen und Schreiben von Videosequenzen

Das direkte Einlesen von Videosequenzen ist mit PUMA leider bisher nicht möglich. Es musste also ein Weg gefunden werden, um solche Videosequenzen unkompliziert verarbeiten zu können. Das Open-Source Projekt *MPlayer*<sup>6</sup> stellt eine Programmsuite zur

---

<sup>5</sup>Dieses Verfahren kann als Java Applet betrachtet werden unter:

<http://www.cs.unc.edu/~mcmillan/comp136/Lecture8/areaFills.html>

<sup>6</sup><http://www.mplayerhq.hu>

```
1 void boundaryFill(int x, int y, int fill, int boundary)
2 {
3     if ((x < 0) || (x >= raster.width)) return;
4     if ((y < 0) || (y >= raster.height)) return;
5     int current = raster.getPixel(x, y);
6     if ((current != boundary) & (current != fill)) {
7         raster.setPixel(fill, x, y);
8         boundaryFill(x+1, y, fill, boundary);
9         boundaryFill(x, y+1, fill, boundary);
10        boundaryFill(x-1, y, fill, boundary);
11        boundaryFill(x, y-1, fill, boundary);
12    }
13 }
```

Listing 5.1: Füllalgorithmus für geschlossene Kantenregionen

Verfügung, die es ermöglicht aus einem Videostream die einzelnen Frames zu extrahieren und separat in beliebige Bildformate zu speichern. Das Script in Listing 5.2 erledigt genau diese Aufgabe. Das Script erlaubt es einen zweiten Parameter  $i$  zu übergeben, dieser ermöglicht nur die ersten  $i$  Frames des Videostreams auszulesen.

Das Rückschreiben der Einzelbilder in einen zusammenhängenden Videostream kann man entweder mittels des zum MPlayer gehörigen `mencoder` oder des Tools `ppmtompeg` ermöglichen. Die Scripte 5.3 und 5.4 werden benötigt um die Einzelbilder zusammenzufügen. Beide Scripte werden in das Verzeichnis mit den entsprechenden Einzelbildern kopiert. Danach ist ein einfacher Aufruf des Scriptes mit dem Dateinamen der Konfigurationsdatei als Parameter notwendig. Die Konvertierung von 1024 Einzelbildern ist so in 45 Sekunden möglich!

```
1 #!/bin/bash
2
3 # Script for extracting every single frame of a video stream
4 # stored in an AVI, MPEG or any other file format readable
5 # by MPlayer. The single frames are stored in the PPM file
6 # format.
7
8 i = $2
9
10 if [ -n "i" ]
11 then
12     echo 'Executing only for the first' i 'frames'
13     /usr/bin/mplayer -vo ppm -frames i $1
14 else
15     /usr/bin/mplayer -vo ppm $1
16 fi
17
18 exit
```

Listing 5.2: Skript zur Extraktion von Einzelbildern aus Videosequenzen

```
1 #!/bin/bash
2
3 # Script for encoding many single frames into a video stream.
4 # This stream is stored in the AVI, MPEG or any other file
5 # format writeable by MPlayer.
6
7 config_file = $1
8
9 if [ -n "i" ]
10     then
11         echo 'Please give the config file as a parameter!'
12     else
13         /usr/bin/ppmtompeg $config_file
14     fi
15
16 exit
```

Listing 5.3: Skript zum Zusammenfügen von Einzelbildern zu einer Videosequenzen

```
1 PATTERN I
2 # define the filename of the output movie
3 OUTPUT sequence.mpg
4 # define the file format of the input images
5 BASE_FILE_FORMAT PPM
6 INPUT_CONVERT *
7 GOP_SIZE 650
8 SLICES_PER_FRAME 1
9 INPUT_DIR .
10 # define the syntax for parser for the input images
11 INPUT
12 newframe*.ppm [0000-1023]
13 END_INPUT
14 # half/full quality measurement
15 PIXEL HALF
16 RANGE 10
17 PSEARCH_ALG LOGARITHMIC
18 BSEARCH_ALG CROSS2
19 IQSCALE 8
20 PQSCALE 10
21 BQSCALE 25
22 REFERENCE_FRAME ORIGINAL
23 # additional documentation: 'man ppmtompeg'
```

Listing 5.4: Beispielkonfiguration für ppmtompeg



## Anhang B - Tabellen

Pixel gesamt:	65.536	(256 * 256)
<b>HSV-Schwellwerte</b>	Glanzlichtpixel	keine Glanzlichtpixel
Glanzlichtpixel detektiert	340	21
kein Glanzlichtpixel detektiert	71	65.104
<b>Farbgradienten</b>	Glanzlichtpixel	keine Glanzlichtpixel
Glanzlichtpixel detektiert	241	421
kein Glanzlichtpixel detektiert	81	64.793

Tabelle 5.2: Übersicht über detektierte Glanzlichtpixel bei einer synthetisch generierten Szene

Glanzlichtregionen gesamt:	4	
<b>HSV-Schwellwerte</b>	Glanzlichtregionen	keine Glanzlichtregionen
Glanzlichtregion detektiert	4	0
keine Glanzlichtregion detektiert	0	1
<b>Farbgradienten</b>	Glanzlichtregion	keine Glanzlichtregionen
Glanzlichtregion detektiert	4	1
keine Glanzlichtregion detektiert	0	1

Tabelle 5.3: Übersicht über detektierte Glanzlichtregionen bei einer synthetisch generierten Szene

Pixel gesamt:	414.720	(720 * 576)
<b>HSV-Schwellwerte</b>	Glanzlichtpixel	keine Glanzlichtpixel
Glanzlichtpixel detektiert	10.170	4.386
kein Glanzlichtpixel detektiert	1.521	398.643
<b>Farbgradienten</b>	Glanzlichtpixel	keine Glanzlichtpixel
Glanzlichtpixel detektiert	2.718	2.613
kein Glanzlichtpixel detektiert	5.130	404.259

Tabelle 5.4: Übersicht über detektierte Glanzlichtpixel bei einer realen Operationsaufnahme

Glanzlichtregionen gesamt:	6	
<b>HSV-Schwellwerte</b>	Glanzlichtregionen	keine Glanzlichtregionen
Glanzlichtregion detektiert	6	2
keine Glanzlichtregion detektiert	0	1
<b>Farbgradienten</b>	Glanzlichtregion	keine Glanzlichtregionen
Glanzlichtregion detektiert	6	2
keine Glanzlichtregion detektiert	0	1

Tabelle 5.5: Übersicht über detektierte Glanzlichtregionen bei einer realen Operationsaufnahme

Pixel gesamt:	307.200	(640 * 480)
<b>HSV-Schwellwerte</b>	Glanzlichtpixel	keine Glanzlichtpixel
Glanzlichtpixel detektiert	200	3
kein Glanzlichtpixel detektiert	126	306.871
<b>Farbgradienten</b>	Glanzlichtpixel	keine Glanzlichtpixel
Glanzlichtpixel detektiert	313	10
kein Glanzlichtpixel detektiert	60	306.817

Tabelle 5.6: Übersicht über detektierte Glanzlichtpixel bei einer realen Laboraufnahme eines Tackers

Glanzlichtregionen gesamt:	3	
<b>HSV-Schwellwerte</b>	Glanzlichtregionen	keine Glanzlichtregionen
Glanzlichtregion detektiert	3	0
keine Glanzlichtregion detektiert	0	1
<b>Farbgradienten</b>	Glanzlichtregion	keine Glanzlichtregionen
Glanzlichtregion detektiert	3	0
keine Glanzlichtregion detektiert	0	1

Tabelle 5.7: Übersicht über detektierte Glanzlichtregionen bei einer realen Laboraufnahme eines Tackers



# Literaturverzeichnis

- [Can86] CANNY, JOHN: *A Computational Approach to Edge Detection*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Seiten 679–698, November 1986.
- [dZ86] ZENZO, SILVANO DI: *Gradient of a multi-image*. Computer Vision, Graphics and Image Processing, 33:116–125, 1986.
- [FVF95] FOLEY, JAMES D., ANDRIES VANDAM und STEVEN K. FEINER: *Computer Graphics: Principles and Practice*. Addison-Wesley Professional, 2. Auflage, August 1995.
- [GS99a] GEVERS, TH. und A.W.M. SMEULDERS: *Color Based Object Recognition*. Pattern Recognition, Seiten 453–464, 1999.
- [GS99b] GEVERS, THEO und HARRO STOKMAN: *Reflectance Based Edge Classification*. In: *Vision Interface*, Trois-Rivieres, 19-21 May 1999.
- [GS00] GEVERS, TH. und H.M.G. STOKMAN: *Classifying color transitions into shadow-geometry, illumination, highlight or material edges*. In: *Proceedings of the International Conference on Image Processing (ICIP)*, Seiten 521–524, Vancouver, BC, September 2000. IEEE Computer Society Press.
- [Haf98] HAFNER, WALTER: *Segmentierung von Video-Bildfolgen durch Adaptive Farbklassifikation*. Doktorarbeit, Institut für Informatik der Technischen Universität München, 1998.

- [Kli01] KLIMOWICZ, CÄSAR: *Lichtfelderzeugung aus endoskopischen Bildfolgen des Bauchraums*. Diplomarbeit, Friedrich-Alexander-Universität Erlangen Nürnberg, Februar 2001.
- [Kos97] KOSCHAN, A.: *Analyse von Glanzlichtern in Farbbildern*. In: PAULUS, D. und T. WAGNER (Herausgeber): *3. Workshop Farbbildverarbeitung*, Seiten 121–127, 95, Erlangen, 1997.
- [Kra02] KRAMME, RÜDIGER (Herausgeber): *Medizintechnik - Verfahren, Systeme, Informationsverarbeitung*. Springer Verlag, Berlin, 2002.
- [LH96] LEVOY, MARC und PAT HANRAHAN: *Light Field Rendering*. *Computer Graphics*, 30(Annual Conference Series):31–42, 1996.
- [MA03] MORELAND, K und E ANGEL: *The FFT on a GPU*. In: *SIGGRAPH/Eurographics Workshop on Graphics Hardware 2003 Proceedings*, Seiten 112–119, Juli 2003.
- [Mac03] MACK, MARTIN: *X-Ray Image Processing on a Standard Graphics Card*. Diplomarbeit, Universität Koblenz, 2003.
- [McC00] MCCOOL, M. D.: *SMASH: A next-generation API for programmable graphics accelerators*. Technischer Bericht, Computer Graphics Lab, University of Waterloo, 2000.
- [Pau01] PAULUS, DIETRICH: *Aktives Bildverstehen*. Der andere Verlag, Osnabrück, 2001. Habilitationsschrift in der Praktischen Informatik, Universität Erlangen-Nürnberg.
- [Pö01] PÖPPL, H (Herausgeber): *5. Workshop Bildverarbeitung für die Medizin*, Heidelberg, 2001. Springer Verlag.
- [pum] *PUMA - Application Programmers Handbook*.
- [RR98] REUTER, MATTHIAS A. und HANS J. REUTER: *Geschichte der Endoskopie*. Krämer, Stuttgart, 1998.

- [Sch03] SCHAER, PHILIPP: *Medizinische Bildverarbeitung - Hauptseminar: Endoskopie*. Technischer Bericht, Universität Koblenz, Juli 2003.
- [Sha85] SHAFER, S. A.: *Using color to separate reflection components*. *COLOR research and application*, 10(4):210–218, 1985.
- [SR96] SAPIRO, G. und D. L. RINGACH: *Anisotropic Diffusion of Multi-valued Images with Applications to Color Filtering*. *IEEE PAMI*, 11(5):1582, 1996.
- [Str00] STROUSTRUP, BJARNE: *Die C++ Programmiersprache*. Addison-Wesley Verlag, München, 4. Auflage, 2000.
- [THO02] THOMPSON, CHRIS J., SAHNGYUN HAHN und MARK OSKIN: *Using Modern Graphics Architectures for General-Purpose Computing: A Framework and Analysis*. In: *International Symposium on Microarchitecture (MICRO)*, Türkei, November 2002.
- [VKNS] VOGT, FLORIAN, SOPHIE KRÜGER, HEINRICH NIEMAND und CHRISTOPH SCHICK: *A System for Real-Time Endoscopic Image Enhancement*. In: *Medical Image Computing & Computer Assisted Intervention (MICCAI 2003)*, Seiten 356–363. Springer.
- [VKP<sup>+</sup>01] VOGT, FLORIAN, CAESAR KLIMOWICZ, DIETRICH PAULUS, W. HOHENBERGER, HEINRICH NIEMANN und CHRISTOF SCHICK: *Bildverarbeitung in der Endoskopie des Bauchraums*. In: PÖPPL, H [Pö01], Seiten 320–324.
- [VPN02] VOGT, FLORIAN, DIETRICH PAULUS und HEINRICH NIEMANN: *Highlight Substitution in Light Fields*. In: *International Conference on Image Processing (ICIP)*, Seiten 637–640, 2002.