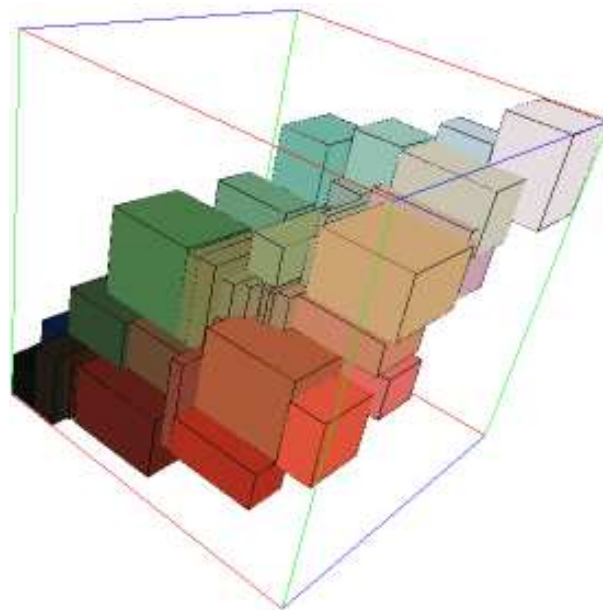


# 10. Workshop Farbbildverarbeitung

7.– 8. Oktober 2004  
Koblenz



**Detlev Droege**  
**Dietrich Paulus**  
(Hrsg.)

10. Workshop Farbbildverarbeitung 2004, Koblenz, Online-Proceedings  
<http://www.uni-koblenz.de/icv/fws2004/>

## Vorwort

Die Aufnahme, Verarbeitung und Analyse farbiger und mehrkanaliger Bilder gewinnt seit Jahren beständig an Bedeutung, wozu sowohl die verbesserten technischen Möglichkeiten als auch die gestiegenen Ansprüche aus den vielfältigen Anwendungsfeldern in der Industrie, der Medizin, der Umwelt und den Medien beitragen. Mit den technischen Möglichkeiten und mit den Wünschen der Anwender steigt auch der Forschungsbedarf in diesem Gebiet.

Im Jahr 1995 wurde erstmals der Workshop Farbbildverarbeitung in Koblenz veranstaltet, der in der Folge jährlich an verschiedenen Orten stattfand. Dieser Workshop bietet ein Diskussionsforum für Forscher, Anwender und Entwickler, das sich den Problemen der Farbbildaufnahme, den Farbräumen und der Farbmessung genauso widmet wie der Entwicklung von neuen Methoden und Algorithmen zur Analyse mehrkanaliger Bilder. Seit seiner Gründung bietet der Workshop ein Forum, in dem Hardware-nahe Untersuchungen ebenso wie Software-Entwicklungen, Grundlagen ebenso wie Anwendungsberichte ihren Platz finden. Damit ist die Möglichkeit zum Informationsaustausch zwischen denjenigen Gruppen gegeben, die sich intensiv mit dem Thema Farbe beschäftigen, auch wenn sie aus sonst getrennten Gebieten wie dem Rechnersehen oder der Beleuchtungstechnik kommen.

Der 10. Workshop findet im Jahr 2004 zum dritten Mal an seinem Gründungsort statt und wird an der Universität Koblenz-Landau im Institut für Computervisualistik in Zusammenarbeit mit der Firma TiTech Visionsort durchgeführt. Der Unterstützung durch die DAGM und durch die Gesellschaft für Informatik (GI, Fachgruppe 1.0.4) sei an dieser Stelle gedankt.

Die Themen des Workshops, wie sie im vorliegenden Tagungsband dokumentiert sind, umfassen alle Bereiche der Erfassung, Verarbeitung, Analyse und Wiedergabe von Farbbildern sowie der Nutzung der industriellen Farbbildverarbeitung zur Qualitätskontrolle, Inspektion, Robotik und Automatisierung.

Koblenz, im September 2004

Detlev Droege

Dietrich Paulus

---

Das Titelbild zeigt eine Visualisierung des Median-Cut Algorithmus zur Farbquantisierung nach P. Heckbert (1982).

## **Organisation**

Der 10. Workshops Farbbildverarbeitung wird von dem Institut für Computervisualistik der Universität Koblenz–Landau, Campus Koblenz, ausgerichtet in Zusammenarbeit mit der Firma TiTech Visionsort GmbH, Andernach, der Gesellschaft für Informatik, Fachgruppe Bildverstehen, Bonn, und der Deutsche Arbeitsgemeinschaft für Mustererkennung (DAGM e.V.).

### **Tagungsleitung**

Tagungsvorsitz: Prof. Dr.-Ing. Dietrich Paulus (Universität Koblenz–Landau)  
Prof. Dr. Lutz Priebe (Universität Koblenz–Landau)  
Dr. Volker Rehrmann (TiTech Visionsort)

### **Programmkomitee**

PD Dr. Karl-Heinz Franke (TU Ilmenau)  
Prof. Dr.-Ing. Bernhard Hill (RWTH Aachen)  
Prof. Dr. Andreas Koschan (University of Tennessee)  
Prof. Dr.-Ing. Dietrich Paulus (Universität Koblenz-Landau)  
Prof. Dr. Lutz Priebe (Universität Koblenz-Landau)  
Dr. Volker Rehrmann (TiTech Visionsort, Andernach)  
Dr. Werner Ritter (Daimler Chrysler, Ulm)  
Prof. Dr. Wolfgang Slowak (Fachhochschule Koblenz)  
Prof. Dr. Gerd Stanke (GFaI, Berlin)

### **Veranstalter**

Institut für Computervisualistik, Universität Koblenz-Landau, Koblenz  
TiTech Visionsort GmbH, Andernach  
Gesellschaft für Informatik, Fachgruppe Bildverstehen, Bonn  
Deutsche Arbeitsgemeinschaft für Mustererkennung e.V.

### **Sponsor**

TiTech Visionsort GmbH, Andernach

## Inhaltsverzeichnis

<b>Die Metamer-Randdeskriptor-Methode zur Farbkorrektur</b> . . . . .	1
<i>Philipp Urban, Rolf-Rainer Grigat</i>	
<b>Realisierung einer farbmtrischen Anbindung an valenzmetrische Schnittstellen ...</b>	10
<i>Rico Nestler (ZBS e.V.), Karl-Heinz Franke (ZBS e.V.), Rainer Jahn (ZBS e.V.)</i>	
<b>Spektral modellierbare Lichtquelle zur Erzeugung beliebiger Spektren durch ...</b>	18
<i>Markus Schnitzlein, Bernhard Frei, Andreas Willert</i>	
<b>Farbbasierte Objekterkennung mit einem omnidirektionalen System</b> . . . . .	27
<i>A. Maas, P. Heim, J. Kaluza, S. Mitnacht, V. Hong, P. Maillard, F. Occelli, C. Kurucz, D. Paulus</i>	
<b>Nonparametric Noise Removal in Color Images</b> . . . . .	35
<i>Bogdan Smolka</i>	
<b>Bestimmungshilfe für Mollusken und Eulenfalter mittels digitaler Bildanalyse</b> . .	43
<i>Fabian Fritzer</i>	
<b>An Efficient Vector Median Filter Computation</b> . . . . .	50
<i>V. Hong, L. Csink, S. Bouattour, D. Paulus</i>	
<b>On Color Image Quantization by the K-Means Algorithm</b> . . . . .	58
<i>Henryk Palus (Silesian University of Technology)</i>	
<b>Symmetrie als kognitives Bildmerkmal</b> . . . . .	66
<i>Kai Hübner</i>	
<b>Shading-Korrektur für Endoskopische Bilder und Fundusbilder</b> . . . . .	74
<i>Annika Hirsch, Christian Münzenmayer, Dietrich Paulus</i>	
<b>Farbkalibrierung mittels linearer Transformation</b> . . . . .	83
<i>J. Michel, V. Hong, D. Paulus, C. Münzenmayer</i>	
<b>Einsatz von diskreten Farbsensoren zur Druckbildkontrolle</b> . . . . .	91
<i>Frank Krumbein, Gunter Sieß, Winfried Mahler</i>	
<b>Robust Color Image Retrieval for the WWW</b> . . . . .	96
<i>Bogdan Smolka</i>	
<b>Assessment of RGB Encoding for Color Imaging</b> . . . . .	103
<i>Úlfar Steingrímsson, Peter Zolliker (EMPA St.Gallen)</i>	
<b>Ein Entwurfswerkzeug für Farbklassifikatoren in Echtzeitanwendungen</b> . . . . .	111
<i>Klaus Arbter und Daniel Kish</i>	

<b>Haustoria Segmentation in Microscope Colour Images of Barley Cells</b> .....	119
<i>Alexander Ihlow, Udo Seiffert</i>	
<b>Farbanalyse von gebackenen oder frittierten Nahrungsmitteln ...</b> .....	127
<i>Walter Hillen, Felix Fischer (Fachhochschule Aachen)</i>	
<b>Farbauswertung in der Mikroskopie - wo liegen die Herausforderungen ...</b> .....	135
<i>Peter Schwarzmann</i>	

## Die Metamer-Randdeskriptor-Methode zur Farbkorrektur

Philipp Urban<sup>1</sup> und Rolf-Rainer Grigat<sup>2</sup>

<sup>1</sup> Ratio Entwicklungen GmbH

<sup>2</sup> Technische Universität Hamburg-Harburg, Vision Systems

### 1 Einleitung

Farbkorrektur beschreibt die Transformation zwischen den Sensorantworten eines Bildaufnahmesystems (z.B.: RGB) und dem CIEXYZ- bzw. CIEL\*a\*b\* Farbraum. In jedem metameren Farbproduktionssystem ist dies die erste Farbtransformation nach der Bildaufnahme.

Es existieren zwei unterschiedliche Klassen von Methoden zur Farbkorrektur für lineare Bildaufnahmesysteme: Target- und modellbasierte Methoden. Targetbasierte Methoden benutzen eine Menge an Trainings- CIEXYZ / CIEL\*a\*b\* Farben mit korrespondierenden Sensorantworten, um eine Approximation der Farbkorrekturabbildung zu bestimmen. Die am weitesten verbreitete Methode ist die lineare Transformation zwischen den Sensorantworten und dem CIEXYZ Farbraum. Falls die CIEXYZ Farben linear abhängig sind von den Sensorantworten ist diese Transformation optimal. In dieser Arbeit werden lineare Bildaufnahmesysteme mit  $q \geq 3$  Kanälen und Sensorantworten aus der Menge  $C := [0, 1]^q$  betrachtet. Die lineare Transformation wird durch eine  $3 \times q$  Matrix  $A$  beschrieben, die mittels linearer Regression der Targetfarben bestimmt werden kann. Die Farbkorrekturabbildung lässt sich in diesem Fall wie folgt schreiben:

$$A \cdot c = (X, Y, Z)^T \rightarrow (L^*, a^*, b^*)^T, \quad (1)$$

wobei  $c \in C$  eine beliebige Sensorantwort ist.

Da die Aufnahme- und Beobachtungslichtart i.A. differieren und gängige Bildaufnahmesysteme die Luther-Bedingung (d.h. die lineare Abhängigkeit der CIE-Normspektralwertkurven von den Kanalempfindlichkeiten, vgl. [9][6]) nicht erfüllen, ist die Farbkorrekturabbildung i.A. nicht linear und die Methode resultiert in großen  $\Delta E$  Fehlern. Eine Verbesserung dieser Methode ist die Benutzung einer polynomialen Abbildung der Ordnung  $p > 1$

$$\begin{bmatrix} P_X(c) \\ P_Y(c) \\ P_Z(c) \end{bmatrix} = (X, Y, Z)^T \rightarrow (L^*, a^*, b^*)^T, \quad (2)$$

wobei

$$P_x(c = (c_1, \dots, c_q)^T) = \sum_{i_1 + \dots + i_q \leq p} a_{(i_1, \dots, i_q)}^x \prod_{j=1}^q c_j^{i_j}, \quad (3)$$

für  $x = X, Y, Z$ . Die Koeffizienten  $a_{(i_1, \dots, i_q)}^x$  lassen sich mittels mehrdimensionaler polynomialer Regression (**MPR**) der Targetfarben bestimmen.

In den beschriebenen Methoden wurde die Regression in den intensitätslinearen CIE $X$ - $Y$ - $Z$  Farbraum durchgeführt. Dies minimiert lediglich den „Root Mean Square“ Fehler zwischen den transformierten Farben und den korrespondierenden Targetfarben und nicht dem Wahrnehmungsfehler. Da die Farbabstandsmetriken jedoch im CIE $L^*a^*b^*$  Farbraum definiert sind, ließe sich eine Verbesserung, im Sinne kleinerer  $\Delta E$  Werte erwarten, wenn die Regression direkt in den CIE $L^*a^*b^*$  Farbraum durchgeführt werden würde.

Hardeberg [3] transformierte  $R, G, B$  Sensorantworten zuerst mit der nichtlinearen Funktion  $g(x) = x^{1/3}$ , um eine annähernd lineare Beziehung zwischen den  $g(R), g(G), g(B)$  Werten und den CIE $L^*a^*b^*$  Farben zu erzielen. Dies führt zu kleineren  $\Delta E$  Fehlern der anschließenden mehrdimensionalen polynomialen Regression. Der Grund für die Wahl von  $g(x) = x^{1/3}$  liegt in der Abbildungsvorschrift zwischen dem CIE $X$ - $Y$ - $Z$  und dem CIE $L^*a^*b^*$  Farbraum. Die Methode wird nachfolgend mit **MPRLabH** (MPR im CIE $L^*a^*b^*$  nach Hardeberg) abgekürzt. Eine Verallgemeinerung auf Systeme mit  $q$  Kanälen kann durch die kanalweise Transformation der Sensorantworten mittels  $g(x)$  durchgeführt werden.

$$\begin{bmatrix} P_{L^*}(g(c_1), \dots, g(c_q)) \\ P_{a^*}(g(c_1), \dots, g(c_q)) \\ P_{b^*}(g(c_1), \dots, g(c_q)) \end{bmatrix} = (L^*, a^*, b^*)^T, \quad (4)$$

wobei  $P_x, x = L^*, a^*, b^*$ , analog zu Gleichung (3) definiert ist.

König [5] transformierte die Sensorantworten wie in Gleichung (1) und führte anschließend eine mehrdimensionale polynomialen Regression auf die in den CIE $L^*a^*b^*$  Farbraum konvertierten Targetfarben durch.

$$A \cdot c = (X, Y, Z)^T \rightarrow (\tilde{L}^*, \tilde{a}^*, \tilde{b}^*)^T \rightarrow \begin{bmatrix} P_{L^*}(\tilde{L}^*, \tilde{a}^*, \tilde{b}^*) \\ P_{a^*}(\tilde{L}^*, \tilde{a}^*, \tilde{b}^*) \\ P_{b^*}(\tilde{L}^*, \tilde{a}^*, \tilde{b}^*) \end{bmatrix} = (L^*, a^*, b^*)^T, \quad (5)$$

wobei  $P_x, x = L^*, a^*, b^*$ , analog zu Gleichung (3) definiert ist. Die Methode wird nachfolgend mit **MPRLabK** (MPR im CIE $L^*a^*b^*$  nach König) abgekürzt.

Wenn rauschbehaftete Systeme betrachtet werden, resultieren Regressionsmethoden in relativ großen Farbfehlern. König [5] propagierte in diesem Fall eine neue matrixbasierte Methode (vgl. Gleichung (1)). Die Koeffizienten der Matrix werden jedoch mittels linearer Programmierung bestimmt. Das Ziel besteht darin, Matrixeinträge zu berechnen, die ungefähr die gleiche Größenordnung besitzen und trotzdem zu kleinen Fehlerraten führen. Die resultierende Matrix verstärkt durch ihre Struktur Rauschen viel weniger als die regressionsbasierten Matrizen. Die Methode wird nachfolgend mit **RM** (Rauschstabile Matrizierung) abgekürzt.

Die andere Klasse von Methoden zur Farbkorrektur beinhaltet die modellbasierten Methoden. Diese Methoden benutzen das physikalische Modell des Bildaufnahmesystems (vgl. Gleichung (6)), um die CIE $L^*a^*b^*$  Farben aus den Sensorantworten zu rekonstruieren. Wie bereits angemerkt, unterscheidet sich i.A. die Beobachtunglichtart von der



Aufnahmelichtart und das Bildaufnahmesystem erfüllt die Luther-Bedingung nicht. Eine eindeutige Rekonstruktion der CIEL\*a\*b\* Farben aus den Sensorantworten ist damit i.A. nicht möglich. Es existiert eine ganze Menge an Farben mit unterschiedlichen Reflexionsspektren, die zwar zur gleichen Sensorantwort führen (Gerätemetamerie), die jedoch unterschiedliche Farbeindrücke bei einem Beobachter unter der Beobachtungslichtart erzeugen.

Finlayson und Morovic [2] versuchten diesen sog. metameren Unterraum, der durch eine Sensorantwort aufgespannt wird, im CIEXYZ Farbraum zu charakterisieren. Hierzu nutzten sie lineare Programmierung, um den kleinsten Kubus zu finden, der den metameren Unterraum einschließt. Als Kandidaten für die Farbkorrektur wählten Sie den Schwerpunkt des Kubus. Die Methode wird nachfolgend mit LPCC (Linear Programming Centre of Cube) abgekürzt.

Ein Nachteil dieser Methode besteht darin, dass die Charakterisierung im intensitätslinearen CIEXYZ Farbraum durchgeführt wird und nicht im CIEL\*a\*b\* Farbraum, in dem die Farbmetriken definiert sind. Außerdem ist die Charakterisierung des metameren Unterraumes durch einen einschließenden Kubus sehr grob.

Im folgenden Text wird eine Methode vorgestellt, die die Charakterisierung im quasi wahrnehmungsgleichabständigen CIEL\*a\*b\* Farbraum mittels einer metameren Randdeskriptor-Matrix durchführt. Der Schwerpunkt der Matrixelemente wird als eine Approximation der optimalen Farbkorrektur im Sinne des kleinsten mittleren Fehlers gewählt (Dies ist die Farbe, die zu allen anderen möglichen Farben im metameren Unterraum den kleinsten  $\Delta E$  Abstand besitzt).

## 2 Der metamere Unterraum einer Sensorantwort

Im Folgenden wird ein lineares Bildaufnahmesystem mit der folgenden Beziehung zwischen Aufnahmelichtart  $L_a$ , den Kanalempfindlichkeiten  $s_1, \dots, s_q$ , dem Reflexionsspektrum der Vorlage und der Sensorantwort  $c = (c_1, \dots, c_q)^T \in C$  betrachtet, mit additivem Rauschen  $\epsilon$  und den Empfindungswellenlängenbereich  $\Lambda = [400\text{nm}, 700\text{nm}]$ :

$$c_i = \int_{\Lambda} s_i(\lambda) L_a(\lambda) r(\lambda) d\lambda + \epsilon, \quad i = 1, \dots, q \quad (6)$$

Die korrespondierende diskrete Formulierung, durch Abtastung der Spektren an  $N$  äquidistanten Wellenlängen ist gegeben durch

$$c = \Omega_a r + \epsilon \quad (7)$$

mit  $r = (r(\lambda_1), \dots, r(\lambda_N))^T$  und der Aufnahme-Lichtmatrix

$$\Omega_a = \begin{bmatrix} s_1(\lambda_1) L_a(\lambda_1) & \cdots & s_1(\lambda_N) L_a(\lambda_N) \\ \vdots & & \vdots \\ s_q(\lambda_1) L_a(\lambda_1) & \cdots & s_q(\lambda_N) L_a(\lambda_N) \end{bmatrix} \quad (8)$$

Im folgenden Text sind die Sensorantwort  $c \in C$ , die Aufnahme-Lichtmatrix  $L_a$  und die Kanalempfindlichkeiten  $s_1, \dots, s_q$  bekannt. In der Praxis lassen sich die Kanalempfindlichkeiten durch Messung oder durch Nutzung einer Menge an Trainingsfarben rekonstruieren. Methoden hierzu sind beispielsweise die Pseudoinverse, die Wiener-Inverse

[8], Eigenvektormethode [3], Projektion auf konvexe Mengen [10], lineare Programmierung [5] oder quadratische Programmierung [1].

Der Farbeindruck  $v_r \in \text{CIEXYZ}$ , der durch ein Reflexionsspektrum  $r$  unter der Beobachtungslichtart  $L_v$  beim Beobachter entsteht, wird mathematisch analog wie beim Bildaufnahmesystem erzeugt. Hier soll gleich der diskrete Fall behandelt werden:

$$v_r = \Omega_v r \quad (9)$$

mit  $\Omega_v$  als Beobachter-Lichtmatrix, die von den CIE Normspektralwertkurven  $\bar{x}, \bar{y}, \bar{z}$  abhängt

$$\Omega_v = \begin{bmatrix} \bar{x}(\lambda_1)L_v(\lambda_1) \cdots \bar{x}(\lambda_N)L_v(\lambda_N) \\ \bar{y}(\lambda_1)L_v(\lambda_1) \cdots \bar{y}(\lambda_N)L_v(\lambda_N) \\ \bar{z}(\lambda_1)L_v(\lambda_1) \cdots \bar{z}(\lambda_N)L_v(\lambda_N) \end{bmatrix} \quad (10)$$

Eine allgemeine Auflösung von Gleichung (7) nach dem Reflexionsspektrum  $r$  lautet

$$r = \Omega_a^\ominus (c - \epsilon) + \text{Kern}(\Omega_a) \quad (11)$$

wobei  $\Omega_a^\ominus$  die Pseudoinverse von  $\Omega_a$  ist und  $\text{Kern}(\Omega_a) := \{w \mid \Omega_a w = 0\}$ . Wenn eine Basis  $w_1, \dots, w_K$  von  $\text{Kern}(\Omega_a)$  zu einer Matrix  $W := (w_1, \dots, w_K)$  zusammengefasst wird, lässt sich (11) wie folgt umschreiben

$$r = \Omega_a^\ominus (c - \epsilon) + Wu, \quad \forall u \in \mathbb{R}^K \quad (12)$$

Die algebraische Lösung muss durch physikalische Restriktionen, wie Positivität, Beschränktheit und Glattheit, von natürlichen Oberflächenspektren beschränkt werden. Damit sind nicht alle  $u \in \mathbb{R}^K$  physikalisch sinnvoll. All die  $u \in \mathbb{R}^K$  in Gleichung (12), die die Bedingungen erfüllen, sollen zur Menge  $U \subset \mathbb{R}^K$  zusammengefasst werden. Falls (12) in (9) eingesetzt wird, lässt sich der metamere Unterraum  $M_{XYZ}^c$  aller möglichen Farbeindrücke, die aus der Sensorantwort rekonstruiert werden können, angeben

$$M_{XYZ}^c = \{\Omega_v \Omega_a^\ominus (c - \epsilon) + \Omega_v Wu \mid u \in U\} \quad (13)$$

Da die Farbmatriken im CIEL\*a\*b\* Farbraum definiert sind, müssen die Elemente von  $M_{XYZ}^c$  in den CIEL\*a\*b\* transformiert werden. Im Folgenden soll die Farbraumtransformation von CIEXYZ nach CIEL\*a\*b\* mit  $\mathcal{L} : \text{CIEXYZ} \mapsto \text{CIEL*a*b*}$  bezeichnet werden. Der metamere Unterraum im CIEL\*a\*b\* Farbraum wird wie folgt definiert

$$M_{Lab}^c = \mathcal{L}(M_{XYZ}^c) \quad (14)$$

### 3 Der metamere Rand-Deskriptor

Der in dieser Arbeit eingeführte metamere Randdeskriptor (**Metamer Boundary Descriptor**, **MBD**, vgl. Urban, Grigat [12]) ist eine  $p \times q$  Matrix, die in jedem Eintrag einen Randpunkt des metameren Unterraumes  $M_{Lab}^c$  abspeichert. Jede Zeile enthält  $q$  Umrisspunkte der Menge für einen festen  $L^*$ -Wert. Die Randpunkte werden im CIEXYZ-Farbraum mit Hilfe der linearen Optimierungsaufgabe 1 berechnet, die den metameren Unterraum entlang der Halbgeraden  $g + \lambda h, \lambda \geq 0$  abtastet (vgl. Abbildung 1a).

**Aufgabe 1** Sei  $\Omega_a \in \mathbb{M}(n \times N)$  die Aufnahme-Lichtmatrix für die Aufnahmelichtart  $L_a$  (vgl. Gleichung (8)) und  $\Omega_v \in \mathbb{M}(3 \times N)$  die Beobachter-Lichtmatrix für die Beobachtungslichtart  $L_v$  definiert (vgl. Gleichung (10)). Ferner sei  $c + \epsilon$  eine Sensorantwort mit additivem Rauschen  $\epsilon$ . Finde

$$- \lambda = \min \tag{15}$$

unter den linearen Nebenbedingungen (Variablenerklärungen nachfolgend)

$$r \geq 0 \tag{16}$$

$$r \leq 1 \tag{17}$$

$$Hr \leq \rho \tag{18}$$

$$-Hr \leq \rho \tag{19}$$

$$\Omega_a r = c + \epsilon \tag{20}$$

$$\Omega_v r = v_r \tag{21}$$

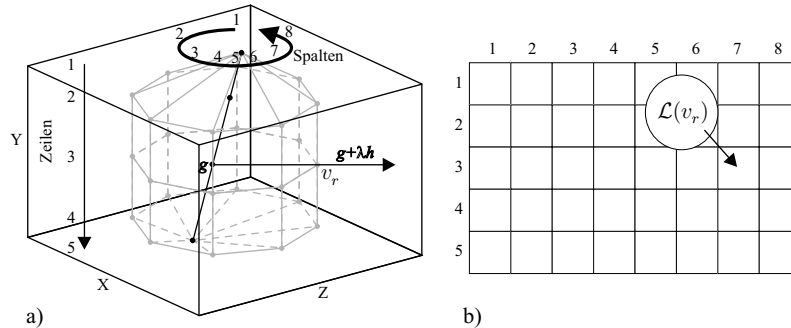
$$g + \lambda h = v_r \tag{22}$$

$$\lambda \geq 0 \tag{23}$$

Nebenbedingung (16) sichert die Positivität der Reflexionsspektren, Nebenbedingung (17) garantiert ihre Beschränktheit, da von nicht-fluoreszierenden Oberflächen ausgegangen wird. Zusätzlich wird die Glattheitsbedingung in (18) und (19) benutzt. Hierbei ist  $\rho > 0$  ein frei wählbarer Glattheitsfaktor und  $H \in \mathbb{M}(N - 2 \times N)$  eine Konvolutionsmatrix, die die Aufgabe hat, die zweite Ableitung auf das Reflexionsspektrum  $r$  anzuwenden

$$H = \begin{bmatrix} 1 & -2 & 1 & 0 \\ & \ddots & \ddots & \ddots \\ 0 & & 1 & -2 & 1 \end{bmatrix} \tag{24}$$

Durch die Nebenbedingung (20) wird sichergestellt, dass nur Reflexionsspektren  $r$  berücksichtigt werden, die zu der gegebenen Sensorantwort  $c + \epsilon$  führen. Die Nebenbedingung (21) führt die CIE Normspektralwertkurven und die Beobachtungslichtart über die Matrix  $\Omega_v$  ein. Hierbei ist die Farbwahrnehmung  $v_r$  eine Hilfsvariable, die für die nächste Nebenbedingung (22) benötigt wird. Diese Bedingung zusammen mit der Zielfunktion (15) und Bedingung (23) ermöglichen ein Abtasten des metameren Unterraumrandes entlang der bereits erwähnten Halbgerade. Für jede lineare Optimierungsaufgabe müssen die Parameter  $g$  und  $h$ , die die Halbgerade definieren, so gewählt werden, dass die Gesamtheit der Punkte im MBD den Rand von  $M_{Lab}^c \subset CIELab$  gleichmäßig überdecken. Wie dies erreicht werden kann, wird in unserer Publikation [12] erläutert. Als Lösung der Aufgabe 1 wird ein physikalisch sinnvolles Reflexionsspektrum  $r$  zurückgegeben sowie die zugehörige Farbwahrnehmung  $v_r$  unter der Beleuchtung  $L_v$ . Diese Farbwahrnehmung entspricht dem Schnittpunkt der Halbgeraden mit dem Rand von  $M_{XYZ}^c$  und ist eindeutig, da alle in Aufgabe 1 vorkommenden Nebenbedingungen (16) - (23) eine konvexe Menge bilden, die bei entsprechend groß gewähltem Glattheitsfaktor  $\rho > 0$  nicht leer ist. Dieser Schnittpunkt  $v_r$  wird anschließend



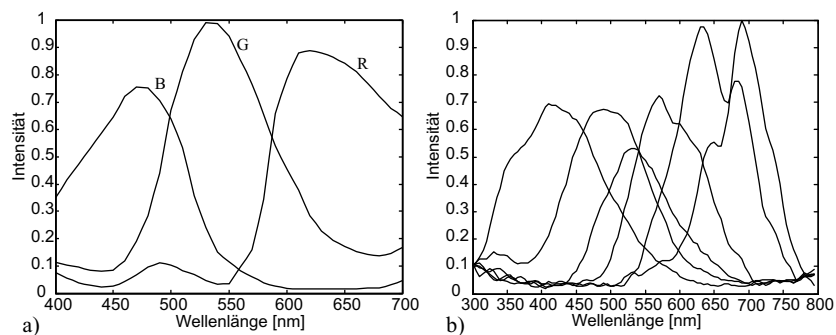
**Abb. 1.** a) Abtastung des metameren Unterraumes  $M_{XYZ}^c$  entlang der Halbgeraden  $g + \lambda h$ ,  $\lambda \geq 0$ . b) Die korrespondierende MBD Matrix besitzt 8 Spalten und 5 Zeilen mit den in den *CIE Lab*-Farbraum transformierten Abtastpunkten.

in den *CIE Lab*-Farbraum transformiert und in der MBD Matrix abgespeichert. Bei verrauschten Systemen lässt sich das unbekannte additive Rauschen  $\epsilon$  in der Nebenbedingung (20) als weitere Variable definieren, die durch zusätzlichen Nebenbedingungen eingeschränkt und additiv in der Zielfunktion gewichtet minimiert wird. Damit wird sichergestellt, dass auch im Rauschfall die durch die Nebenbedingungen definierte Menge nicht leer ist und damit eine Lösung garantiert.

Ist die MBD Matrix bestimmt, so lässt sich ein guter Kandidat für die optimale Farbkorrektur hinsichtlich des kleinsten mittleren Fehlers durch die Wahl des Schwerpunktes der Matrixelement bestimmen.

## 4 Resultate und Diskussion

Die (MBD) Methode wurde mit den anderen Methoden, die in der Einleitung beschrieben worden sind, mit Hilfe von Simulationsexperimenten verglichen. Hierfür wurde ein Sony Sensor [11] und ein experimenteller 6 Kanalsensor aus Jülich [5] [4] mit den Empfindlichkeiten aus Abbildung 2 benutzt.



**Abb. 2.** Kanalempfindlichkeiten des a) Sony Sensors und b) 6 Kanalsensors aus Jülich

Als Aufnahmeleuchtart wurde CIE-**F11** gewählt, die der Charakteristik einer Leuchtstoffröhre entspricht. Als Beobachtungslichtarten wurden die folgenden CIE-Lichtarten gewählt: **A, C, E, D50, D65, F2, F11**. Als Testspektren wurden verschiedene Datenbanken mit Reflexionsspektren genommen. Vrhel Datenbank [13]: 64 Spektren von MUNSSELL-Chips, DUPONT: 120 Spektren von Dupont-Farb-Chips und OBJECTS: 170 Spektren natürlicher und künstlicher Oberflächen sowie ein spektral vermessenes IT8.7/2 Target [7] mit 288 Farbfeldern.

Für alle targetbasierten Methoden wurden 288 Farben eines weiteren IT8.7/2 Targets als Trainingsmenge genommen. Für die regressionsbasierten Methoden wurden Polynome der Ordnung 3 gewählt. Für den Sony Sensor mussten mit dieser Wahl 20 Koeffizienten und für den 6 Kanalsensor 84 Koeffizienten pro Polynom bestimmt werden. Die Basis der Reflexionsspektren, die für die **LPCC** Methode benötigt wird, bestand aus 8 Spektren, die 99.9% der Energie der Trainingsmenge enthielten.

Die Anzahl an Spalten in der MBD Matrix wurde auf  $m = 8$  gesetzt. Die Anzahl an Zeilen ist variabel und hängt von der Luminanzspanne von  $M_{Lab}^c$  ab. Mit  $L_{max}^*$  und  $L_{min}^*$  dem größten und kleinsten  $L^*$ -Wert von  $M_{Lab}^c$  wird die Zeilenanzahl wie folgt gewählt

$$n = \text{Round} \left[ 2 \cdot (L_{max}^* - L_{min}^*) + \frac{1}{2} \right]. \quad (25)$$

Durch die Analyse verschiedener Datenbanken von Reflexionsspektren wurde der Glattheitsparameter auf  $\rho = 0.0035$  gesetzt. Dieser Wert hängt von der Abtastung der Spektren, d.h. von  $N$  ab. In unserer Untersuchung wurde  $N = 61$  gewählt, dies entspricht einer Abtastung in 5 nm Schritten.

In einer zweiten Simulation wurde additives, binäres, gleichverteiltes Rauschen auf die Sensorantwort gelegt. Die Rauschamplitude entsprach 1% der maximalen Sensorantwort.

Die Ergebnisse lassen sich den Tabellen 1 und 2 entnehmen.

	Rauschfrei			1% Rauschen		
	$E(\Delta E_{94}^*)$	$\text{Std}(\Delta E_{94}^*)$	$\text{Max}(\Delta E_{94}^*)$	$E(\Delta E_{94}^*)$	$\text{Std}(\Delta E_{94}^*)$	$\text{Max}(\Delta E_{94}^*)$
<b>MBD</b>	1.93	1.19	10.18	4.91	0.93	38.07
<b>MPRXYZ</b>	2.69	1.42	10.06	5.39	0.61	40.42
<b>MPRLabH</b>	2.71	1.47	9.89	5.34	1.35	51.20
<b>MPRLabK</b>	2.74	1.53	10.28	5.38	0.90	47.87
<b>RM</b>	4.26	2.60	17.42	5.97	2.65	40.94
<b>LPCC</b>	2.39	1.72	22.34	5.04	2.19	42.88

**Tabelle 1.** Ergebnisse für den Sony ICX434DQN Sensor. Werte beziehen sich auf alle Lichtarten und Testspektren.

Die Simulationsergebnisse für den Sony RGB Sensor zeigen, dass die **MBD** Methode, sowohl im rauschfreien als auch im verrauschten System, die anderen untersuchten Methoden an mittlerer Genauigkeit übertrifft. Die **MBD** Methode wird gefolgt von der **LPCC** Methode, deren mittlerer Fehler im rauschfreien Fall nahezu  $\Delta E_{94}^* = 0.5$  größer

	Rauschfrei			1% Rauschen		
	$E(\Delta E_{94}^{**})$	$Std(\Delta E_{94}^{**})$	$Max(\Delta E_{94}^{**})$	$E(\Delta E_{94}^{**})$	$Std(\Delta E_{94}^{**})$	$Max(\Delta E_{94}^{**})$
<b>MBD</b>	1.10	0.93	22.27	5.98	5.11	35.19
<b>MPRXYZ</b>	0.75	0.61	11.06	45.97	32.11	167.98
<b>MPRLabH</b>	1.31	1.35	35.56	36.53	34.91	240.73
<b>MPRLabK</b>	0.93	0.90	27.83	14.72	12.46	112.02
<b>RM</b>	3.87	2.65	18.35	6.31	3.99	36.94
<b>LPCC</b>	2.20	2.19	14.95	17.18	12.69	86.60

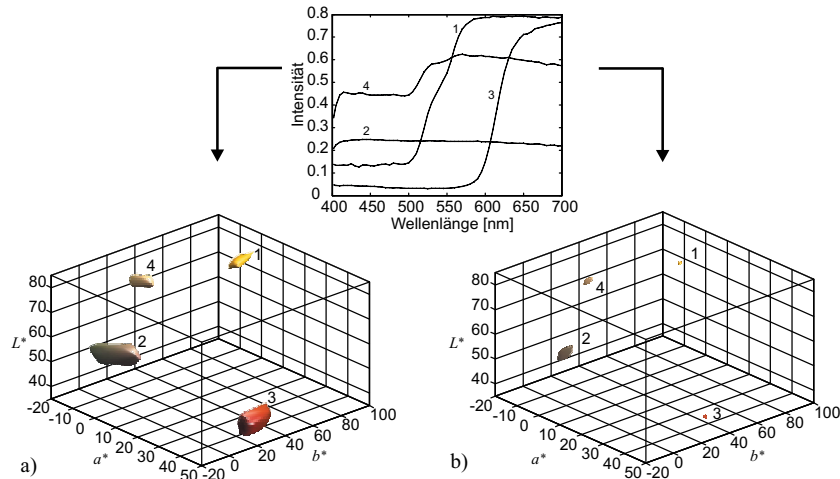
**Tabelle 2.** Ergebnisse für den 6 Kanalsensor. Werte beziehen sich auf alle Lichtarten und Testspektren. Die MPR Methoden sind extrem rauschempfindlich und in diesem Fall für das verrauschte System nicht anwendbar.

ist. In dieser Simulation erzielen die **MPR** Methoden, deren Polynomordnung jeweils 3 entspricht, annähernd gleiche Ergebnisse. Es zeigt sich kein Vorteil durch Nutzung des CIEL\*a\*b\* Farbraumes zur Regression. Die **RM** Methode resultiert in den größten mittleren Fehlern für beide untersuchten Rauschlevel.

Für den 6 Kanalsensor aus Jülich bietet sich ein ganz anderes Bild. In der rauschfreien Simulation erzielen die **MPR** Methoden extrem kleine mittlere Fehler, die im Falle der **MPRXYZ** und **MPRLabK** Methode sogar etwas unter den der **MBD** Methode liegen. Dieses Verhalten ändert sich jedoch unter dem Einfluß von Rauschen. Die Terme höherer Ordnung der verwendeten Polynome verstärken die Rauschfehler extrem. Von einem Einsatz dieser Methoden ist deshalb bei verrauschten mehrkanaligen Bildaufnahmesystemen abzuraten. Die **LPCC** Methode liegt bei den Ergebnissen für das rauschfreie System im unteren Mittelfeld der untersuchten Methoden. Im Rauschfall führt diese Methode zu unakzeptabel großen Fehlern. Ursache hierfür ist das Fehlen jeglicher Rauschbetrachtung in den linearen Optimierungsaufgaben. Die durch die Nebenbedingungen definierte Menge kann dadurch in einigen Fällen leer sein, was dazu führt, dass die Methode keine vernünftigen Ergebnisse mehr liefern kann. Die **RM** Methode ist im rauschfreien Fall, ähnlich wie beim Sony RGB Sensor, die schlechteste der analysierten Methoden. Ihre Stärke offenbart diese Methode erst in der Rauschsimulation. Die Fehler sind deutlich kleiner als bei den **MPR** Methoden und nur unmerklich größer als bei der **MBD** Methode.

## 5 Zusammenfassung

Eine neue modellbasierte Methode zur Farbkorrektur wurde vorgestellt und mit anderen Methoden durch Simulationsexperimente verglichen. Die neue Methode charakterisiert den metameren Unterraum der Sensorantwort im CIEL\*a\*b\* Farbraum durch die Berechnung einer metameren Randdeskriptormatrix. Durch die Berechnung des Schwerpunktes lässt sich eine gute Approximation der optimalen Farbkorrektur hinsichtlich des kleinsten mittleren Fehlers bestimmen. Zusammenfassend lässt sich aus den Simulationsergebnissen die interessante Schlußfolgerung ziehen, dass die **MBD** Methode sowohl in nahezu rauschfreien als auch in verrauschten Systemen zu den genauesten Methoden zählt. Andere Verfahren erzielen lediglich in einem dieser Fälle gute Ergebnisse.



**Abb. 3.** Beispiele einiger Munsell-Spektren mit zugehörigen metameren Unterräumen, unter der Beobachtungsbeleuchtung CIE-F2 und Aufnahmebeleuchtung CIE-F11, berechnet für a) den Sony ICX434DQN Sensor [11] und b) den 6 Kanalsensor aus Jülich [5] [4].

## Literatur

1. G. D. Finlayson, S. Hordley, and P. M. Hubel. Recovering device sensitivities with quadratic programming. In *IS&T/SID*, pages 90–95, Scottsdale Ariz., 1998.
2. G. D. Finlayson and P. M. Morovic. Metamer constrained color correction. *Journal of Imaging Science and Technology*, 44:295–300, 2000.
3. J. Y. Hardeberg. *Acquisition and reproduction of colour images: colorimetric and multispectral approaches*. PhD thesis, Ecole Nationale Supérieure des Télécommunications, France, 1999.
4. P. G. Herzog, D. Knipp, H. Stiebig, and F. König. Colorimetric characterization of novel multiplechannel sensors for imaging and metrology. *Journal of Electronic Imaging*, 8:342–353, 1999.
5. F. König. *Die Charakterisierung von Farbsensoren*. PhD thesis, Rheinisch-Westfälische Technische Hochschule Aachen, Deutschland, 2001.
6. F. König and P. G. Herzog. On the limitations of metameric imaging. *PICS: Image Processing, Image Quality, Image Capture, Systems Conference*, pages 163–168, 1999.
7. NPES. *Graphic technology - color reflection target for input scanner calibration*, volume IT8.7/2-1993. Reston, Virginia, 1999.
8. W. K. Pratt. *Digital Image Processing*, volume 2. John Wiley & Sons, inc., 1991.
9. G. Sharma. *Digital Color Imaging Handbook*. CRC PRESS, USA, 1 edition, 2003.
10. G. Sharma and H. J. Trussell. Characterization of scanner sensitivity. In *IS&T/SID*, pages 103–107, Scottsdale Ariz., 1993.
11. Sony. Icx434dqn, diagonal 5.68mm (type 1/3.2) frame readout ccd image sensor with square pixel for color cameras. *Datasheet*.
12. P. Urban and R.-R. Grigat. Color correction by calculating a metamer boundary descriptor. In *CGIV*, pages 536–541, Aachen, Germany, 2004.
13. M.J. Vrhel, R. Gershon, and L.S. Iwan. Measurement and analysis of object reflectance spectra. *Color Research and Applications*, 19:4–9, 1994.

## Realisierung einer farbmtrischen Anbindung an valenzmetrische Schnittstellen mittels targetbezogener, lokal linearer Abbildung

R. Nestler, K.-H. Franke, R. Jahn

Zentrum für Bild- und Signalverarbeitung e.V.,  
Gustav-Kirchhoff-Str. 5, D-98693 Ilmenau  
eMail: [info@zbs-ilmenau.de](mailto:info@zbs-ilmenau.de),  
Web: [www.zbs-ilmenau.de](http://www.zbs-ilmenau.de)

**Zusammenfassung.** Der Gegenstand dieses Beitrages ist ein flexibles Verfahren, dass die Anbindung eines gerätebezogenen Sensorfarbraumes an eine valenzmetrische Schnittstelle durch lokal lineare Farbraumabbildung vornimmt. Die Grundlage hierfür stellt eine korrespondierende geometrische Zerlegung des Ist- und Soll-Farbraumes in sogenannte 3-Simplexe (Tetraeder) dar, die anhand bekannter Stützstellen (Farbtargets) realisiert wird. Die zwischen den resultierenden Simplexpaarungen bestehende Abbildungsvorschrift ist linear und, da interpolierend, bezüglich der Stützstellen fehlerfrei. Dieser „geometrische“ Kalibrieransatz ermöglicht es, die Vorzüge der linearen Abbildung im Lokalen mit einer sehr hohen Approximationsgüte im gesamten triangulierten Farbraumausschnitt selbst bei eindeutig nichtlinearer Beziehung zwischen Ist- und Soll-Farbraum zu verbinden. Anhand eines Applikationsbeispiels, welches die Realisierung eines standardisierten Farbraumes unbedingt voraussetzt, werden die Eigenschaften dieses Ansatzes dargestellt und Aspekte des praktischen Einsatzes, wie Auswirkungen der irregulären Stützstellenverteilung, Kompressionsstrategien und die zeiteffiziente Realisierung in der Kannphase, diskutiert.

### 1 Einleitung

Beim Einsatz kommerzieller Farbkameras für die Farbmessung nach dem Dreibereichsprinzip besteht das Problem, dass die primären Sensordaten im Allgemeinen keiner valenzmetrischen Schnittstelle<sup>1</sup> entsprechen bzw. die Sensorempfindlichkeiten die notwendigen Lutherbedingungen nicht erfüllen. Für reale Messsysteme ist zusätzlich immer davon auszugehen, dass bei der Signalentstehung und im Aufnahme Kanal auch nichtlineare Übertragungskennlinien wirksam sind, die eine farbreizabhängige, nichtlineare Beeinflussung der primären Farbwerte und somit eine lokale Verzerrung des Sensorfarbraumes nach sich ziehen können. Aber selbst bei speziellen Dreibereichsensoren mit sehr gut an die gewünschten Spektralwertkurven angepassten Empfindlichkeitsfunktionen und weitestgehend linearem Übertragungskanal werden die applikationsbedingt oft sehr hohen Genauigkeitsansprüche aufgrund von Fertigungstoleranzen der optischen Komponenten des Messsystems nicht in jedem Fall erreicht. Hierbei

<sup>1</sup> z.B. dem XYZ-Normvalenzsystem (CIE 1931) bzw. einer davon abgeleiteten Farbvalenzbeschreibung



weisen die toleranzbedingten Farbortabweichungen im primären Farbraum einen ebenfalls farbreiz- und damit von der Applikationsklasse abhängigen, sehr lokalen Charakter auf.

Die Farbmessung erfordert neben der Realisierung rückführbarer Messbedingungen das Herstellen eines exemplarbezogenen Bezugs zu einem standardisierten Farbraum (Kalibrierung). Die dazu notwendige Transformationsvorschrift wird, da ein vollständig analytisches Modell meist nicht zur Verfügung steht, für einen gewählten Modellansatz auf der Grundlage einer endlichen Zahl bekannter Testfarben (Targets) bestimmt.

## 2 Überblick zu targetbezogenen Kalibrieransätzen

Die Verfahren zur targetbezogenen Kalibrierung lassen sich nach verschiedenen Gesichtspunkten systematisieren. Einerseits ist eine Unterteilung anhand des mathematischen Charakters des gewählten Modellansatzes in **lineare** und **nicht lineare** sowie in **approximierende** oder, einschränkend, **interpolierende** Verfahren möglich, andererseits kann eine Gliederung nach dem Farbraum, der zur Bewertung des resultierenden Abbildungsfehlers verwendet wird, vorgenommen werden<sup>2</sup>. Im Zusammenhang mit der Berücksichtigung auch lokaler Deformationen der zugrunde liegenden Farbräume sind die Verfahren, deren Ergebnis eine für den gesamten Farbraumbereich gültige Transformationsvorschrift ist, **global**, oder im anderen Fall **lokal**.

Für die Eingangs skizzierten Systeme führt das Anwenden einer globalen Abbildungsvorschrift zu einem systematischen Restfehler selbst für die zur Kalibrierung verwendeten und somit bekannten Farbtargets, der durch die Abweichung der realen Verhältnisse vom gewählten Modellansatz begründet ist. Ohne differenzierte Wichtung der verwendeten Targets verringern lokale Deformationen der Farbortverteilung die Approximationsgüte auch im übrigen Farbraum erheblich. Nicht lineare Ansätze weisen zwar eine verbesserte Reproduktion in der Umgebung der Stützstellen im Farbraum auf, neigen jedoch im Gegensatz zu den linearen Verfahren mit zunehmender Komplexität<sup>3</sup> zu einem im Vorfeld nicht bestimmten, instabilen Verhalten bei der Abbildung unbekannter Farbvalenzen. Insgesamt bleibt die gewünschte Genauigkeit durch den Einsatz eines globalen Korrekturansatzes oft auf sehr enge Farbraumabschnitte und damit klar definierte Applikationsfälle begrenzt.

Das Prinzip der lokalen Korrektur ist es, den durch die Kalibriertargets bestimmten Bereich des Ausgangs- oder Sensorfarbraumes durch das Anwenden von Transformationsvorschriften, die nur in regional begrenzten Raumabschnitten gültig und möglichst beim Übergang zwischen den Abschnitten stetig sind, in den Soll-Farbraum zu überführen. Diesem Ansatz liegt die Vorstellung zugrunde, dass sich jede stetige Funktion durch stückweise (lineare) Elemente approximieren lässt. Damit lassen sich lokale Eigenschaften der Farbräume ohne Rückwirkung auf den übrigen Farbraumbereich berücksichtigen und gleichzeitig eine hohe Approximationsgüte im gesamten betrachteten Farbraumabschnitt erzielen.

<sup>2</sup> Der mittlere quadratische Abbildungsfehler in diesem Farbraum (zumeist der Soll-Farbraum oder aber ein Farbraum mit dem Farbpfeifen angepasster Metrik) stellt das Optimierungskriterium für Gestaltung der Transformationsvorschrift dar.

<sup>3</sup> Koeffizientenanzahl

Diese Verfahrensgruppe wird bereits seit vielen Jahren zur Charakterisierung von Ausgabemedien, wie z.B. Druckern, verwendet, um die korrekte Reproduktion einer gegebenen Farbvalenz in einem zur Farbkommunikation verwendeten geräteunabhängigen Standardraum, z.B. CIE-Lab, im physikalischen Farbraum des Ausgabegerätes, z.B. CMYK, zu realisieren [3]. Die theoretischen Zusammenhänge zur Herstellung eines bestimmten Farbeindrucks<sup>4</sup> sind aufgrund ihrer starken Nichtlinearität und Komplexität nicht vollständig analytisch modellierbar. Die Beschreibung erfolgt daher durch die gleichmäßige Belegung des Druckergammuts mit einer großen Zahl von Testfarben<sup>5</sup>, der Raumzerlegung und der Anwendung raumabschnittsbezogener Abbildungsvorschriften. Dieses Vorgehen hat den Vorteil, dass neben der gewünschten Ist-Sollwert-Transformation auch eine Inverse unmittelbar zur Verfügung steht.

Die Methoden dieser Verfahrensgruppe unterscheiden sich in der Art der Raumzerlegung, der Methodik zur Extraktion eines zu einer Ist-Farbvalenz gehörenden Raumelementes sowie im Ansatz zur Ermittlung der lokalen Ist-Soll-Transformationsbeziehung. Ein aus Sicht der Flexibilität, des Beschreibungsaufwandes, der algorithmischen Komplexität und erzielbaren Genauigkeit geeigneter Kompromiss stellt hierbei die tetraedrische Zerlegung und Interpolation dar. Es besteht daher das Interesse zu klären, ob und unter welchen Bedingungen dieser Ansatz zur Kalibrierung von Dreibereichsfarbmesssystemen geeignet ist. Hierbei ist zu berücksichtigen, dass zur Charakterisierung in diesem Fall zumeist nur eine begrenzte Anzahl kommerziell und/oder applikativ verfügbarer Testfarben mit irregulärer und spärlicher Farbraumverteilung zur Verfügung stehen.

### 3 Tetraedrische Raumzerlegung und Interpolation

Die tetraedrische Partitionierung eines durch die Testfarben belegten Farbraumbereichs erfolgt durch dreidimensionale Delauney-Triangulation der Testfarborte im gewählten Bezugsfarbraum (Ist-Farbraum). Das Triangulationsergebnis wird geometrisch korrespondierend im jeweils zugehörigen Farbraum (Soll-Farbraum) aufgebaut. Die dreidimensionale Triangulation (simplicialer Komplex) besteht aus Tetraedern, dem sogenannten 3-Simplex (einfachster konvexer Baustein) des dreidimensionalen Raumes und bildet im Bezugsraum einen geschlossenen, konvexen Körper, der aus Sicht der Regeln der Delauneytriangulation optimal ist. Dies folgt nicht notwendig für die Gestalt im Soll-Farbraum, wogegen die simplicialen Raumabschnitte ihren geometrischen Charakter unabhängig vom tatsächlichen Zusammenhang zwischen Ist- und Soll-Farbraum beibehalten (lokale Ähnlichkeit). Im Vergleich dazu würden andere zunächst „einfache“ Raumprimitive des Bezugsfarbraumes, z.B. Quader bei kubischer Zerlegung, zu nicht vorher bestimmten, u.U. konkaven Körpern entarten und damit einen erheblich größeren Aufwand zur Beschreibung und Extraktion nach sich ziehen. Die Interpolationsvorschrift eines Vektors  $P$  innerhalb eines Tetraeders (siehe Abb. 1), das durch die

<sup>4</sup> die Wechselwirkung zwischen Farbstoffen und Medien sowie technologische Einflüsse

<sup>5</sup> einige Tausend [1]

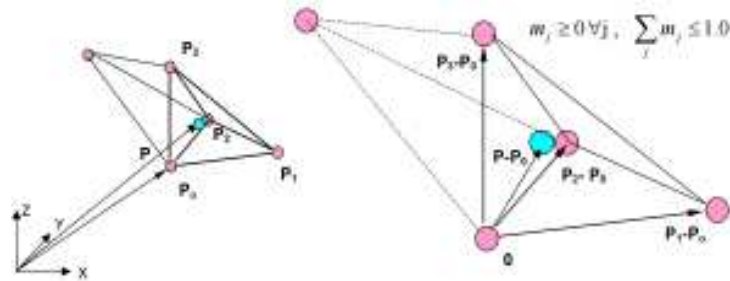


Abb. 1. Interpolation im Simplex der tetraedrischen Raumzerlegung

Stützstellen  $(P_0, P_1, P_2, P_3)$  gebildet wird, lautet

$$P = P_0 + \tilde{T} \cdot M = P_0 + \begin{pmatrix} (P_1 - P_0)_x & (P_2 - P_0)_x & (P_3 - P_0)_x \\ (P_1 - P_0)_y & (P_2 - P_0)_y & (P_3 - P_0)_y \\ (P_1 - P_0)_z & (P_2 - P_0)_z & (P_3 - P_0)_z \end{pmatrix} \cdot \begin{pmatrix} m_0 \\ m_1 \\ m_2 \end{pmatrix}, \quad (1)$$

$$M : m_j \geq 0 \quad \forall j, \quad \sum_j m_j \leq 1. \quad (2)$$

Verschiebt man den Tetraederaufpunkt  $P_0$  in den Koordinatenursprung, so gibt der Vektor  $M$  die Anteile des Vektors  $(P - P_0)$  als Linearkombination der Spaltenvektoren der Matrix  $\tilde{T}$  wider. Die Spaltenvektoren der Matrix  $\tilde{T}$  beinhalten hierbei die mit dem Betrag der Seiten  $(P_1 - P_0, P_2 - P_0, P_3 - P_0)$  skalierten Einheitsvektoren eines Tetraederkoordinatensystems bezüglich des Aufpunktes  $P_0$  in Koordinaten des zugrunde liegenden Raumes.

Anhand der weiteren Zerlegung der Matrix  $\tilde{T}$  in Einheitsvektor- und Skaliermatrix

$$\tilde{S} = \begin{pmatrix} |P_1 - P_0| & 0 & 0 \\ 0 & |P_2 - P_0| & 0 \\ 0 & 0 & |P_3 - P_0| \end{pmatrix}$$

$$\tilde{T} = (\tilde{T} \cdot \tilde{S}^{-1}) \cdot \tilde{S} = \tilde{E} \cdot \tilde{S}$$

lassen sich Kriterien ableiten, die sich zur Beurteilung der Regularität eines Tetraeders<sup>6</sup> eignen. Diese Bewertung schließt den Test der Tetraedereckpunkte auf Komplanarität, Kolinearität und Identität<sup>7</sup> ein. Dies ist insbesondere für die zum triangulierten Bezugsraum korrespondierende geometrische Raumzerlegung notwendig. Daneben ist das in [3] vorgeschlagene normalisierte Volumenverhältnis von umfassender und eingeschlossener Kugel bezüglich des idealen, gleichseitigen Tetraeders ein geeigneter Anhaltspunkt zur heuristischen Bewertung der Gestaltungseigenschaften.

Der Test, ob ein Vektor innerhalb eines Tetraeders  $i$  der Raumzerlegung liegt, erfolgt unter Verwendung der Bedingung laut Gl. (2) für alle oder eine zuvor eingeschränkte

<sup>6</sup> Spatvolumen  $|e_0 \cdot (e_1 \times e_2)| > \epsilon$ ,  $\epsilon \in (0, 1]$  der Einheitsvektoren  $\tilde{E} : e_j$  (Grad der linearen Abhängigkeit), Quantität und Homogenität der Skalierung  $\tilde{S} : s_{i,i}$  bzw.  $\max |s_{i,i} - s_{j,j}|$

<sup>7</sup> im Fall metamerer Testfarbtargets

Menge der beteiligten Raumsimplexe.

Die durch die Eckpunktkorrespondenz bedingte Äquivalenz des Vektors  $M$  eines beliebigen inneren Punktes  $P$  zweier Tetraeder des Ist- und Sollraums führt auf die lineare Transformationsbeziehung zwischen Ist- und Sollraum der betrachteten Raumabschnitte. Die Abbildung ist im Falle regulärer Tetraeder eineindeutig und somit in beliebiger Richtung möglich.

$$\begin{aligned} (\tilde{\mathbf{T}}^{\text{Soll}})^{-1} \cdot (\mathbf{P}^{\text{Soll}} - \mathbf{P}_0^{\text{Soll}}) &= \mathbf{M} = (\tilde{\mathbf{T}}^{\text{Ist}})^{-1} \cdot (\mathbf{P}^{\text{Ist}} - \mathbf{P}_0^{\text{Ist}}) \\ \rightarrow \mathbf{P}^{\text{Soll}} &= (\tilde{\mathbf{T}}^{\text{Soll}}) \cdot (\tilde{\mathbf{T}}^{\text{Ist}})^{-1} \cdot (\mathbf{P}^{\text{Ist}} - \mathbf{P}_0^{\text{Ist}}) + \mathbf{P}_0^{\text{Soll}} \end{aligned} \quad (3)$$

Die Stetigkeit an den Grenzen zweier benachbarter Tetraeder wird über ihre gemeinsamen Tetraedereckpunkte hergestellt. Unter Verwendung der Matrixzerlegung  $\tilde{\mathbf{T}} = \tilde{\mathbf{E}} \cdot \tilde{\mathbf{S}}$  können gestaltsbedingte Auswirkungen auf die lokalen Transformationseigenschaften im praktischen Einsatz (Rauschskalierung und -korrelation zwischen Ist- und Sollraum) dargestellt werden.

#### 4 Applikationsbeispiel

Zum Test der Eigenschaften des lokal linearen Korrekturansatzes soll das folgende Szenario dienen. Die Aufgabe besteht in der empfindungsgemäßen Zuordnung (Klassifikation) eines mit einer kommerziellen Farbkamera unter definierten Messbedingungen (Weißlicht-LED,  $0^\circ/d$ - Messgeometrie) erfassten Messfeldes bezüglich eines gegebenen applikationsspezifischen Referenzfarbsatz. Zu diesem Zweck ist die Anbindung des gerätespezifischen Farbraumes der Kamera an einen standardisierten Raum zur objektiven Farbmessung notwendig. Hierfür wurde der CIE-Lab-Raum für die DIN-Lichtart *D65* gewählt. Die zu approximierende Transformationsbeziehung ist unabhängig von verzerrenden Einflüssen auf den primären Farbraum durch die Kamera somit eindeutig nicht linear.

Der Referenzfarbsatz umfasst 15 weitestgehend unbunte Farben aus einem begrenzten Gebiet des Soll-Farbraumes. Aufgrund der vergrößerten Wahrnehmungsempfindlichkeit in diesem Farbraumbereich sind die Anforderungen an die Approximationsgüte im Sollfarbraum besonders hoch. Dies gilt insbesondere für die Abbildung und das Wiedererkennen der Referenzfarben. Zur stabilen Charakterisierung der Abbildungsvorschrift in diesem Raumgebiet wurden darüber hinaus 84 Hilfsfarben u.a. des kommerziellen *Munsell Nearest Neutral*-Farbsatzes ausgewählt, um die stark variierende Stützstellendichte der Referenzfarben möglichst auszugleichen und deren Verteilung in ausreichendem Maße zu umgeben. Als notwendige Randbedingung zur Gewährleistung regulärer Tetraeder wurde sichergestellt, dass sowohl Referenz- als auch Hilfsfarben im primären Farbraum der Sensorik eindeutig trennbar und somit nicht metamer sind. Zum Test der Approximationsgüte des Korrekturansatzes im realisierten Soll-Farbraum wurden aus dem Referenzdatensatz 4 Testfarben ( $T$ ) entfernt und nicht in die Raumzerlegung einbezogen. Die Triangulation umfasst somit insgesamt 95 Stützstellen (siehe Abb. 2).

Zum Verfahrensvergleich wurden zusätzlich die Ergebnisse bei Einsatz zweier alternativer, aber **globaler** Korrekturverfahren, linear - affin und nicht linear - polynomial 3. Ordnung, ermittelt. Die mittleren und maximalen Farbabstände zwischen Referenzfarben im realisierten Soll-Farbraum und Sollfarborten sind in Tabelle 1, die Farbabstände

der approximierten Testfarben ( $T^*$ ) zum Sollfarbort ( $T$ ) und dem erwarteten bzw. ermittelten empfindungsgemäß nächsten Nachbarn ( $NN$  bzw.  $NN^*$ ) in Tabelle 2 dargestellt.

**Tabelle 1.** Approximationsgüte der 11 Referenzfarben

	lokal, tetraedrisch	global, affi n	global, polynomial
$\overline{\Delta E}_{\text{Lab}}^{\text{D65}}$	<b>0.00</b>	1.75	0.99
$\max [\Delta E_{\text{Lab}}^{\text{D65}}]$	<b>0.00</b>	3.75	2.02

**Tabelle 2.** Approximations- und Messgüte der Testfarben

Testfarbe $T$ , ( $NN$ , $\Delta E_{\text{Lab}, NN-T}^{\text{D65}}$ )	$\Delta E_{\text{Lab}, T^*-T}^{\text{D65}}$ , ( $NN^*$ , $\Delta E_{\text{Lab}, NN^*-T^*}^{\text{D65}}$ , $\Delta E_{\text{Lab}, NN-T^*}^{\text{D65}}$ )		
	lokal, tetraedrisch	global, affi n	global, polynomial
A3 (A2, 2.46)	<b>1.82, (A2, 2.42, 2.42)</b>	2.17, (A2, 2.21, 2.21)	2.18, (A2, 2.78, 2.78)
B3 (B4, 2.84)	<b>2.11, (B4, 3.19, 3.19)</b>	3.76, (A2, 3.06, 3.22)	3.52, (B4, 3.01, 3.01)
C3 (A35, 2.97)	<b>0.18, (A35, 2.95, 2.95)</b>	2.14, (C2, 2.11, 4.62)	0.84, (C2, 3.71, 3.72)
D3 (C2, 2.68)	<b>0.68, (C1, 2.69, 2.75)</b>	1.04, (C2, 2.18, 2.18)	1.55, (C2, 2.64, 2.64)

Obwohl die Ergebnisse des lokalen Ansatzes vergleichsweise sehr gut erscheinen, bedürfen diese zur korrekten Bewertung einiger Erläuterungen. Die Raumzerlegung für die gewählten Farbtargets im Ist-Farbraum (siehe Abb. 2, links) enthält zahlreiche „deformierte“ Tetraeder insbesondere im Randbereich des konvexen Körpers. Dies ist insofern problematisch, da im Falle einer sehr lang gestreckten Gestalt das Lokalisierungsprinzip verletzt wird. Der lineare Modellansatz auf der Grundlage von 4 räumlich nicht lokalen Targetpunkten widerspiegelt im Inneren des Tetraeders die realen Abbildungsverhältnisse dann nur ungenügend (Testfarben A3, B3). Darüber hinaus wird beim stochastisch induziertem Übergang zwischen irregulären und regulären Raumabschnitten die Rauschstatistik wesentlich verzerrt, so dass eine erwartungstreue Mittelwertschätzung von Einzelmessungen im Sollraum nicht in jedem Raumbereich möglich ist.

Der gesamte Farbkörper im Ist-Farbraum ist insgesamt stark abgeplattet, was auf die Existenz numerisch problematischer, nahezu komplanarer Tetraeder schließen lässt. Um die für die Testfarben C3 und D3 erzielten Resultate im gesamten Raumabschnitt zu realisieren, besteht daher Bedarf zur Modifikation und Erweiterung der noch nicht optimalen Stützstellenverteilung. Sofern nicht durch die Applikation vorbestimmt, sollte die Auswahlgrundlage für physisch verfügbare Kalibriertargets immer deren gleichmäßige Verteilung im triangulierten Raum (Regularität) sein.

Da durch den begrenzten Stützstellenumfang nicht sicher gestellt werden kann, dass Messwerte immer im Inneren des triangulierten Farbkörpers liegen, ist die Behandlung dieses Sonderfalls im praktischen Einsatz bedeutsam. Hier ergeben sich neben der Rückweisung zwei Varianten:

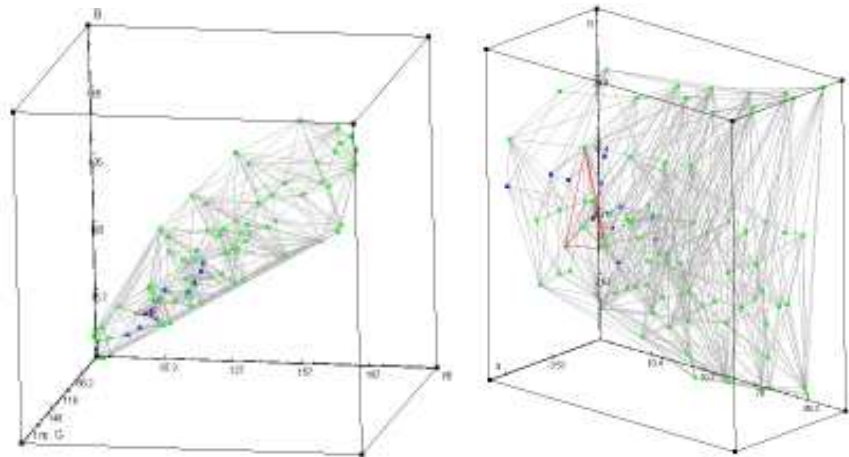
- **Extrapolation:** Farbvalenzen außerhalb des Triangulationskörpers, deren Abstand  $d$  entlang einer heuristischen Geraden zur konvexen Hülle kleiner als eine vorzuziehende Schwelle ist, werden im Soll-Farbraum linear extrapoliert (siehe Abb. 3,

links). Liegt der Abstandsentscheidung ein Sollraum mit empfindungsgemäß gleichabständiger Metrik zugrunde (Abstand zur zugehörigen Tetraeder„außenfläche“), so folgt daraus eine farbortabhängige Ausdehnung des konvexen Triangulationskörpers im Istraum.

- **Heuristisches Mapping auf die konvexe Hülle:** Extrapolierte Farbvalenzen im Soll-Farbraum werden auf den Schnittpunkt der heuristischen Geraden mit der zugehörigen Außenfläche gemappt (siehe Abb. 3, rechts). In Soll-Farbräumen mit dekorrelierten Farbvalenzkomponenten kann neben der Projektion entlang der Flächennormalen auch ein Mapping unter Beibehaltung des Bunttons oder der Helligkeit erfolgen. Die Entscheidung ist applikationsspezifisch zu treffen und zu bewerten.

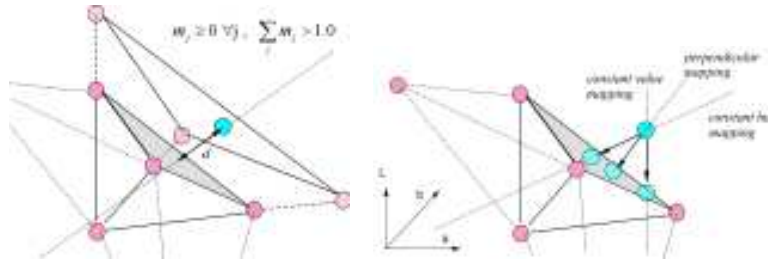
Im Rahmen des Applikationsbeispiels konnte festgestellt werden, dass die Approximationsgüte der Helligkeitskomponente  $L$  höher ist, als die der übrigen Komponenten des Soll-Farbraums. Aus diesem Grund erfolgte das Mapping „außenliegender“ Farbvalenzen auf die konvexe Hülle unter Beibehaltung der Helligkeit.

Für die Zuordnung (Klassifikation) der approximierten Farbvalenz zum Referenzfarbsatz könnte die geometrische Partitionierung des Raumes nutzbringend eingesetzt werden, da sich die Menge der zu testenden Punkte zunächst auf die der Approximation zugrunde liegenden Tetraedereckpunkte beschränken lässt und nur im Bedarfsfall erweitert werden muss.



**Abb. 2.** Triangulation der Referenz- (blau) und Hilfsfarben (grün) im Sensorfarbraum (links) sowie korrespondierende geometrische Struktur im Sollfarbraum (rechts).

Für das hier beschriebene Beispiel ist jedoch infolge der Verletzung des Lokalisierungsprinzips und der geringen Anzahl der Referenzfarben gegenüber den Hilfsfarben hieraus nur geringer Vorteil zu ziehen.



**Abb. 3.** Behandlung von Sonderfällen außerhalb des Triangulationskörpers: Extrapolation (links), Kompression (rechts)

## 5 Zusammenfassung und Ausblick

Gegenstand dieses Beitrages war ein Verfahren, dass die farbmtrische Anbindung an eine valenzmetrische Schnittstelle durch Anwenden einer lokal linearen Farbraumabbildung vornimmt. Das große Potential dieses Ansatzes konnte im Rahmen des Applikationsbeispiels bisher nur teilweise ausgeschöpft werden. Der Schlüssel für den erfolgreichen Einsatz dieser Verfahrensgruppe ist die Bereitstellung von Kalibriertargets in ausreichender Zahl. Da die lokale Abbildungsvorschrift auf lediglich 4 Punkten beruht, ist die geeignete Verteilung der Targets zur Gewährleistung des Lokalisprinzipis und zur Realisierung regulärer Raumabschnitte in den Farbräumen die Grundvoraussetzung. Aufgrund der im Vorfeld nicht bekannten Komplexität des zu approximierenden Zusammenhangs, ist eine gleichmäßige, höchstmögliche Stützstellendichte hierbei der Idealfall. Das zwangsläufige Auftreten irrelevanter Stützstellen wirkt aufgrund des lokalen Ansatzes dabei nicht störend. Bei Annäherung an das Ideal, steht mit dieser Verfahrensgruppe ein leistungsfähiger, flexibler Ansatz zur Realisierung von standardisierten Farbräumen für farbmtrische Aufgaben zur Verfügung, der insbesondere beim Vorliegen lokaler Verzerrungen einer ansonsten „glatten“ Abbildung nutzbringend eingesetzt werden kann.

## Literatur

1. Groff R.E., Koditschek D.E. Khargonekar P.P.: Representation of Color Space Transformations for Effective Calibration and Control, J. of S&T's NIP16, Int. Conf. on Digital Printing Technologies Vancouver, B.C., Canada, Nov.2000, S.:255-260, ISBN/ISSN: 0-89208-230-5
2. Hung P-C.: Colorimetric calibration in electronic imaging devices using look-up-table model and interpolations, J. of Electronic Imaging, Vol.:2(1), 1993, S.:53-61
3. Kasson, J. M., Plouffe W., Nin, S.I.: A tetrahedral interpolation technique for color space conversion, Proc. of SPIE, Vol.:1909, S.:127-138



## Spektral modellierbare Lichtquelle zur Erzeugung beliebiger Spektren durch Einsatz eines „Digital Mirror Device“

Markus Schnitzlein<sup>1</sup>, Bernhard Frei<sup>1</sup> und Andreas Willert<sup>2</sup>

<sup>1</sup> Océ Document Technologies GmbH, 78467 Konstanz  
Abteilung Scanner-Technik und Bildverarbeitung,  
[markus.schnitzlein@odt-oce.com](mailto:markus.schnitzlein@odt-oce.com)  
[www.odt-oce.com](http://www.odt-oce.com)

<sup>2</sup> TU Chemnitz – Institut für Print- und Medientechnik  
[www.tu-chemnitz.de/pm](http://www.tu-chemnitz.de/pm)

**Zusammenfassung.** Die mehrkanalige bzw. multispektrale Bildgewinnung gewinnt zunehmend ein größeres Interesse - vor allem in Bezug auf die spektrale Repräsentation der zu erfassenden Objekte. Die Kenntnis der spektralen Eigenschaften ermöglicht eine differenzierte farborientierte Verarbeitung der erzeugten Bilder. Neben der spektralen Erfassung von Vorlagen, die heute durch den Einsatz mehrkanaliger Kamerasysteme erfolgt und durch die Verwendung von speziellen mathematischen Verfahren, die zu relativ guten Schätzungen der spektralen Eigenschaften der Vorlage führen, ergibt sich ein weiteres bedeutendes Verbesserungspotenzial durch Verwendung von Lichtquellen mit definierten Spektren. Die im Rahmen dieser Arbeit vorgestellte spektral modellierbare Lichtquelle ermöglicht die Erzeugung beliebiger spektraler Verteilungen mit einer Genauigkeit von ca.  $\pm 1nm$  und einer Intensitätsgenauigkeit von  $\pm 1\%$ . Die Lichtquelle wird durch einen Digital Light Processor (DLP) gesteuert, mit dessen Hilfe die Anteile von einzelnen Basisspektren kombiniert werden. Die Ermittlung der Arbeitsparameter des DLPs zur Erzeugung eines beliebigen Beleuchtungsspektrums wird mittels eines iterativen Vektorraum-Projektionsverfahrens gelöst. Die Lichtquelle eignet sich sowohl als Ersatz für optische Farbfilter bei der digitalen Bildaufnahme, aber auch als Lichtquelle zur multispektralen Farbwiedergabe in Projektionssystemen. Im Rahmen des Vortrags werden sowohl Ergebnisse von spektralen Messungen an der modellierbaren Lichtquelle gezeigt, als auch die mathematischen Verfahren zur Ansteuerung des DLPs diskutiert.

## 1 Einleitung

### 1.1 Aufbaukonzept der Lichtquelle

Alle bisher gezeigten Ansätze zur Farberfassung und Farbwiedergabe werden primär mit dem Ziel eingesetzt, den visuellen Eindruck, den ein Farbreiz verursacht, nachzubilden, wobei der erste Anspruch darin besteht, die Tristimulus-Werte der visuellen Wahrnehmung berechnen zu können.

Da sich dieses Ziel letztlich nur durch die spektrale Nachbildung erreichen lässt, muss versucht werden, diese spektralen Eigenschaften möglichst exakt zu bestimmen.



Mit der etablierten Technik ist dies derzeit nicht möglich, da Sensoren und Wiedergabefarben noch bevorzugt in X,Y,Z - Werten beschrieben werden. Die Mehrdeutigkeit dieser Beschreibung führt einerseits zu Metamerie-Effekten, andererseits zu der Notwendigkeit Erfassungs- und Wiedergabesysteme durch empirische Methoden aufeinander abzustimmen, um im Verlauf des Bearbeitungsprozesses eine konstante Farbbehandlung zu ermöglichen (Color Management).

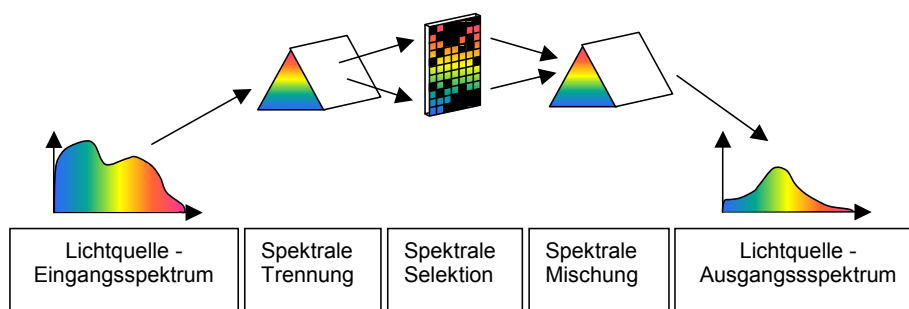
Mit der nachfolgend beschriebenen spektral modellierbaren Lichtquelle wird eine spektral exakte Farbwiedergabe und Farberfassung möglich, wobei die Genauigkeit mit der die Farbe dargestellt oder erfasst werden kann, im System wählbar ist.

Die Basisidee der spektralen modulierbaren Lichtquelle ist die spektrale Aufteilung der Emission der Primärlichtquelle mit Hilfe von dispersiven optischen Elementen (Blaze-Gitter oder Prisma) und die spektral selektive Zusammenführung einzelner Wellenlängenbereiche zu der gewünschten Spektralverteilung. Die spektrale Selektion erfolgt durch einen DMD (Digital Mirror Device - in der Literatur oft auch als DLP „Digital Light Processor“ bezeichnet).

Da Dispersionselemente naturgemäß eine winkelabhängige Spektralverteilung produzieren, hängt die Güte der spektralen Trennung direkt vom Divergenzwinkel des einfallenden Lichtbündels ab. Je größer der Divergenzwinkel ist, umso größer wird die spektrale Breite des selektierten Wellenlängenbereiches. Die Parallelisierung des einfallenden Lichtes erfolgt durch eine Spaltblende. Die Güte der Fokussierung des von der Primärlichtquelle emittierten Lichtes auf diese Spaltblende bestimmt direkt die Effizienz des Gesamtsystems.

Durch Ansteuerung des DLP-Chips können, während der Bilderfassung, beliebige Wellenlängenbereiche (und deren Kombination) auf die Vorlage gebracht werden. Die auf diese Weise erhaltenen mehrkanaligen Bilder enthalten pixelweise, in unterschiedlichen Bildebenen, die spektralen Remissionseigenschaften der Vorlage.

In den nachfolgenden Abbildung ist der einfacheren Darstellung wegen, der reflektiv arbeitende DMD durch ein allgemeinen SLM (Spatial Light Modulator) ersetzt.



**Abb. 1.** Methodischer Aufbau des Selektionssystems

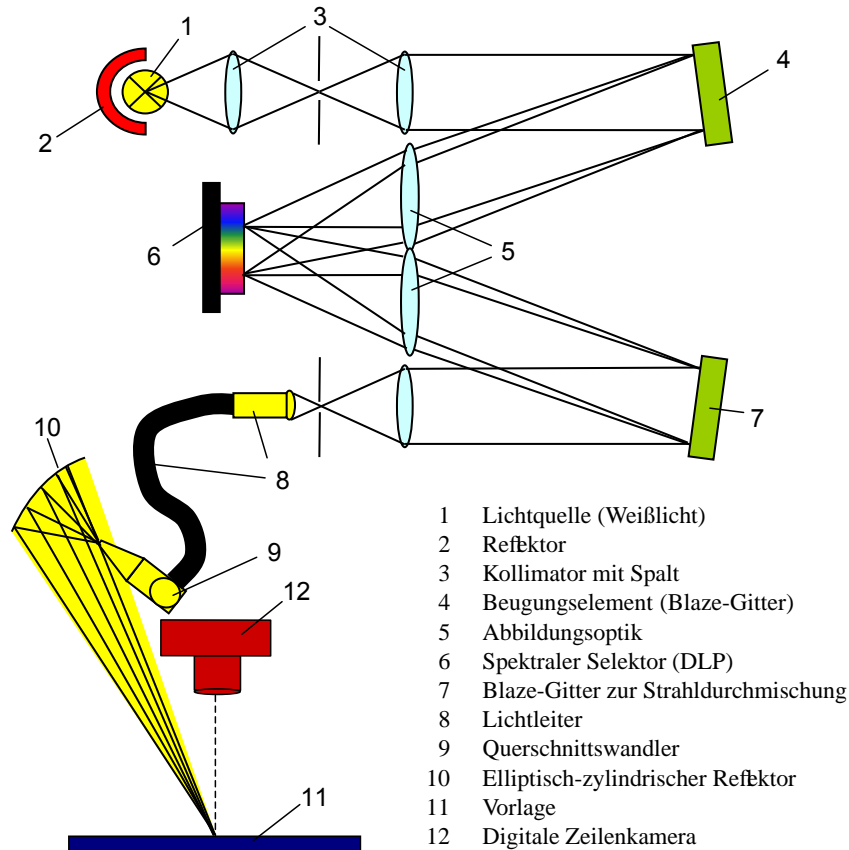


Abb. 2.

Die Lichtquelle (1) wird durch eine effiziente Xenon-Hochdrucklampe realisiert, deren emittiertes Licht über einen elliptischen Spiegel (2) auf einen Spalt fokussiert wird. Das durch den Spalt hindurch tretende Licht wird durch eine Kollimatoroptik (3) parallel gerichtet und trifft auf ein Beugungselement (4), das durch seine Gitterstruktur (Blaze) das polychromatische Licht der Lichtquelle wellenlängenabhängig in unterschiedlichen Winkeln ablenkt. Das hiermit spektral aufgefächerte Lichtbündel wird über eine Abbildungsoptik (5) auf den DLP-Chip (6) abgebildet. Hier können selektiv einzelne Wellenlängenbereiche ausgeblendet werden, sodass nur das aktiv benötigte Licht durch die zweite Abbildungsoptik hindurch gesammelt wird. Das auf diese Weise erhaltene Strahlbündel enthält immer noch winkelabhängige Spektralanteile, die erst durch das zweite Blaze-Gitter (7) in eine gemeinsame Richtung gebeugt werden. Durch die Kollimatoroptik (8) wird das Licht in einen faseroptischen Lichtleiter (9) eingekoppelt, der durch eine Umsortierung der Lichtleitfasern einen spaltförmigen Austrittsquerschnitt (10) realisiert. Das hier divergent austretende Licht wird mit Hilfe eines

lang gestreckten elliptisch-zylindrischen Spiegels auf die Vorlage (11) fokussiert. Das Beleuchtungssystem ist derartig konzipiert, dass das vom Lichtleiter emittierte Licht streifenförmig auf das Objekt projiziert wird. Der Austrittsspalt des Lichtleiters befindet sich im ersten Fokus der Ellipse - die Vorlage im zweiten Brennpunkt. Durch die langen Übertragungswege zwischen der Quelle und der Vorlage, kommt es zu einer sehr guten Lichtdurchmischung entlang des Beleuchtungsstreifens auf dem Objekt. Diese Durchmischung ist sowohl hinsichtlich der Abstrahlungsinhomogenitäten des Lichtleiters selbst, als auch hinsichtlich der spektralen Durchmischung des Lichtbündels, günstig. Das von der Vorlage remittierte Licht wird mit einer Zeilenkamera (12) mit 10 Bit Grauwertaufösung detektiert.

Die folgenden technischen Aspekte dieses Erfassungssystems müssen genauer betrachtet werden:

1. Effizienz:

Bei der Parallelisierung des von der Lichtquelle divergent emittierten Lichts entsteht ein hoher Energieverlust, der in der geringen Größe der Spaltblende begründet liegt. Je größer die Spaltblende realisiert werden kann, umso größer ist die Effizienz des Gesamtsystems, aber in gleicher Weise verschlechtert sich die spektrale Trennung des Systems. Die lichttechnische Effizienz des Gesamtsystems bewegt sich zwischen 1% - 3%.

2. Ansteuerung des DLP

Die Berechnung der Einstellparameter der einzelnen DLP-Spiegelelemente ist sehr komplex und bedarf einer algorithmischen Optimierung. Die Rechenzeit für eine Optimaleinstellung, hängt stark von der spektralen Homogenität des Zielspektrums ab. Bei Berechnung aller Einstellparameter der ca. 800.000 Mikrospiegel bewegt sich die Rechenzeit im Bereich von 5 - 60 min.

3. Spektrale Durchmischung

Die spektrale Durchmischung des Lichtbündels ist von großer Bedeutung, um eine spektral homogene Beleuchtung der Szene zu realisieren. Alternative Möglichkeiten ohne zweites (rückführendes) Beugungselement müssen untersucht werden.

4. Separate Verwendbarkeit der Lichtquelle

Die Verwendung der Lichtquelle, als Normlichtquelle mit einstellbarem Spektrum ist in Laufe der Forschungsarbeit an dem beschriebenen Scansystem immer wichtiger geworden. Eine spektral einstellbare und steuerbare Lichtquelle ist nicht nur im Umfeld der Bilderfassung, sondern auch im Bereich der Messtechnik von Bedeutung.

## 1.2 Spektrale Approximation von Spektren

Die Berechnungsverfahren, die zur Anwendung kommen, um die optimalen Beiträge der erzeugten Bandpässe zu ermitteln, führen zu einer sehr guten Annäherung der Zielspektren. Es wurden testweise die Optimaleinstellungen für alle Munsellfarben aus „Book of Colors“, die „Natural Colors“ aus der Pointer-Datenbank und die standardisierten Lichtfarben (A, B, C, D50, D65, F2, F11) berechnet. Es zeigt sich, dass das beschriebene Verfahren alle o. g. Spektren mit einer Abweichung deutlich besser als  $\Delta E = 0,5$  beschreiben kann.

Besondere Bedeutung gewinnt dieses Ergebnis vor dem Hintergrund, dass bereits mit sehr großen Bandbreiten von ca. 50 nm sehr gute Approximationen erzeugt werden konnten.

Mit dem beschriebenen spektralen Trennungssystem erhalten wir eine 2-dimensionale Spektralverteilung  $S(\lambda, \sigma)$  in der DMD-Ebene, die durch die einzelnen Spiegel des DMDs in ihrem Beitrag zum Gesamtspektrum gesteuert werden können. Die Zielsetzung besteht nun darin, eine vorgegebenes Spektrum  $E(\lambda)$  (Zielspektrum) durch Superposition der einzelnen Spektralverteilungen  $S_{pq}(\lambda_p, \sigma_q)$  linear zu kombinieren.

$$E(\lambda) = \sum_{p=1}^s \sum_{q=1}^t x_{pq} S_{pq}(\lambda_p, \sigma_q)$$

der Index  $p = 1, \dots, s$  beschreibt die Position der Spiegel des DMDs, über die das Spektrum in Richtung der Wellenlängenachse verteilt ist - der Index  $q = 1, \dots, t$  beschreibt die Position Spiegel des DMDs, senkrecht zur Wellenlängenachse - in dieser Richtung ändert sich die Bandbreite der Spektralverteilung bei fester Mittenwellenlänge  $\lambda_m$  der Verteilung.

Von H. Stark und Y. Yang [2], [1] wurde ein iteratives Lösungsverfahren zur auf Basis von Vektorraum-Projektionsmethode vorgeschlagen, das für diesen Ansatz sehr gut geeignet ist. Das Verfahren wird in den angegebenen Literaturstellen sehr gut beschrieben.

In unserem Fall soll eine Linearkombination von spektralen Basisfunktionen gefunden werden, die zur besten Approximation eines Zielspektrums führt, wobei die Beiträge der breitbandigen (und damit effizienten) Basisfunktionen möglichst groß werden soll.

Es ist schnell einsichtig, dass sich bei Verwendung von allen Einzelspiegeln (ca. 800.000), als parametrisierbares Gewicht einer spektralen Basisfunktion, ein sehr rechenzeitintensiver Prozess ergibt.

Bei einer Spiegelanzahl von  $1024 \times 768 = 786432$  Elementen ergibt eine Anzahl  $2 \times (786432)^2$  Multiplikation =  $1,2 \cdot 10^{+12}$  und ebensovielen Multiplikationen pro Iterationsschritt. Voraussichtlich liegt die Anzahl an Iterationsschritten, die für eine „gute Approximation“ benötigt werden - in Abhängigkeit von der Komplexität des spektralen Verlaufs - etwa im Bereich der Vektorlänge, sodass sich eine Anzahl von etwa  $10^{+18}$  Operationen zur Berechnung eines Zielspektrums ergibt.

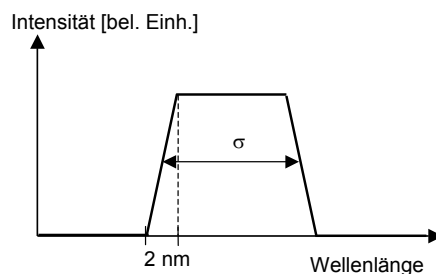
Die Anzahl an Bandpässen, die modelliert werden, wurde auf 12 Bereiche reduziert. Die Anzahl an Stützstellen des Spektrums wurde von 401 auf 101 reduziert. Dies ist keine signifikante Restriktion, da auf Basis reale Spektren auf dieser Auflösungs-skala als glatt angesehen werden können. Exemplarisch wurden einige Spektren mit der vollen Stützstellenanzahl modelliert, wobei sich keine signifikanten Ergebnisveränderungen ergaben.

Für die Approximation wurden 12 Bandpässe in einer 4 nm Abstufung der Bandbreite vorgeben: 4 nm, 8 nm, 12 nm, 16 nm, 20 nm, 24 nm, 28 nm, 32 nm, 36 nm, 40 nm, 44 nm, 48nm. Die Mittelwellenlängen der Bandpässe wurden im Bereich von 380 nm - 780 nm im 4 nm-Abstand (101 Stützstellen) im Spektrum festgelegt.

## 2 Ergebnisse

Die nachfolgend dargestellten Ergebnisse zeigen die Approximation von zwei Normlichtspektren: D65 und F11.

Zielsetzung ist es, eine möglichst gute spektrale Näherung des Zielspektrums zu erreichen - durch die Linearkombination von Basisspektren mit möglichst großer Bandbreite. Die Basisspektren sind hierbei symmetrische Trapezfunktionen mit einer mit einer Halbwertsbreite von 48nm, 44nm, 40nm, 36nm, 32nm, 28nm, 24nm, 20nm, 16nm, 12nm, 8nm und 4nm. Die Flankensteilheit beträgt immer 4nm.



**Abb. 3.** Grundform der Basisfunktionen. Die Bandbreite  $\sigma$  variiert von 48 nm bis 4 nm in Schritten 4 nm

Die Darstellungen zeigen:

**Abb. 4, 7:** Die Approximation des Normlichtspektrums D65 bzw. F11 (schwarze Linie) durch bis zu 12 Bandpässe. Beginnend mit der breitesten Basisfunktion (48 nm) werden schrittweise Bandpässe mit geringerer Bandbreite angeboten, sodass sich letztlich die beste Näherung durch die Linearkombination der Basisfunktionen mit der geringsten Bandbreite ergibt.

**Abb. 5, 8:** Die Abweichung der Intensitätswerte der Zielfunktion von den Approximationsfunktionen für Normlicht D65 und F11 bei Bandbreiten von bis zu 40nm, 20nm und 4nm.

**Abb. 6, 9:** Die beiden Kurven zeigen den Verlauf der spektralen Abweichung (Summe der stützstellenweise berechneten Abweichungsquadrate zwischen Zielfunktion und Approximationsfunktion). Diese Funktion muss prinzipbedingt monoton fallen. Die zweite Kurve zeigt den Verlauf des  $\Delta E_{ab}$  Wertes. Dieser Wert zeigt zwar auch eine fallende Tendenz - die unterschiedliche lokale Näherungen der Approximationsfunktionen an das Zielspektrum führt jedoch zu einem nicht monotonen Verhalten.

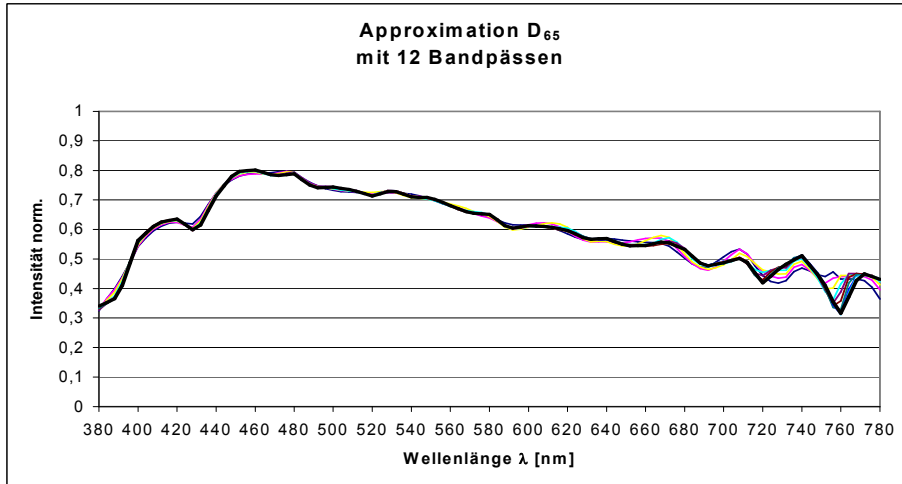


Abb. 4.

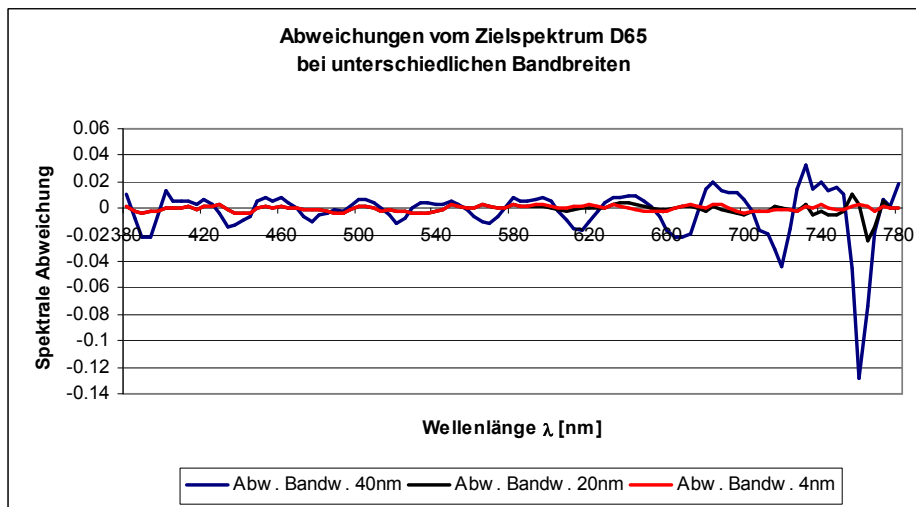


Abb. 5.

### 3 Zusammenfassung

Mit der beschriebenen Lichtquelle ist es möglich beliebige Zielspektren zu erzeugen und in schneller Folge zu wechseln, sodass sich über die reine Lichtquelle hinaus, eine Vielzahl von Applikationen ergeben, in denen eine wählbares Emissionsspektrum entscheidende technische Vorzeile bieten würde. Als Beleuchtungssystem für Kamera- und Scannersysteme zur Gewinnung hochwertiger und hoch aufgelöster Bilder von

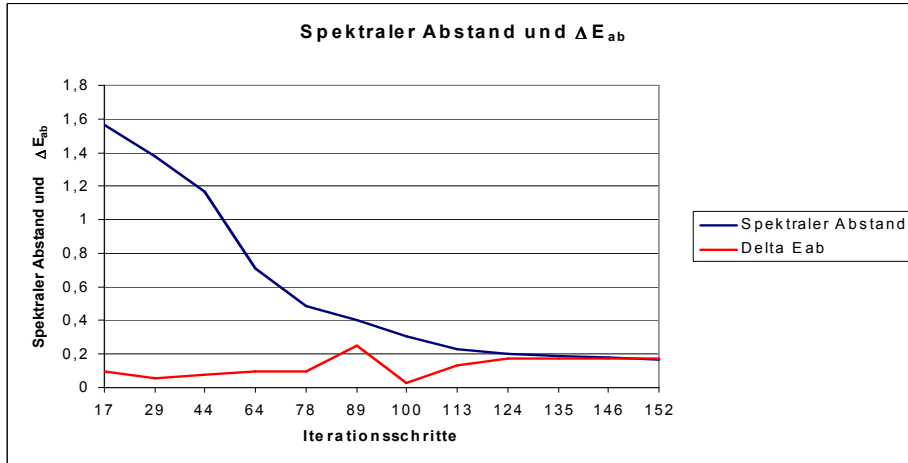


Abb. 6.

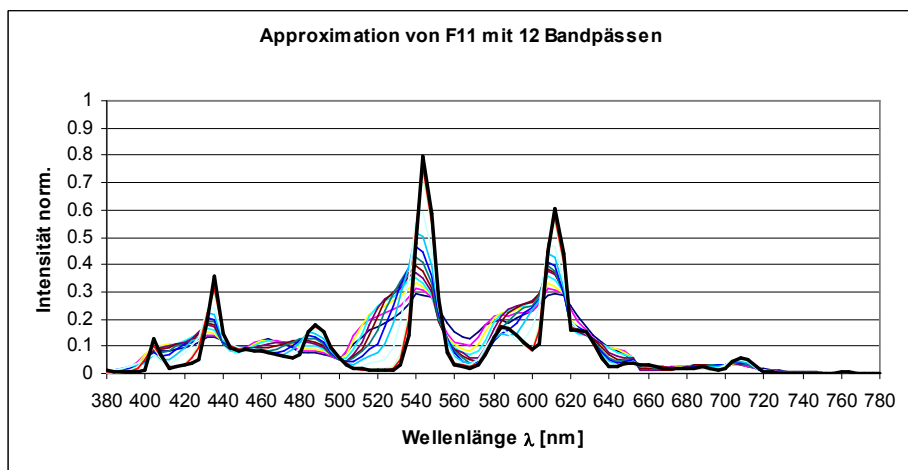


Abb. 7.

großformatigen Vorlagen, ist diese Lichtquelle bereits in Erprobung. Die Lichtquelle eignet sich sowohl als Ersatz für optische Farbfilter bei der digitalen Bildaufnahme, aber auch als Lichtquelle zur multispektralen Farbwiedergabe in Projektionssystemen. Die Berechnung der spektral optimalen und lichttechnisch effizienten Linearkombination von Basisspektren ist derzeit noch aufwendig. Aus heutiger Sicht bietet sich auf der Basis der Vektorraum-Projektionsmethode der am besten nutzbare Ansatz. Weitere lineare Optimierungsverfahren müssen im Hinblick auf die beschriebene Anwendung aber noch untersucht werden.

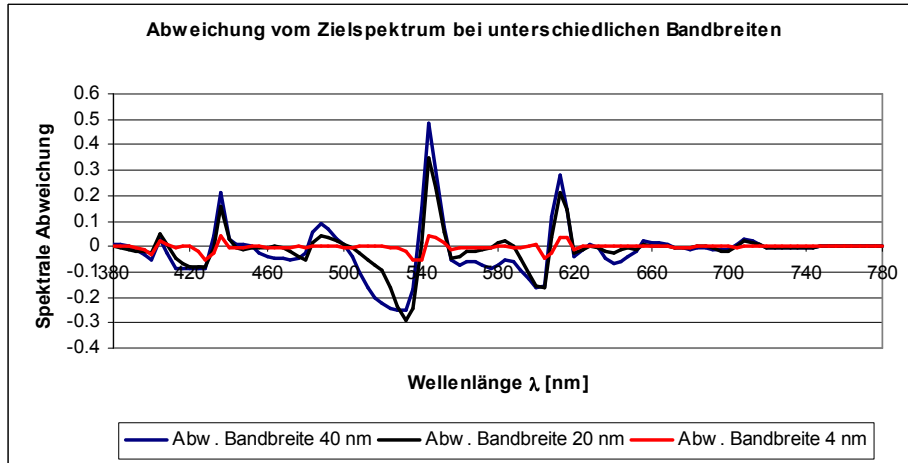


Abb. 8.

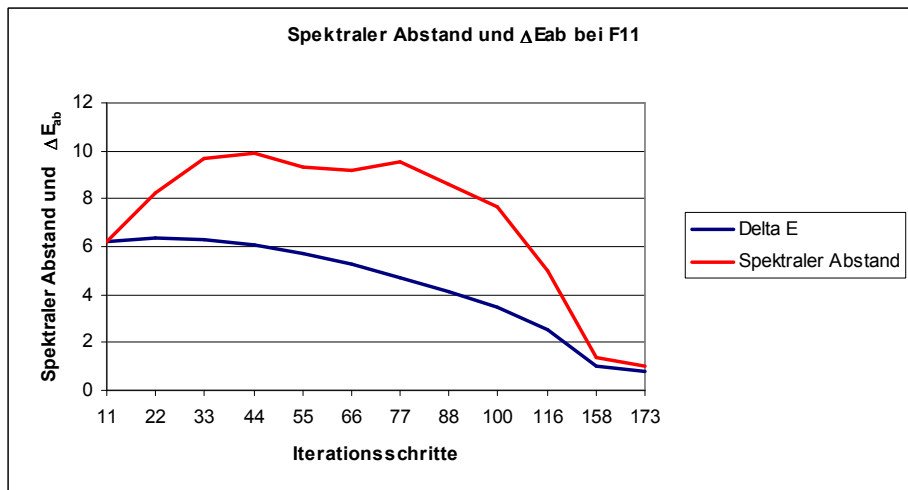


Abb. 9.

## Literatur

1. A. Levi and H. Stark. Signal restoration from phase by projection onto convex sets. *J. Opt. Soc. Am.*, 73:810–822, 1983.
2. H. Stark. *Vector Space Projections*. John Wiley&Sons, 1998.



## Farbbasierte Objekterkennung mit einem omnidirektionalen System

A. Maas<sup>1</sup>, P. Heim<sup>1</sup>, J. Kaluza<sup>1</sup>, S. Mitnacht<sup>1</sup>, V. Hong<sup>1</sup>, P. Maillard<sup>2</sup>, F. Occelli<sup>2</sup>,  
C. Kurucz<sup>1</sup>, D. Paulus<sup>1</sup>

<sup>1</sup> Institut für Computervisualistik, Universität Koblenz-Landau, Universitätsstr. 1, 56070  
Koblenz – Germany, [agas@uni-koblenz.de](mailto:agas@uni-koblenz.de)

<sup>2</sup> Université de Poitiers, 15, rue de l'Hôtel Dieu, 86034 Poitiers Cedex – France,  
{[pmaillard](mailto:pmaillard@gphy.campus.univ-poitiers.fr), [foccelli](mailto:foccelli@gphy.campus.univ-poitiers.fr)}@gphy.campus.univ-poitiers.fr

**Zusammenfassung.** In diesem Beitrag wird der Einfluß der Farbnormierung und der Farbkalibrierung auf die Performanz der farbbasierten Objekterkennung am Beispiel eines Roboters mit einem omnidirektionalen Kamerasystem untersucht. Der Roboter soll ein Objekt, in diesem Fall einen roten Ball, in einem langen Flur mit stark variierender Beleuchtung finden. Zur Lokalisation wird die Histogrammrückprojektion nach [10] eingesetzt. Um den Einfluß der Beleuchtungsänderungen auf die Lokalisation des Objekts zu reduzieren, werden Verfahren zur Farbnormierung und Farbkalibrierung vorgestellt und getestet.

### 1 Einleitung

Objekterkennung wird in zahlreichen Anwendungsbereichen wie z. B. der Wertstoffsortierung [1] und der Robotik [6] benötigt. Die Lokalisation und Identifikation von Objekten sind hierbei wichtige Aspekte [10]. Für die Objekterkennung können verschiedene Merkmale wie z. B. Farbe, Form und Textur verwendet werden. Die Farbe eines Objekts ist weitgehend invariant gegenüber Rotation, Translation und Skalierung des Objekts in der Szene [10]. Ein Problem des farbbasierten Ansatzes ist jedoch der Einfluss der variierenden Beleuchtungsverhältnisse, welcher durch die Anwendung von Verfahren zur Farbnormierung und Farbkalibrierung reduziert werden kann.

In diesem Paper wird als Anwendungsszenario ein autonomer mobiler Roboter vorgestellt, der in einem langen Flur mit unterschiedlicher Beleuchtung nach einem Ball sucht. Der Roboter nimmt mit einem omnidirektionalen Kamerasystem seine Umgebung in einer 360° Rundumsicht wahr. Der wesentliche Vorteil dieses Systems besteht darin, dass der Roboter ohne Änderung seiner aktuellen Fahrtrichtung Objekte vor, neben und hinter sich erkennen kann. Da so jedoch sehr verschieden ausgeleuchtete Bereiche der Umgebung gleichzeitig erfasst werden, wird das genannte Problem der unterschiedlichen Beleuchtungsbedingungen in der Szene verstärkt.

Im weiteren Verlauf wird in Abschnitt 2 das allgemeine Vorgehen zur Lokalisation des Balls in der Szene vorgestellt. In Abschnitt 3 werden Verfahren zur Farbnormierung und Farbkalibrierung erläutert. Im experimentellen Teil 4 werden die in Abschnitt 3 vorgestellten Verfahren an unterschiedlichen Stellen im Flur für verschiedene Ballpositionen getestet und ausgewertet. Der Beitrag schließt in Abschnitt 5 mit einer Zusammenfassung und einem Ausblick für zukünftige Arbeiten ab.

## 2 Vorgehen

In Abb. 1 ist die Verarbeitungskette für die Lokalisation des Balls in der Szene aufgeführt. Während der Ballsuche nimmt der Roboter mit einem omnidirektionalen System Bilder von seiner Umgebung mit variierenden Beleuchtungsbedingungen auf. Eine aufgenommene Szene wird zunächst in eine beleuchtungsinvariante Szene transformiert. Dieser Schritt wird in Abschnitt 3 näher vorgestellt. Im nächsten Schritt wird aus der Szene mit Hilfe einer Maske derjenige Ausschnitt entfernt, in dem sich der Roboter befindet. Die eigentliche Lokalisation des Balls in der Szene wird mit der Histogrammrückprojektion nach [10] realisiert. Zur Verbesserung der Lokalisation werden weitere Nachverarbeitungsschritte wie morphologische Operationen ausgeführt.

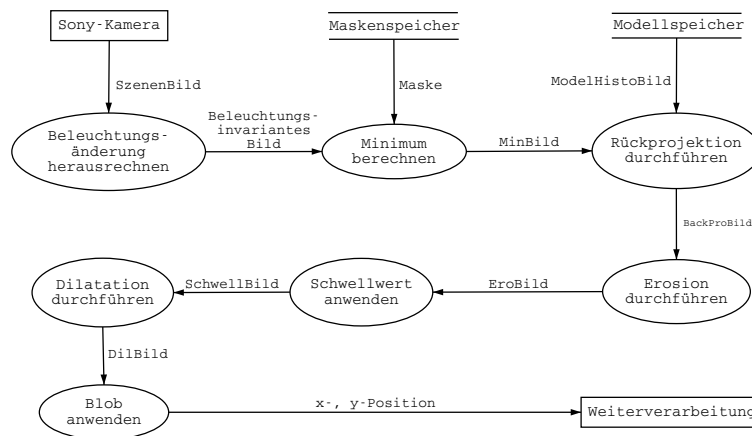


Abb. 1. Verarbeitungskette zur Lokalisation des Balls

## 3 Farbkonstanz

Der Mensch verfügt über die Fähigkeit, die Farben eines Objekts unter verschiedenen Lichtverhältnissen als dieselbe wahrzunehmen [8]. Dieses als Farbkonstanz bekannte Phänomen lässt sich jedoch nicht ohne Weiteres auf den Rechner übertragen. Im Folgenden werden Verfahren zur Farbnormierung (s. Abschnitt 3.1) und Farbkalibrierung (s. Abschnitt 3.2) vorgestellt, die das aus den Beleuchtungsänderungen resultierende Problem reduzieren bzw. eliminieren.

### 3.1 Farbnormierung

Ziel der Farbnormierung ist es, eine Szene mit unbekannter Beleuchtung in eine beleuchtungsinvariante Szene zu transformieren [2, 5]. Zur Farbnormierung existieren

zahlreiche Ansätze [3]. Eine detaillierte Beschreibung der in Abschnitt 4 evaluierten Verfahren findet sich in [8, S. 63 – 70].

*Chromatizitätsnormierung* Die Chromatizitätsnormierung errechnet aus einer gegebenen Szene ein von der Beleuchtungsintensität unabhängiges Bild (vgl. z. B. [4]), indem jeder Farbpixel durch seine Intensität dividiert wird.

*CCN (Comprehensive Color Normalization)* Das in [4] vorgestellte, iterative Verfahren erweitert die Chromatizitätsnormierung, indem es aus einem Bild sowohl den Einfluß der Beleuchtungsgeometrie als auch den der Beleuchtungsfarbe auf die Szene entfernt.

*Pomierski* Das physiologisch motivierte Verfahren nach Pomierski aus [9] betrachtet die im Bild vorkommenden Farbwerte als Punktwolke. In Anlehnung an die Grauweltannahme [8, S. 105] wird hierzu die erste Hauptachse des Farbclusters auf die Weiß-Schwarz-Achse (WB-Achse) im Gegenfarbraum rotiert.

*Rotation im RGB* Das in [8, S. 106 ff.] vorgestellte Verfahren lehnt an das Verfahren aus [9] an. Die erste Hauptkomponente des Farbclusters wird hierbei jedoch nicht auf die WB-Achse des physiologisch motivierten Raums, sondern auf die Hauptdiagonale des RGB-Raums rotiert. Im Allgemeinen liefert dieser Ansatz wie das Verfahren von Pomierski für den menschlichen Betrachter intuitivere Ergebnisse.

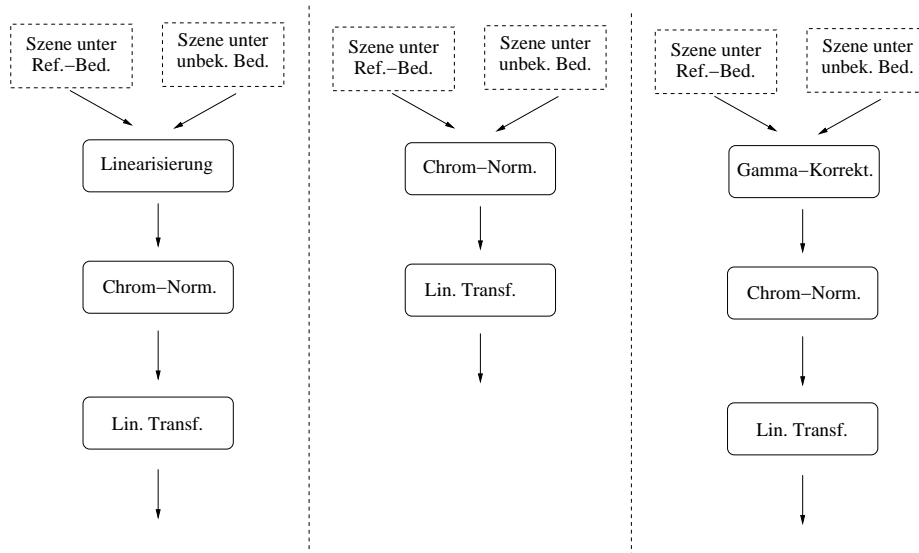
### 3.2 Farbkalibrierung

Die in diesem Paper verwendete Farbkalibrierung dient dazu, ein Bild einer Szene mit unbekannter Beleuchtung in ein Bild mit einer zuvor definierten Beleuchtungssituation zu überführen. Für die Kalibrierung werden Farbcharts eingesetzt, die aus Farbflächen bzw. Farbpatches bestehen. Die Farbwerte und spektralen Reflektivitäten der einzelnen Farbpatches sind bekannt oder können mittels eines Spektralphotometers bestimmt werden. Diese Angaben beziehen sich auf eine bestimmte Beleuchtungssituation. Das allgemeine Vorgehen besteht darin, eine Szene, welche ein bekanntes Kalibrieremuster enthält, unter einer unbekanntem Beleuchtung aufzunehmen. Aus den bekannten Informationen der Charts und den ermittelten Farbwerten des Charts aus dem aufgenommenen Bild wird eine Transformationsmatrix berechnet. Mit Hilfe dieser Transformationsmatrix wird das Bild in ein Bild mit bekannter Beleuchtung überführt.

Die in diesem Paper verwendeten Verfahren basieren auf [7], in dem eine Farbkalibrierung mittels linearer Transformation vorgestellt wird. Im weiteren Verlauf wird nicht auf das Verfahren aus [7] näher eingegangen, da dieses den Rahmen des Beitrags sprengen würde.

Für die Lokalisation des Balls wird die in Abb. 2 aufgeführte, auf [7] basierende Verarbeitungskette zugrundegelegt. Die lineare Transformation wird zum einen mit einer reinen Chromatizitätsnormierung, zum anderen mit einer RGB-Linearisierung mit anschließender Chromatizitätsnormierung eingesetzt. Weiter wird an Stelle der Linearisierung die Gammakorrektur mit der Chromatizitätsnormierung kombiniert.

Es wurde ein eigenes Kalibrieremuster entwickelt, welches auf dem Roboter montiert wurde, damit es in jedem aufgenommenen Bild enthalten ist (vgl. Abb. 5). Die



**Abb. 2.** Angepasste Verarbeitungskette zur Farbkalibrierung

Farbpatches wurden unter der Standardbeleuchtung D50 mit einem Spektralphotometer vermessen<sup>3</sup>. Zwar sind zahlreiche (standardisierte) Farbkalibriermuster von verschiedenen Herstellern erhältlich, allerdings haben sich diese auf Grund der Konstruktion des Roboters mit dem omnidirektionalen System als ungeeignet erwiesen. Das selbst entwickelte Kalibriermuster enthält u. A. eine Graupatches und weitere Farbpatches, die das komplette sichtbare Spektrum möglichst gut repräsentieren sollen.

## 4 Experimente

Für die Experimente wurde der Roboter wie in Abb. 3 an vier Positionen im Flur mit unterschiedlichen Beleuchtungsbedingungen plaziert. An jeder der vier Positionen wurde ein roter, matter Ball in unterschiedlichen Entfernungen (1m hinter, 0m, 1m, 2m und 3m vor dem Roboter) und Ausrichtungen (links, mittig, rechts) zum Roboter positioniert (s. Abb. 4).

Insgesamt wurden  $4 \cdot 5 \cdot 3 = 60$  Szenenbilder für die Evaluation des Einflusses verschiedener Farbnormierungs- und Farbkalibrierungsverfahren auf die Lokalisation des Balls in der Szene eingesetzt (vgl. Abb. 5). Die aufgenommenen Bilder werden dabei nach der in Abb. 1 aufgeführten Verarbeitungskette in Kombination mit den in Abschnitt 3.1 und 3.2 vorgestellten Verfahren bearbeitet.

Zur Bewertung der Güte der Lokalisation wird die Erkennungsrate des Balls in der Szene in Abhängigkeit von der Distanz des Balls zum Roboter ermittelt. Die Ergebnisse

<sup>3</sup> Dank an Herrn Prof. W. Slowak, Fachhochschule Koblenz

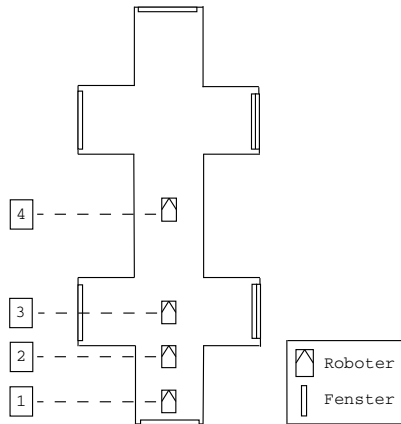


Abb. 3. Positionen des Roboters im Flur

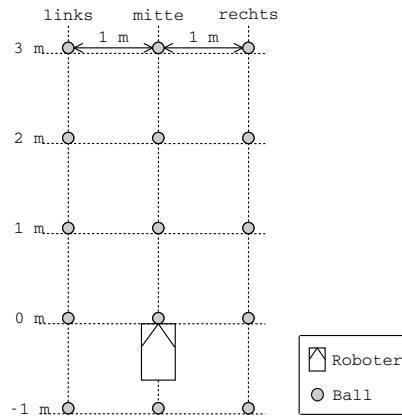


Abb. 4. Relative Ballposition zum Roboter

der Experimente sind in Abb. 6 und 7 jeweils separat nach der Ausrichtung des Balles aufgeführt. Dabei ist auf der  $x$ -Achse die Entfernung des Balles von der Nulllinie des Roboters (Abb. 4) in Meter aufgetragen. Auf der  $y$ -Achse ist die durchschnittliche Anzahl der erfolgreichen Lokalisationen an den vier Positionen im Flur in Abhängigkeit von der Ballposition aufgeführt.

#### 4.1 Experimente zur Farbnormierung

Die Diagramme aus Abb. 6 stellen die Ergebnisse der getesteten Verfahren für die drei möglichen Ausrichtungen des Balles dar.

Die geringsten Erkennungsraten weisen die Farbclusterrotationen im RGB- und Gegenfarbraum auf, wobei hier die Ergebnisse teilweise sogar schlechter ausfallen, als wenn keine Farbnormierung eingesetzt wird.

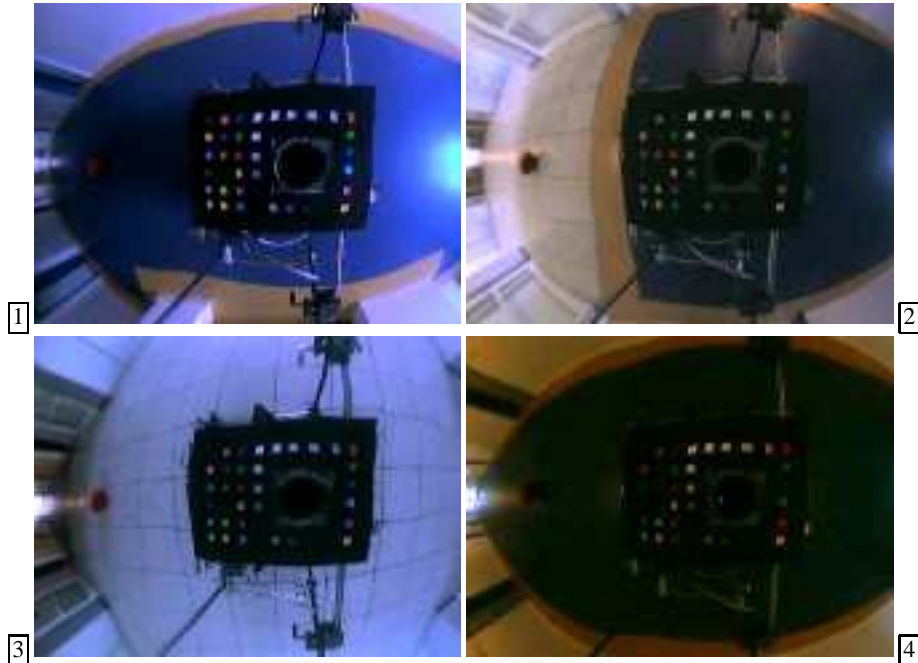
Bessere Ergebnisse liefert die Chromatizitätsnormierung, da diese den Einfluß der Beleuchtungsintensität auf die Szene reduziert.

Die besten Ergebnisse liefert das CCN-Verfahren, da dieses im Vergleich zur Chromatizitätsnormierung zusätzlich den Einfluß der Beleuchtungsfarbe auf die Szene berücksichtigt. Für alle drei Ballausrichtungen liefert letztgenanntes Verfahren die maximale Trefferquote. Damit kann ein in der Umgebung beliebig positionierter Ball mittels eines omnidirektionalen Systems stets gefunden werden.

#### 4.2 Experimente zur Farbkalibrierung

Die Diagramme aus Abb. 7 repräsentieren die Ergebnisse der Balldetektion mittels Farbkalibrierung<sup>4</sup>.

<sup>4</sup> Die hier aufgeführte Chromatizitätsnormierung und die Chromatizitätsnormierung der Farbnormierung sind verschieden.



**Abb. 5.** Beispiel-Eingabebilder an den vier Positionen

Die Linearisierung liefert die schlechteste Erkennungsrate, welche allerdings besser ist als diejenige, die ohne Farbkalibrierung erreicht wird. Ein eindeutiger Zusammenhang zwischen Erkennungsrate und Ballabstand oder Ballausrichtung ist bei diesem Verfahren nicht erkennbar.

Mittels der Chromatizitätsnormierung erhält man sehr gute Ergebnisse, falls sich der Ball weniger als 1m vor dem Roboter befindet. Mit zunehmender Entfernung nimmt jedoch die Erkennungsrate teilweise drastisch ab.

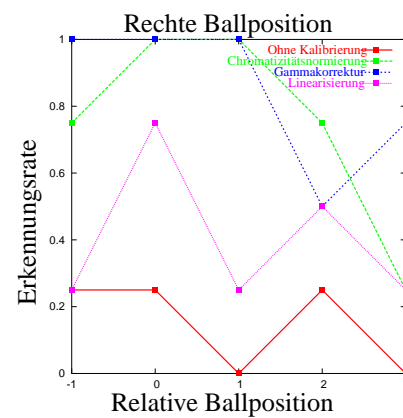
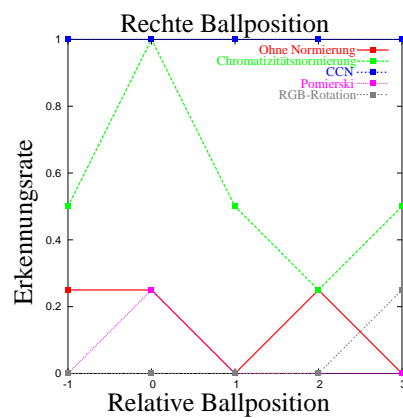
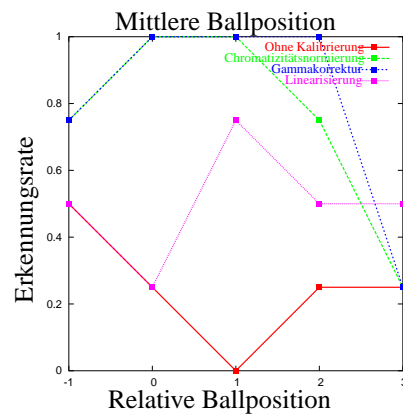
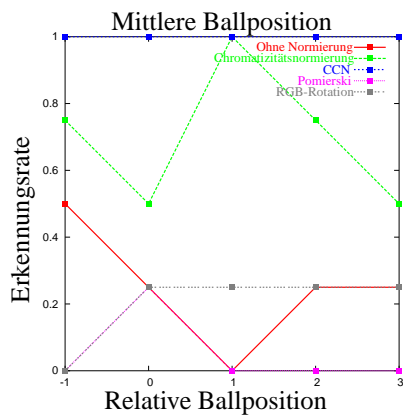
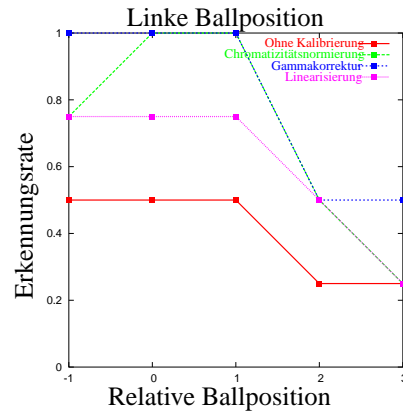
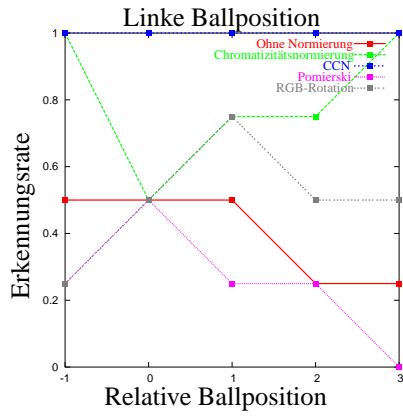
Die Gammakorrektur liefert ähnliche Ergebnisse wie die Chromatizitätsnormierung, jedoch sind die Lokalisationsraten für Entfernungen größer als 1m in der Regel besser als die der Chromatizitätsnormierung.

## 5 Fazit

In diesem Paper wurde der Einfluß verschiedener Verfahren zur Farbnormierung und Farbkalibrierung auf die Lokalisation eines roten, matten Balls in einem Flur untersucht.

Die Evaluation hat ergeben, daß die insgesamt besten Ergebnisse das CCN-Verfahren aus [4] liefert.

Beim Vergleich der Farbnormierungsverfahren fällt auf, daß die Farbclusterrotationsverfahren aus [8] und [9] bei der Objekterkennung deutlich schlechter abschneiden als die anderen Verfahren.



**Abb. 6.** Lokalisation mittels Farbnormierung

**Abb. 7.** Lokalisation mittels Farbkalibrierung

Bei den Farbkalibrierungsverfahren liefern die Gammakorrektur und die Chromatizitätsnormierung ähnlich gute Ergebnisse.

Zur Steigerung des Einflusses der Farbkalibrierungsverfahren auf die Lokalisation eines Objekts wäre eine weitere Evaluation hinsichtlich der Materialien der Farbcharts und der Komposition des Kalibrieremusters sinnvoll.

Da sich die vorgestellten Ergebnisse auf ein anfangs festgelegtes, eindeutig wiedererkennbares Objekt beziehen, ist in zukünftiger Arbeit zu erproben, wie die Ergebnisse der Experimente für die Lokalisation eines komplexeren Objekts ausfallen.

## Literatur

1. Dirk Balthasar, B Erdmann, Johannes Pellenz, Volker Rehrmann, J. Zeppen, and Lutz Priebe. Real-time detection of arbitrary objects in alternating industrial environments. In *Proceedings Scandinavian Conference on Image Analysis*, 1:321–328, 6 2001.
2. Kobus Barnard, Vlad Cardei, and Brian V. Funt. A comparison of computational color constancy algorithms – part i: Methodology and experiments with synthesized data. In *Transactions on Image Processing*, number 9/11. IEEE Computer Society, 2002.
3. Detlev Droege, Vinh Hong, and Dietrich Paulus. Farbnormierung auf langen Bildfolgen. In Karl-Heinz Franke, editor, *9. Workshop Farbbildverarbeitung*, pages 107–112, Ilmenau, 2003. Zentrum für Bild- und Signalverarbeitung e.V.
4. Graham D. Finlayson, Bernt Schiele, and James L. Crowley. Comprehensive colour image normalization. *Lecture Notes in Computer Science*, 1406:475–490, 1998. 1406.
5. Graham D. Finlayson and Ruixia Xu. Non-iterative comprehensive normalisation. In *Proceedings of the First international Conference CGIV*, pages 159–163, Poitiers, France, 4 2002. The Society for Imaging Science and Technology. Uni Koblenz Bib Signatur: INF 2003/4765.
6. Matthias Hagler. Reaktive hindernisvermeidung für einen mobilen roboter. Master's thesis, Technische Universität Graz, 2004.
7. Jochen Michel. Farbkalibrierung mittels linearer Transformation. Studienarbeit, 2004.
8. Dietrich Paulus. *Aktives Bildverstehen*. Der Andere Verlag, Osnabrück, 2001.
9. T. Pomierski and H. M. Groß. Verfahren zur empfindungsgemäßen Farbumstimmung. In G. Sagerer, S. Posch, and F. Kummert, editors, *Musterverkennung 1995*, pages 473–480, Berlin, 9 1995. Springer Verlag.
10. Michael J. Swain and Dana H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 11 1991.



## Nonparametric Noise Removal in Color Images

Bogdan Smolka \*

Polish-Japanese Institute of Information Technology  
Koszykowa 86, 02-008, Warsaw  
bsmolka@pjwstk.edu.pl

**Abstract.** In this paper the problem of nonparametric impulsive noise removal in multichannel images is addressed. The proposed filter class is based on the nonparametric estimation of the density probability function in a sliding filter window. The obtained results show good noise removal capabilities and excellent structure preserving properties of the new impulsive noise reduction technique.

### 1 Introduction

The majority of the nonlinear, multichannel filters are based on the ordering of vectors in a sliding filter window. The output of these filters is defined as the lowest ranked vector according to a specific vector ordering technique.

Let the color images be represented in the commonly used RGB color space and let  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$  be  $N$  samples from the sliding filter window  $W$ . Each of the  $\mathbf{x}_i$  is an  $m$ -dimensional multichannel vector, (in our case  $m = 3$ ). The goal of the vector ordering is to arrange the set of  $N$  vectors  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$  belonging to  $W$  using some sorting criterion.

In [1, 2] the ordering based on the cumulative distance function  $R(\mathbf{x}_i)$  has been proposed:  $R(\mathbf{x}_i) = \sum_{j=1}^N \rho(\mathbf{x}_i, \mathbf{x}_j)$ , where  $\rho(\mathbf{x}_i, \mathbf{x}_j)$  is a function of the distance among  $\mathbf{x}_i$  and  $\mathbf{x}_j$ . The ordering of the scalar quantities according to  $R(\mathbf{x}_i)$  generates the ordered set of vectors. The most commonly used measure to quantify distance between two multichannel signals is the Minkowski norm  $\rho_\gamma(\mathbf{x}_i, \mathbf{x}_j) = [\sum_{k=1}^m |x_{ik} - x_{jk}|^\gamma]^{1/\gamma}$ . The Minkowski metric includes the city-block distance ( $\gamma = 1$ ), Euclidean distance ( $\gamma = 2$ ) and chess-board distance ( $\gamma = \infty$ ) as the special cases.

One of the most important noise reduction filter is the vector median. In the case of gray scale images, given a set  $W$  containing  $N$  samples, the median of the set is defined as  $x_{(1)} \in W$  such that

$$\sum_j |x_{(1)} - x_j| < \sum_j |x_i - x_j|, \quad \forall x_i, x_j \in W. \quad (1)$$

Median filters exhibit good noise reduction capabilities, (especially when long tailed distribution noise is involved) and outperform simple nonadaptive linear filters in preserving signal discontinuities. As in many applications the signal is multidimensional,

---

\* This research has been supported by a grant No PJ/B/01/2004 from the Polish-Japanese Institute of Information Technology

in [3] the *Vector Median Filter* (VMF) was introduced, by generalizing the definition (1) using a suitable vector norm. Given a set  $W$  of  $N$  vectors, the vector median of the set is defined as  $\mathbf{x}_{(1)} \in W$  satisfying

$$\sum_j \|\mathbf{x}_{(1)} - x_j\| < \sum_j \|\mathbf{x}_i - \mathbf{x}_j\|, \quad \forall \mathbf{x}_i, \mathbf{x}_j \in W. \quad (2)$$

The orientation difference between two vectors can also be used as their distance measure. This so-called vector angle criterion is used by the *Vector Directional Filters* (VDF), to remove vectors with atypical directions, [4]. The *Basic Vector Directional Filter* (BVDF) is a ranked-order, nonlinear filter which parallelizes the VMF operation. However, a distance criterion based on the angles between vectors is utilized. To improve the efficiency of the directional filters, another method called *Directional-Distance Filter* (DDF) was proposed. This filter retains the structure of the BVDF, but utilizes the combined distance criterions to order the vectors inside the processing window, [4, 5].

## 2 Nonparametric Estimation

Applying statistical pattern recognition techniques requires the estimation of the probability density function of the data samples. Nonparametric techniques do not assume a particular form of the density function since the underlying density of the real data rarely fits common density models.

*Nonparametric Density Estimation* is based on placing a kernel function on every sample and on the summation of the values of all kernel function values at each point in the sample space, [6]. The nonparametric approach to estimating multichannel densities can be introduced by assuming that the color space occupied by the multichannel image pixels is divided into  $m$ -dimensional hypercubes. If  $h_N$  is the length of an edge of a hypercube, then its volume is given by  $V_N = h_N^m$ . If we are interested in estimating the number of pixels falling in the hypercube of volume  $V_N$ , then we can define the window function  $\phi(\mathbf{x}_i) = 1$ , if  $|x_{ij}| \leq 1/2, j = 1, \dots, m$  and 0 otherwise, which defines a unit hypercube centered in the origin.

The function  $\phi(\|\mathbf{x} - \mathbf{x}_i\|/h_N)$  is equal to unity if the pixel  $\mathbf{x}_i$  falls within the hypercube  $V_N$  centered at  $\mathbf{x}$  and is zero otherwise. The number of pixels in the hypercube with the length of edges equal to  $h_N$  is then  $k_N = \sum_{i=1}^N \phi(\|\mathbf{x} - \mathbf{x}_i\|/h_N)$  and the estimate of the probability that a sample  $\mathbf{x}$  is within the hypercube is  $p_N = k_N/NV_N$ , which gives

$$p_N(\mathbf{x}) = (NV_N)^{-1} \sum_{i=1}^N \phi(\|\mathbf{x} - \mathbf{x}_i\|/h_N). \quad (3)$$

This estimate can be generalized by using a smooth kernel function  $K$  in place of  $\phi(\cdot)$  and the width parameter  $h_N$  satisfying:  $K(\mathbf{x}) = K(-\mathbf{x})$ ,  $K(\mathbf{x}) \geq 0$ ,  $\int K(\mathbf{x}) d\mathbf{x} = 1$  and  $\lim_{N \rightarrow \infty} h_N = 0$ ,  $\lim_{N \rightarrow \infty} h_N^m = \infty$ .

The multivariate estimator in the  $m$ -dimensional case is defined as

$$p_N^*(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^N \frac{1}{h_1 \dots h_m} \mathcal{K} \left( \frac{|x_1 - x_{i1}|}{h_1}, \dots, \frac{|x_m - x_{im}|}{h_m} \right), \quad (4)$$

with  $\mathcal{K}$  denoting a multidimensional kernel function  $\mathcal{K} : \mathbb{R}^m \rightarrow \mathbb{R}$ ,  $h_1, \dots, h_m$  denoting bandwidths for each dimension and  $N$  being the number of samples in  $W$ . A common approach to build multidimensional kernel functions is to use a *product kernel*  $\mathcal{K}(u_1, \dots, u_m) = \prod_{i=1}^m K(u_i)$ , where  $K$  is a one-dimensional kernel function

$$p_N^*(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^N \prod_{j=1}^m \left( \frac{|x_{ij} - x_j|}{h_j} \right). \quad (5)$$

The shape of the approximated density function depends heavily on the bandwidth chosen for the density estimation. Small values of  $h$  lead to spiky density estimates showing spurious features. On the other hand, too big values of  $h$  produce over-smoothed estimates that hide structural features.

If we chose the Gaussian kernel, then the density estimate of the unknown probability density function at  $\mathbf{x}$  is obtained as a sum of kernels placed at each sample  $\mathbf{x}_i$

$$p_N(\mathbf{x}, h) = \frac{1}{N (h\sqrt{2\pi})^m} \sum_{i=1}^N \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{2h^2}\right). \quad (6)$$

The smoothing parameter  $h$  depends on the local density estimate of the sample data. The form of the data dependent smoothing parameter is of great importance for the non-parametric estimator.

Choosing the Gaussian kernel function for  $\mathcal{K}$ , the optimal bandwidth is

$$h^* = (4/(m+2))^{-\frac{1}{m+4}} \hat{\sigma} N^{-\frac{1}{m+4}}, \quad (7)$$

where  $\sigma$  denotes the approximation of the standard deviation of the samples. In one dimensional case (7) reduces to the well known, 'rule of thumb',  $h^* = 1.06N^{-\frac{1}{5}}\hat{\sigma}$ , [6]. A version which is more robust against outliers in the sample set can be constructed if the interquartile range is used as a measure of spread instead of the variance. This modified estimator is  $h^* = 0.79\varrho N^{-\frac{1}{5}}\hat{\sigma}$ , where  $\varrho$  is the inter-quartile range. Another robust estimate of the optimal bandwidth is  $h^* = 0.9AN^{-\frac{1}{5}}\hat{\sigma}$  with  $A = \min(\hat{\sigma}, \varrho/1.34)$ . Generally the simplified rule of choosing the optimal bandwidth  $h$  can be written as

$$h_1^* = C \hat{\sigma} N^{-\frac{1}{m+4}}, \quad (8)$$

where  $C$  is an appropriate weighting coefficient.

From the maximum likelihood principle and assuming independence of the samples, one can write the likelihood of drawing the complete dataset as the product of the densities of one sample

$$\mathcal{L}(h) = \prod_{j=1}^N p_N(\mathbf{x}_j, h) = \prod_{j=1}^N \frac{1}{N} \sum_{i=1}^N \frac{1}{(h\sqrt{2\pi})^m} \exp\left(-\frac{\|\mathbf{x}_j - \mathbf{x}_i\|^2}{2h^2}\right). \quad (9)$$

As this likelihood function has a global maximum for  $h=0$ , in [7] a modified approach has been proposed

$$\mathcal{L}^*(h) = \left[ \prod_{j=1}^N \frac{1}{N} \sum_{i=1, i \neq j}^N \frac{1}{(h\sqrt{2\pi})^m} \exp\left(-\frac{\|\mathbf{x}_j - \mathbf{x}_i\|^2}{2h^2}\right) \right]^{\frac{1}{m}}. \quad (10)$$

This function has one maximum for  $h$ , which can be found by setting to 0 the derivative of the logarithm of  $\mathcal{L}^*(h)$  with respect to  $h$ . A crude but rather fast way to obtain an approximate solution hereof is by assuming that the density estimate of Eq. (5) on a certain location  $\mathbf{x}$  in the feature space is determined by the nearest kernel only, [7]. In this paper we use the optimal  $h$  derived from (10) defined as

$$h_2^* = C \left( (mN)^{-1} \sum_{j=1}^N \|\tilde{\mathbf{x}}_j - \mathbf{x}_j\|^2 \right)^{\frac{1}{2}}, \quad (11)$$

where  $\tilde{\mathbf{x}}_i$  represents the nearest neighbor of the sample  $\mathbf{x}_i$ , and  $C$  is a tuning parameter.

### 3 Proposed Algorithm

Let us assume a filtering window  $W$  containing  $N$  image pixels,  $\{x_1, \dots, x_N\}$  and let us define the similarity function  $\mu : [0; \infty) \rightarrow \mathbb{R}$  which is non-ascending and convex in  $[0; \infty)$  and satisfies  $\mu(0) = 1$ ,  $\mu(\infty) = 0$ . The similarity between two pixels of the same intensity should be 1, and the similarity between pixels with minimal and maximal gray scale values should be very close to 0. The function  $\mu(x_i, x_j)$  defined as  $\mu(x_i, x_j) = \exp\{-[(x_i - x_j)/h]^2\}$ , where  $h$  is the bandwidth of the Gaussian kernel, defined by (8) or (11), satisfies the required conditions.

Let us additionally define the cumulated sum  $M$  of similarities between a given pixel and all other pixels belonging to window  $W$ . For the central pixel  $x_1$  we introduce  $M_1$  and for the neighbors of  $x_1$  we define  $M_k$  as

$$M_1 = \sum_{j=2}^N \mu(x_1, x_j), \quad M_k = \sum_{j=2, j \neq k}^N \mu(x_k, x_j), \quad k > 1, \quad (12)$$

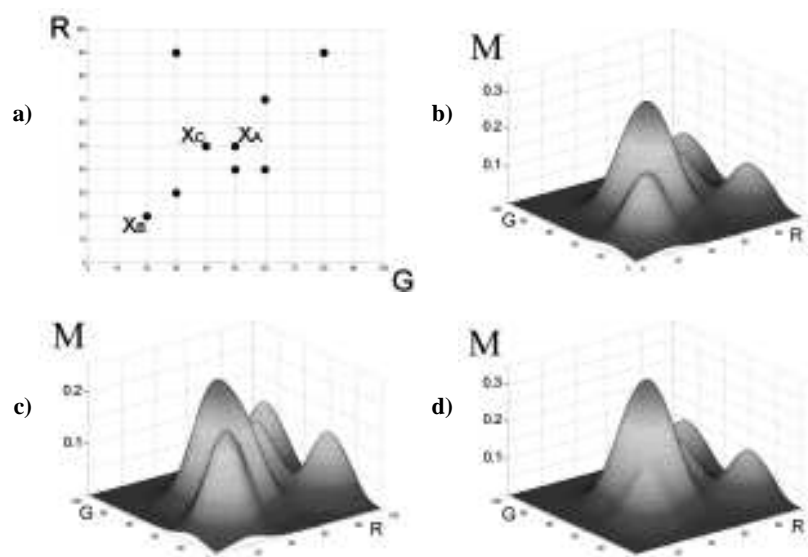
which means that for  $x_k$ , which are neighbors of  $x_1$ , we do not take into account the similarity between  $x_k$  and  $x_1$ , which is the main idea of this algorithm. The omission of the similarity  $\mu(x_k, x_1)$  when calculating  $M_k$ , privileges the central pixel, as in the calculation of  $M_1$  we have  $N - 1$  similarities  $\mu(x_1, x_k)$ ,  $k > 2$  and for  $M_k$ ,  $k > 1$  we have only  $N - 2$  similarity values, as the central pixel  $x_1$  is excluded from the calculation of  $M_k$ , [8, 9].

In the construction of the new filter, the reference pixel  $x_1$  in the window  $W$  is replaced by one of its neighbors if  $M_1 < M_k$ ,  $k = 2, \dots, N$ . If this is the case, then  $x_1$  is replaced by that  $x_{k^*}$  for which  $k^* = \arg \max M_k$ ,  $k = 2, \dots, N$ . In other words  $x_1$  is detected as being corrupted if  $M_1 < M_k$ ,  $k = 2, \dots, N$  and is replaced by its neighbors  $x_k$  which maximizes the sum of similarities  $M$  between all the pixels from  $W$  excluding the central pixel.

The basic assumption is that a new pixel must be taken from the window  $W$ , (introducing pixels, that do not occur in the image is prohibited like in the VMF). For this purpose  $\mu$  must be convex, which means that in order to find a maximum of the sum of similarity functions  $M$  it is sufficient to calculate the values of  $M$  only in points  $x_1, x_2, \dots, x_N$ .

The presented approach can be applied in a straightforward way to multichannel images using the similarity function defined as  $\mu(\mathbf{x}_i, \mathbf{x}_j) = \exp\{-[\|\mathbf{x}_i - \mathbf{x}_j\|/h]^2\}$ ,

where  $\| \cdot \|$  denotes the specific vector norm and  $h$  denotes the bandwidth. Now in exactly the same way we can maximize the total similarity function  $M$  for the vector case.



**Fig. 1.** Impulsive noise removal technique in the 2D case.

The working scheme of the new filter is presented in Fig. 1 for the two-dimensional data. Fig. 1 a) depicts the arrangement of pixels in  $W$  and Fig. b) their nonparametric probability density estimation. Figs. c) and d) present the density plots for the cases when the central pixels  $x_A$  and  $x_B$  are removed from  $W$ . It can be seen that in the first case c) the pixel  $x_1 = x_A$  will be retained and in the second case d) the pixel  $x_1 = x_B$  will be replaced by  $x_A$ . The pixel  $x_A$  will be preserved, as in Fig. c) the plot attains its maximum at  $x_C$ , but this maximum is less than the maximum for  $x_A$  in Fig. b). Regarding sample  $x_B$ , its rejection causes that the maximum is attained at  $x_A$  and this pixel will replace the central pixel  $x_B$ .

## 4 Results

The performance of the proposed impulsive noise reduction filters was evaluated using the widely used PSNR quality measure. Figure 2a) shows the dependence of the noise attenuation capability of the proposed filter class on the bandwidth type  $h_1^*$  and  $h_2^*$  defined by (8) and (11). Clearly the filter based on the  $h_2^*$  outperforms the technique based on the  $h_1$  bandwidth for the whole range of used contamination probabilities  $p$ , ( $p = 0.01 - 0.1$ ).

Figure 2b) presents the dependence of the PSNR restoration quality measure on the kind of the Minkowski norm. Surprisingly, the  $L_\infty$  norm yields significantly better results than the  $L_1$  or  $L_2$  norms. This is the result of the construction of the  $h_2^*$  bandwidth,

which depends on the nearest neighbor in the sliding filter window. This behavior is advantageous, as the calculation of the  $L_\infty$  norm is much faster than the evaluation of distances determined by  $L_1$ ,  $L_2$  norms.

The efficiency of the filters based on adaptive  $h_1^*$  and  $h_2^*$  bandwidths are dependent, (especially for very small noise contamination) on the coefficient  $C$  in (8) and (11). Figure 2c) shows the dependence of PSNR for the filter based on  $h_2^*$  as a function of  $C$  in (11). For low noise intensity the parameter  $C$  should be significantly larger than for the case of images corrupted by heavy noise process. However, setting  $C$  to 4 is an acceptable trade-off, as can be seen in Fig. 2 d), which depicts the efficiency of the proposed filter in comparison with VMF, AMF and BVDF. It can be observed that although the  $C = 4$  is not an optimal setting for the whole range of tested noise intensities, nevertheless the described filter yields much better results than the traditional techniques.

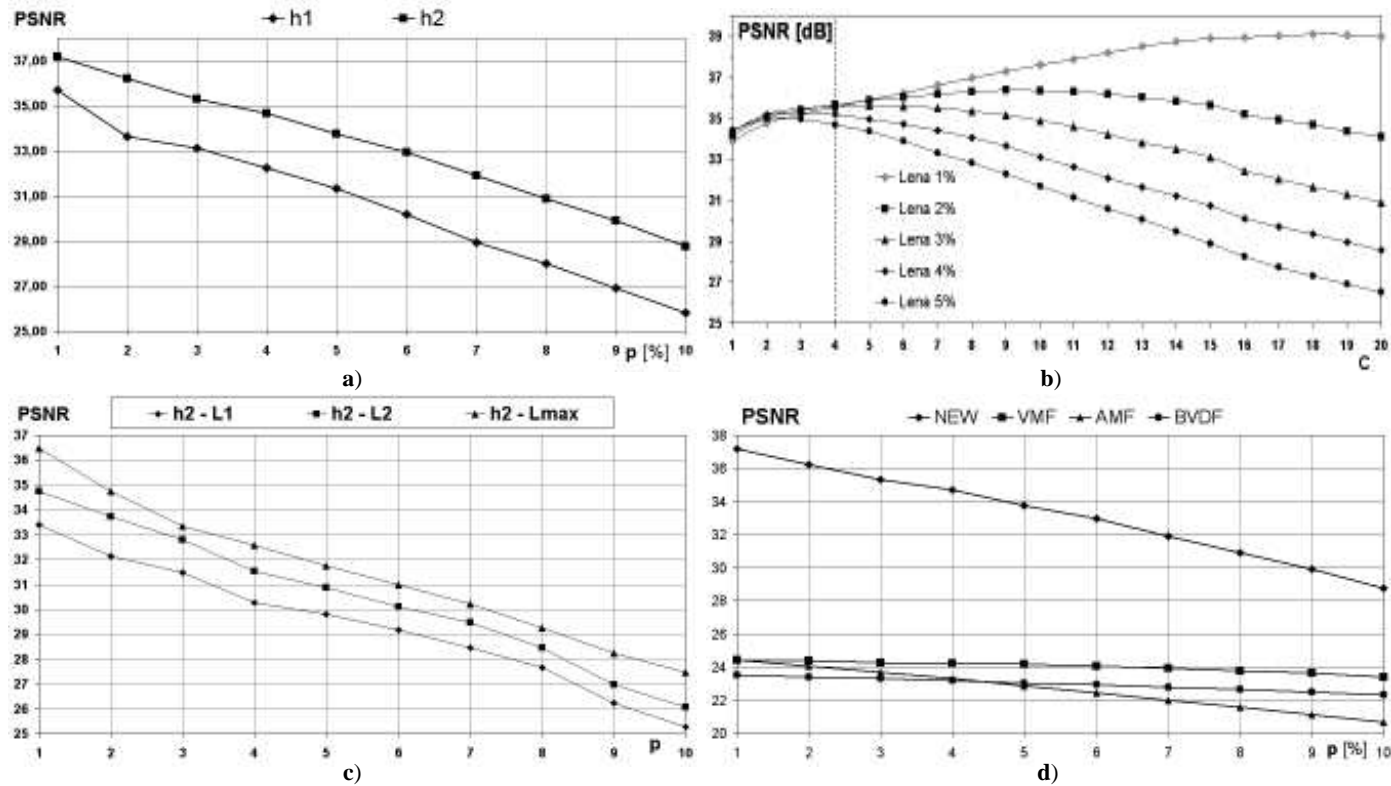
This is also testified by Fig. 3, which compares the filtering results obtained by the filter based on adaptive  $h_2^*$  bandwidth with the performance of the *reference* VMF, BVDF, DDF filter. As can be observed the new filtering has much better detail preserving properties than VMF, BVDF and DDF.

## 5 Conclusions

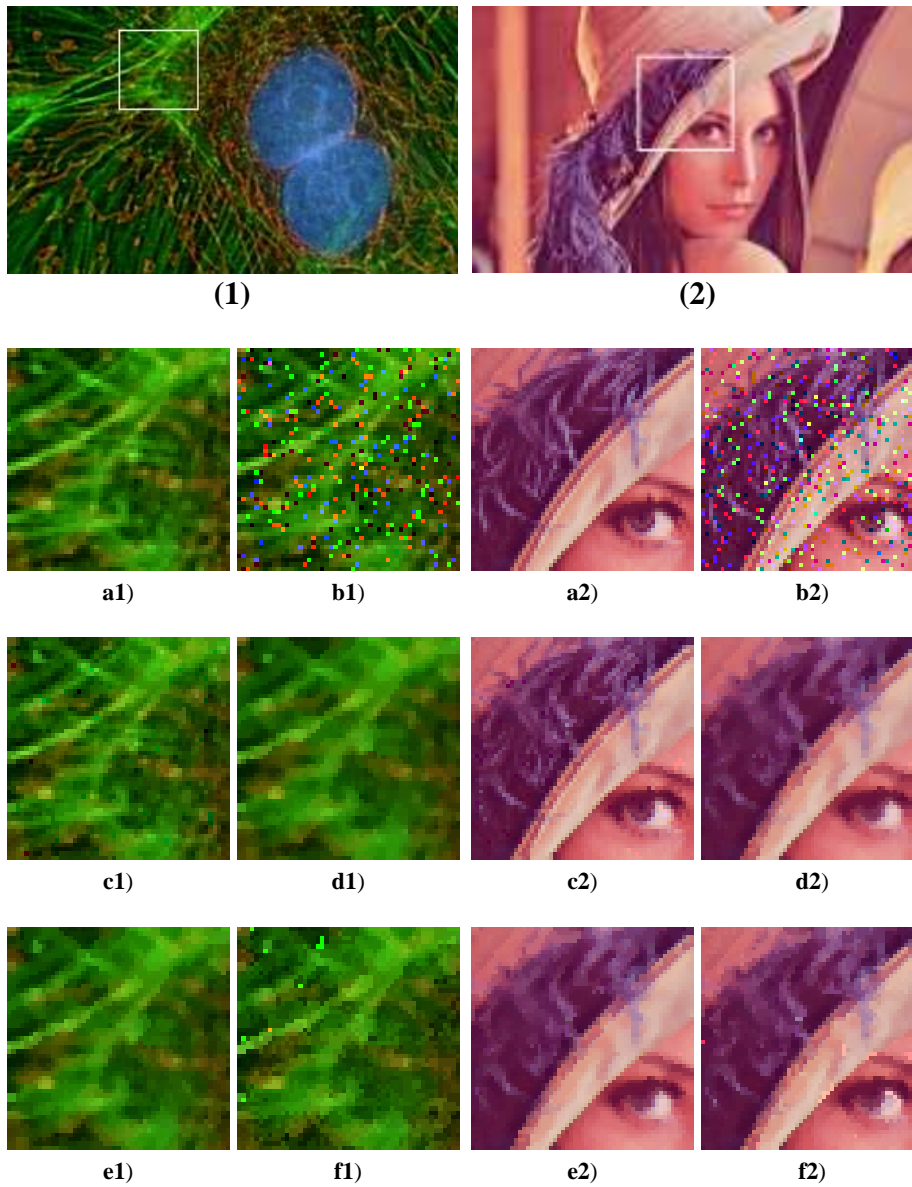
In this paper a new nonparametric technique of impulsive noise removal in multichannel images has been proposed. The described filter class is based on the estimation of the kernel bandwidth using the technique proposed in [7]. The experiments revealed, that the proposed algorithm yields the best results when applying the  $L_\infty$  norm, which makes the filter computationally very attractive. The obtained results show that the proposed technique excels significantly over the standard techniques like VMF, BVDF and DDF. The future work will focus on the automatic adjustment of the tuning parameter  $C$  in (8) and (11).

## References

1. I. Pitas, P. Tsakalides, Multivariate ordering in color image processing, IEEE Trans. on Circuits and Systems for Video Technology, 1, 3, 247-256, 1991.
2. K. Tang, J. Astola, Y. Neuvo, Nonlinear multivariate image filtering techniques, IEEE Trans. on Image Processing, 4, 6, 788-797, 1995.
3. J. Astola, P. Haavisto, Y. Neuvo, Vector median filters, Proceedings of the IEEE, 78, 678-689, 1990.
4. P.E. Trahanias, A.N. Venetsanopoulos, Vector directional filters: a new class of multichannel image processing filters, IEEE Trans. on Image Processing, 2, 4, 528-534, 1993.
5. K.N. Plataniotis, A.N. Venetsanopoulos, "Color Image Processing and Applications", Springer Verlag, August 2000.
6. D.W. Scott, "Multivariate Density Estimation", New York, John Wiley, 1992.
7. M.A. Kraaijveld, A Parzen classifier with an improved robustness against deviations between training and test data, Pattern Recognition Letters, 17, 679-689, 1996.
8. B. Smolka, K.N. Plataniotis, A. Chydzinski, M. Szczepanski, A.N. Venetsanopoulos, K. Wojciechowski, Self-adaptive algorithm of impulsive noise reduction in color images, Pattern Recognition, 35, 1771-1784, 2002.
9. B. Smolka, R. Lukac, A. Chydzinski, K.N. Plataniotis, K. Wojciechowski, Fast adaptive similarity based impulsive noise reduction filter, Real Time Imaging, 9, 261-276, 2003.



**Fig. 2.** Dependence of the efficiency of the proposed filtering scheme on  $h_1^*$  (8) and  $h_2^*$  (11) - (a), besides the dependence of the PSNR on the tuning parameter  $C$  in (11) - (b) and the dependence on the kind of Minkowski norm for the bandwidth  $h_2^*$  - (c). Figure (d) shows the comparison of results obtained using the  $h_2^*$  bandwidth,  $L_\infty$  norm and  $C = 4$  with the standard multichannel filters VMF and BVDF, (test were performed on the color image *LENA*);  $p$  denotes the probability of a pixel corruption - to RGB channels random, uniformly distributed values from the interval [0,255] were assigned.



**Fig. 3.** Illustrative example of the efficiency of the proposed algorithm: **a)** zoomed parts of the test color images, **b)** image corrupted by 3% of impulsive noise, **c)** image after filtering with the proposed filter, **d)** VMF output, **e)** DDF output, **f)** BVDF output.



# Bestimmungshilfe für Mollusken und Eulenfalter mittels digitaler Bildanalyse

Fabian Fritzer

Universität Koblenz-Landau

**Zusammenfassung.** Zur Bestimmung von Tier- und Pflanzenarten arbeitet das Unternehmen science4you derzeit an digitalen Bestimmungshilfen, die in Form von Bestimmungs-CDs oder als Internetanwendungen zur Verfügung stehen sollen. Die Merkmale der Tiere sollen anhand von mehreren tausend bereits vorhandenen Farbbildern automatisch ermittelt und zunächst als Rohdaten in einen xml-Reader, dann in eine Datenbank geschrieben werden. Dieser Artikel befasst sich mit einigen Algorithmen, die bei der Bildvorverarbeitung und der Merkmalsextraktion zum Einsatz kommen.

## 1 Motivation

Der Begriff „Mollusken“ bezeichnet Weichtiere, d. h. alle Schnecken und Muscheln und unter „Eulenfalter“ versteht man eine Nachtflatterfamilie. Viele auf den ersten Blick ähnliche Arten (456 Eulenfalter allein in Baden-Württemberg) sind festzustellen mit einem Blick allein ist eine Erkennung nicht möglich. Item Beobachtungen von Tieren und Pflanzen sind ohne Bestimmung für den Wissenschaftler nicht verwertbar und für den Anfänger weniger interessant. Daher ergibt sich der Bedarf einer automatischen Bestimmungshilfe.

Bislang verwenden Biologen Bestimmungsbücher bzw. neuerdings Bestimmungsforen im Internet.<sup>1</sup> Das funktioniert gut und die Foren sind sehr beliebt; aber es gibt auch Nachteile (die an dieser Stelle nicht näher erläutert werden).

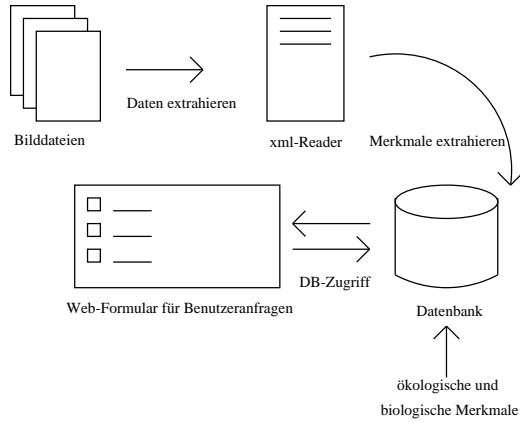
Abbildung 1 zeigt ein Architekturdiagramm der Bestimmungshilfe, wie sie im Rahmen einer Studienarbeit am Institut für Computervisualistik der Universität Koblenz-Landau entstand. Aufgabe war das Extrahieren der Daten aus den Bilddateien. Die restlichen Aufgaben werden vom Unternehmen science4you übernommen. Zur Programmierung wurde das PUMA-Framework (Programmierungsumgebung für die Musteranalyse) der Uni Koblenz eingesetzt. Ein Vorteil der computergestützten Bestimmung ist, dass gleichzeitig auf die ökologischen Merkmale (u.a. Höhenverbreitung, Flugzeit) und die Aussehensmerkmale zugegriffen werden kann.

Die Eingabebilder sind bezüglich mancher Eigenschaften bereits normiert, haben aber ihre „Tücken“: Z.B. bei den Mollusken Blitzlichtartefakte, die in Abbildung 2 zur Verdeutlichung schwarz dargestellt wurden.

Die Falterbilder haben einen Farbstich, der Hintergrund ist nicht exakt einfarbig, und die Bereiche (Vorderflügel, Hinterflügel, Körper) sind natürlich nicht eingezeichnet.

---

<sup>1</sup> [www.schmetterling-raupe.de](http://www.schmetterling-raupe.de), [www.nafoku.de](http://www.nafoku.de), [www.s2you.com](http://www.s2you.com)

**Abb. 1.** Gesamtarchitektur der Bestimmungshilfe**Abb. 2.** Beispielbild einer Schnecke**Abb. 3.** Beispielbild eines Eulenfalters

## 2 Vorverarbeitung

Die folgenden Schritte werden zur Vorverarbeitung der Bilder durchgeführt:

1. Blitzlichtartefakte (Mollusken)
2. Hintergrund vom Vordergrund trennen (v.a. Eulenfalter), da der Hintergrund nicht exakt einfarbig ist
3. Farbkorrektur
4. Bounding Box ermitteln

5. Bildbereiche erkennen (Eulenfalter): Ein Merkmal ist ohne Zuordnung zu den Vorder- bzw. Hinterflügeln nur halbsoviel wert
6. Downsampling
7. Histogramme erstellen

Von diesen Vorverarbeitungsschritten wird im Folgenden die Trennung von Hintergrund und Vordergrund vorgestellt.

## 2.1 Trennung von Hintergrund und Vordergrund



Abb. 4. ohne Flooding

Fast der gesamte Hintergrund der Eulenfalterbilder hat exakt die gleiche Hintergrundfarbe (RGB-Wert 96,129,196). Die Hintergrundpixel an der Grenze zum Objekt weichen von diesem Farbwert jedoch leicht ab, siehe Abbildung 4 (die exakt hintergrundfarbenen Pixel sind zur Verdeutlichung schwarz gemacht). Würde man bei der Histogrammerstellung also nur den exakten Farbwert 96,129,196 aussparen, dann würden viele Hintergrundpixel mit ins Histogramm aufgenommen. Ein einfacher Ansatz zur Lösung dieses Problems wäre es, einen ganzen Farbbereich als Hintergrund zu werten, d.h. bei der Histogrammerstellung zu ignorieren. Dadurch würden jedoch auch Pixel innerhalb des Flügels verlorengehen. Deshalb wurde der „Floodfill-Algorithmus“ eingesetzt, der ansonsten z.B. auch zur Wegsuche verwendet wird. Dieser Algorithmus „frisst sich von aussen ran“.

Die Variable „stapel“ in Abbildung 5 wurde mit den Koordinaten sämtlicher Nachbarpixel von fast exakt hintergrundfarbenen Pixeln initialisiert. „allowediff“ ist der Parameter, mit dem reguliert werden kann, wie groß der Farbabstand zur Hintergrundfarbe maximal sein darf, damit ein geflutetes Pixel noch als Hintergrundpixel zählt. „coldif- feuklidforflood“ ist ein einfaches Farbabstandsmaß. „isforeground“ ist eine zweidimensionale bool-Matrix von der Größe des Bilds. Zu jeder Pixelposition wird hier eingetragen, ob das dort befindliche Pixel zum Vordergrund zu rechnen ist oder nicht. Dadurch müssen die Farbwerte nicht manipuliert werden, was diesen ansonsten ziemlich aufwendigen Algorithmus stark beschleunigt.

WHILE stapel.size > 0	
y = stapel.pop_back()	
x = stapel.pop_back()	
IF	$x \geq xsize \vee x < 0 \vee y \geq ysize \vee y < 0$
THEN	continue
IF	!isforeground[x][y]
THEN	continue
IF	coldiffeuklidforfbod(x,y) <= alloweddiff
THEN	isforeground[x][y] = false stapel.push_back(x+1) stapel.push_back(y) stapel.push_back(x-1) stapel.push_back(y) stapel.push_back(x) stapel.push_back(y+1) stapel.push_back(x); stapel.push_back(y-1);

Abb. 5. Floodfill-Algorithmus

Das Ergebnis ist in fast allen Fällen zufriedenstellend, siehe Abbildung 6.



Abb. 6. nach Flooding

Anmerkung: Auch das Problem der Blitzlichtartefakte auf den Molluskenbildern wird auf ähnliche Art und Weise gelöst. Dort wird ausgehend von den rein weißen Pixeln (RGB-Wert 255,255,255) geflutet.

### 3 Bildauswertung und Dokumentation

Die folgenden Schritte sind zur Auswertung erforderlich:

1. Grundfarben, vorhandene Farben, ähnliche Farben
2. Kontrast ermitteln
3. Form ermitteln
4. Durchschnittsfarbe
5. Transparenz (nur Eulenfalter)

Bei den Eulenfaltern werden diese Schritte für Vorder- und Hinterflügel separat durchgeführt. Verwendet wurden RGB-Histogramme mit 8 Abschnitten pro Farbachse. Insgesamt werden also 512 Farben unterschieden. Im Folgenden wird das Ermitteln der Grundfarben und der vorhandenen Farben vorgestellt.

### 3.1 Grundfarben ermitteln

Ein Problem stellt es dar, eine mathematische Definition für „Grundfarbe“ zu finden. Eine Lösung besteht darin, eine Farbe (d. h. ein Bin im Histogramm) genau dann Grundfarbe zu bezeichnen, wenn sie mindestens 30 % der Pixelanzahl der häufigsten Farbe (d. h. der Bin mit der höchsten Pixelanzahl) hat.



**Abb. 7.** einfarbige Muschel *Unio crassus*

Bei mehr oder weniger einfarbigen Objekten verteilen sich die Pixel auf nur wenige Bins im Histogramm. Die Bin mit der höchsten Pixelanzahl ist stets hoch gefüllt; der resultierende Schwellwert (30 % dieser Pixelanzahl) ist also ebenfalls relativ hoch. Daraus folgt, dass einfarbige Objekte nur wenige Einträge für „Grundfarbe“ haben, z. B. die oben abgebildete Muschel *Unio crassus*. Das zugehörige Ergebnis im xml-Reader sieht folgendermaßen aus:

```
<maxcolor anteil="41">303030</maxcolor>
```

```
<basecolors count="3">
  <color anteil="17">101010</color>
  <color anteil="41">303030</color>
  <color anteil="11">505050</color>
</basecolors>
```

Bei bunten Objekten hingegen verteilen sich die vorhandenen Farben weiter über das Histogramm. Die Bin mit der höchsten Pixelanzahl ist niedrig gefüllt; der resultierende Schwellwert (30 % dieser Pixelanzahl) ist also ebenfalls relativ niedrig. Daraus folgt, dass bunte Objekte mehr Einträge für „Grundfarbe“ haben, z.B. die Muschel *Charonia tritonis* in Abbildung 8.

```
<maxcolor anteil="11">705030</maxcolor>
<basecolors count="13">
  <color anteil="4">503010</color>
  <color anteil="6">503030</color>
  <color anteil="3">505030</color>
  <color anteil="11">705030</color>
  <color anteil="5">705050</color>
  <color anteil="4">707050</color>
  <color anteil="8">907050</color>
  <color anteil="6">907070</color>
  <color anteil="4">909070</color>
  <color anteil="4">909090</color>
  <color anteil="4">B09070</color>
  <color anteil="3">B09090</color>
  <color anteil="3">B0B0B0</color>
</basecolors>
```

Bei Untersuchung der Farbwerte sieht man, dass Weiss, Rotbraun *und* Blauschwarz vertreten sind. Dies ist wichtig, denn diejenige Farbe, die später vom Benutzer der Bestimmungshilfe als Grundfarbe angegeben wird, muss auf jeden Fall auch als Grundfarbe im Reader und in der Datenbank stehen. Es gilt grundsätzlich, dass zu weit gefasste Kriterien besser sind als zu enge. Bei zu weit gefassten Kriterien hat man später im Ergebnis einer Benutzeranfrage zu viele in Frage kommende Arten, aber die gesuchte Art ist dabei. Bei zu eng gefassten Kriterien hingegen kann es passieren, dass die vom Benutzer gesuchte Art aus der Ergebnismenge eliminiert wird!

### 3.2 Vorhandene Farben ermitteln

Als Schwellwert für einen Eintrag bei den „vorhandenen Farben“ wurde nach empirischen Tests 0.4 % der Gesamtanzahl der im Histogramm erfassten Pixel gewählt. Bei einem höheren Schwellwert wäre der Farbwert D09050 (orange) in Bild 9 nicht in die vorhandenen Farben aufgenommen worden. Dieser niedrige Schwellwert erklärt sich daraus, dass kleine Farbbereiche mit abweichender Farbe dem menschlichen Betrachter stark auffallen, tatsächlich aber nur wenige Pixel einnehmen. Hinzu kommt, dass

sich diese Pixel oftmals auf mehrere Bins im Histogramm verteilen, da sie etwas unterschiedliche Farben haben. Da es nur sehr wenige Arten mit beispielsweise orangeroten Flecken im Vorderflügel gibt, ist dieses Merkmal ein guter Filter, der nur wenige der 456 Eulenfalter durchlässt. In einem Bestimmungsbuch die etwa vier Arten mit diesem Merkmal zu finden, bedeutet hingegen viel Hin- und Herblättern.



**Abb. 8.** Bunte Muschel *Charonia tritonis*



**Abb. 9.** Sehr wenige Arten haben für die Vorderflügel den Farbeintrag `< hascolor > D09050` (orange). Darunter auch die Steineule.

## Literatur

1. Srikanth Bandi and Daniel Thalmann. The use of space discretization for autonomous virtual humans. pages 336–337, 1998.
2. G. Buchsbaum. A spatial processor model for object colour perception. *Journal of the Franklin Institute*, 310:1–26, 1980.
3. M.P. Wand. Data-based choice of histogram bin width. *The American Statistician*, 51:59–64, 1997.

## An Efficient Vector Median Filter Computation

V. Hong<sup>1</sup>, L. Csink<sup>2</sup>, S. Bouattour<sup>1</sup>, D. Paulus<sup>1</sup>

<sup>1</sup> Institut für Computervisualistik, Universität Koblenz-Landau,  
Universitätsstr. 1, 56070 Koblenz – Germany,  
{hong,bouattour,paulus}@uni-koblenz.de

<sup>2</sup> John von Neumann Faculty of Informatics, Budapest Polytechnic,  
H1037 Budapest, Nagyszombat 19 – Hungary,  
csink.laszlo@nik.bmf.hu

**Abstract.** The vector median filter (VMF) is a useful and common tool for noise removal in color image processing. The drawback of the approach in its general form is the high computational cost if implemented naively. Indeed it takes  $O(n \times m \times m_s^4)$  distance evaluations for computing the filter response, where  $m_s$  is the mask size and  $n \times m$  the dimension of the image. In this contribution we propose a general algorithmic optimization framework for reducing the complexity to  $O(n \times m \times m_s^3)$ . We discuss the dependency of the VMF on the distance measure and demonstrate that for the square Euclidian distance, the VMF becomes a vector *near-mean* filter with just  $O(n \times m \times m_s^2)$  distance evaluations.

### 1 Introduction

By definition the VMF as used in [2, 3, 1] of an  $n \times m$  RGB image  $f$  is computed as follows: Go through the image with a square mask  $W$  of size  $m_s \times m_s$ . The VMF  $g$  of  $f$  at the center of the mask  $(i, j)$  is

$$g_{i,j} = f(i_{Min}, j_{Min}) \quad (1)$$

where  $(i_{Min}, j_{Min})$  is the position of the pixel with the minimal distance sum to all other pixels within the mask:

$$R_{i_{Min}, j_{Min}} = \min_{(\mu, \nu) \in W} R_{\mu, \nu} \quad (2)$$

where

$$R_{\mu, \nu} = \sum_{(k, l) \in W} \text{Distance}(f_{\mu, \nu}, f_{k, l}) \quad (3)$$

and *Distance* is the distance function between two pixels.

According to this definition, a mask of size  $m_s \times m_s$  is applied to each pixel in the image of size  $n \times m$ . For each pixel within the mask the distance between this pixel to all other pixels in the mask have to be computed. This costs  $O(m_s \times m_s)$  evaluations. This has to be repeated  $m_s^2$  times within the mask and over the whole image. VMF's computation needs thus  $O(n \times m \times m_s^4)$  distance evaluations. This high computational cost is the main drawback of the VMF, especially when it is applied to real time applications.



In the following we discuss two approaches to reduce the computational complexity. In section 2 we propose a general algorithmic framework for reducing the number of evaluations by preserving the computations performed in one position and using it in the following one. This leads to a reduction of the order of  $O(n \times m \times m_s^3)$ . In section 3 we discuss the dependency of VMF on a special distance measure: the square Euclidian distance, and demonstrate that the minimum of Eq. 2 can be computed analytically. This yields a smoothing filter which is actually the best approximation of the mean filter. We call this the vector near-mean filter. Section 4 summarizes the paper and describes future work.

## 2 Algorithmic Optimization

The most expensive part of the original VMF is the computation of the pixel distances. For each mask applying the distances between each pixel in the mask and its neighbours are computed completely new, although most of the distances are already computed before. In our approach we reduce the number of pixel distance computations by reusing computed distances between pixels and those neighbours. We introduce a distance matrix and a distance cube.

While applying masks, we go through the image columnwise from left to right. Furthermore we ignore border cases.

### 2.1 Distance Matrix

To reduce the frequency of the expensive distance computation between pixels we introduce a symmetric matrix  $D = [D(i, j, k, l)]_{i, j, k, l=0, \dots, m_s-1}$ . In this matrix the distances between pixels in a mask are stored. The pixel distances computed in this matrix can be reused for several mask operations. A matrix element  $D(i, j, k, l)$  represents the distance between the pixels at mask position  $(i, j)$  and  $(k, l)$ . The distance between two pixels is direction independent, that means  $D(i, j, k, l) = D(k, l, i, j)$  is valid for all pixels in the mask. To increase the reuseability of this matrix we make a difference between horizontal and vertical pixelwise moving of the mask.

**Initialization** When the mask is applied on the image for the first time the distance matrix has to be completely computed. The complete computation of the mask is done only once. This process is divided up into two parts, determine the distances between pixels being in the same and different rows. In Fig. 1 the calculation of the distances between pixels at the same row in the mask are presented. The computational time of this step is  $O(m_s^3)$ .

In Fig. 2 the calculation of the distances between pixels at different rows in the mask are shown. We iterate through each row in the mask from top to bottom and compute foreach pixel the distance to each neighbours on the lower rows. The computational time of this step is  $O(m_s^4)$ .

$\forall i = 1..m_s$ // Iterate through each row
$\forall j = 1..m_s - 1$ // Iterate through columns
$\forall l = j + 1..m_s$ // Iterate through columns
$D(i, j, i, l) = \text{Distance}(f(i, j), f(i, l));$
$D(i, l, i, j) = D(i, j, i, l);$ // Symmetrical case

**Fig. 1.** Computation of distances between pixels at the same row

$\forall i = 1..m_s - 1$ // Iterate through rows
$\forall j = 1..m_s - 1$ // Iterate through columns
$\forall k = i + 1..m_s$ // Iterate through rows
$\forall l = 1..m_s$ // Iterate through columns
$D(i, j, k, l) = \text{Distance}(f(i, j), f(k, l));$
$D(k, l, i, j) = D(i, j, k, l);$ // Symmetrical case

**Fig. 2.** Computation of distances between pixels at the different rows

**Horizontal moving** After scanning an image along a column  $j$  we continue the scanning with the next right column  $j + 1$  from the top. Originally we have to compute the whole distance matrix at the beginning of a new column. But luckily, we can reduce the costs of building this matrix. Before scanning an image vertical we compute recursively the distance matrices for each starting column from left to right and store these results. If we move the mask from a starting column to the next, most of the already determined distances in the previous distance matrix can be reused. We cannot reuse the distance for the pixels at the left mask side of the previous mask, because these pixels are no more in the current mask. The other distances can be reused in the way that the column index of the elements in previous matrix are increased by 1 (see Fig. 3). Finally we have to compute for each pixel at the right mask side the distances to its neighbours (see Fig. 4).

The computation cost of Fig. 3 is  $O(m_s^4)$ .

$\forall i = 1..m_s$ // Iterate through each row
$\forall j = 1..m_s - 1$ // Iterate through columns
$\forall k = 1..m_s$ // Iterate through rows
$\forall l = 1..m_s - 1$ // Iterate through columns
$D_{New}(i, j, k, l) = D(i, j + 1, k, l + 1)$

**Fig. 3.** Reusing the distance information of the previous distance matrix – horizontal moving

The computation cost of Fig. 4 is  $O(m_s^4)$ . Here  $sM_{Col}$  is an auxiliary variable keeping track of the column position to start from.

$\forall i = 1..m_s$ // Iterate through rows					
<table border="1" style="border-collapse: collapse; width: 100%;"> <tr> <td style="padding: 2px;"><math>\forall k = 1..m_s</math> // Iterate through rows</td> </tr> <tr> <td style="padding: 2px;"> <table border="1" style="border-collapse: collapse; width: 100%;"> <tr> <td style="padding: 2px;"><math>\forall l = 1..m_s</math> // Iterate through columns</td> </tr> <tr> <td style="padding: 2px;"><math>D(i, m_s, k, l) = \text{Distance}(f(i, m_s + sM_{Col} - 1), f(k, l + sM_{Col} - 1));</math></td> </tr> <tr> <td style="padding: 2px;"><math>D(k, l, i, m_s) = D(i, m_s, k, l);</math> // Symmetrical case</td> </tr> </table> </td> </tr> </table>	$\forall k = 1..m_s$ // Iterate through rows	<table border="1" style="border-collapse: collapse; width: 100%;"> <tr> <td style="padding: 2px;"><math>\forall l = 1..m_s</math> // Iterate through columns</td> </tr> <tr> <td style="padding: 2px;"><math>D(i, m_s, k, l) = \text{Distance}(f(i, m_s + sM_{Col} - 1), f(k, l + sM_{Col} - 1));</math></td> </tr> <tr> <td style="padding: 2px;"><math>D(k, l, i, m_s) = D(i, m_s, k, l);</math> // Symmetrical case</td> </tr> </table>	$\forall l = 1..m_s$ // Iterate through columns	$D(i, m_s, k, l) = \text{Distance}(f(i, m_s + sM_{Col} - 1), f(k, l + sM_{Col} - 1));$	$D(k, l, i, m_s) = D(i, m_s, k, l);$ // Symmetrical case
$\forall k = 1..m_s$ // Iterate through rows					
<table border="1" style="border-collapse: collapse; width: 100%;"> <tr> <td style="padding: 2px;"><math>\forall l = 1..m_s</math> // Iterate through columns</td> </tr> <tr> <td style="padding: 2px;"><math>D(i, m_s, k, l) = \text{Distance}(f(i, m_s + sM_{Col} - 1), f(k, l + sM_{Col} - 1));</math></td> </tr> <tr> <td style="padding: 2px;"><math>D(k, l, i, m_s) = D(i, m_s, k, l);</math> // Symmetrical case</td> </tr> </table>	$\forall l = 1..m_s$ // Iterate through columns	$D(i, m_s, k, l) = \text{Distance}(f(i, m_s + sM_{Col} - 1), f(k, l + sM_{Col} - 1));$	$D(k, l, i, m_s) = D(i, m_s, k, l);$ // Symmetrical case		
$\forall l = 1..m_s$ // Iterate through columns					
$D(i, m_s, k, l) = \text{Distance}(f(i, m_s + sM_{Col} - 1), f(k, l + sM_{Col} - 1));$					
$D(k, l, i, m_s) = D(i, m_s, k, l);$ // Symmetrical case					

**Fig. 4.** Computation of the distances between pixels in the last row mask and its neighbours – horizontal moving

**Vertical Moving** For the vertical moving of the mask the approach is very similar to the horizontal moving. We can keep most of the distance information of the previous computed matrix (see Fig. 5). Only for the pixels in the last row of the mask the distances to its neighbours must be computed(see Fig. 6).

The computation cost of Fig. 5 is  $O(m_s^4)$ .

$\forall i = 1..m_s - 1$ // Iterate through each row					
<table border="1" style="border-collapse: collapse; width: 100%;"> <tr> <td style="padding: 2px;"><math>\forall k = 1..m_s - 1</math> // Iterate through columns</td> </tr> <tr> <td style="padding: 2px;"> <table border="1" style="border-collapse: collapse; width: 100%;"> <tr> <td style="padding: 2px;"><math>\forall j = 1..m_s</math> // Iterate through rows</td> </tr> <tr> <td style="padding: 2px;"><math>\forall l = 1..m_s - 1</math> // Iterate through columns</td> </tr> <tr> <td style="padding: 2px;"><math>DNew(i, j, k, l) = D(i + 1, j, k + 1, l)</math></td> </tr> </table> </td> </tr> </table>	$\forall k = 1..m_s - 1$ // Iterate through columns	<table border="1" style="border-collapse: collapse; width: 100%;"> <tr> <td style="padding: 2px;"><math>\forall j = 1..m_s</math> // Iterate through rows</td> </tr> <tr> <td style="padding: 2px;"><math>\forall l = 1..m_s - 1</math> // Iterate through columns</td> </tr> <tr> <td style="padding: 2px;"><math>DNew(i, j, k, l) = D(i + 1, j, k + 1, l)</math></td> </tr> </table>	$\forall j = 1..m_s$ // Iterate through rows	$\forall l = 1..m_s - 1$ // Iterate through columns	$DNew(i, j, k, l) = D(i + 1, j, k + 1, l)$
$\forall k = 1..m_s - 1$ // Iterate through columns					
<table border="1" style="border-collapse: collapse; width: 100%;"> <tr> <td style="padding: 2px;"><math>\forall j = 1..m_s</math> // Iterate through rows</td> </tr> <tr> <td style="padding: 2px;"><math>\forall l = 1..m_s - 1</math> // Iterate through columns</td> </tr> <tr> <td style="padding: 2px;"><math>DNew(i, j, k, l) = D(i + 1, j, k + 1, l)</math></td> </tr> </table>	$\forall j = 1..m_s$ // Iterate through rows	$\forall l = 1..m_s - 1$ // Iterate through columns	$DNew(i, j, k, l) = D(i + 1, j, k + 1, l)$		
$\forall j = 1..m_s$ // Iterate through rows					
$\forall l = 1..m_s - 1$ // Iterate through columns					
$DNew(i, j, k, l) = D(i + 1, j, k + 1, l)$					

**Fig. 5.** Reusing the distance information of the previous distance matrix – vertical moving

The computation cost of Fig. 6 is  $O(m_s^2)$ . Here  $sM_{Row}$  and  $sM_{Col}$  are auxiliary variables keeping track of the row and column positions, respectively, to start from.

$\forall j = 1..m_s - 1$ // Iterate through rows			
<table border="1" style="border-collapse: collapse; width: 100%;"> <tr> <td style="padding: 2px;"><math>\forall l = j + 1..m_s</math> // Iterate through columns</td> </tr> <tr> <td style="padding: 2px;"><math>D(m_s, j, m_s, l) = \text{Distance}(f(m_s + sM_{Row} - 1, j + sM_{Col} - 1), f(m_s + sM_{Row} - 1, l + sM_{Col} - 1));</math></td> </tr> <tr> <td style="padding: 2px;"><math>D(m_s, l, m_s, j) = D(m_s, j, m_s, l);</math> // Symmetrical case</td> </tr> </table>	$\forall l = j + 1..m_s$ // Iterate through columns	$D(m_s, j, m_s, l) = \text{Distance}(f(m_s + sM_{Row} - 1, j + sM_{Col} - 1), f(m_s + sM_{Row} - 1, l + sM_{Col} - 1));$	$D(m_s, l, m_s, j) = D(m_s, j, m_s, l);$ // Symmetrical case
$\forall l = j + 1..m_s$ // Iterate through columns			
$D(m_s, j, m_s, l) = \text{Distance}(f(m_s + sM_{Row} - 1, j + sM_{Col} - 1), f(m_s + sM_{Row} - 1, l + sM_{Col} - 1));$			
$D(m_s, l, m_s, j) = D(m_s, j, m_s, l);$ // Symmetrical case			

**Fig. 6.** Computation of the distances between pixels in the last row mask and its neighbours – vertical moving

## 2.2 E-Cube

In the standard VMFD we have to compute for each pixel in the mask the sum of the row distances between this pixel and its neighbours in the corresponding mask row. In

general, the computed row distances can be used many times again. For this purpose we introduce the E-Cube that is a matrix of the dimension  $m_s^3$ . Each element  $E(i, j, s)$  represents the distance between the element  $(i, j)$  in the mask to all its neighbours in the mask row  $s$ .

**Initialization** For each element in the row  $i$  and column  $j$  the distance  $E(i, j, k)$  to all its neighbours in the row  $k$  is computed by using the distances matrix (see Fig. 7).

$\forall s = 1..m_s$ // Iterate through each slice
$\forall i = 1..m_s$ // Iterate through rows
$\forall j = 1..m_s$ // Iterate through columns
$\forall l = 1..m_s$ // Iterate through columns
$E(i, j, s) = E(i, j, s) + D(s, j, i, l)$

Fig. 7. Initialization of the E-Cube

**Horizontal Moving** If the mask is shifted to the next vertical position, then most of the computed distances in the E-Cube can be reused in the following way. All elements in the E-Cube are moved in the way that the column index is decreased by one. The value of each element  $ENew(i, j, s)$  must be updated in two steps. First, the distance between the pixel  $(s, j)$  and  $(i, 1)$  is removed, because the pixel  $(i, 1)$  is no longer valid. Second, the distance between the pixel  $(s, j)$  and  $(i, m_s)$  is added, because the pixel  $(i, m_s)$  is new in the mask.

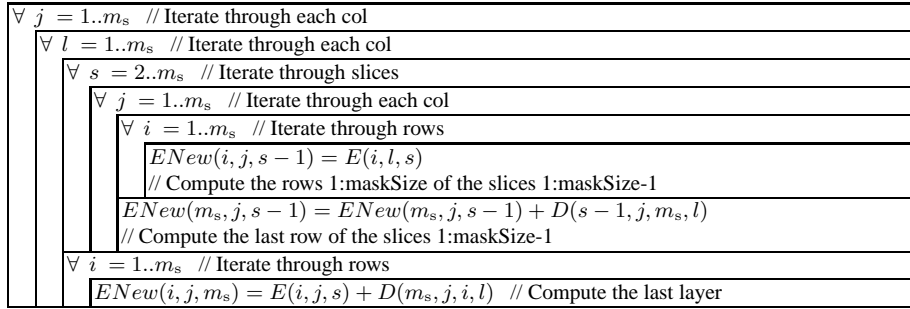
Additionally, for each pixel in the right column the distances pixels have to be computed new.

In Fig. 8 is the algorithm of the horizontal shifting described.

$\forall s = 1..m_s$ // Iterate through each slice
$\forall i = 1..m_s$ // Iterate through each row
$\forall j = 1..m_s - 1$ // Iterate through columns
$ENew(i, j, s) = E(i, j + 1, s) - DOld(s, j + 1, i, 1) + D(s, j, i, m_s)$
// Remove old distance between current pixel and left pixel in previous (left) mask
// Add distance between current pixel and right pixel in current mask
$\forall l = 1..m_s$ // Iterate through columns
$ENew(i, m_s, s) = E(i, j + 1, s) + D(s, m_s, i, l)$
// Compute the right pixel in current row new

Fig. 8. Horizontal Shifting of the E-Cube

**Vertical Moving** The vertical shifting of the E-Cube is analogous to the horizontal shifting in 2.2.



**Fig. 9.** Vertical Shifting of the E-Cube

### 2.3 Main Algorithm

Fig. 10 outlines the main algorithm of our approach. This algorithm consists of the following parts: first initialization, then columnwise scanning of the image from left to right.

In the initialization step the mask starts at the position  $(startRow, startCol)$ . The distance matrix and the E-Cube are created (see sect. 2.1 and 2.2) and stored in the auxiliary lists *DInits* resp. *CubeList*. After this step the mask is moved horizontal from left to right. For each mask moving the distance matrix and E-Cube are updated by using the horizontal moving algorithm (see sect. 2.1 and 2.2). The results are appended to the lists *DInits* resp. *CubeList*.

After the initialization the mask is moved to the position  $(startRow, startCol)$ . Then the image is scanned columnwise. For each new column the distance matrix and E-Cube are recomputed by using the corresponding element of the lists *DInits* resp. *CubeList*. For each column the new value of the applied pixel is computed in the *computeMinPixel* function using current distance matrix and E-Cube. After each pixel apply the distance matrix and E-Cube are updated by using the in sect. 2.1 and 2.2 presented algorithms.

The function *computeMinPixel* returns the pixel in the mask with minimal distance to all its neighbours by using the e-cube.

## 3 Analytic Optimization

When the VMF is applied on one-dimensional pixel values, the computation of the distances over each two pixels becomes unnecessary since it can be shown that the

$sM_{Col} = 1$
$startCol = (m_s + 1)/2$
$endOfColLoop = cols - (m_s - 1)/2$
$startRow = (m_s + 1)/2$
$endOfRowLoop = rows - (m_s - 1)/2$
$stripeSize = endOfColLoop - startCol + 1$
$DInits[1] = createDistMatrix(m_s, \mathbf{f})$ // Create first distance matrix
$CubeList[1] = createECube(m_s, \mathbf{f})$ // Create first e-cube
$\forall i = 2..stripeSize$ // Iterate through the stripes
// Compute recursive the other distance matrices
$DInits[i] = shiftHorDistMatrix(i + sM_{Col} - 1, m_s, D, \mathbf{f})$
// Compute recursive the other e cubes
$CubeList[i] = shiftHorECube(i + sM_{Col} - 1, m_s, DInits[i - 1], DInits[i], \mathbf{f})$
$\forall s = 1..m_s$ // Iterate through each slice
$\forall i = 1..m_s$ // Iterate through rows
$\forall j = 1..m_s$ // Iterate through columns
$\forall l = 1..m_s$ // Iterate through columns
$E(i, j, s) = E(i, j, s) + D(s, j, i, l)$
$\forall j = 1..endOfColLoop$ // Iterate through the stripes
$sM_{Row} = 1$
$sM_{Row} = 1$
$\mathbf{g}(startRow, j) = computeMinPixel(sM_{Row}, sM_{Col}, E(, , m_s, , )\mathbf{f})$ // Compute new pixel
$\forall i = 1 + startRow..endOfRowLoop$ // Iterate through the rows of the stripe
$sM_{Row} ++$
$DInits[j] = updateDistMatrix(sM_{Row}, sM_{Col}, DInits[j], m_s, \mathbf{f})$
$DInits[j] = updateECube(sM_{Row}, sM_{Col}, CubeList[j], m_s, \mathbf{f})$
$\mathbf{g}(startRow, j) = computeMinPixel(sM_{Row}, sM_{Col}, E(, , m_s, , )\mathbf{f})$
// Compute new pixel
$sM_{Col} ++$

Fig. 10. Main Algorithm

*median* of the mask is the solution of the optimization problem of equation 2. It has to be emphasized that this is fulfilled, only if the the absolute values of differences between each pixel pair is the used distance measure in equation 3.

For color images, pixel values are three-dimensional vectors. The application of the median rule is no more valid. In the following we will demonstrate that even in this case we can solve the optimization problem analytically under the assumption of a special distance measure: the *square Euclidian distances*.

The equation to be minimized in the case of square Euclidian distance is:

$$R_{\mu, \nu} = \sum_{k, l \in W} \|\mathbf{f}_{\mu, \nu} - \mathbf{f}_{k, l}\|^2. \quad (4)$$

The minimum can be computed by finding the vector that sets the first derivative with respect to  $\mathbf{f}_{\mu, \nu}$  to zero:

$$\frac{d\mathbf{R}_{\mu,\nu}}{d\mathbf{f}_{\mu,\nu}} = \sum_{k,l \in W} 2(\mathbf{f}_{\mu,\nu} - \mathbf{f}_{k,l}) \stackrel{!}{=} 0 \quad (5)$$

$$\implies m_s^2 \hat{\mathbf{f}}_{\mu,\nu} - \sum_{k,l \in W} \mathbf{f}_{k,l} = 0$$

$$\implies \hat{\mathbf{f}}_{\mu,\nu} = \frac{1}{m_s^2} \sum_{k,l \in W} \mathbf{f}_{k,l} \quad (6)$$

Eq. 6 shows that the vector that minimizes Eq. 2 for the square Euclidian distance is the *mean* vector within the mask. As the mean value can have values which are not existent in the image, the best approximation is the vector which has the smallest square Euclidian distance to the mean vector in the mask. In this case the number of evaluations of the distance measure over the whole image is reduced to  $O(n \times m \times m_s^2)$ . The computation of the mean vectors grows linearly with the mask size.

Analytical solutions for other distance measures have to be considered in dependency with the properties of the measure. If it is not possible to compute the solution in a closed form, the algorithmic optimization decreases in any case the computational costs.

## 4 Conclusion

In our paper we propose two different ways of improving the computational time of the VMF ( $O(n \times m \times m_s^4)$ ). Although there are some elements of the algorithm of  $O(m_s^4)$ , but these are not multiplied by  $n \times m$  - this is a key point of the algorithmic improvement.

The algorithmic way is independent of the used distance function. It reduces the computational time by reusing computed pixel distances ( $O(n \times m \times m_s^3)$ ). The analytic way works only for the square Euclidian distance. This approach reduces the computational costs of the VMF to  $O(n \times m \times m_s^2)$ .

Further work remains to be done on extending the analytic way to any distance function.

## References

1. Jaakko Astola, Pekka Haavisto, and Yrjo Neuvo. Vector median filters. *Proceedings of the IEEE*, 78:678–689, 1990.
2. Vinh Hong, Henryk Palus, and Dietrich Paulus. Edge preserving filters on color images. In *ICCS2004 – International Conference on Computational Science*, volume 4, pages 35–42, Berlin, Germany, 6 2004. Academic Computer Centre CYFRONET AGH, Springer Verlag.
3. Konstantinos N. Plataniotis and Anastasios N. Venetsanopoulos. *Color Image Processing and Applications*. Springer Verlag, 2000. INF 2003/3326.

## On Color Image Quantization by the K-Means Algorithm

Henryk Palus

Institute of Automatic Control, Silesian University of Technology,  
Akademicka 16, 44-100 GLIWICE Poland,  
[hpalus@polsl.gliwice.pl](mailto:hpalus@polsl.gliwice.pl)

**Abstract.** In this paper we show the main properties of k-means algorithm as a tool for color image quantization. All experiments have been carried out on color images with different number of unique colors and different colorfulness. We have tested the influence of methods of determination of initial cluster centers, of choice of distance metric, of choice of color space. In our tests we have used two dimensions of palette (256,16) and three different measures for quantization errors. The results of k-means technique have been compared with quantized images from commercial programs.

### 1 Introduction

There is not a problem to observe that in each digital color image appears only a small subset of all possible 16,7 millions colors. Color image quantization is the process of transformation of a true color image (eight bit onto each color component) into an image consisting of a small number of specially selected colors (color palette). This process is very often used as an auxiliary operation in computer vision and computer graphics. Color quantization is widely presented in color image processing handbooks [3, 2]. New colors are selected by minimizing the color difference between the original image and the quantized image. Low value of color difference (quantization error) needs a palette designed for the particular image. The quantization error depends on the number of colors in the palette (e.g. 256, 64, 16 colors), the method of choice of colors for the palette and the pixel classifying technique. Color quantization is a kind of lossy compression technique. Many color image quantization techniques based on hierarchical or iterative (partitional) scheme have been proposed in the past. As a generally statement, it may be found that the hierarchical methods are faster than the iterative methods but they have larger quantization errors. Frequently used iterative technique is named k-means algorithm.

The purpose of this paper is to show the main properties of k-means algorithm as a tool for color image quantization. This is done by testing a few color images with different number of unique colors. Three methods of determination of initial cluster centers: randomly from color space, randomly from image, and by uniform partitioning of gray scale are considered. Two metrics in two color spaces (RGB, CIELAB) are tested. We propose to use different measures of quantization errors. First of all, measures that are popular in the color community i.e. MSE and  $\Delta E$ . Apart from these classical measures we try to use other evaluation function: the colorfulness of image [4]. This approach



assumes that the colorfulness of the quantized image should be equal to the colorfulness of the original image. Additionally we called attention on color quantization as presegmentation, which can reduce the complexity of the segmentation problem. We also present the quantization errors as a function of decreasing palette size (256 colors, 16 colors). Quantization results are compared with results from such commercially available programs as PhotoPaint and Photoshop.

## 2 Short description of k-means technique

K-means as clustering technique has been developed in the sixties and already has been described in classical Anderberg's handbook [1]. During the first step of k-means algorithm a fixed number of clusters and initial cluster centers in the color space are chosen. The main idea is to modificate the positions of cluster centers so long as the sum of distances between all points of clusters and their cluster centers will be minimal. During these modifications all points are allocated to closest cluster centers using a predefined metric. The most typical used metrics are: the Euclidean distance and the City Block metric. After each allocation a new positions of cluster centers are computed as arithmetical means of cluster points. The algorithm usually stops if the difference between new and old positions of cluster centers is too small. K-means converges to a locally optimal solution. Received results depend on different factors such as method of determination of initial cluster centers, used color space, applied metric etc. On the other hand there is one of the fastest methods to perform clustering.

## 3 The choice of test images

For investigations a representative set of five color images has been chosen. These images have the same spatial resolution (640×480 pixels) and are presented in Fig. 1. The choice was based on calculating the number of unique colors present in the image. Table 1 presents data about chosen images.

**Table 1.** Number of colors in the test images

	Chart	Duck	Landscape	Characters	Mountains
Number of colors	40523	49759	66634	78805	95481

In our set of images we have images with small number of unique colors (Chart, Duck), with middle number (Landscape) and with large number of unique colors (Characters, Mountains). From other point of view, we can characterize color images using their perceptual features e.g. colourfulness. Therefore we computed colorfulness of images using metric proposed in [4] (more details about metric see in Section 7). The results of this computations are presented in Table 2.

**Table 2.** Values of colorfulness in the test images

	Chart	Duck	Landscape	Characters	Mountains
Colorfulness	96,56	15,60	35,92	88,31	46,04

If we use semantic attributes of colorfulness, then we can correlate the names of images with these attributes: Duck - slightly colorful, Landscape - moderately colorful, Mountains - averagely colorful, Characters - highly colorful and Chart - extremely

colorful. Now we are certain that our set of color images is representative: our images "stretch" along the whole scale of colorfulness. By the way, the extremely colorful image Chart has the smallest number of unique colors.

#### 4 Methods of determination of initial cluster centers

However classical k-means technique has few "degree of freedom". First of all we should determine the initial cluster centers i.e. starting points of algorithm. We can find these points in whole color space or in set of colors of image pixels or on the gray scale line by its dividing into equal segments. We have tested these three approaches during color quantization to 256 and 16 colors. In each case we evaluated the results using two kinds of values:

$$RSME = \frac{1}{MN} \sum_{j=1}^M \sum_{i=1}^N \sqrt{(R_{ij} - R_{ij}^*)^2 + (G_{ij} - G_{ij}^*)^2 + (B_{ij} - B_{ij}^*)^2} \quad (1)$$

$$\Delta E = \frac{1}{MN} \sum_{j=1}^M \sum_{i=1}^N \sqrt{(L_{ij} - L_{ij}^*)^2 + (a_{ij} - a_{ij}^*)^2 + (b_{ij} - b_{ij}^*)^2} \quad (2)$$

where:  $R_{ij}, G_{ij}, B_{ij}$  are color components in original image,  $R_{ij}^*, G_{ij}^*, B_{ij}^*$  are color components in quantized image and similarly for CIE Lab components.

Tables 3–6 contain the results of experiment. We can see that in the majority of cases the smaller values of quantization error are achieving for initial cluster centers coming from image. Additionally, we observe that color quantization error is growing if the number of colors in the palette is decreasing. The Fig. 2 shows it for three different palette (256, 64, 16) colors.

**Table 3.** RMSE values for images quantized to 256 colors

	Chart	Duck	Landscape	Characters	Mountains
From space	3,78	5,62	5,52	6,25	7,17
From image	2,37	2,27	2,84	4,23	3,34
From gray-scale	2,92	2,54	3,12	4,70	3,41

**Table 4.**  $\Delta E$  values for images quantized to 256 colors

	Chart	Duck	Landscape	Characters	Mountains
From space	1,63	2,12	1,92	9,02	3,52
From image	1,13	1,11	1,17	3,83	1,83
From gray-scale	1,40	1,26	1,30	6,15	2,00

**Table 5.** RMSE values for images quantized to 16 colors

	Chart	Duck	Landscape	Characters	Mountains
From space	18,80	12,19	11,56	14,37	14,17
From image	21,61	7,22	8,68	14,71	11,33
From gray-scale	25,38	7,24	8,91	13,76	11,13

**Table 6.**  $\Delta E$  values for images quantized to 16 colors

	Chart	Duck	Landscape	Characters	Mountains
From space	6,91	3,53	3,66	14,38	5,52
From image	8,21	2,52	3,15	11,49	5,03
From gray-scale	9,48	2,47	2,79	13,49	4,69

## 5 The choice of metric

Other "degree of freedom" is related with the choice of distance metric in clustering process. We considered two classical metrics for clustering in RGB space: L1 (city block metric) and L2 (Euclidean metric):

$$L1 = |R_1 - R_2| + |G_1 - G_2| + |B_1 - B_2| \quad (3)$$

$$L2 = \sqrt{(R_1 - R_2)^2 + (G_1 - G_2)^2 + (B_1 - B_2)^2} \quad (4)$$

Tables 7–10 show the results for different dimensions of palette (256,16) and different measures of quantization error (RMSE,  $\Delta E$ ). Almost in all cases the quantization error is smaller for L2 metric. In further investigations we used L2 metric and started with clustering process from initial cluster centers that were randomly chosen from all colors of the image.

**Table 7.** RMSE values for images quantized to 256 colors using different metrics

	Chart	Duck	Landscape	Characters	Mountains
L2	2,37	2,27	2,84	4,23	3,34
L1	2,42	2,32	2,92	4,34	3,41

**Table 8.**  $\Delta E$  values for images quantized to 256 colors using different metrics

	Chart	Duck	Landscape	Characters	Mountains
L2	1,13	1,11	1,17	3,83	1,83
L1	1,16	1,12	1,21	3,92	1,87

**Table 9.** RMSE values for images quantized to 16 colors using different metrics

	Chart	Duck	Landscape	Characters	Mountains
L2	21,61	7,22	8,68	14,71	11,33
L1	22,84	7,08	8,92	14,68	11,39

**Table 10.**  $\Delta E$  values for images quantized to 16 colors using different metrics

	Chart	Duck	Landscape	Characters	Mountains
L2	8,21	2,52	3,15	11,49	5,03
L1	8,82	2,53	3,31	11,72	5,07

## 6 The choice of color space

The clustering process can be organized in different color spaces. Among color spaces a special role plays the CIE Lab color space. The Euclidean distance in this space is approximately equal to the perceptual difference between colors. Therefore we investigated color quantization using both spaces: RGB and CIELAB. Tables 11–14 show the results for different dimensions of palette (256,16) and different measures of quantization error (RMSE,  $\Delta E$ ). We can observe here some ambivalence: if we use for evaluation RMSE values (based on RGB space) then we achieve that quantization errors for RGB-based k-means clustering are smaller. But if we use for evaluation  $\Delta E$  values (based on CIELAB space) then we achieve that quantization errors for CIELAB-based k-means clustering are smaller.

**Table 11.** RMSE values for images quantized to 256 colors using different color spaces

	Chart	Duck	Landscape	Characters	Mountains
RGB	2,37	2,27	2,84	4,23	3,34
CIELAB	2,90	2,85	3,61	5,54	4,19

**Table 12.**  $\Delta E$  values for images quantized to 256 colors using different color spaces

	Chart	Duck	Landscape	Characters	Mountains
RGB	1,13	1,11	1,17	3,83	1,83
CIELAB	0,89	0,93	0,97	3,03	1,52

**Table 13.** RMSE values for images quantized to 16 colors using different color spaces

	Chart	Duck	Landscape	Characters	Mountains
RGB	21,61	7,22	8,68	14,71	11,33
CIELAB	23,46	8,05	9,64	21,15	13,61

**Table 14.**  $\Delta E$  values for images quantized to 16 colors using different color spaces

	Chart	Duck	Landscape	Characters	Mountains
RGB	8,21	2,52	3,15	11,49	5,03
CIELAB	6,48	2,42	2,36	10,31	4,26

## 7 Colorfulness for evaluation of color quantization results

In order to decide above presented dilemma we can use additional evaluation function. This function can be the colorfulness  $M$  of image [4]. The quantization error can be expressed as colorfulness difference between original and quantized images. We calculated colorfulness using following formula:

$$M = \sqrt{\sigma_{rg}^2 + \sigma_{yb}^2} + 0.3 \cdot \sqrt{\mu_{rg}^2 + \mu_{yb}^2} \quad (5)$$

where:  $\sigma$  and  $\mu$  are the standard deviation and the mean value of the pixel cloud along direction described by subscripts,  $rg = R - G$ ,  $yb = 0.5(R + G) - B$ . Tables 15–18 contain the results of experiment. We can see that in the majority of cases the smaller values of colorfulness difference are achieving for CIELAB color space. Color quantization can be also used as tool for presegmentation. We can see this potential application in Fig. 3 if we use pseudocolorizing for quantized images, particularly for 16 colors.

**Table 15.** Values of colorfulness in the original and quantized images (RGB space)

	Chart	Duck	Landscape	Characters	Mountains
Original	96,56	15,60	35,92	88,31	46,04
RGB 256	96,49	15,20	35,75	87,90	45,77
RGB16	90,60	13,61	34,09	84,21	42,94

**Table 16.** Values of colorfulness in the original and quantized images (CIELAB space)

	Chart	Duck	Landscape	Characters	Mountains
Original	96,56	15,60	35,92	88,31	46,04
LAB 256	96,47	15,36	35,77	87,93	45,84
LAB16	93,47	14,33	35,38	80,45	44,59

## 8 Comparison between the k-means quantization and techniques used in commercial programs

Many commercial graphical programs enable users to reduce the number of image colors. We can compare the results from k-means algorithm to the popular programs: Corel

PhotoPaint v.11 and Adobe Photoshop v.7. From few algorithms used in these programs we have chosen following: Optimized (PhotoPaint), Adaptive (PhotoPaint) and Adaptive (Photoshop). Tables 17–20 show the results of comparison for different dimensions of palette (256,16) and different measures of quantization error (RMSE,  $\Delta E$ ). In the case of 256 colors the quantization errors (RMSE,  $\Delta E$ ) of k-means algorithm are smaller than in the case of commercial programs. In the case of 16 colors the superiority of k-means is not so distinctly expressed. The general drawback of k-means technique in comparison with commercial algorithms of color quantization is its longer execution time.

**Table 17.** RMSE values for images quantized to 256 colors using different methods

	Chart	Duck	Landscape	Characters	Mountains
k-means	2,37	2,27	2,84	4,23	3,34
Paint_optim	2,54	2,31	2,96	3,92	3,55
Paint_adapt	4,65	4,14	4,40	5,38	4,29
Photo_adapt	4,05	2,68	4,33	9,91	4,87

**Table 18.**  $\Delta E$  values for images quantized to 256 colors using different methods

	Chart	Duck	Landscape	Characters	Mountains
k-means	1,13	1,11	1,17	3,83	1,83
Paint_optim	1,21	1,16	1,23	5,61	1,95
Paint_adapt	2,15	2,03	1,81	9,74	2,38
Photo_adapt	1,98	1,30	1,76	12,81	2,52

**Table 19.** RMSE values for images quantized to 16 colors using different methods

	Chart	Duck	Landscape	Characters	Mountains
k-means	21,61	7,22	8,68	14,71	11,33
Paint_optim	22,15	7,38	8,62	13,60	13,20
Paint_adapt	32,15	10,97	10,53	20,53	13,25
Photo_adapt	24,86	6,81	10,21	17,74	11,91

**Table 20.**  $\Delta E$  values for images quantized to 16 colors using different methods

	Chart	Duck	Landscape	Characters	Mountains
k-means	8,21	2,52	3,15	11,49	5,03
Paint_optim	7,00	2,63	2,62	12,58	5,01
Paint_adapt	11,06	3,23	3,49	16,82	5,52
Photo_adapt	7,44	2,39	3,00	16,73	4,82

## 9 Conclusions

K-means clustering technique used as tool for color quantization produces good quality results. They are not bad from images quantized by commercial programs. Researched method works good with initial cluster centers chosen from image colors, L2 distance metric and clustering in CIELAB color space. Colorfulness can be used for choice of representative set of color images and as additional measure of quality of color quantization.

## References

1. M. Anderberg. *Cluster Analysis for Applications*. Academic Press, 1973.

2. Luc Brun and Alain Trémeau. Color quantization. In G. Sharma, editor, *Digital Color Imaging Handbook*, Electrical and Applied Signal Processing, chapter 9 : Color quantization, pages 589–637. CRC Press, 2002.
3. M. Domanski and Bartkowiak M. Compression. In S.Sangwine and R.E.N.Horne, editors, *Colour Image Processing Handbook*, pages 242–304. Chapman and Hall, 1998.
4. D. Hasler and S. Süsstrunk. Measuring colourfulness in natural images. In *Proc. IS&T/SPIE Electronic Imaging*, volume 5007, pages 87–95, 2003.



a)



b)



c)

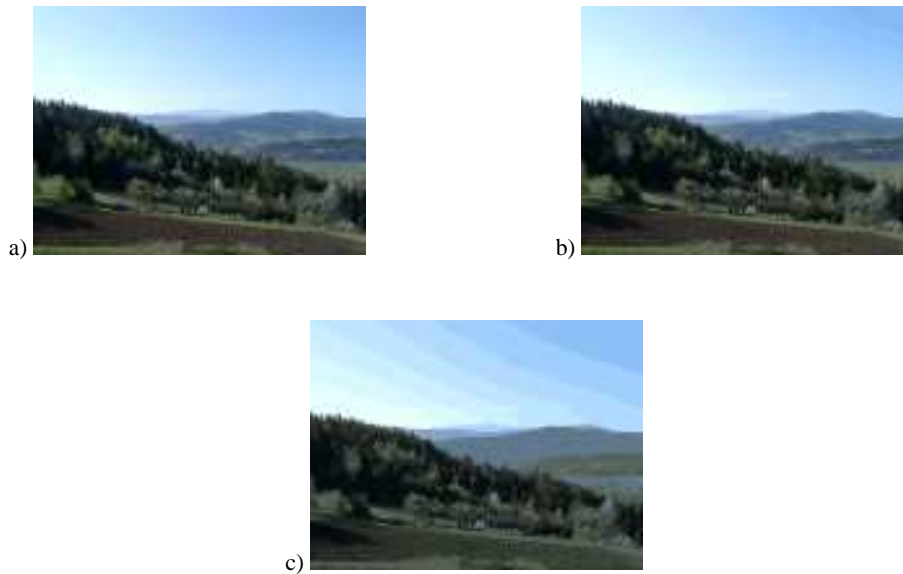


d)

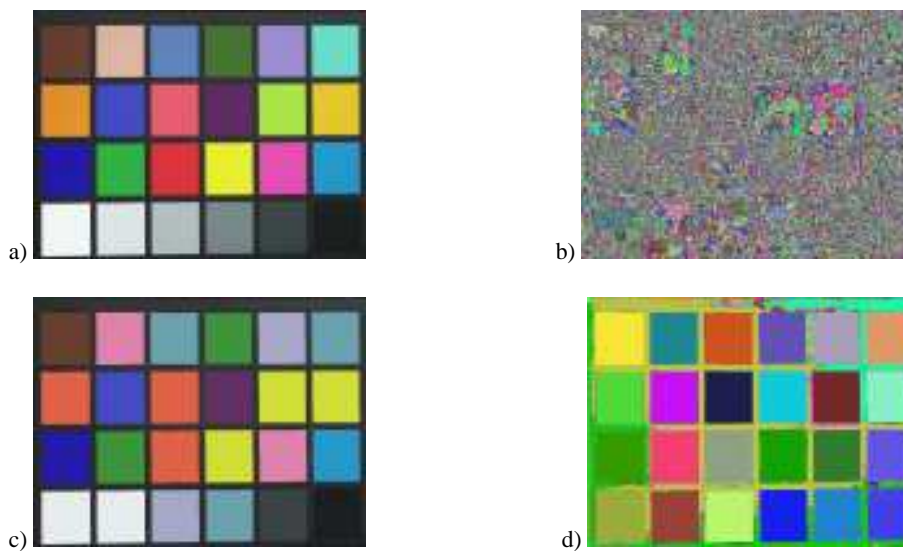


e)

**Fig. 1.** Color images chosen for tests: a) Chart, b) Duck, c) Landscape, d) Characters, e) Mountains. Image resolution:  $640 \times 480$  pixels.



**Fig. 2.** Results of k-means based color image quantization of the image Mountains: a) quantization to 256 colors, b) quantization to 64 colors, c) quantization to 16 colors.



**Fig. 3.** Different versions of the image Chart: a) True color version of the image Chart quantized to 256 colors, b) Pseudocolor version of the image Fig. 3a contains 123633 regions, c) True color version of the image Chart quantized to 16 colors, d) Pseudocolor version of the image Fig. 3c contains 3330 regions.



## Symmetrie als kognitives Bildmerkmal

Kai Hübner

Bremer Institut für Sichere Systeme, FB3  
Universität Bremen, Postfach 330440, 28334 Bremen  
khuebner@informatik.uni-bremen.de

**Zusammenfassung.** Symmetrie ist in der Natur sowie im alltäglichen Leben ein in vielfältiger Weise vorhandenes und weitverbreitetes Merkmal, an dessen Ergründung und Verwertung bereits in unterschiedlichen Domänen geforscht wurde. Auf dem Gebiet der maschinellen Bildverarbeitung beschreibt die Symmetrie im Gegensatz zu bekannteren Verfahren der Merkmalsextraktion wie der Kantenerkennung ein regionales Merkmal, dessen Aussagekraft mit der betrachteten Region wächst. Der daraus entstehende Nachteil vieler Ansätze ist daher im hohen Aufwand an Rechenzeit zu sehen, begründet durch die notwendigerweise weitreichenderen Operatoren zur Symmetrienerkennung. In der Literatur aufgezeigte psychophysische Experimente zur Symmetrieanalyse bekräftigen die These, dass vertikale und horizontale Spiegelsymmetrien die bedeutendsten Symmetriemerkmale in der menschlichen Wahrnehmung darstellen. Auf der Basis dieser Ergebnisse wurden in bisheriger Arbeit diese Symmetriearten in einen eigenen Ansatz eingebunden, mit dem somit ein einfacher, schneller und kompakter Operator zur Verfügung gestellt wird. Im diesem Beitrag wird Symmetrie als Merkmal motiviert, sowie erzielte Ergebnisse des Symmetrieoperators vorgestellt.

### 1 Einführung und Motivation

Viele Gegenstände des täglichen Gebrauchs besitzen eine auf die ein oder andere Art symmetrische Struktur. Tiere und Menschen nutzen Symmetrie als Landmarke zur visuellen Exploration von Objekten und Umgebungen. Bis heute wurden viele unterschiedliche Aspekte der Symmetrie in verschiedenen Domänen erforscht, angefangen in der Biologie [12, 8, 5], auf den Gebieten der Psychologie und Psychophysik [15, 14] und der Mathematik<sup>1</sup>, sowie auch in der maschinellen Bildverarbeitung. In Letzterem wird die Symmetrie bevorzugt als zweidimensionales Aufmerksamkeitsmerkmal zur Detektierung von signifikanten Regionen (sog. „Regions-of-Interest“) oder zur Orientierungserkennung von Objekten eingesetzt. Dabei werden in erster Linie reflektive Symmetrien (Spiegelsymmetrien) genutzt, aber auch Ansätze zur Erkennung rotationaler Symmetrien (Drehsymmetrien) sind in der Literatur zu finden.

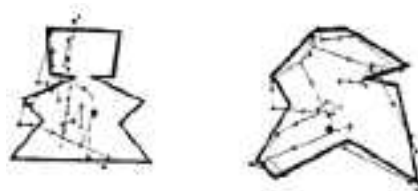
Locher und Nodine [14] ermittelten anhand von Testreihen, dass insbesondere Spiegelsymmetrien und deren Orientierung von wichtiger Bedeutung für das menschliche Sehen sind. Es konnte gezeigt werden, dass Menschen bei Vorhandensein von horizontaler oder vertikaler Symmetrie in einer Szene die Symmetrieachsen äußerst schnell

---

<sup>1</sup> Viele Aspekte, die sich auf Natur und Mathematik beziehen, finden sich z.B. im Buch „Denkt Gott symmetrisch? - Das Ebenmaß in Mathematik und Natur“ von Stewart und Golubitsky [18]



erkennen und zur weiteren visuellen Exploration der Szene ausnutzen können (siehe Abb. 1). An diesem Beispiel lässt sich die Existenz und Effektivität der Eigenschaft des menschlichen Auges, Symmetrien zu erkennen und auch auszunutzen, deutlich nachvollziehen.



**Abb. 1.** Visuelle Exploration einer vertikal symmetrischen (links) und einer asymmetrischen Figur (rechts) [14].

Palmer und Hemenway [15] betrachteten in ähnlichen Experimenten die Erkennungszeit von unterschiedlich orientierten Polygonen. Als Ergebnis konnte festgestellt werden, dass besonders der Detektierung von Spiegelsymmetrien eine hohe Bedeutung zukommt. Die Unterschiede in der Erkennung der Figuren zeigen auf, dass vom Menschen vertikale Spiegelsymmetrie<sup>2</sup> sehr oft und wesentlich schneller visuell erkannt wird als horizontale Spiegelsymmetrie, diese wiederum besser und schneller als sonstige Spiegelsymmetrien.

Diese beiden Arbeiten sind beispielhaft für weitere, die Symmetrie als visuelles Merkmal aus biologischer und psychophysischer Sicht motivieren (siehe z.B. Arbeiten von Barlow und Reeves [1], Csathó et al. [4] oder ein gesammeltes *Special Issue on The Perception of Symmetry* [19]).

## 2 Verwandte Arbeiten

Neben ihrem Einfluss auf die menschliche Wahrnehmung wurde die Symmetrie bereits in verschiedensten Anwendungen der maschinellen Bildverarbeitung betrachtet. Es finden sich einige einführende Arbeiten, die Symmetrie als Merkmal in unterschiedlichsten Anwendungen motivieren [20, 17, 13]. Im Folgenden werden Ideen, Ergebnisse und Probleme einer Auswahl einschlägiger Ansätze zur Symmetriestimmung vorgestellt.

Reisfeld et al. [17] definieren eine verallgemeinerte Symmetrietransformation, die Spiegelsymmetrie zur Extraktion von Aufmerksamkeitsbereichen verwendet. Der zweidimensionale Operator beinhaltet sowohl Symmetrie- als auch Gradienteninformation. Dadurch entsteht neben dem Problem der Komplexität (2D-Faltung) zudem der Nachteil, dass symmetrische, aber schwach kontrastierte Regionen (z.B. Wände) kaum Beachtung finden (Beispiel in Abb. 3(b)).

Di Gesù und Valenti präsentieren eine diskrete Symmetrie-Transformation (DST), beschleunigt durch die Auswahl nicht uniformer Bildbereiche. Das hieraus resultierende Symmetriebild wird hier für Aufgaben der Gesichtserkennung, Bildsegmentierung

<sup>2</sup> Unter vertikaler Spiegelsymmetrie wird durchgängig eine Spiegelsymmetrie an einer vertikalen Achse verstanden (siehe auch Beispiel in Abb. 1).

und Objektklassifikation [6] sowie der Bewegungsanalyse [7] weiterverwendet. Wie beim vorigen Ansatz auch, bestehen die wesentlichen Probleme in der Mächtigkeit und Parametrisierbarkeit des Operators, die sich in langsamer Berechnung und Unhandlichkeit niederschlagen (Beispiel in Abb. 3(c)).

Von Chetverikov [2, 3] wird als Basis der Symmetrienerkennung ein Regularitätswert anhand der symmetrischen Struktur einer Textur bestimmt. Dieser Ansatz erweist sich als sehr aufwändig, da für jeden Bildpunkt eine Umgebung bezüglich ihrer Anisotropie (Richtungsabhängigkeit) analysiert werden muss. Erst basierend auf diesem Ergebnis wird eine Symmetriestruktur bestimmt, welche die Texturorientierung beschreibt. Für die Extraktion von Merkmalspunkten ist dieser Ansatz eher ungeeignet, nicht nur wegen der Aufwändigkeit des Verfahrens, sondern auch aufgrund der Tatsache, dass das hier extrahierte Merkmal eine Textur als Gesamtbild beschreibt (siehe Abb. 4).

Bereits aus diesen ausgewählten Arbeiten ist zu erkennen, dass sich viele Ansätze zur Symmetriestimmung teilweise stark in den angewandten Methoden und somit auch in der Art der Resultate unterscheiden. Einige Ergebnisse beschreiben die Symmetriermittlung für ein Bild einer Textur oder eines Objektes vor einem einfarbigen Hintergrund und sind damit ohne objektextrahierende Vorverarbeitungsschritte eher ungeeignet für die im Bereich Robotersehen erwünschte Bestimmung von Landmarken aus Szenenbildern. Viele Methoden lassen Spiegelsymmetrie beliebiger Orientierung einfließen oder geben erste Ansätze zur Extraktion drehsymmetrischer Eigenschaften (siehe z.B. Arbeiten von Johansson et al. [11]) und sind somit sehr rechenaufwändig.

### 3 Methode

Mit dem im Folgenden beschriebenen Ansatz wird dagegen ein einfacher, schneller und kompakter Operator zur Verfügung gestellt, der Symmetriewerte beliebiger Objekte ermittelt und den Echtzeitanforderungen von Roboteranwendungen trotzdem entgegenkommt. Die genannten psychologischen Experimente zeigen, dass vertikale und horizontale Spiegelsymmetrien die wichtigsten Symmetriemerkmale in der menschlichen Wahrnehmung sind. Auf der Basis dieser Ergebnisse werden nur diese speziellen Symmetriearten in den Operator eingebunden. Diese Auswahl zeigt sich auch als vorteilhaft, wenn in Betracht gezogen wird, dass ein digitales Bild durch ein zweidimensionales Feld horizontal und vertikal angeordneter Bildelemente dargestellt wird und Probleme wie Interpolation und die Benutzung trigonometrischer Funktionen nicht auftreten. Somit müssen zur Bestimmung der vertikalen Symmetrieeigenschaften<sup>3</sup> eines Pixels  $p_{x,y}$  nur Bildpunkte in derselben Bildreihe  $R(y)$  in Betracht gezogen werden. Entsprechend für horizontale Symmetrie die Bildpunkte in derselben Bildspalte  $C(x)$ . Da sich die Methoden aber entsprechen, wird im Weiteren nur auf den vertikalen Fall eingegangen.

Ein wichtiger Aspekt des Robotersehens ist die Verarbeitung von Realbildern. Da diese im Allgemeinen verrauscht sind, würde ein Operator zur Bestimmung exakter Symmetrie inakzeptable Ergebnisse liefern. Aus diesem Grund wird der folgende Operator, basierend auf einer normalisierten mittleren quadratischen Fehlerfunktion, zur Ermittlung von vertikaler Symmetrie vorgeschlagen:

$$S_v(p_{x,y}, m) = 1 - \frac{1}{N \cdot m} \sum_{i=1}^m \sigma(i, m) \cdot g_v(p_{x-i,y}, p_{x+i,y})^2 \quad (1)$$

<sup>3</sup> Im Folgenden wird unter Symmetrie generell Spiegelsymmetrie verstanden.

$N$  beschreibt eine Normalisierungskonstante, die gemäß der Farbtiefe des Bildes und abhängig von einer zur Gewichtung der Fehler genutzten radialen Basisfunktion  $\sigma(i, m)$  gewählt werden muss. Hierdurch wird ein normierter Symmetriewert gewährleistet, der für Analyse und Weiterverarbeitung von Vorteil ist. Die Maskengröße  $m$  gibt die für die Symmetriebestimmung einzubeziehende Umgebung an, in der ein Symmetriewert für  $p(x, y)$  bestimmt werden soll. Die Differenz zwischen zwei sich gegenüberliegenden Bildpunkten wird durch die Gradientenfunktion  $g_v$  bestimmt, gängigerweise als Euklidische Distanz zwischen den entsprechenden Farbvektoren  $p_{x-i,y}$  und  $p_{x+i,y}$ :

$$g_v(p_{x-i,y}, p_{x+i,y}) = \begin{cases} \|p_{x-i,y} - p_{x+i,y}\| & \text{if } p_{x-i,y} \in R(y) \wedge p_{x+i,y} \in R(y) \\ c & \text{sonst,} \end{cases} \quad (2)$$

wobei  $c$  den maximal errechenbaren Fehler beschreibt. Dadurch wird die Gesamtfehlerfunktion anteilig stark angehoben, sobald ein Bildpunkt in Betracht gezogen wird, der außerhalb des Bildes liegt. Ein Beispielresultat des Operators ist in Abb. 3(d) zu sehen. Insbesondere sind hier folgende Punkte die Operatorgröße betreffend zu beachten:

- Symmetrie ist ein Merkmal, das mit der Größe der betrachteten Region an Bedeutung gewinnt. Je kleiner  $m$ , desto weniger aussagekräftig ist der ermittelte Wert.
- Größere Masken fallen häufiger neben den sichtbaren Bildbereich. Durch diese Unsicherheiten werden Werte am Bildrand stärker geschwächt.
- Je größer  $m$  gewählt wird, desto aufwändiger die Berechnung und desto höher die Störungsanfälligkeit, z.B. durch Schattenwurf, Verdeckung oder perspektivische Verzerrung.

Die diesbezügliche Schwäche weisen die meisten der Ansätze auf, daher beschreibt die Wahl von  $m$  den wesentlichen Nachteil der bekannten Methoden aus der Bildverarbeitung. Fast durchgängig wird ein Operator einer auszuwählenden Maskenweite verwendet, um Symmetrie in einem zuvor definierten Bereich zu ermitteln. A priori ist es aber ohne Vorwissen unmöglich, genau die Maskenweite zu wählen, die das beste Ergebnis liefert. Die Qualität des Ergebnisses hängt von der Übereinstimmung der Maskenweite mit der im Bild anzufindenden Weite der symmetrischen Region ab.

Eine interessantere Möglichkeit der Merkmalsextraktion besteht in der Detektierung von Symmetrieachsen als visuelle Landmarken. In den durch den Operator erzeugten Symmetriebildern können die Achsen nicht notwendigerweise dort gefunden werden, wo hohe Symmetriewerte auftreten, aber dort, wo diese ein Maximum annehmen. Obwohl die Extraktion von Extremwerten eines Symmetriebildes leichtes Rauschen in den erzeugten Binärbildern verursacht, ist sie wesentlich signifikanter als die Anwendung eines Schwellwertes, da der optimale Schwellwert von Anwendung zu Anwendung oder sogar von Bild zu Bild variieren kann.

## 4 Anwendung

Ein Großteil der bisherigen Anwendungen des beschriebenen Operators nutzt Symmetrieachsen als Merkmal statt der Symmetriewerte an sich [9]. In [10] wurde für die Domäne des RoboCup<sup>4</sup> ein Verfahren vorgestellt, welches Linien basierend auf diesem

<sup>4</sup> RoboCup ist eine internationale Initiative zur Förderung der Forschung in den Bereichen Künstliche Intelligenz und Autonome Mobile Roboter, die vor allem durch ihre Aktivität im Roboterfußball großes Interesse hervorruft.

Symmetrieoperator extrahiert. Die durch Symmetrie erkannten Linien können sinnvoll als neues Perzept genutzt werden und sich in weiteren Anwendungen einbringen lassen. Während die zuvor genannten Arbeiten die Ausarbeitung und die Fähigkeiten des entwickelten Symmetrieoperators an sich wiedergeben, wurden Symmetriemerkmale auch auf Panoramabildern berechnet und in darauf aufbauenden Verfahren verwendet. In [16] wurden in einem omnidirektionalen Sichtsystem Symmetriemerkmale für Anwendungen der mobilen Robotik genutzt.

Als neue Anwendung soll hier die Idee motiviert werden, Symmetrie als Datenreduktionsansatz zur Farbklassifikation zu nutzen. Dabei wird die These vertreten, dass durch Symmetrie ausgezeichnete Bildpunkte einen Großteil der signifikanten Farbinformation des Gesamtbildes beinhalten. Zur Prüfung dieses Ansatzes wurde ein semi-automatischer Linde-Buzo-Gray-Algorithmus auf Basis des  $k$ -means-Clusterverfahrens zur Farbklassifikation implementiert. In einem Quellbild vorhandene Farbwerte werden in den YUV-Farbraum übertragen und die sich dort bildenden Punktwolken entsprechend des LBG-Algorithmus als Normalverteilungen angenähert. Die Approximation geschieht automatisch, die Beurteilung und eventuelle Aufspaltung zur erneuten automatischen Anpassung der gefundenen Farbklassen dagegen manuell. Im ersten Schritt (Abb. 5 oben) wird diese Klassifikation auf den Gesamtdaten ausgeführt, im zweiten (Abb. 5 mittig) dann separat mit den auf Symmetrieachsen reduzierten Daten. Abschließend wird eine Farbklassifizierung der Gesamtdaten anhand der im zweiten Schritt ermittelten Farbklassen durchgeführt (Abb. 5 unten).

	Vollständig	Reduziert	Anteil
<i>Datenpunkte</i>	25.344	1.619	6 %
<i>Farbeinträge</i>	752	186	25 %
<i>Ermittelte Klassen</i>	7	13(8)	186 %
<i>Klassifikationsrate</i>	1	0.96	96 %

**Tabelle 1.** Vergleich vollständiger und auf Symmetrien reduzierter Klassifikation (siehe Abb. 5).

Wie auch im Vergleich in Tab. 1 zu erkennen, ist es durch Symmetrieachsendetektion in diesem Beispiel möglich, durch einen Bruchteil der Daten immer noch eine gute Klassifikation zu erreichen. Mit einem optimal klassifizierten Bild wurden die Ergebnisse nicht verglichen, es fällt allerdings auf, dass viele der im vollständigen Fall falsch klassifizierten Daten, die häufig auf den unscharfen Randbereichen liegen, im reduzierten Falle in fünf unbestimmte Klassen aussortiert werden konnten.

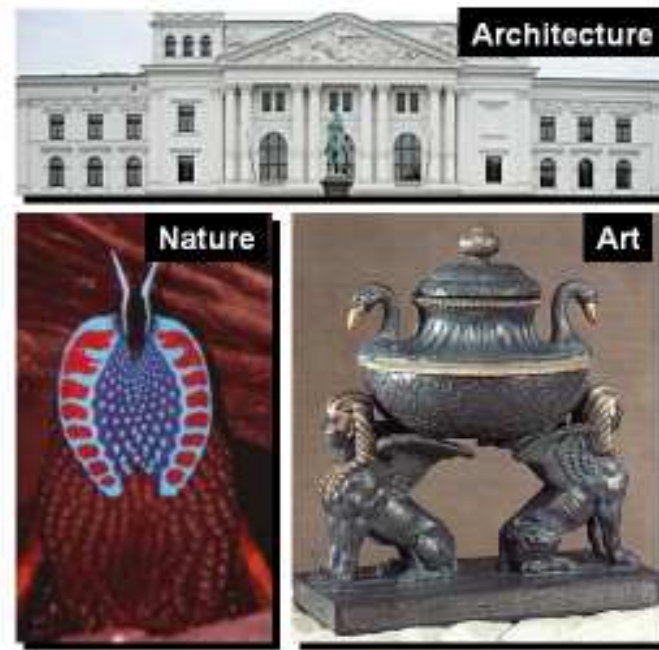
## 5 Zusammenfassung

Symmetrie ist ein nützliches, biologisch motiviertes Merkmal. Die Signifikanz spezieller Spiegelsymmetrien spiegelt sich auch in den Ergebnissen des implementierten Operators wider. Jedoch besitzt der vorgestellte Operator wie auch verwandte Ansätze das Problem der Wahl der Operatorgröße. In zukünftiger Arbeit soll ein laufweitenbasierter Operator entwickelt werden, der als Merkmal die Größe der Umgebung zurückgibt, in der noch ein annehmbares Maß an Symmetrie herrscht, um das genannte Problem zu umgehen. Der Fokus der auf einem Symmetrieoperator basierenden Anwendungen

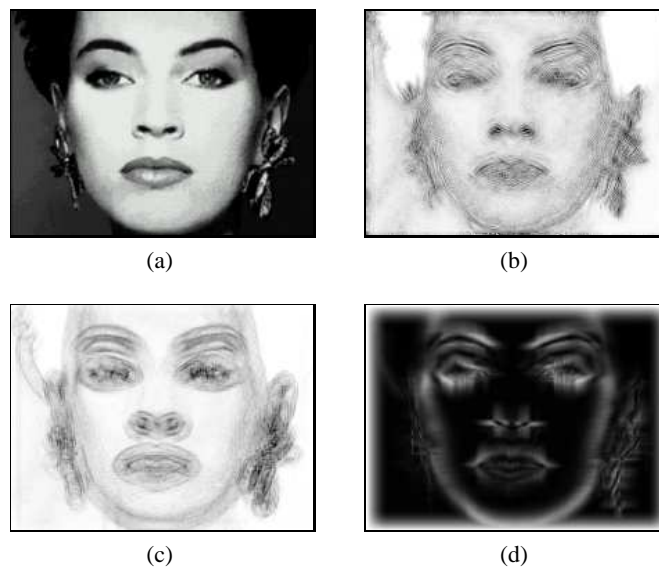
liegt auf der Merkmalsextraktion zur Bestimmung von Landmarken. Mit der vorgestellten Idee zur Farbklassifikation sind die Eigenschaften der Datenreduktion anhand von Symmetrieachsen an einem Beispiel aufgezeigt worden. Eine weitere Ausarbeitung dieses Ansatzes ist allerdings nicht geplant, da sich weitere Ziele auf die Bestimmung von Symmetrielandmarken für Anwendungen der mobilen Robotik konzentrieren.

## Literatur

1. H. B. Barlow and B. C. Reeves. The Versatility and Absolute Efficiency of Detecting Mirror Symmetry in Random Dot Displays. *Vision Research*, 19:783–793, 1979.
2. D. Chetverikov. Fundamental Structural Features in the Visual World. In *Proceedings of the International Workshop on Fundamental Structural Properties in Image and Pattern Analysis*, pages 47–58, Budapest, 1999.
3. D. Chetverikov. Texture Analysis Using Feature Based Pairwise Interaction Maps. *Pattern Recognition, Special Issue on Color and Texture*, 32:487–502, 1999.
4. Á. Csathó, G. van der Vloed, and P. A. van der Helm. Blobs strengthen repetition but weaken symmetry. *Vision Research*, 43:993–1007, 2003.
5. J. D. Delius and B. Nowak. Visual Symmetry Recognition by Pigeons. In *Psychological Research*, volume 44, pages 199–212, 1982.
6. V. Di Gesù and C. Valenti. The Discrete Symmetry Transform in Computer Vision. Technical report, DMA Università di Palermo, 1995.
7. V. Di Gesù and C. Valenti. A New Symmetry Operator for the Analysis of Sequences of Images. 1996.
8. M. Enquist and A. Arak. Symmetry, beauty and evolution. *Nature*, 372:169–172, 1994.
9. K. Huebner. A 1-Dimensional Symmetry Operator for Image Feature Extraction in Robot Applications. *The 16th Int. Conf. on Vision Interface (VI'03)*, pages 286–291, June 2003.
10. K. Huebner. A Symmetry Operator and its Application to the RoboCup. In D. Polani, B. Browning, A. Bonarini, and K. Yoshida, editors, *RoboCup 2003: Robot Soccer World Cup VII*, Lecture Notes in Artificial Intelligence. Springer-Verlag, 2004.
11. B. Johansson. Rotational Symmetries, a Quick Tutorial. December 2003.
12. M. Kirkpatrick and G. G. Rosenthal. Symmetry without fear. *Nature*, 372:134–135, 1994.
13. Y. Liu. *Computational Symmetry*, chapter 21 of *Symmetry 2000*, pages 231–245. Wenner-Gren International Series, Portland Press, London, December 2000. CMU RI-00-31.
14. P. Locher and C. Nodine. The Perceptual Value of Symmetry. *Comput. Math. Applic.*, 17:475–484, 1989.
15. S. E. Palmer and K. Hemenway. Orientation and Symmetry: Effects of Multiple, Rotational, and Near Symmetries. *Journal of Experimental Psychology: Human Perception and Performance*, 4(4):691–702, 1978.
16. D. Popescu, K. Huebner, and J. Zhang. Real-time Intelligent Vision Sensor for Robot Navigation Using Symmetry Features. In *5th International FLINS Conference on Computational Intelligent System for Applied Research*, pages 421–428, 2002.
17. D. Reissfeld, H. Wolfson, and Y. Yeshurun. Context Free Attentional Operators: the Generalized Symmetry Transform. *International Journal of Computer Vision*, 14:119–130, 1995.
18. I. Stewart and M. Golubitsky. *Denkt Gott symmetrisch? Birkhäuser Verlag*, 1992. German translation of the book *Fearful Symmetry: Is God a Geometer?* by Stewart/Golubitsky, ISBN 3-7643-2783-9.
19. C. W. Tyler, editor. *Spatial Vision*, volume 8(4), chapter Special Issue on The Perception of Symmetry. 1994.
20. H. Zabrodsky. Symmetry - A Review. Technical report, Department of Computer Science, The Hebrew University of Jerusalem, May 1990.



**Abb. 2.** Beispiele für Symmetrie im Alltag aus Architektur, Natur und Kunst.



**Abb. 3.** Symmetrieegebnisse verschiedener Operatoren: (a) Originalbild [6] (b) Generalized Symmetry Transform (Reisfeld et al. [17]) (c) Discrete Symmetry Transform (Di Gesù und Valenti [6]) (d) 1D-Operator [9]. Dunklere Grautöne entsprechen jeweils höheren Symmetriewerten.

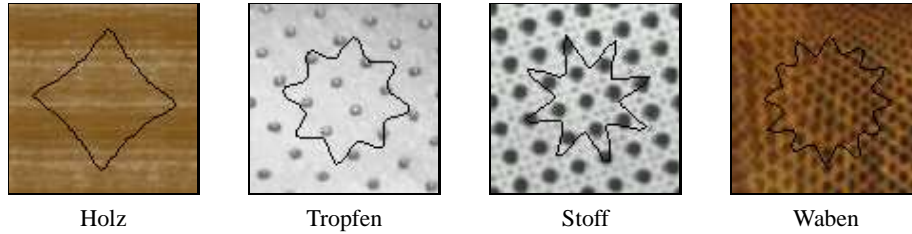


Abb. 4. Symmetriediagramme einiger Texturbeispiele (Algorithmus von Chetverikov [3]).

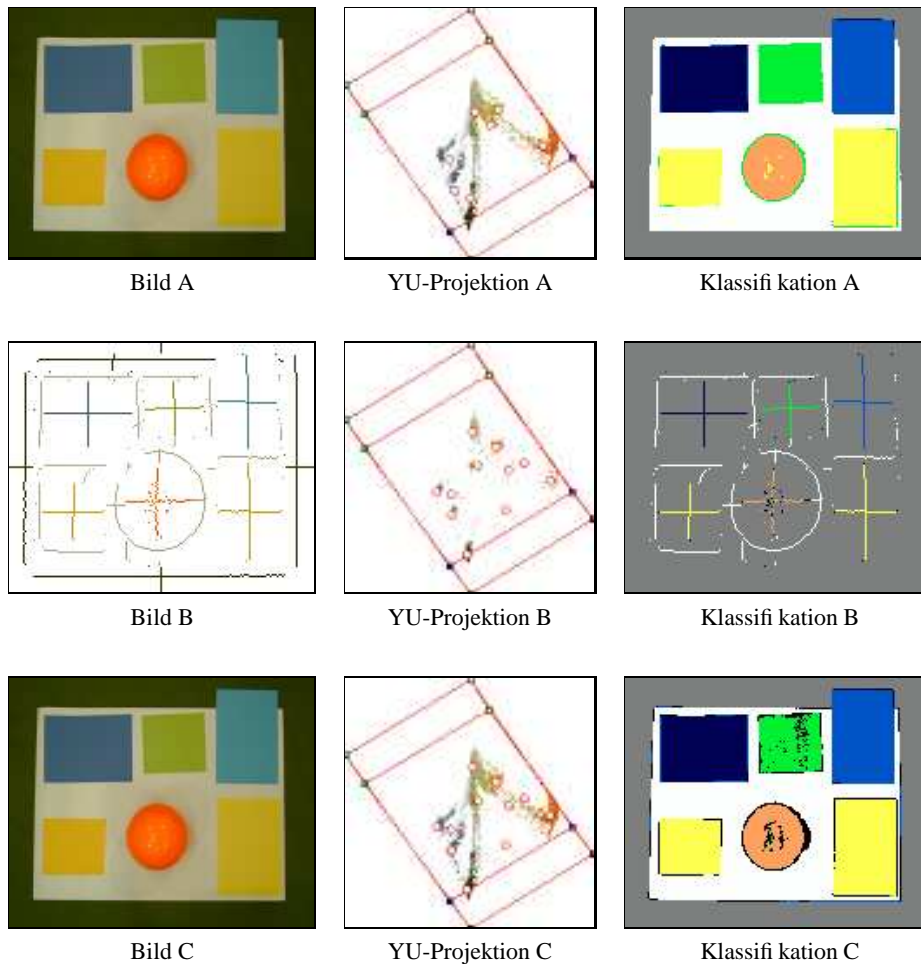


Abb. 5. Oben: Vollständige Klassifikation. Mitte: Auf Symmetrieachsen reduzierte Klassifikation. Unten: Klassifikation des vollständigen Bildes anhand der reduzierten Klassen. In der mittleren Spalte jeweils die Projektion des YUV Farbraums auf die YU-Ebene inklusive der ermittelten Farbklassen als Kreise.



## Shading-Korrektur für Endoskopische Bilder und Fundusbilder

Annika Hirsch<sup>1</sup>, Christian Münzenmayer<sup>2</sup> und Dietrich Paulus<sup>1</sup>

<sup>1</sup> Institut für Computervisualistik  
Universität  $\frac{1}{2}$ Koblenz-Landau  
56070 Koblenz  
[paulus@uni-koblenz.de](mailto:paulus@uni-koblenz.de)

<sup>2</sup> Fraunhofer Institut für Integrierte Schaltungen,  
Am Wolfsmantel 33,  
D-91058 Erlangen

**Zusammenfassung.** In der medizinischen Diagnostik werden Farbkameras eingesetzt. Bei Endoskopen zur Betrachtung des Körperinneren ist eine Beleuchtungsquelle erforderlich. Ebenso wird beim Blick in das Auge mit einer Funduskamera eine Beleuchtung benötigt. In beiden Fällen ist das entstehende Bild nicht gleichmäßig ausgeleuchtet, was durch die Geometrie der Aufnahme bedingt ist. Im Beitrag werden Verfahren vorgestellt, wie diese Beleuchtungsvariation automatisch erkannt und reduziert werden kann, so dass nachfolgende rechnergestützte Verarbeitungsschritte von einer gleichmäßigen Beleuchtung ausgehen können.

### 1 Einleitung

In medizinischen Anwendungen unterliegen endoskopische Bilder speziellen Beleuchtungsschwankungen, die durch die Geometrie der Optik und Lampe bedingt sind. Ähnliche Probleme ergeben sich in der Diagnostik des Auges, wenn so genannte Fundusbilder verwendet werden. Hier ist ebenfalls eine Änderung der Beleuchtung festzustellen, die zusätzlich noch durch die Geometrie des Auges bedingt ist. Auf die Notwendigkeit der Normierung der Eingabebilder in der Ophthalmologie wird beispielsweise in [4, 1] hingewiesen.

Werden so geartete Bilder mittels Bildverarbeitung und Bildanalyse untersucht, so sind die Beleuchtungsänderungen oft Grund für Schwierigkeiten der Verfahren. Es ist daher wünschenswert, die Bilder so zu normalisieren, dass eine Unabhängigkeit von der Beleuchtung entsteht. Zum Ausgleich dieser Schwankungen in endoskopischen Bildern wurden in [5] Verfahren vorgestellt, die auf einer so genannten retrospektiven Filterung beruhen. Im Folgenden werden diese Verfahren auch auf Fundusbildern angewendet. Ein neuer Ansatz basierend auf einer angenommenen Beleuchtungsverteilung wird dann vorgestellt und evaluiert.

Im Abs. 2 stellen wir den Stand der Technik zum Thema Shading-Korrektur vor. In Abs. 3.1 führen wir ein neues lineares Modell ein, das in Abs. 3.2 auf Quadriken erweitert wird. Ergebnisse werden in Abs. 4 vorgestellt. Eine Zusammenfassung in Abs. 5 beschließt den Beitrag.



## 2 Shading-Korrektur

### 2.1 Grauwertbilder

Eine ungleichmäßige Ausleuchtung eines Bilds stellt für den menschlichen Beobachter nicht immer ein Problem dar, da das Gehirn diese Unterschiede auszugleichen scheint. Anders ist es bei der automatischen Analyse mit einem Rechner. Hier müssen diese Unterschiede ausgeglichen werden, falls die ermittelten Merkmale nicht gegenüber Beleuchtungsunterschieden invariant sind. Es ist dabei festzustellen, dass die Beleuchtung über das Bild hinweit meist langsam variiert. Abb. 1 zeigt hierzu ein Beispiel eines Fundus-Bildes.

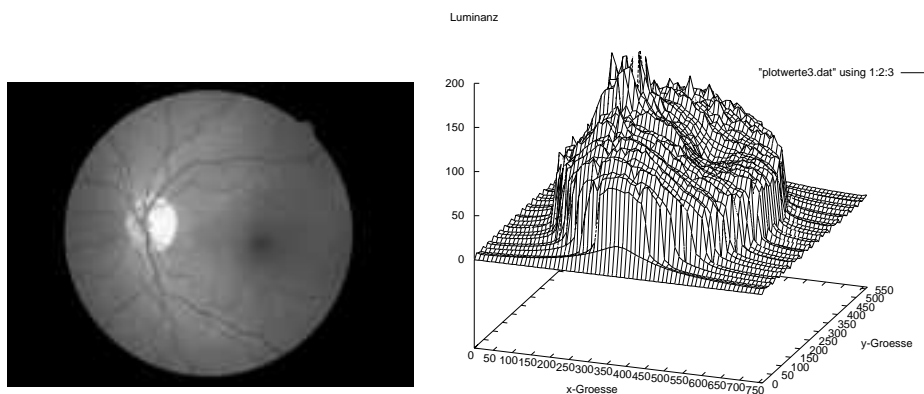


Abb. 1. Fundus-Bild zur Erläuterung der Beleuchtungsunterschiede, rechts ein Plot der Intensität

In der Literatur wird der Zusammenhang zwischen idealem Grauwert-Bild  $f_{ij}$  und verwaschenem, beleuchtungsvariantem Bild  $g_{ij}$ , das aus der realen Kamera stammt, durch ein lineares Modell angegeben [8]:

$$g_{ij} = f_{ij}\gamma_{ij} + \beta_{ij} \quad (1)$$

Die Parameter  $\gamma_{ij}$  und  $\beta_{ij}$  sind positionsabhängige multiplikative und additive Komponenten, die im Allgemeinen unbekannt sind.

Während es beispielsweise in der Industrie möglich ist, mittels Kalibrierenaufnahmen von weißen und schwarzen Oberflächen und von Testobjekten zunächst Daten zu erhalten, aus denen dann die Parameter  $\gamma_{ij}$  und  $\beta_{ij}$  geschätzt werden, ist diese Vorgehensweise in der Medizin oft nicht möglich [5]. In [5] werden daher für endoskopische Bilder des Ösophagus so genannte retrospektive Verfahren angewendet, die die Beleuchtung nur auf Basis des Eingabebilds  $g_{ij}$  schätzen.

Unter der Annahme, dass sich die Beleuchtung langsamer verändert als der Bildinhalt, kann das ideale Bild aus (1) mittels homomorpher Filterung ermittelt werden [8, 5].

In einem Ansatz aus [3] werden Polynome zweiten Grads zur Modellierung der Beleuchtung eingesetzt.

## 2.2 Farbbilder

Wie in [5] dargestellt, lassen sich die Verfahren für Grauwertbilder auch auf Farbbildern anwenden, wenn die üblichen drei-kanaligen Bilder aus dem RGB-Farbraum zunächst in einen Farbraum konvertiert werden, in dem Farbe und Intensität  $\frac{1}{2}$  getrennt sind. Die Verfahren aus Abs. 2.1 werden auf dem Intensitäts-Kanal angewendet und das Bild wird dann in den RGB-Farbraum zurückgeführt. Für die homomorphe Filterung und den YUV-Farbraum ist dies als Diagramm in Abb. 2 dargestellt.

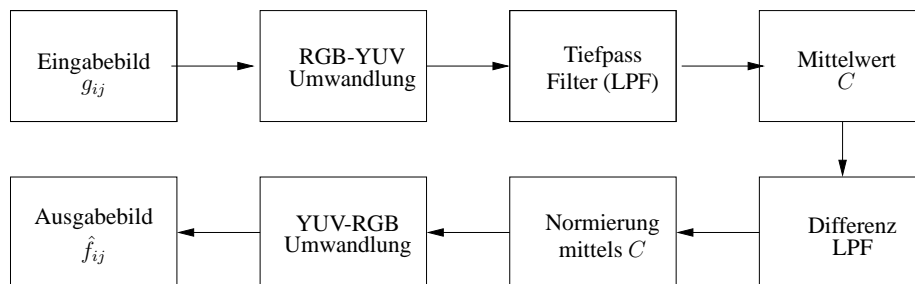


Abb. 2. Farbbild-Shading-Korrektur im Intensitätskanal nach [5]

Ein Ergebnis auf einem endoskopischen Bildern aus [5] ist in Abb. 9 auf Seite 80 dargestellt.

## 3 Beleuchtungskorrektur durch Oberflächenanpassung

### 3.1 Shading-Korrektur durch Ebenenanpassung

Wie im unteren Teil von Abb. 9 zu sehen, ergibt sich im korrigierten Bild bei einem horizontalen Schnitt ein Profil, das durch eine annähernd horizontale Gerade angenähert werden kann. Im Original-Bild ist die Ausgleichsgerade dagegen gekippt. Dieser Effekt rührt von der räumlich langsamen Änderung der Beleuchtung, die im Beispiel von links nach rechts abnimmt.

Hieraus ergibt sich ein neuer Ansatz. Unter der Annahme, dass das Bild eine makroskopisch annähernd homogene Struktur hat, lässt sich durch eine Ebenen-Approximation der Intensitätswerte eine zweidimensionale Beleuchtungsverteilung approximieren.

Mit Pixelpositionen  $(i, j)$ , Pixelgröße  $\Delta_x \times \Delta_y$  und Grauwerten, bzw. Intensitäten  $f_{ij}$  ergibt sich für jeden Bildpunkt die Ebenengleichung

$$a_1 \Delta_x j + a_2 \Delta_y i + a_3 f_{ij} + a_4 = 0 \quad , \quad (2)$$

wobei die Parameter der Ebene  $a_1, a_2, a_3, a_4$  zu bestimmen sind. Zur Berechnung dieser Ebene werden die schwarzen und weißen Pixel des Hintergrunds (vgl. Abb. 9 und Abb. 1) nicht berücksichtigt. Die Lösung mit minimalem quadratischen Fehler führt zu

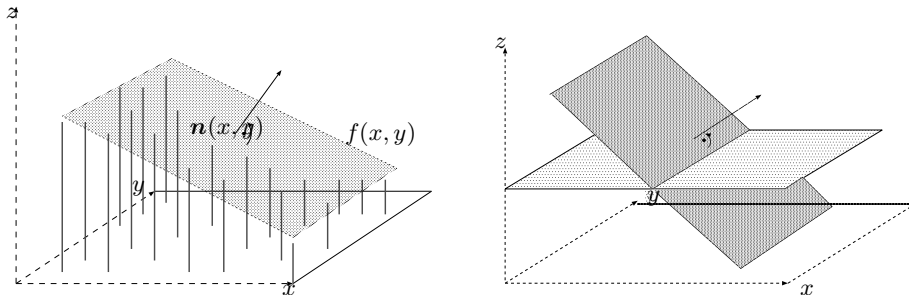


Abb. 3. Illustration Ausgleichsebene

einem Parametervektor  $(a_1, a_2, a_3)^T$ . Bei quadratischen Pixeln kann dabei die Pixelgröße  $\Delta_x \times \Delta_y$  ignoriert werden. Für rechteckige Pixel muss dagegen das Seitenverhältnis  $\Delta_x/\Delta_y$  aus den Kameraangaben entnommen werden.

Die Intensitäten werden nun nach

$$\hat{f}_{ij} = \frac{-\mu a_3}{1 + a_1 j + a_2 i} f_{ij} \quad (3)$$

korrigiert, d. h., jeder Pixel wird in Abhängigkeit seines Abstands zur Ausgleichsebene korrigiert (vgl. Abb. 3).

### 3.2 Shading-Korrektur durch Quadrik-Anpassung

An Stelle der Annahme einer planaren Verteilung in (2) lässt sich auch eine hyperbolische Annahme treffen, wie sie bei einer nahen Punktlichtquelle zu beobachten ist. Die Ebenenanpassung muss dann eine Quadrik approximieren, wobei hier auf die Angabe der Pixelgröße  $(\Delta_x, \Delta_y)$  verzichtet wird. Die allgemeine Quadrik

$$a_1 x^2 + a_2 y^2 + a_3 z^2 + a_4 xy + a_5 xz + a_6 yz + a_7 x + a_8 y + a_9 z + a_{10} = 0 \quad (4)$$

wird für Graph-Oberflächen, die als Funktion nach  $z = f(x, y)$  auflösbar sind, unter Verwendung der diskreten Pixelpositionen  $(i, j)$  zur Formel

$$a_1 j^2 + a_2 i^2 + a_4 ij + a_7 j + a_8 i + a_9 f_{ij} + a_{10} = 0 \quad (5)$$

In den Experimenten führte eine weitere Einschränkung zu

$$a_1 j^2 + a_2 i^2 + a_7 j + a_8 i + a_9 f_{ij} + a_{10} = 0 \quad (6)$$

in der Regel zu besseren Ergebnissen. Die Parameter  $(a_1, a_2, a_7, a_8, a_9, a_{10})$  lassen sich wie in Abs. 3.1 durch ein lineares Gleichungssystem mit minimalem quadratischen Anpassungsfehler lösen. In beiden Anpassungsproblemen ist es numerisch jedoch bedeutsam, die Gleichungen (2) und (6), durch geeignete Skalierung in einen geeigneten Bereich zu bringen; wie in [2] wurde beispielsweise zur Lösung von (2) der Messwertvektor  $\mathbf{m} = (j, i, f_{ij}, 1)$  normiert auf  $\mathbf{m}' = \sqrt{2}\mathbf{m}/\|\mathbf{m}\|$ .

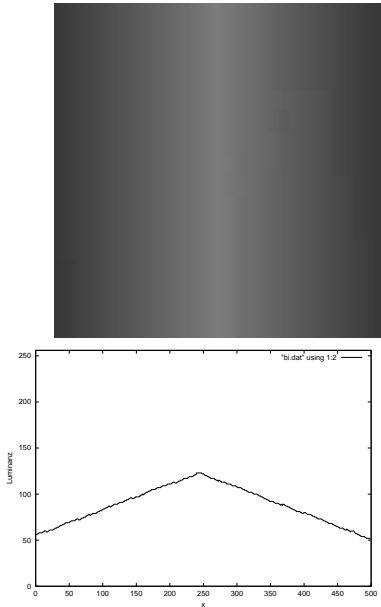


Abb. 4. Bilinearer Verlauf

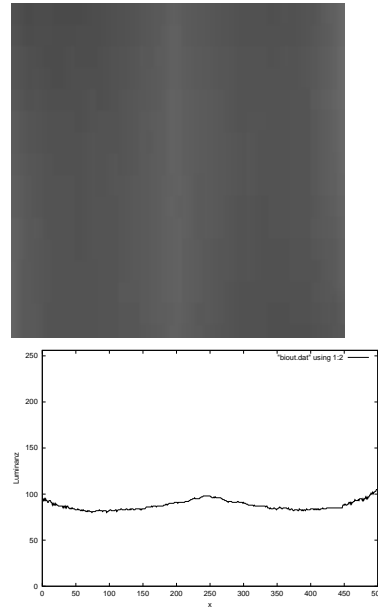


Abb. 5. Quadrikausgleich

## 4 Experimente und Ergebnisse

Abb. 4 zeigt oben ein synthetisches Bild, das horizontal ein dachförmiges Intensitätsprofil hat (Abb. 4 unten). Die Korrektur nach (6) ist in Abb. 5 rechts dargestellt. Die Korrektur eines endoskopischen Bilds ist in Abb. 6 zu sehen. Ausgangsbild und Ergebnisse auf Fundusbildern sind in Abb. 7 und Abb. 8 dargestellt.

Koeff.	Abb. 7	Abb. 7	Abb. 8	Abb. 8
$a_1$	5.93296e-06	-3.76443e-06	-2.30964e-06	3.47189e-06
$a_2$	2.39738e-06	-1.52782e-06	-8.77195e-07	1.3389e-06
$a_4$	0	-2.41213e-06	-2.12514e-06	0
$a_7$	-0.00460803	0.0038473	0.00294476	-0.00316883
$a_8$	-0.00191254	0.00213184	0.00160293	-0.00120388
$a_9$	0.000889694	-0.000428865	0.00142442	-0.00181368
$a_{10}$	1	1	1	1

Tabelle 1. Quadrik-Anpassung mit (6)

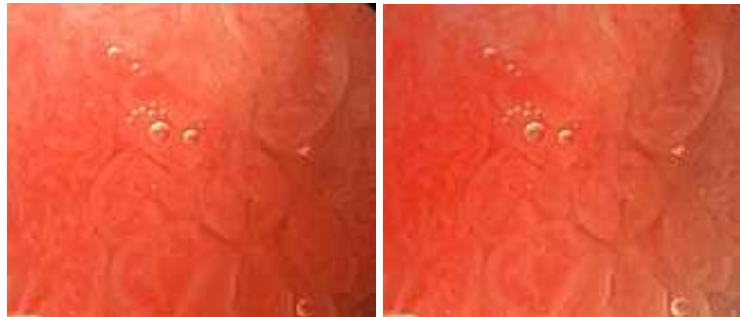
Für Abb. 7 sind die berechneten Parameter wie folgt: der Ausgleich mittels (2) liefert

$$-0.000788394x + -0.000415322y + -0.00519192z + 1 = 0 \quad .$$

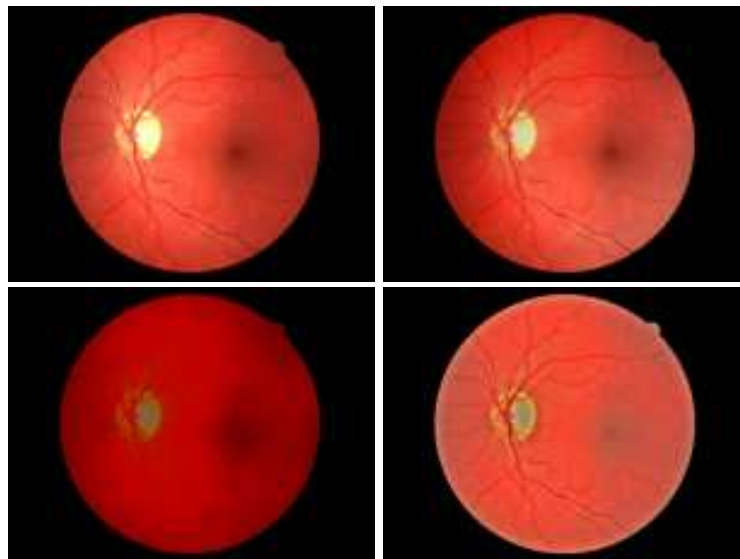
Die entsprechende Gleichung für Abb. 8 ist

$$-0.001316x + -0.000449263y + -0.00454655z + 1 = 0 \quad .$$

Die Koeffizienten der Anpassung mit (5) und (6) für die beiden Bilder sind in Tabelle 1 zusammengesfasst.

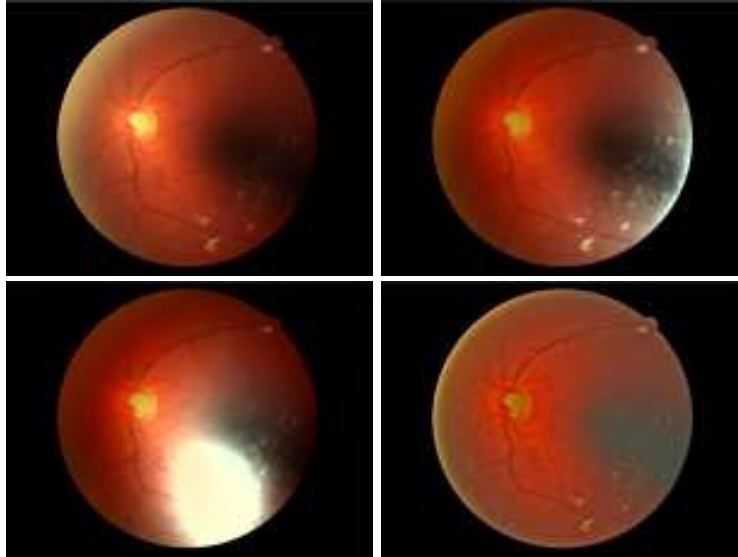


**Abb. 6.** Eingangsbild (links) und korrigiertes Bild (rechts) unter Verwendung von (2)

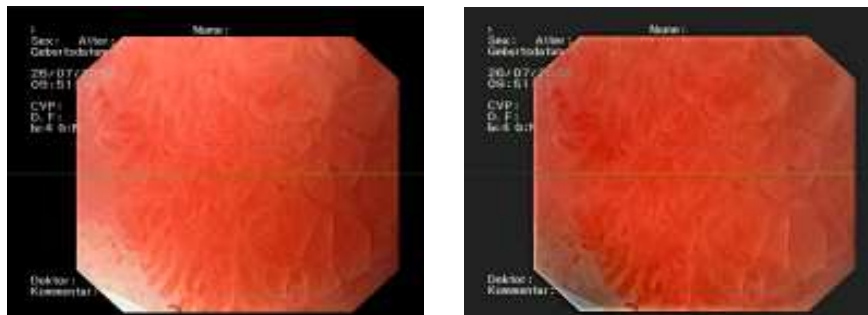


**Abb. 7.** Eingangsbild (oben links) und korrigierte Bilder unter Verwendung von (2) oben rechts, (6) unten links, homomorpher Filterung (Abb. 2) unten rechts

Abb. 7 zeigt, dass homomorphe Filterung und Ebenenanpassung zu einem erkennbaren Helligkeitsausgleich im Bild führen. Bei der Ebenenanpassung ist im Bereich des Schattens der Iris, der als dunkler Bereich rechts im Bild zu sehen ist, ein höherer Kontrast festzustellen, als bei der homomorphen Filterung. Letzere führt jedoch zu einer

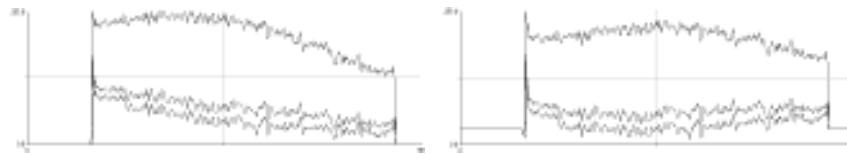


**Abb. 8.** Eingangsbild (oben links) und korrigierte Bilder unter Verwendung von (2) oben rechts, (6) unten links, homomorpher Filterung (Abb. 2) unten rechts.



(a) Eingangsbild

(b) YUV Glättung



(c) RGB Linien Profil

(d) RGB Linien Profil

**Abb. 9.** Ergebnis auf einem endoskopischen Bild

deutlich homogeneren Helligkeit im Bild. Diese Ergebnisse waren auch auf anderen Bildern festzustellen.

Wie in Abb. 8 zu sehen ist, führen bei extremen Beleuchtungen die Ebenenanpassungen zu schlechteren Ergebnissen, als die homomorphe Filterung, da es durch die großen Anpassungsfaktoren aus (3) zu Über- und Untersteuerung kommt. Aufnahmen dieser Qualität sind jedoch in der ärztlichen Praxis selten und werden in der Regel ohnehin wiederholt.

Die Anpassung einer Quadrik führt in Abb. 7 zu einem gleichmäßigen, aber relativ kontrastarmen Bild. In Abb. 8 führt der dunkle Schattenbereich zu einer Oberflächenform, die bei der Helligkeitsanpassung zu einem starken Übersprechen führt.

## 5 Zusammenfassung und Ausblick

An zwei medizinischen Anwendungsbeispielen aus Endoskopie und Ophthalmologie wurde gezeigt, wie Farbbilder so normiert werden können, dass Beleuchtungsunterschiede eliminiert werden. Die vorgestellten neuen Verfahren zur Ebenenanpassung wurden qualitativ verglichen mit dem bekannten Verfahren der homomorphen Filterung.

In [5] wurde gezeigt, wie Texturklassifikation von endoskopischen Bildern mit Merkmalen aus [6] und [7] durch die Anwendung von homomorpher Filterung verbessert werden kann. Ein quantitativer Vergleich der Klassifikationsraten nach Shading-Korrektur mittels Ebenenanpassung steht noch aus.

Ebenso ist es geplant, einen Vergleich von Registrierungsverfahren von Fundusbildern mit HRT Bildern mit und ohne Shading-Korrektur durchzuführen. Für intensitätsabhängige Abstandsmaße ist hier mit einer Verbesserung der Registrierungsergebnisse zu rechnen.

## Dank

Die Autoren danken Frau PD Dr. B. Mayinger und Herrn PD Dr. S. Mühldorfer für die freundliche Überlassung von endoskopischem Bildmaterial. Die Bilder der Augen wurden uns dankenswerter Weise von der Universitätsklinik Erlangen (Prof. Dr. G. Michelson) und der Praxis Dr. Schmitz-Valckenberg & Dr. Derse, Koblenz überlassen.

## Literatur

1. Radim Chrastek, Matthias Wolf, Klaus Donath, Heinrich Niemann, Torsten Hothorn, Berthold Lausen, Robert Lämmer, C. Y. Mardin, and Georg Michelson. Automated segmentation of the optic nerve head for glaucoma diagnosis. In Thomas Wittenberg, Peter Hastreiter, Ulrich Hoppe, Heinz Handels, Alexander Horsch, and H. P. Meinzer, editors, *Bildverarbeitung für die Medizin 2003: Algorithmen - Systeme - Anwendungen*, Informatik aktuell, pages 338–342, Erlangen, 2003. Springer-Verlag, Berlin, Heidelberg, New York.
2. Richard I. Hartley and Andrew Zisserman. *Multiple View Geometry in computer vision*. Cambridge University Press, Cambridge, 2000.

3. B. Likar, J.B.A. Maintz, M.A. Viergever, and F. Pernus. Retrospective shading correction based on entropy minimization. *Journal of Microscopy*, 197(3):285–295, 2000.
4. Georg Michelson, M. Groh, P. Pogorelov, M. Scibor, Radim Chrastek, K. Keppler, and J. Keller. Projekt e-EyeCare: Augenheilkunde im Web. *Deutsches Ärzteblatt, Supplement: Praxis Computer*, 99(11):20–22, 2002.
5. Christian Münzenmayer, Frederic Naujokat, Steffen Mühldorfer, and Thomas Wittenberg. Enhancing texture analysis by color shading correction. In Karl-Heinz Franke, editor, *9. Workshop Farbbildverarbeitung*, Ilmenau, 2003. Zentrum für Bild- und Signalverarbeitung e.V.
6. C. Münzenmayer, H. Volk, C. Küblbeck, K. Spinnler, and T. Wittenberg. Multispectral texture analysis using interplane sum- and difference-histograms. In Luc Van Gool, editor, *Pattern Recognition - Proceedings of the 24th DAGM Symposium Zurich, Switzerland, September 2002*, pages 25–31, Berlin, 2002. Springer.
7. C. Münzenmayer, H. Volk, D. Paulus, F. Vogt, and Wittenberg. Statistical geometrical features for texture analysis and classification. In *8. Workshop Farbbildverarbeitung, Autorenvorträge*, pages 87–94, Ilmenau, 2002. Zentrum für Bild- und Signalverarbeitung e.V. Ilmenau.
8. D. Tomazevic, B. Likar, and F. Pernus. A comparison of retrospective shading correction techniques. In *International Conference on Pattern Recognition (ICPR'00)*, volume 3, pages 564–567. IEEE, 2000.



## Farbkalibrierung mittels linearer Transformation

J. Michel<sup>1</sup>, V. Hong<sup>1</sup>, D. Paulus<sup>1</sup>, C. Münzenmayer<sup>2</sup>

<sup>1</sup> Institut für Computervisualistik, Universität Koblenz-Landau, Universitätsstr. 1, 56070  
Koblenz – Germany, {jmichel,hong,paulus}@uni-koblenz.de

<sup>2</sup> Fraunhofer-Institut für Integrierte Schaltungen, Am Wolfsmantel 33,  
91058 Erlangen mzn@iis.fraunhofer.de

**Zusammenfassung.** In diesem Paper wird ein zweistufiges Verfahren zur Farbkalibrierung vorgestellt, welches auf [2, 6, 5] basiert. Unter Farbkalibrierung ist in diesem Kontext die farbmetrische Charakterisierung und die Normierung von Bilderfassungsgeräten zu verstehen, die dem Ziel dient, aufgenommene Szenen in Szenen unter zuvor definierten Referenzbedingungen zu überführen. Ausgangspunkt für das Verfahren ist eine Referenzszene, die unter zuvor festgelegten Bedingungen aufgenommen wurde. Weiterhin existiert eine Menge von zu kalibrierenden Szenen, die unter gleichen Bedingungen aufgenommen wurden. Im ersten Schritt des Verfahrens wird eine lineare Transformation bestimmt, die die Farbwerte der zu kalibrierenden Szenen auf die der Referenzszene abbildet. Im zweiten Schritt werden mit der ermittelten Transformation die zu kalibrierenden Szenen in Szenen unter Referenzbedingungen überführt.

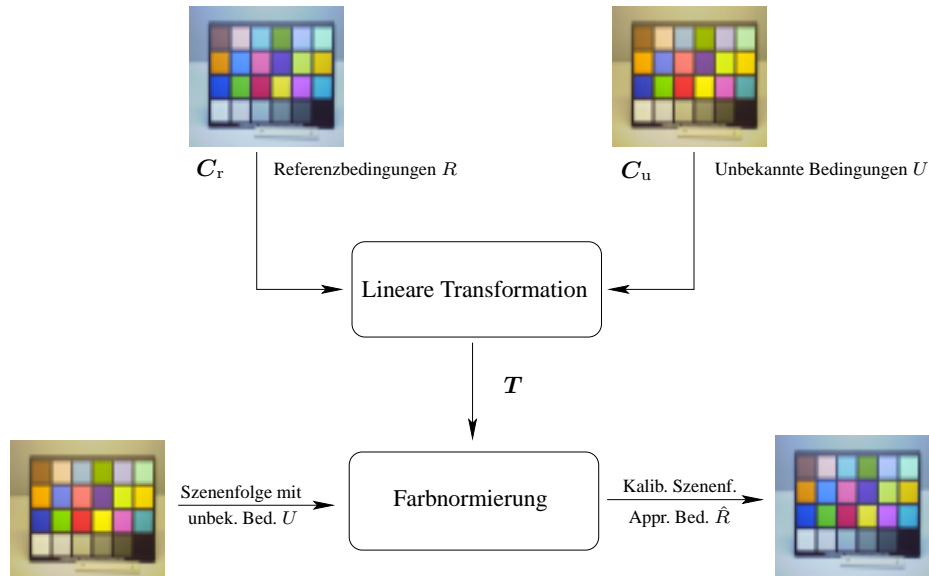
### 1 Einleitung

Eine in der digitalen Bildverarbeitung häufig vorkommende Aufgabe ist der direkte Vergleich von Farbbildern. Beispiele hierfür sind medizinische Aufnahmen wie z. B. mikroskopische und endoskopische Bilder, die Objekterkennung oder auch die Qualitätskontrolle in Bereich der Industrie, bei der es um die Erkennung von Produktionsfehlern geht. Als sehr störend werden hier Faktoren empfunden, die die Farbinformationen ungewollt beeinflussen oder verfälschen. Diese Störeinflüsse können unter anderem in sich verändernden Beleuchtungsverhältnissen oder dem Einsatz verschiedener Bilderfassungsgeräte begründet sein. Mit einer Farbkalibrierung können diese negativen Faktoren kompensiert werden.

Konkret geht es in diesem Beitrag darum, Bilderfassungsgeräte wie Scanner oder digitale (Video-) Kameras farbmetrisch zu charakterisieren. Diese Charakterisierung ermöglicht es letztendlich, die erzeugten Farbwerte, die in einem gerätespezifischen Farbraum<sup>3</sup> vorliegen, in einen geräteunabhängigen Normfarbraum<sup>4</sup> gemäß wohldefinierter Bedingungen zu überführen. Die hier vorgestellte Farbkalibrierung wird im RGB-Farbraum durchgeführt, ermöglicht aber dennoch eine Normlicht-Szenenbeleuchtung zu approximieren und vom Bilderfassungsgerät zu abstrahieren. Es wird also in erster Linie eine Invarianz bezüglich der Szenenbeleuchtung und des verwendeten Bilderfassungsgerätes angestrebt.

<sup>3</sup> Der geräteabhängige Farbraum wird von den Sensorelementen (CCD-Chips) aufgespannt.

<sup>4</sup> Normfarbräume sind unter anderem CIE<sup>5</sup> XYZ oder CIE L\*a\*b\*.



**Abb. 1.** Ein schematischer Überblick über das grundlegende Verfahren.

In Abschnitt 2 werden das Vorgehen und die verschiedenen Vorverarbeitungsschritte im Einzelnen beschrieben. In Abschnitt 3 werden die durchgeführten Experimente und deren Ergebnisse vorgestellt und Abschnitt 4 gibt eine kurze Zusammenfassung.

## 2 Vorgehen

In Abb. 1 ist das in diesem Beitrag verwendete Verfahren dargestellt. Ausgangspunkt ist zum einen eine Szenenaufnahme unter Referenzbedingungen  $R$  und zum anderen die zu kalibrierende Szenenaufnahme mit unbekanntem Bedingungen  $U$  gegeben. Beide Aufnahmen enthalten dasselbe Farbkalibrieremuster. Bei der in der Referenzszene vorherrschenden Beleuchtung kann es sich um eine Normlichtart wie CIE D65, D50, A, B, C oder eine beliebige andere handeln, die approximiert werden soll. Zunächst wird zwischen einer zu kalibrierenden Szene und der Referenzszene ein linearer Zusammenhang in Form einer linearen Transformationsmatrix hergestellt (vgl. Abschnitt 2.1). Im Anschluß daran können Szenen mit denselben unbekanntem Bedingungen  $U$  in Szenen unter ähnlichen Bedingungen  $\hat{R}$  wie die der Referenzbedingungen mittels dieser Transformationsmatrix überführt werden (vgl. Abschnitt 2.2).

### 2.1 Bestimmung der Transformationsmatrix

Die Grundannahme des Verfahrens ist, dass eine Funktion  $\Phi$  existiert, die Farbvektoren  $c_u \in [0, 255]^3$  unter unbekanntem Aufnahmebedingungen<sup>6</sup> Farbvektoren  $c_r \in [0, 255]^3$

<sup>6</sup> Dies umfasst sowohl die unbekanntem Beleuchtungscharakteristik in der Szene, als auch die Sensitivitätsfunktionen der Sensorelemente des Bilderfassungsgeräts.

unter Referenzbedingungen zuordnet (s. Gleichung 1).

$$\Phi : [0, 255]^3 \rightarrow [0, 255]^3 \quad \text{mit} \quad \Phi(\mathbf{c}_u) = \mathbf{c}_r \quad (1)$$

Aus den  $n$  Farbflächen eines Farbkalibrierungsmusters werden jeweils  $n$  RGB-Farbvektoren  $\mathbf{c}_{ri}$  aus der Referenzszene und  $\mathbf{c}_{ui}$  aus der zu kalibrierenden Szene mit  $i = 1, \dots, n$  gewonnen.  $\mathbf{c}_{ri}$  bzw.  $\mathbf{c}_{ui}$  ist dabei der Farbvektor der  $i$ -ten Farbfläche des Kalibrierungsmusters aus Referenzszene bzw. der zu kalibrierenden Szene. Diese Farbvektoren werden in den Matrizen  $\mathbf{C}_r$  und  $\mathbf{C}_u$  zusammengefasst, wobei  $\mathbf{C}$  für „color“, der Index  $r$  für die Referenzszene und der Index  $u$  für die unkalibrierte Szene stehen:

$$\mathbf{C}_r = [\mathbf{c}_{ri}]_{i=1, \dots, n} \quad \text{mit} \quad \mathbf{c}_{ri} = (R_{ri}, G_{ri}, B_{ri})^T \quad (2)$$

$$\mathbf{C}_u = [\mathbf{c}_{ui}]_{i=1, \dots, n} \quad \text{mit} \quad \mathbf{c}_{ui} = (R_{ui}, G_{ui}, B_{ui})^T \quad (3)$$

In diesem Beitrag wird, in Anlehnung an das vereinfachte Bildentstehungsmodell wie z. B. aus [7], von einer linearen Zuordnungsfunktion  $\Phi$  mit  $\Phi(\mathbf{c}) = \mathbf{T}\mathbf{c}$  ausgegangen. D. h. gesucht ist eine Transformationsmatrix  $\mathbf{T}$ , die für jeden Farbvektor  $\mathbf{c}_u$  eine möglichst gute Approximierung des korrespondierenden Farbvektors  $\mathbf{c}_r$  unter der Referenzbedingung liefert.

$$\forall i \in \{1, \dots, n\} : \mathbf{c}_r \approx \Phi(\mathbf{c}_u) \quad (4)$$

Die Referenzbedingungen werden durch die RGB-Ausgabe des Erfassungsgerätes für die Referenzszene dargestellt, die sowohl Beleuchtungseinflüsse, Sensorcharakteristiken, als auch Objekteigenschaften der Szene enthält[7].

Die Transformationsmatrix  $\mathbf{T}$  kann mit Matrizen verschiedener Dimension wie z. B.  $3 \times 3$ ,  $3 \times 4$  und  $3 \times 10$  beschrieben werden[6, 5]. Bei der  $3 \times 3$  Transformationsmatrix können beliebige affine Transformationen im RGB-Raum mit Ausnahme der Translation beschrieben werden. Bei  $3 \times 4$  Version werden die RGB-Vektoren als homogene Koordinaten ausgedrückt, damit auch eine konstante Translation im RGB-Raum in der Transformationsmatrix möglich ist. Bei der Berechnung einer Transformationsmatrix der Dimension  $3 \times 10$  wird die Zuordnungsfunktion  $\Phi$  durch ein Polynom zweiten Grades beschrieben. Die Transformationsmatrix  $\mathbf{T}$  wurde mit Hilfe der Singulärwertzerlegung bestimmt.

## 2.2 Der Normierungsschritt

In dem Normierungsschritt werden Szenen, die unter denselben Bedingungen wie die der im vorherigen Schritt zu kalibrierenden Szene entstanden sind, in Szenen unter der Referenzbedingung überführt. Zu diesem Zweck wird die in 2.1 erwähnte lineare Transformation  $\Phi$  auf alle Pixel eines Eingabebildes angewendet. Man erhält die zugehörige Approximation der Referenzbedingungen, die auf den  $n$  Patches des Kalibrierungsmusters als Stützstellen basiert. Die Approximation ist nur dann hinreichend genau, wenn die Szene statisch bleibt. Wenn sich beispielsweise die Beleuchtung in der zu normierenden Szene ändert, muss neu kalibriert werden. Ansonsten kann das in der zu normierenden Szene eingebrachte Kalibrierungsmuster nach dem Kalibrierungsschritt entfernt werden.

### 2.3 Vorverarbeitung und Nachverarbeitung

Um die Qualität und die Robustheit der Kalibrierung zu erhöhen, können zusätzliche Vor- und Nachverarbeitungsschritte in das in Abb. 1 vorgestellte Verfahren eingebaut werden.

Als Vorverarbeitung werden eine Linearisierung der RGB-Werte des Eingabegeräts und die Chromatizitäts-Normierung berücksichtigt, die sich auf die RGB-Matrizen  $C_r$  und  $C_u$  beziehen.

Als Nachverarbeitung wird die Intensitäts-Normierung verwendet.

*RGB-Linearisierung:* Da die radiometrischen Antwortfunktionen<sup>7</sup> der Sensoren vieler Aufnahmesysteme nicht linear sind [2, 8], müssen diese erst linearisiert werden. Dies ist insbesondere wichtig, da sonst die Annahme einer linearen Zuordnungsfunktion  $\Phi$  nicht zutrifft. Zu diesem Zweck wurden die sechs Graupatches des GretagMacbeth ColorChecker Kalibrierungsmusters mit Hilfe eines Spektralphotometers vermessen und mit der mittleren Intensität der RGB-Ausgabe für diese Patches in Beziehung gesetzt<sup>8</sup>. Die sechs Graupatches des Kalibrierungsmusters weisen den gleichen Abstand bezüglich ihrer spektralen Reflektivitäten zueinander auf – die Reflektivitäten von 0 bis 1 sind also äquidistant auf sie verteilt.

*Chromatizitäts-Normierung:* Ziel dieser Normierung ist es, die Einflüsse der Beleuchtungsintensität in der Szene für die Transformation zu eliminieren. Dabei wird angenommen, dass die Antwortfunktionen der Kamera für die einzelnen Farbkanäle linear sind. D. h. eine Skalierung der Beleuchtungsintensität um einen Faktor beeinflusst die Farbwerte in dem Bild um denselben Faktor [1]. Der Einfluß der Beleuchtungsintensität kann dadurch reduziert werden, dass jeder Farbpixel durch seine Intensität dividiert wird.

*Intensitäts-Normierung:* Die Intensitäts-Normierung soll sicherstellen, dass die Helligkeit in der Szene konstant bleibt. Dafür wird jedes Zeilenelement von  $T$  durch seine Zeilensumme dividiert [6]. Die Zeilensummen der so normierten Transformationsmatrizen beträgt 1, wodurch sichergestellt ist, dass durch die Transformation mit dieser Matrix der ursprüngliche Wertebereich erhalten bleibt.

## 3 Experimente

Für die Experimente wurden verschiedene Szenen unter einer Just Pantone Color Viewing Light Normlicht-Box mit einer CCD<sup>9</sup>-Chip-Kamera (Sony DFW-X700) erfasst. Zur Evaluierung wurden sieben Testszenen jeweils unter den Standardbeleuchtungen CIE D50 und CWF<sup>10</sup> aufgenommen. Als Farbkalibrierungsmuster kam ein GretagMacbeth ColorChecker zum Einsatz.

<sup>7</sup> Relation zwischen Kamera-Output und Intensität des einfallenden Lichtes [3]

<sup>8</sup> Dank an Herrn Prof. Slowak (Fachhochschule Koblenz) für die photometrische Vermessung des Kalibrierungsmusters

<sup>9</sup> Charge Coupled Device

<sup>10</sup> Cool White Fluorescent-Store



**Abb. 2.** Aufnahmen von der zweiten Szene unter verschiedenen Bedingungen (Links: CWF-Beleuchtungsbedingungen; Mitte: Referenzbedingungen (D50); Rechts: Approximation)



**Abb. 3.** Aufnahmen von der vierten Szene unter verschiedenen Bedingungen (Links: CWF-Beleuchtungsbedingungen; Mitte: Referenzbedingungen (D50); Rechts: Approximation)

Auf Basis der ersten Szene der Testreihe wurde die Transformationsmatrix bestimmt, die bei gegebener CWF-Beleuchtung die D50-Referenzbeleuchtung approximiert. Die restlichen Testszenen enthielten unterschiedliche Zusammenstellungen von alltäglichen Büroartikeln.

Für die Bewertung der Kalibrierungsgüte wurden zwei Bewertungsfunktionen verwendet, CIELAB und MSE<sup>11</sup>. Mit der Farbmessung CIELAB wird der euklidische Farbabstand zweier Farbvektoren im CIE  $L^*a^*b^*$ -Farbraum bestimmt. In diesem Paper wurde der gemittelte CIELAB-Wert zwischen zwei Bildern verwendet. Beim MSE wird der gemittelte, quadratische euklidische Abstand zweier RGB-Bilder betrachtet.

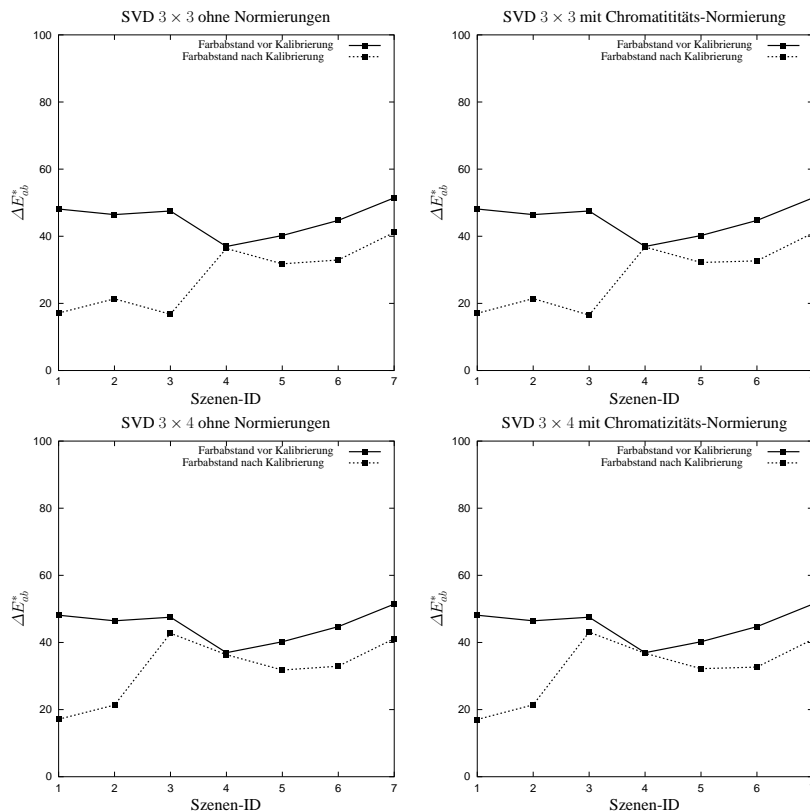
Für die Evaluation wurden nur die Transformationsmatrizen der Dimensionen  $3 \times 3$  und  $3 \times 4$  näher betrachtet. Außerdem wurde die Intensitäts-Normierung in diesem Paper nicht berücksichtigt, da die Experimente in [4] zeigten, dass die Kalibrierungsergebnisse durch diese Normierung signifikant verschlechtert wurden.

In Abb. 2 und 3 sind exemplarisch das Ausgangsbild (links), das Referenzbild (mitte) und rechts dessen Approximation mit dem SVD  $3 \times 4$ -Verfahren<sup>12</sup> und der Chromatizitäts-Normierung der zweiten und vierten Szene der Testreihe aufgeführt. Während mit der zweiten Szene gute Kalibrierergebnisse erzielt wurden, liefert die vierte Szene die schlechtesten Ergebnisse im Experiment.

Die Ergebnisse der Kalibrierung für die Testszenen sind in Abb. 4 mit CIELAB ( $\Delta E_{ab}^*$ ) als Farbmessung und in Abb. 5 mit MSE als Farbabweichung aufgeführt. Für die

<sup>11</sup> Mean Square Error

<sup>12</sup> Berechnung einer Transformationsmatrix der Dimension  $3 \times 4$  mit Hilfe der Singulärwertzerlegung (SVD).



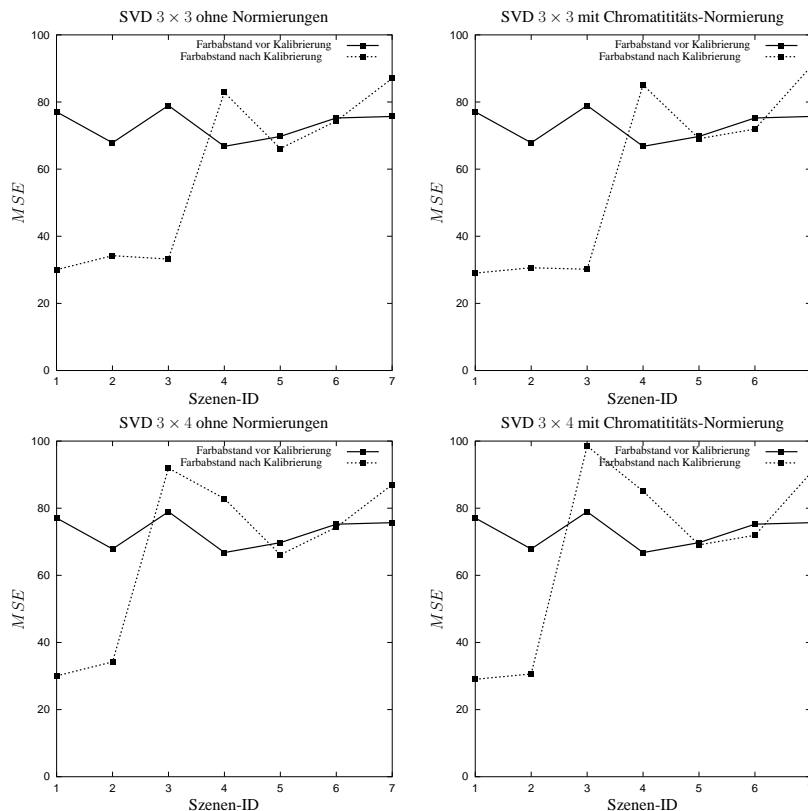
**Abb. 4.** Kalibrierergebnisse beurteilt nach CIELAB

Berechnung dieser Farbabstände wurden jeweils die gesamten Aufnahmen pixelweise miteinander verglichen. Die durchgezogene Linie beschreibt den Farbabstand zwischen den beiden Beleuchtungsarten vor der Kalibrierung pro Szene an. Die gestrichelte Linie gibt den Farbabstand zwischen der D50-Szenenaufnahme und deren Approximation an.

In Abb. 4 ist erkennbar, dass der Farbabstand zwischen Approximation und Referenzbild (D50-Beleuchtung) kleiner ist, als dies vor der Kalibrierung zwischen Referenzbild und zu kalibrierender Szene (CWF-Beleuchtung) der Fall war. Weiterhin fällt auf, dass die Verwendung der Chromatizitäts-Normierung keinen großen Einfluß auf die Kalibrierungsergebnisse hat (vgl. linke Spalte mit rechter Spalte). Die Wahl des Transformationsverfahrens (SVD  $3 \times 3$ <sup>13</sup> vs. SVD  $3 \times 4$ ) zeigt mit einer einzigen Ausnahme (s. Szenen-ID 3) keinen nennenswerten Unterschied in Bezug auf die Kalibrierung.

In Abb. 5 sticht hervor, dass die Ergebnisse mit MSE als Bewertungskriterium sehr stark szenenabhängig sind. Bei einigen Szenen wie z. B. die Szene 4 war der betrachtete

<sup>13</sup> Berechnung einer Transformationsmatrix der Dimension  $3 \times 3$  mit Hilfe der Singulärwertzerlegung (SVD).



**Abb. 5.** Kalibrierergebnisse beurteilt nach MSE

Farbabstand nach der Kalibrierung sogar größer als zuvor. Analog zu Abb. 4 fällt auf, dass die Verwendung der  $3 \times 4$ - und  $3 \times 3$ -Transformationsmatrizen ähnliche Ergebnisse liefert. Der Einsatz der Chromatizitäts-Normierung wirkte sich bei beiden eingesetzten Transformationsverfahren kaum auf die Performanz der Farbkalibrierung aus.

## 4 Fazit

In diesem Beitrag wurde ein zweistufiges Verfahren zur Farbkalibrierung mittels linearer Transformation vorgestellt. Im ersten Schritt wurde eine Transformationsmatrix ermittelt, die einen linearen Zusammenhang zwischen einer Szene unter unbekanntem Bedingungen und einer Szene mit Referenzbedingungen herstellt. Mithilfe dieser Transformationsmatrix konnten Szenen mit den gleichen unbekanntem Bedingungen in Szenen mit Bedingungen ähnlich denen der Referenzbedingungen überführt werden.

Die Evaluation hat gezeigt, dass die Ergebnisse der Farbkalibrierung sehr stark szenenabhängig sind. Dies fällt besonders bei Verwendung des MSE als Evaluationskri-

terium auf. Insgesamt sind die ermittelten Ergebnisse der Experimente nicht sehr zufriedenstellend. Eine mögliche Ursache hierfür ist, dass die Lichtbox die Szene nicht homogen ausleuchtet. Denn in [4] wurde gezeigt, dass Szenen, die homogener als die verwendete Lichtbox ausgeleuchtet waren, deutlich bessere Ergebnisse liefern.

In den hier vorgestellten Ergebnissen wurde ein GretagMacbeth ColorChecker Kalibriermuster mit 24 Farbpatches verwendet. Eine weitere Evaluation bzgl. des Einsatzes anderer Farbkalibriermustern (mit mehr Farbpatches und somit mehr Stützstellen für die Transformationsberechnung) wäre anzudenken.

In diesem Beitrag wurden lediglich Szenen unter CFW-Beleuchtung in Szenen unter D50-Beleuchtung überführt. Man sollte daher auch andere Beleuchtungen in weiteren Tests mitberücksichtigen.

Bei den Experimenten wurde lediglich zwischen den Beleuchtungsarten transformiert, wobei das Bilderfassungsgerät gleich blieb. Nach dem Bildentstehungsmodell [7] lassen sich auch Änderungen der Sensorcharakteristik in einer linearen Transformation ausdrücken.

## Literatur

1. Graham D. Finlayson, Bernt Schiele, and James L. Crowley. Comprehensive colour image normalization. *Lecture Notes in Computer Science*, 1406:475–490, 1998. 1406.
2. Jon Y. Hardeberg. Transformations and colour consistency for the colour facsimile. Master's thesis, Norwegian Institute of Science and Technology; Trondheim, Norway, 1995.
3. Corvin Idler. Farbkalibrierung. Studienarbeit, 2003.
4. Jochen Michel. Farbkalibrierung mittels linearer Transformation. Studienarbeit, 2004.
5. Christian Münzenmayer, Frederic Naujokat, Steffen Mühldorfer, B Mayinger, and Thomas Wittenberg. Lineare Farbkorrektur zur automatischen Gewebeerkenkung in der Endoskopie des Ösophagus. In *Bildverarbeitung für die Medizin 2004, Proc. des Workshops vom 29.-30. März 2004 in Berlin*, pages 15–19, Berlin, 2004. Springer-Verlag, Berlin, Heidelberg, New York.
6. Frederic Naujokat. Automatische Farbkalibrierung für Endoskopische und Mikroskopische Aufnahmesysteme. Bachelorarbeit, 2003.
7. Dietrich Paulus, Joachim Hornegger, and L. Csink. Linear approximation of sensitivity curve calibration. In K.-H. Franke, editor, *8. Workshop Farbbildverarbeitung*, pages 3–10, Ilmenau, 2002. Schriftenreihe des Zentrums für Bild- und Signalverarbeitung.
8. F. Schmidt and U. Krueger. Eigenschaften von digitalen 1Chip-CCD-Kameras bei der Anwendung zur Farbbeurteilung und Farbdarstellung, 2000.



## Einsatz von diskreten Farbsensoren zur Druckbildkontrolle

Frank Krumbein, Gunter Sieß und Winfried Mahler

MAZeT GmbH, 07745 Jena

[www.mazet.de](http://www.mazet.de)

**Zusammenfassung.** Nachfolgender Beitrag beschreibt den Einsatz von Farbsensoren zur Druckbildkontrolle im Tuben- und Verpackungsdruck. Die Besonderheit der gewählten Lösung besteht in der Auswahl einer diskret aufgebauten Sensorzeile und dem Verzicht auf handelsübliche Sensorzeilen oder Imagerchips. Die Firma ZIND Entwicklungssysteme GmbH bietet das OLIMAT Druckbildkontrollsystem zum Erkennen von unregelmäßig oder schlecht bedruckten Tuben, Dosen, Hülsen und Bechern für den Einsatz direkt an der Druckmaschine an. Des Weiteren werden un-, halb- und schwach lackierte sowie verschmutzte Produkte erkannt.

### 1 Aufbau

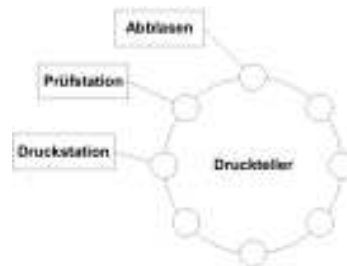


Abb. 1. Druckbildkontrollsystem OLIMAT

Das Druckbild jeder Tube wird eine Station nach der Druckstation mit Lichtsensoren auf der ganzen Länge abgetastet. Jede als schlecht erkannte Tube wird mit einem Magnetventil einen oder mehrere Takte später vom Dorn abgeblasen. Die Abweichung jeder Tube vom zuvor aufgenommenen Original wird auf einem Display dargestellt.

## 2 Diagnose

Das Druckbildkontrollsystem überwacht das Druckbild auf

- Farbschwankungen,
- Farbfehler,
- Ausfall von Farben
- Schriftversatz.

Das heißt, es findet eine komplette Überwachung der Druckmaschine statt.

Weiterhin können auch Fehler im Druckbild selbst, d.h. Fehler, die unabhängig von der Druckmaschine entstanden sind, überwacht werden. Beispiele hierfür sind:

- Fehler in der Lackierung
- Klischeefehler
- Ölspritzer
- Innenlackdruckstellen
- fehlende Buchstaben
- Standversatz
- umgeschlagene Ränder.

## 3 Aufbau und Wirkungsweise der Sensorik

Das Druckbildkontrollsystem OLIMAT ist für zwei verschiedene Sensoroptiken ausgelegt:

**Zeilensensor:** In dieser Lösung kommen Standardbauelemente zum Einsatz. Deshalb soll darauf nicht näher eingegangen werden.

**Einzelsensoren:** Als alternative Sensoroptik bietet das Druckbildkontrollsystem OLIMAT mit Einzelsensoren an. Je nach Länge der Tube, Dose oder Hülse können beliebig viele Einzelsensoren (JENCOLOUR-Farbsensoren) eingesetzt werden. Die Einzelsensoren besitzen eine Breite von 30mm (Gehäuseabmessung) und scannen die Tube, Dose oder Hülse jeweils mit einem Lichtfleck von 4,5mm Breite. D.h. bei einer Tube, Dose oder Hülse mit einer Länge von 180mm können z.B. sechs Sensoren eingesetzt werden, die dann das Druckbild an sechs verschiedenen Stellen überwachen.

Mit Hilfe der Einzelsensoren können somit alle Fehler der Farbübertragung, die durch die Druckmaschine entstehen, erkannt werden. Weiterhin werden auch Fehler in der Lackierung zuverlässig erkannt.

Klischeefehler, Ölspritzer, Innenlackdruckstellen, fehlende Buchstaben, Standversatz oder umgeschlagene Ränder werden nur dann erkannt, wenn sie im Lichtfleck der Sensoren auftreten.

## 4 Maschinenablauf und Funktion

Nachdem das Druckbild korrekt eingestellt wurde und die Tube in Ordnung ist, wird die Musteraufnahme gestartet. Automatisch werden der Tastfleck und das Farbmuster auf der ganzen Länge der Tube, Dose oder Hülse mit Hilfe der Sensoroptik aufgenommen. Danach erfolgt die Einstellung der Warngrenze und der Schlichtgrenze für die Tube. Ist die Mustererkennung beendet, wird automatisch der Vergleichsmodus eingeschaltet. Ab jetzt werden **alle** nachfolgenden Tuben mit dem Original verglichen. Wird eine Tube erkannt, deren Druckbild außerhalb der zuvor festgelegten Toleranz liegt, erfolgt einen oder mehrere Takte später deren automatische Entfernung.

Im Betrieb der Druckmaschine bestehen folgende Einflüsse auf die Qualität des Tubendruckes:

- Beim Anfahren oder Anhalten der Druckmaschine besteht die Gefahr, dass das Druckbild bei den ersten Tuben etwas versetzt gedruckt wird. Dieser Druckversatz bzw. Geisterdruck wird vom Druckbildkontrollsystem zuverlässig erkannt. Das Druckbildkontrollsystem passt die gesetzten Toleranzgrenzen automatisch dem jeweiligen Maschinenzustand an, d.h. wird die Maschine gestartet oder angehalten, werden die Toleranzgrenzen automatisch enger gestellt, um den Druckversatz zuverlässig erkennen zu können.
- Durch leer Dorne in unregelmäßiger Reihenfolge können große Farbschwankungen auftreten. Da das Farbwerk abgerückt wird, bleibt die Farbe, die für diese Tube, Dose oder Hülse vorgesehen war, auf dem Drucktuch. Bei der nächsten Tube, Dose oder Hülse des gleichen Drucktuches wird aber zusätzlich Farbe aufgetragen. Das Druckbild wird somit zum Teil deutlich dunkler. Auch diese Abweichung des Druckbildes wird zuverlässig erkannt und die Tuben werden aussortiert.
- Liegen deutlich dunklere Tuben noch innerhalb der vom Kunden vorgegeben Toleranz, so kann das Druckbildkontrollsystem so eingestellt werden, dass nach jedem leeren Dorn automatisch für ein oder zwei Takte die Dunkel-Toleranzgrenzen weiter gestellt werden und diese dunkleren Tuben damit noch innerhalb der Toleranz liegen.

Die Bewertung der Sensorsignale erfolgt in zwei Bereichen (Abbildung 2).



Abb. 2. Erkennungsschwellen

Wird die erste Toleranzschwelle nach Hell oder Dunkel *leicht* überschritten, so wird am Bildschirm eine **WARNUNG** angezeigt.

Wird die 2. Toleranzschwelle nach Hell oder Dunkel überschritten, wird dies am Bildlauf als **SCHLECHT** dargestellt. Jede als schlecht erkannte Tube, Dose oder Hülse wird mit einem Magnetventil vom Dorn abgeblasen.

Nachfolgende Abbildungen illustrieren das Funktionsprinzip und belegen anhand praktischer Beispiele erreichbare Ergebnisse.

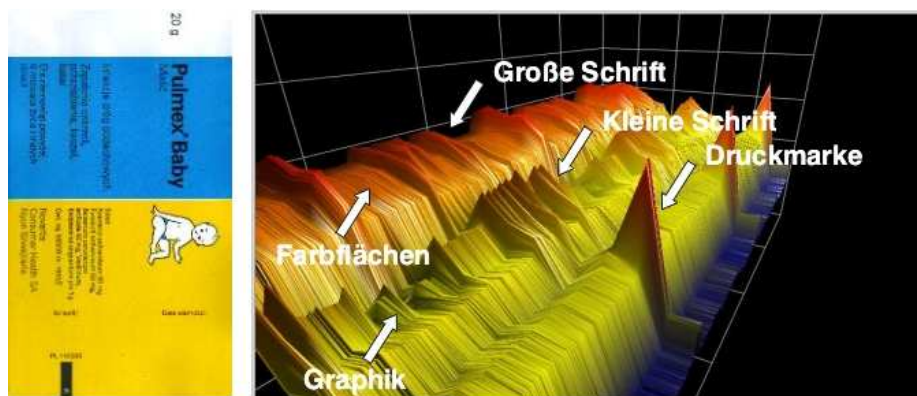


Abb. 3. Ergebnis des Scannens einer Verpackung



Abb. 4. Erkennen von Farbunterschieden beim Verpackungsdruck



Abb. 5. Erkennen von Fehlern beim Verpackungsdruck

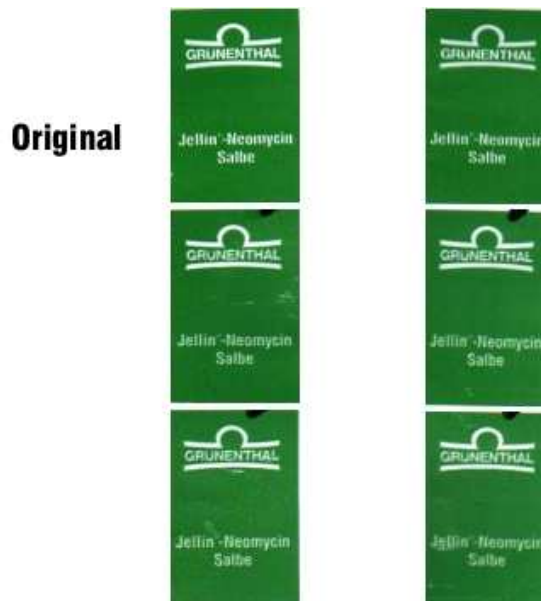


Abb. 6. Erkannte Schriftversatzfehler

## Literatur

1. Zind Systementwicklungs GmbH. Produktbeschreibung OLIMAT.  
[www.ZindSystemEntwicklung.de](http://www.ZindSystemEntwicklung.de).

## Robust Color Image Retrieval for the WWW

Bogdan Smolka \*

Polish-Japanese Institute of Information Technology  
Koszykowa 86, 02-008, Warsaw  
[bsmolka@pjwstk.edu.pl](mailto:bsmolka@pjwstk.edu.pl)

**Abstract.** The rapid growth of image archives increases the need for efficient and fast tools that can retrieve and search through large amount of visual data. In this paper an efficient method of extracting the image color content is proposed, which can serve as an image digital signature, allowing to efficiently index and retrieve large multimedia Internet based databases. The proposed method was applied using the images from the *WEBMUSEUM* Internet database containing the collection of images of fine arts. The results show that the new method of image color representation is robust to image resizing and compression and can be incorporated into existing web-based image retrieval systems.

### 1 Introduction

Successful queries on large, distributed databases cannot rely on textual information only and therefore color image indexing is one of the most important methods used for automatic content based retrieval. In this paper we focus on the image indexing, based on the global color distribution, which is applied for cases when the user provides a sample image for the query.

The majority of the systems exploiting the image color information work using various kinds of color histograms. However the histogram based approach has many drawbacks, as the histogram representation is sensitive to illumination changes, image resizing through interpolation and compression induced artifacts. Therefore, in this paper we propose a nonparametric approach to the problem of the estimation of the distribution of image colors.

### 2 Color Histograms

Color indexing is a process through which the images in a database are retrieved on the basis of their color content. The indexing process must enable the automatic extraction of features, efficient assigning of digital signatures to images and effective retrieval of images within a database.

In order for an image retrieval system to retrieve images that are visually similar to the given query, a proper representation of the visual features is needed and a measure that can determine the similarity between a given query and the images from a database

---

\* This research has been supported by a grant No PJ/B/01/2004 from the Polish-Japanese Institute of Information Technology

set has to be chosen. Assuming that no textual information about the image content are given, image features such as color [1, 2, 3], texture [4] and shape [5, 6] are commonly used.

These features are dependent on illumination, shading, resizing manipulations and compression induced artifacts. Thus, the visual appearance of an image is better described by the distribution of features, rather than by individual feature vectors.

Color feature has proven to be efficient in discriminating between relevant and non-relevant images. One of the widely used tools for image retrieval is the color histogram, which describes the distribution of colors in an image using a specific color space. The colors of an image are mapped into a discrete color space containing  $m$  colors. In this way, a color histogram is an  $m$ -dimensional vector, whose elements represent the number of pixels of a given color in an image.

In this paper we use the RGB color space, which although not perceptually uniform, is the most commonly used, primarily to retain compatibility with computer display systems. Let us assume a color image  $\mathbf{F}$  of size  $n_1 \times n_2 = N$ , composed of three RGB channels  $\mathbf{F} = \{F_{i,j}^R, F_{i,j}^G, F_{i,j}^B\}$ ,  $i = 1, \dots, n_1$ ,  $j = 1, \dots, n_2$ .

An image histogram  $H$  in the RGB color space is the simplest approximation of the density function of the image RGB channels intensities

$$H(\rho, \gamma, \beta) = \# \{F_{i,j}^R = \rho, F_{i,j}^G = \gamma, F_{i,j}^B = \beta\} / N, \quad (1)$$

where  $N$  is the total number of image pixels, and  $\#$  denotes the number of pixels with color channel values  $\{\rho, \gamma, \beta\}$ .

For the analysis of colors, which is independent of image brightness, it is convenient to transform the RGB values into normalized components  $r, g, b$  defined as:  $r = R/I$ ,  $g = G/I$ ,  $b = B/I$ ,  $I = R + G + B$ , where  $R, G, B \in [0, 255]$ . The normalized color values can be expressed using only  $r$  and  $g$  values as  $g = 1 - r - b$  and the normalization makes the  $r, g$  variables non-dependent on the brightness  $I$ .

Using the normalized  $rg$  reduced color space, we can map the color pixel on a two-dimensional plane and obtain a two-dimensional discrete histogram

$$\begin{aligned} \Phi(x, y) &= \# \{ \text{int} (MF_{i,j}^R/I_{i,j}) = x, \text{int} (MF_{i,j}^G/I_{i,j}) = y \} / N \\ &= \# \{ \text{int} (Mr_{i,j}) = x, \text{int} (Mg_{i,j}) = y \} / N, \end{aligned} \quad (2)$$

$x, y = 0, \dots, M$ , where  $M + 1$  is the dimension of the 2-dimensional histogram, (for true-color images  $M = 255$ ). The likeness between two images is often expressed through the similarity of their color histograms. One of the most popular ways to measure the similarity between two histograms is the Minkowski distance or the histogram intersection, which were also used in this work, [1, 7].

### 3 Nonparametric Color Distribution

The drawback of the histogram representation is that the shape of the histogram strongly depends on the method used for lossy image representation and on the number of image pixels, as for small image sizes there is too few points to build the 3-dimensional color histogram, which makes that the comparison of histograms is inapplicable.

To alleviate the problems, we propose in this paper to estimate the color distribution not through the discrete histogram, but to use a smooth nonparametric estimate, based on the concept of nonparametric density estimation, [8, 9]. In this formulation, the similarity measure between two estimates of the color distribution will be expressed as the distance between two surfaces of the two-dimensional kernel density estimation in the normalized  $rg$  color space.

*Density Estimation* describes the process of modelling the probability density function  $f(x)$  of a given sequence of sample values drawn from an unknown density distribution. The simplest form of density estimation is the histogram, however its main disadvantage is its strong dependence on the chosen bin-width and the origin of the histogram grid.

Kernel Density Estimation, avoids this disadvantage by placing a kernel function on every sample value in the sample space and then summing the values of all functions for every point in the sample space, (Figs. 1, 2, 3). This results in a smooth density estimates that are not affected by an arbitrarily chosen partition of the sample space. The multivariate kernel density estimator in the  $q$ -dimensional case is defined as

$$\hat{f}_{\mathbf{h}}(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^n \frac{1}{h_1 \cdots h_q} \mathcal{K} \left( \frac{x_{i_1} - x_1}{h_1}, \dots, \frac{x_{i_q} - x_q}{h_q} \right), \quad (3)$$

with  $\mathcal{K}$  denoting a multidimensional kernel function  $\mathcal{K} : \mathbb{R}^q \rightarrow \mathbb{R}$  and  $h_1, \dots, h_q$  denoting bandwidths for each dimension and  $n$  is the number of samples in the sliding window. A common approach to build multidimensional kernel functions is to use a *product kernel*  $\mathcal{K}(u_1, \dots, u_q) = \prod_{i=1}^q K(u_i)$ , where  $K$  is a one-dimensional kernel function. Intuitively, the kernel function determines the shape of the 'bumps' placed around the sample values and the bandwidths  $h_1, \dots, h_q$  their width in each dimension. If bandwidth is the same in all dimensions, multivariate radial-symmetric kernel functions can be used,

$$\hat{f}_h(\mathbf{x}) = \frac{1}{nh^q} \sum_{i=1}^n K \left( \frac{\|\mathbf{x}_i - \mathbf{x}\|}{h} \right). \quad (4)$$

The shape of the approximated density function depends heavily on the bandwidth chosen for the density estimation. Small values of  $h$  lead to spiky density estimates showing spurious features. On the other hand too big values of  $h$  produce over-smoothed estimates that hide structural features.

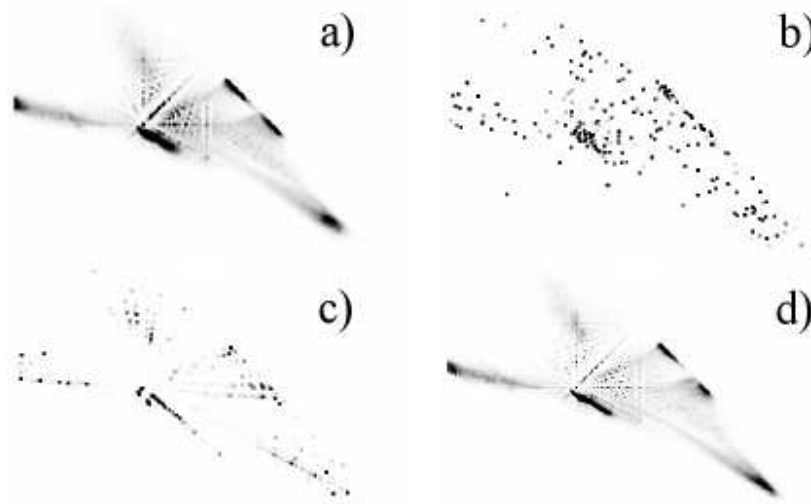
The unknown density function is commonly assumed to be the normal distribution and choosing the Gaussian kernel for  $K$ , the optimal bandwidth in the one-dimensional case is:  $h_{opt} = 1.06\hat{\sigma}n^{-\frac{1}{5}}$ , where  $\hat{\sigma}$  denotes the standard deviation, and for the  $q$ -dimensional case, [8]

$$h_{opt} = (4/(q+2))^{\frac{1}{q+4}} \hat{\sigma} n^{-\frac{1}{q+4}}. \quad (5)$$

In this paper we use the  $rg$  color space, so  $q = 2$  and we used the Gaussian kernel, although we have obtained similar results using other kernel shapes commonly used in nonparametric density estimation.

Using the kernel based estimation, a smooth estimate of the color distribution is obtained as shown in Figs. 2, 3, 4. As can be seen in Fig. 4, the density distribution is





**Fig. 1.** Influence of the compression methods on the color distribution in the normalized  $rg$  color space: a) test image PILLS (Fig. 2) of size 512x512, b) PILLS in GIF format, c) PILLS in JPEG format (compression ratio 78), d) PILLS in JPEG2000 (compression ratio 120).

insensitive to resizing and lossy image coding, which are the basic operations performed when preparing large Internet multimedia databases. This distribution can be used for the image retrieval purposes, as it can serve as an image signature, as depicted in Fig. 3, which shows the  $rg$  nonparametric distributions of some well known color test images.

## 4 Results

To evaluate the efficiency of the proposed color density estimation, we used as the testbed the collection of 3000 images comprising the well-known *WEBMUSEUM* Internet database, ([www.ibiblio.org/wm/](http://www.ibiblio.org/wm/)) and the collection of 10 000 low resolution web-crawled jpeg images from the database of J.Z. Wang, ([wang.ist.psu.edu/docs/related/](http://wang.ist.psu.edu/docs/related/)).

The first database contains a collection of about 3000 images of fine arts of various famous artists. Each image is coded in JPEG of moderate compression ratio, (the blocking artifacts are hardly visible) with width or height of about 1000 pixels. Each image is accompanied by a thumbnail of width or size of 100 pixels, also compressed with JPEG.

From the database, the image of the painting "Starry Night" of V. van Gogh was chosen as the query image, (see Fig. 5 left). Using the kernel density estimation, we used the Euclidean distance between the surfaces of the color distribution estimate as the similarity measure and ordered the retrieved images according to the distance values. The results are very promising, as the second image, most similar to the query, was another painting of van Gogh, "Road with Cypress and Star", (Fig. 5, second image in the row).

In the second experiment, (Fig. 6) we used the thumbnail of the *Starry Night* image as a query image. Although this picture is small ( $122 \times 100$ ) and heavily jpeged, the proposed scheme was able to find the image of full resolution, (first image in the ordered sequence) and the majority of images retrieved using the full resolution images. Very similar results were obtained using the histogram intersection method, so as expected the two methods of similarity evaluation yield comparable results.

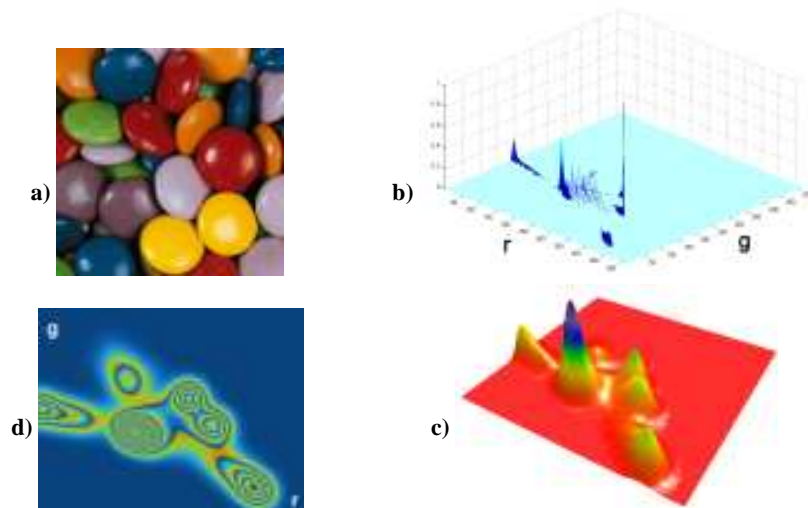
Fig. 7 shows the most similar images from the WEBMUSEUM database to the well known test image LENA. Surprisingly the most similar image was the "Girl with a Pearl Earring" of Vermeer. Additionally, Fig. 8 shows the most similar images to the "Starry Night" from the database of Wang.

## 5 Conclusions

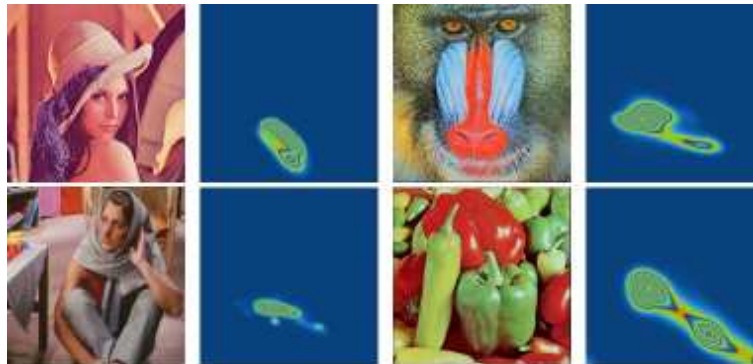
In this paper we proposed a robust way of color density estimation. To enable fast retrieval of large databases we used the normalized  $rg$  color space. The experiments show that the method of nonparametric density estimation is insensitive to image compression and resizing. This makes the proposed framework interesting for image retrieval applications. Especially, the ability to retrieve images using a heavily distorted thumbnail is interesting, as it enables extremely fast retrieval of large databases.

## References

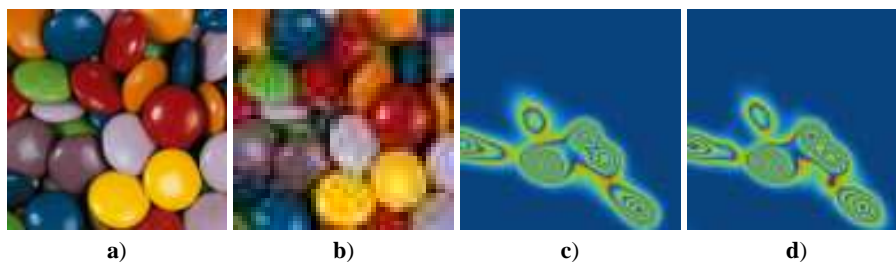
1. Swain M., D. Ballard, Color indexing, International Journal of Computer Vision, 7, 1, 11-32, 1991.
2. M. Stricker, M. Orengo, Similarity of color images, in SPIE Conference on Storage and Retrieval for Image and Video Databases III, 2420, 381-392, February 1995.
3. X. Wan, C.C.J. Kuo, Color distribution analysis and quantization for image retrieval, Proceedings of SPIE, Vol. 2670, February 1996.
4. B.S. Manjunath, W.Y. Ma, Texture features for browsing and retrieval of image data, IEEE Transactions on Pattern Analysis and Machine Intelligence, 18, 8, 837-842, 1996.
5. J.E. Gary, R. Mehrotra, Similar shape retrieval using a structural feature index, Information Systems, 18, 7, 525-537, October 1990.
6. A.K. Jain and A. Vailaya. Image retrieval using color and shape. Pattern Recognition, 29, 8, 1233-1244, 1996.
7. B.S. Manjunath, J.R. Ohm, V.V. Vasudevan, A. Yamada, Color and texture descriptors, IEEE Trans. CSVT, 11, 6, 703-715, June 2001.
8. D.W. Scott, "Multivariate Density Estimation", New York, John Wiley, 1992.
9. B.W. Silverman, "Density Estimation for Statistics and Data Analysis", London, Chapman and Hall, 1986.



**Fig. 2.** Illustration of the nonparametric probability density estimation: **a)** test image PILLS and its histogram in the rg color space (**b**), **c)** and **d)** present the visualization of the smooth kernel based estimation.



**Fig. 3.** Examples of density estimation of the color distribution in the rg space.



**Fig. 4.** Robustness of density estimation to image resizing and compression: **a)** test image of size 512x512, **b)** resized with bilinear interpolation (48x48) and then compressed with JPEG, (compression ratio 8.3), **c)**, **d)** pseudo-color representation of the normalized color densities of **a)** and **b)** respectively).



**Fig. 5.** Results for the query for images similar to the van Gogh "Starry Night" painting (first image) from the WEBMUSEUM database.



**Fig. 6.** Results for the query for images similar to the thumbnail of size (122 x 100) of the van Gogh "Starry Night" painting, from the WEBMUSEUM.



**Fig. 7.** Results for the query for images from the WEBMUSEUM database similar to the LENA color test image.



**Fig. 8.** Results for the query for images from the database of Wang similar to the full resolution "Starry Night" image.

## Assessment of RGB Encoding for Color Imaging

Úlfar Steingrímsson and Peter Zolliker

Swiss Federal Laboratories for Material Testing and Research (EMPA),  
St.Gallen, Switzerland

[ulfar.steingrimsson@empa.ch](mailto:ulfar.steingrimsson@empa.ch),

WWW home page: [www.empa.ch](http://www.empa.ch)

**Abstract.** This work presents validation measures for the optimization of RGB color data encoding. Various RGB specifications had been promoted in the past for standard use in imaging, most of them adapted to application or equipment specific requirements. Although some, as sRGB for instance, are already widely used, none of them has been accepted yet as a universal standard. All of them are subject to limitations regarding to the spatial scope of the color space, making it impossible to satisfy any required conditions in general.

At present, a big variety of different RGB specifications for digital imaging might easily lead to color misinterpretations, because the specific RGB specifications are often missing in the image data. If either the locations of the primary colors or the gamma values are mixed up, color deficiencies are unavoidable. Since the acceptance of a global standard could help to eliminate potential confusions, attempts for the promotion of universal RGB specifications are frequently accomplished.

In the present study, the specific requirements of an RGB in respect of the coding range are examined. A concept for quantitative examination of the encodable color space extent considering the visually perceivable resolution is proposed. Different types of parameters are gathered. First, the fraction of all physically feasible surface colors contained in a certain RGB space is accounted. Next, the visibility of the quantization due to the restricted resolution of the RGB coding is measured in terms of  $\Delta E_{76}$ . Finally, some widely used RGB definitions are analyzed and assessed on the basis of this evaluation scheme.

### 1 Introduction

RGB is undoubtedly the most popular color encoding form, having its wide acceptance due to the analogy with the human color perception, as well as to picture tubes where color reproduction is achieved by three separate electron beams onto red, green and blue phosphor targets. However, a generally accepted all-purpose color data RGB is not yet at hand. The present work explicitly addresses the question of an optimally qualified RGB being suitable for universal color data encoding.

RGB is often regarded as a colorimetric de facto standard and thus, many important color image data formats such as JPEG and TIFF refer to RGB without providing precise specifications. However, since RGB primaries, white point as well as gamma exponent making up the accurate definition of an RGB might disagree considerably,



serious color shifts must be expected if its appropriate specifications are missing. RGB encoding is usually achieved by 8 bits per channel. Depending on the specific settings, any RGB suffers from certain color restrictions in respect of the color range, the discretization, or both.

The problem of RGB confusion might be approached by providing sufficient information about the RGB by including appropriate standard header entries. The Exif format for instance — actually providing an extended JPEG format — supplies this requirement. Exif is already widely being used by many of the latest digital cameras.

On the other hand, a standard RGB image data format with the prospect of worldwide acceptance is another desirable way to affirm color authenticity. In present practice though, image data originating from digital cameras and scanners are based on substantially varying primaries. The same holds true for image reproduction devices such as monitors and printers. Fig. 1 shows the locations of the primaries for some important RGB color spaces, pointing out the manifold variety. A universally approved RGB standard providing a well defined data link between miscellaneous input and output RGB devices would make up an excellent way to inhibit confusions in RGB data interpretation.

Since all of the currently well known RGB standard color spaces suffer from considerable restrictions in respect of the codable color range and the discretization, none of them could make it to the ultimately acknowledged RGB standard yet. Nevertheless, various standardization attempts had already been promoted more or less successfully. At present, mainly sRGB [1] and Adobe98 RGB are essentially predominant in graphic arts, printing and digital camera industry. Since both include serious limitations of the codable saturation range, most of the digital camera manufacturers additionally provide raw RGB images, being heavily device specific and thus baffling the goal of uniformity.

RGB primaries can be optimized in different ways. Finlayson [2] tweaks for hue constancy, which was also the criterion for the development of the oversized ROMM RGB [11]. Kang [4] reports the computational accuracy of various RGB encoding standards by measuring a printed set of color patches. Katajamäki [5] estimates the optimal gamma value for a specific RGB image in respect of a most even distribution of the image colors. Hill [3] exposes the fundamentals for comparative color space estimations based on examinations in visually equispaced color systems such as CIE Lab. On this background, Braun and Spaulding [10] propose volume based metrics for rating different color encoding types, considering up to which degree a color encoding encloses the “real world surface colors”. Separately, they evaluate the quantization related color space limits by determining the maximum and the average quantization errors depending on the bit resolution. The latter works include metrics which are qualified for assessing various types of color spaces such as YCC, CIE Lab, RGB and e-sRGB. Though the same holds true for the parameters proposed herein after, the specific goal of this work is to rank color spaces of type RGB only, in order to account for the worldwide promotion of a general color data RGB.

Since on the one hand all devices providing input images usually feature their own well adapted RGB specifications, and on the other hand the same holds true for all image reproducing devices, it is assumed that some kind of color conversion and some device specific gamut mapping has to be performed anywhere within the workflow from

the input image to the output reproduction. Standard RGB interfaces for image input or output devices — such as sRGB for monitor output — might in fact be convenient. Yet, they imply restrictions of the achievable color range and hence involve undesirable and oblique extra gamut mapping. A RGB particularly optimized for data encoding should thus satisfy in the first place the demand of being large enough for comprising all existing surface colors and secondly, of enabling a better color resolution than the visually perceivable limits. On the other hand since a data RGB is not intended to release any image reproduction workflow from gamut mapping, its primaries don't necessarily need to agree with any output device.

In the following section, metrics for both features — the color volume and the quantization limits in common — are proposed and the ratings for various RGB types compared with each other.

## 2 Methods

In the best case, a RGB color space being explicitly designed for image data encoding satisfies three main objectives:

1. It is large enough for encoding all real world surface colors (maximal colors) and ideally, for all theoretically existing surface colors [9] (optimal colors).
2. Colors outside the optimal color gamut are out of our scope since they cannot be realized except by self illuminants. They should thus be avoided.
3. The visual color distances between two adjacent color codes — being limited due to employing usually 8 bits per RGB channel — should be minor than the visually perceivable minimal color distance, that is, less than  $1 \Delta E_{76}$  [8] for all color domains. Ideally due to the Shannon sampling theorem, it would be wise to keep them even below  $0.5 \Delta E_{76}$ .

For the determination of the maximal colors, a large set of real world color samples had been collected and its convex hull defined in XYZ (see section 3). The real world colors include Pointer [7], Pantone, Munsell and SOCS [13] colors which are converted to the specifically required illuminants by using Bradford color adaptation [6].

Previous estimation metrics [3, 10] are taking consideration of the main objectives 1-3 separately. The goal of this work is to propose one single metric which incorporates all of them.

The basic idea for a model for the evaluation of the three main objectives (points 1-3) includes two issues:

1. The color volume as well as the discretization considerations are accomplished in CIE Lab which is assumed to be perceptibly uniform.
2. Each data sample encodes the color scope of a parallelepiped with a side lengths of  $\Delta E_{76} = 1$  at most, assuming that the eye is capable to perceive color differences of  $\Delta E_{76} > 1$ . That means if adjacent color codes are by less than  $\Delta E_{76} = 1$  apart from each other, the spanned color volume corresponds to the volume of the parallelepiped. If one or more color distances are larger than 1, they are truncated. Thus, each particular color code may cover the volume of a parallelepiped with maximal side lengths of  $1 \Delta E_{76}$ .

The spacings between two adjacent color codes

$$\underline{\Delta}_{RGB} = \begin{pmatrix} R \\ G \\ B \end{pmatrix} - \begin{pmatrix} R + \Delta_r \\ G + \Delta_g \\ B + \Delta_b \end{pmatrix} = \underline{RGB} - \underline{RGB}_{\Delta_{r,g,b}}, \quad \Delta_{r,g,b} \in [0, 1] \quad (1)$$

are converted to CIE Lab resulting in three vectors limited to 1  $\Delta E_{76}$  each, spanning a single color fragment, yielding

$$\underline{\Delta Lab}_r = \min \left( \left| \underline{Lab}_{(RGB)} - \underline{Lab}_{(RGB+\Delta_r)} \right|, 1 \right), \quad \Delta_r = (1, 0, 0) \quad (2)$$

$$\underline{\Delta Lab}_g = \min \left( \left| \underline{Lab}_{(RGB)} - \underline{Lab}_{(RGB+\Delta_g)} \right|, 1 \right), \quad \Delta_g = (0, 1, 0) \quad (3)$$

$$\underline{\Delta Lab}_b = \min \left( \left| \underline{Lab}_{(RGB)} - \underline{Lab}_{(RGB+\Delta_b)} \right|, 1 \right), \quad \Delta_b = (0, 0, 1) \quad (4)$$

The sum of all color fragments being encodable with  $N$  bits and additionally being located within the range of the maximal colors  $V_{ref}$  yields the codable color volume of a specific RGB space

$$V_{Lab} = \sum_{\underline{Lab}_{(RGB)} \in V_{ref}} \det \left[ \underline{\Delta Lab}_r, \underline{\Delta Lab}_g, \underline{\Delta Lab}_b \right] \quad (5)$$

A second significant metric is the maximal color distance between two adjacent codes in a color space, being given by

$$d_{Lab} = \max \left[ \underline{\Delta Lab}_r, \underline{\Delta Lab}_g, \underline{\Delta Lab}_b \right]_{\forall \underline{Lab}_{(RGB)} \in V_{ref}} \quad (6)$$

### 3 Verification

According to section 2, determining the codable color volume by summing up the volume fragments provides the advantage of being able to truncate the side lengths of each single encoding fraction to a well defined maximum size which corresponds with the visual capabilities.

Without providing this benefit, the color volume could be calculated much more efficiently by using convex hull techniques. Since any color space in the CIE Lab system may easily contain concave fragments, a gimmick is required if convex hull techniques are used. According to Grassmann's law the CIE XYZ color space obeys the rules of linearity and additivity. Hence, a color space in XYZ is supposed to be convex provided that additive color mixtures are allowed. The transformation of a convex surface from CIE XYZ into CIE Lab then brings back the concavities, provided that the surface is sufficiently well sampled.

Based on comparisons with the results of the above convex hull technique, the accuracy of the color volume determination by summing up discrete parallelepipeds according to equations 2-5 — while omitting the constraints to 1  $\Delta E_{76}$  — has been examined. The average of the two unequal calculations disagrees by  $\ll 1\%$  only, making evident that the summing up approach yields results of satisfactory accuracy.



## 4 Results

The following well known RGB color spaces [2, 4, 12] have been judged according to the metrics of section 2:

Adobe 1998	Generic EBU Monitor	PAL
Adobe Monitor	Generic Monitor	ROMM
Apple	Kodak DC	SMPTE
Bruce Fraser	Kodak Open Interchange	sRGB
Color Match	NEC Multi Sync Monitor	Wide Gamut
ECI	NTSC 1953	

The encodable color volumes have been evaluated by applying equation 5 and the findings have been compared to conventional color space computations. Fig.2 points out the relative difference of conventional volume estimations to those according to equation 5 implicitly considering quantization. The comparatively large ROMM RGB space has been evaluated at variable bit rates. The results make evident that assigning less than 8 bits per RGB channel yields serious deficits in the color encoding capability since the color graduation becomes too coarse. Encoding with 8 bits seems to be just the minimally required resolution. The evaluation is based on the discernible color limit of  $\Delta E_{76} = 1$ .

The color gaps effected when using 8 bits per channel are evaluated in Fig.3 for all considered RGB definitions. The graph points out the mean values and the variations, as well as the minimal and maximal values of all color distances between two adjacent color encodings. Only those colors within the maximal color space have been considered, since the colors outside this range cannot be reproduced by any image reproduction technique yet and thus they are non-essential. The RGB definitions are sorted by the maximum color distances ranging from 1.2 at the smallest RGB spaces up to more than 2.0 at some large spaces, yielding insufficient resolution in certain color areas.

Two significant metrics — firstly, the relative coverage of all maximal colors and secondly, the mean color distance of every two adjacent color codes — have been chosen to classify RGB specifications regarding to suitability for data encoding. Fig.4 displays the rankings of all RGB types listed at the beginning of this chapter. As outlined in section 2, an optimal RGB encoding would provide both, a high coverage of all maximal colors as well as a preferably high resolution of the color encodings. The second goal agrees with a low mean color distance between adjacent codes. The two intentions however conflict with each other as is manifested by the relatively high correlation rate of  $\rho = 0.79$ , signifying fairly contradictive objectives.

In Fig.4, the scope which is favorable for RGB data encoding is outlined with a shaded area. It includes the four RGB specifications *Adobe 1998*, *NTSC 1953*, *ECI*, as well as *Kodak Open Interchange* RGB. Though the two candidates *ROMM* and *Wide Gamut* RGB include large parts of the maximal colors, its mean color distance is too large. RGB candidates such as *sRGB*, *SMPTE*, *Kodak DC* and *PAL* RGB are not optimal in respect of both ratings.

## 5 Conclusions

A valuation strategy has been proposed to judge RGB specifications in respect of exclusive data encoding. The metrics combine the codable color gamut size with attributes associated with the discretization.

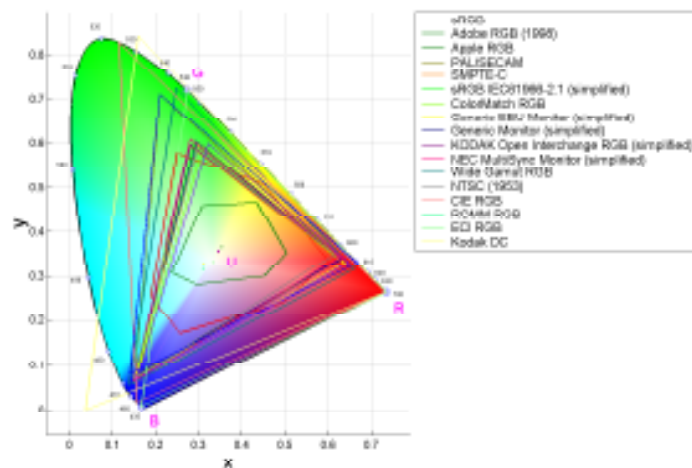
Various common RGB definitions have been examined according to this evaluation scheme. It comes out that in particular four RGB candidates are well qualified for data encoding purpose, namely *Adobe 1998*, *NTSC 1953*, *ECI*, and *Kodak Open Interchange RGB*.

In this work, only existing RGB specifications have been assessed. However by arbitrary variation of RGB primaries and the gamma value, an optimization in terms of color data codability could well be achieved by using the proposed metrics.

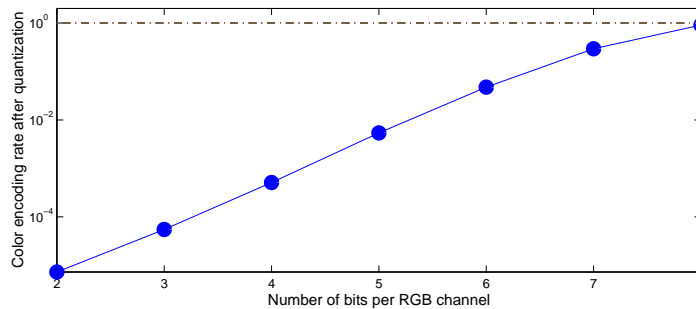
## References

1. M. Anderson, R. Motta, S. Chandrasekar, and M. Stokes. Proposal for a Standard Default Color Space for the Internet: sRGB. In *Color Science, Systems, and Applications*, volume 4, pages 238–245, Scottsdale, Arizona, 1995. IS&T's 4th Color Imaging Conference.
2. G.D. Finlayson and S. Süsstrunk. Optimization for Hue Constant RGB Sensors. In *Color Science and Engineering Systems*, volume 10, pages 343–348, Scottsdale, Arizona, November 2002. IS&T's 10th Color Imaging Conference.
3. B. Hill, Th. Roger, and F.W. Vorhagen. Comparative Analysis of the Quantization of Color Spaces on the Basis of the CIELAB Color-Difference Formula. *ACM Transactions on Graphics*, 16(2):109–154, April 1997.
4. H.R. Kang. Computational Accuracy of RGB Encoding Standards. volume 16, pages 661–664, Vancouver, Canada, November 2000. IS&T's NIP16: International Conference on Digital Printing Technologies.
5. J. Katajamäki and P. Laihanen. Image Dependent Gamma Selection Based on Color Palette Equalization and a Simple Lightness Model. In *Color Science, Systems, and Applications*, volume 7, pages 301–306, Scottsdale, Arizona, November 1999. IS&T's 7th Color Imaging Conference.
6. K.M. Lam. Metamerism and Colour Constancy. Ph.D. Thesis, 1985. University of Bradford.
7. M.R. Pointer. The Gamut of Real Surface Colours. *Color Research and Application*, 5(3):145–155, Fall 1980.
8. A.R. Robertson. The CIE 1976 Color-Difference Formulae. *Color Research and Application*, 2(1):7–11, Spring 1977.
9. E. Schrödinger. Theorie der Pigmente von grösster Leuchtkraft. *Annalen der Physik*, 4(62):603–622, 1920.
10. K. Spaulding and G. Braun. Method for Evaluating the Color Gamut and Quantization Characteristics of Output-Referred Extended-Gamut Color Encodings. In *Color Science and Engineering Systems*, volume 10, pages 99–105, Scottsdale, Arizona, November 2002. IS&T's 10th Color Imaging Conference.
11. K.E. Spaulding, G.J. Woolfe, and E.J. Giorgianni. Optimized Extended Gamut Color Encoding for Scene-Referred and Output-Referred Image States. *Journal of Imaging Science and Technology*, 45(5):418–426, September 2001.
12. S. Süsstrunk, R. Buckley, and S. Swen. Standard RGB Color Spaces. In *Color Science, Systems, and Applications*, volume 7, pages 127–134, Scottsdale, Arizona, November 1999. IS&T's 7th Color Imaging Conference.

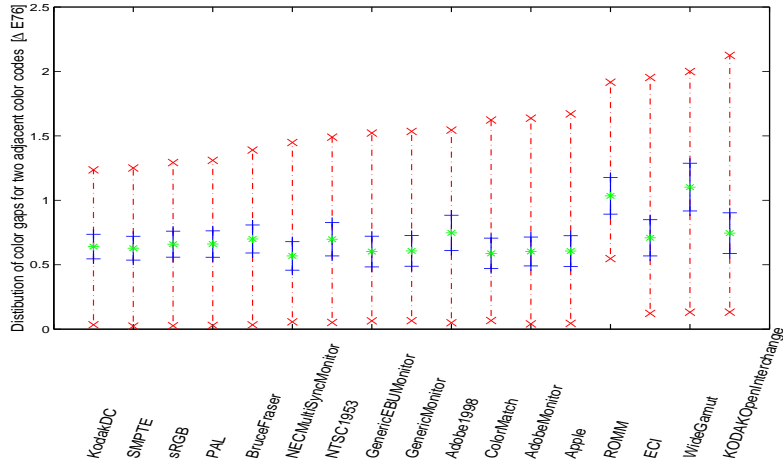
13. J. Tajima, H. Haneishi, N. Ojima, and M. Tsukada. Representative Data Selection for Standard Object Colour Spectra Database (SOCS). In *Color Science and Engineering Systems*, volume 10, pages 155–160, Scottsdale, Arizona, November 2002. IS&T's 10th Color Imaging Conference.



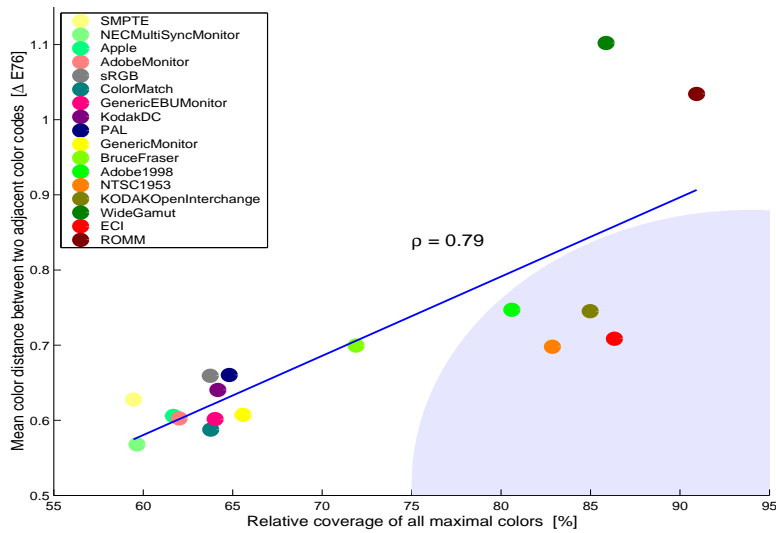
**Fig. 1.** Common RGB spaces being adopted by digital cameras, scanners, monitors, or merely serving as working color spaces for image analysis and archiving purpose. The figure shows the well known horseshoe shaped xy color space representation.



**Fig. 2.** This figure illustrates the impact of the quantization according to equation 5. The limitation to  $1/\Delta E_{76}$  of the color space fraction recordable with a single code yields substantial color space shortcomings whenever the bit rate of the RGB channels is too low. The figure presents the effective color encoding rates depending on the number of coding bits. The comparatively large ROMM RGB has been evaluated. Even at a bit rate of 7 bits, only about one third of the real world surface colors (maximal colors) are encodable due to the too large color gaps between two adjacent codes. At a bit rate of 8 bits, the sampling rate is reasonably close to the perceptible color resolution.



**Fig. 3.** For the RGB list at the top of section 4, the distribution of the color distances of two adjacent color codes is given by its mean values (displayed by stars), scattering intervals (lines delimited by '+') and minimal / maximal value intervals (delimited by crosses). The RGB characteristics are sorted by the maximal color encoding distances.



**Fig. 4.** Evaluation of multiple RGB specifications according to the metrics given in equation 5 and 6. The abscissa shows the percentual portion of all maximal colors being encodable with the respective RGB specifications. The ordinate points out the mean color difference between two adjacent color codes in  $\Delta E_{76}$ .

## Ein Entwurfswerkzeug für Farbklassifikatoren in Echtzeitanwendungen

Klaus Arbter und Daniel Kish

Deutsches Zentrum für Luft- und Raumfahrt e.V.  
Institut für Robotik und Mechatronik Oberpfaffenhofen  
D-82230 Weßling  
[klaus.arbter@dlr.de](mailto:klaus.arbter@dlr.de)  
[www.robotic.dlr.de](http://www.robotic.dlr.de)

**Zusammenfassung.** Farbsegmentierung in Echtzeitanwendungen konnte bis vor wenigen Jahren nur unter Verwendung teurer Hardware für Kameras, Framegrabber und Spezialcomputer realisiert werden. Heute ist leistungsfähige Hardware sehr viel billiger erhältlich, und die Leistung eines modernen Personal Computers ist in vielen Fällen ausreichend. Eine effiziente Software-Implementierung besteht aus einer Color-Look-Up-Tabelle (CLUT) gefolgt von einem morphologischen Filter. Dabei ist die Belegung der CLUT anwendungsspezifisch. Die richtige Belegung zu finden, kann sehr mühsam sein. In dieser Arbeit wird ein interaktives Entwurfswerkzeug vorgestellt, das über eine grafische Benutzeroberfläche bedient wird und es bequem ermöglicht, aus mehreren Bildern eine CLUT abzuleiten. Die Verwendung von Farbklassifikatoren in Echtzeitanwendungen wird an Beispielen aus der Robotik gezeigt.

### 1 Einleitung

Ein Zugang zur Lösung von Problemen der schritthaltenden Bildverarbeitung in Echtzeitanwendungen besteht in einer möglichst raschen Datenreduktion. Der erste Schritt ist dabei oft die Umwandlung des ankommenden Bildes in ein Kantenbild oder Binärbild, wobei unterstellt wird, dass Bildkanten zu Objektkanten und Bildregionen zu Objektflächen korrespondieren. Seit den frühen Tagen des maschinellen Sehens werden Grauwertbilder auf diese Weise vorverarbeitet. Der nahe liegenden Idee, Farbbilder entsprechend zu behandeln, standen bis vor kurzem sehr hohe Implementierungskosten für Farbkameras und für Spezialhardware zur Bildvorverarbeitung im Wege. Die rasante technologische Entwicklung der letzten Jahre hat dazu geführt, dass heute Farbkameras erhältlich sind, die weniger kosten als eine Tankfüllung Benzin, und dass die Rechen- und Datentransferleistung moderner PCs ausreichen, um die Farbklassifikation im Videotakt auszuführen [2].

Eine bewährte Implementierung des Farbklassifikators besteht aus einer Look-Up-Tabelle (CLUT) gefolgt von einem binären Ortsfilter (closing) [6] und einem zeitlichen Tiefpassfilter mit binärem Ausgang [9]. Die CLUT weist jedem Pixel aufgrund seiner

Farbe eine von zwei Bedeutungen zu (Farbe bekannt/unbekannt), sodass als Zwischenergebnis ein Binärbild entsteht. Die Anpassung an eine konkrete Aufgabenstellung erfolgt durch die Belegung der CLUT, die auch Klassifikatorentwurf genannt wird. Das kann vollautomatisch, unter Verwendung von beispielhaftem Bildmaterial, geschehen. Dazu muss zuerst das Entwurfsziel durch die Angabe von Entwurfsparametern festgelegt werden, was den Nichtexperten vor ein schwieriges Problem stellen kann. Dann muss der Anpassungsprozess durch den Anwender initialisiert werden, z. B. indem er Regionen markiert, in welchen wieder zu erkennende Farben vorkommen. Und schließlich benötigt er noch die Möglichkeit, das Ergebnis des automatischen Entwurfs zu korrigieren.

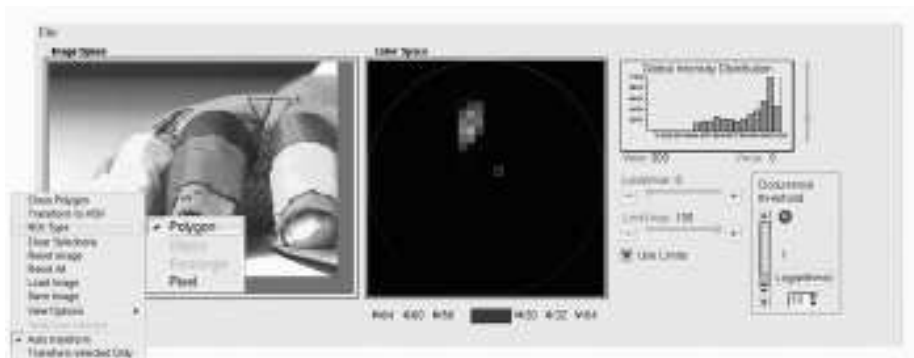
Es erscheint daher attraktiv, auf den vollautomatischen Entwurf zu verzichten und dafür Werkzeuge einzusetzen, die den interaktiven Entwurf unterstützen. In der vorliegenden Arbeit wird das interaktive Entwurfswerkzeug CCDT (Color Classifier Design Tool, DLR) vorgestellt. Auch hier beschreibt der Anwender im Bildraum die Regionen, welche die Farben enthalten, die wieder erkannt werden sollen. Die Klassengrenzen im Farbraum kann er weitgehend frei festlegen, aber auch an Schwellwerte in Farbhistogrammen koppeln. Zu jeder Zeit kann er seinen Entwurf an Testbildern überprüfen. Als Ergebnis einer Sitzung erhält er eine Datei, welche die CLUT in einem dicht komprimierten Format enthält. Diese Daten kann er unmittelbar in seine Anwendung übertragen. Anwendungsbeispiele aus der Robotik schließen die Arbeit ab.

## 2 Das Entwurfswerkzeug CCDT

Jeder interaktive Entwurf erfolgt im Wechsel zwischen den Entwurfsphasen Analyse, Synthese und Verifikation. Dementprechend ist das Entwurfswerkzeug CCDT (Color Classifier Design Tool) konzipiert. Es stellt Hilfsmittel zur Farbbildanalyse, zur Klassifikatorsynthese und zur Anwendung der aktuellen Klassifikatorauslegung an Testmaterial zur Verfügung. Diese sind funktional voneinander unabhängig und können daher in beliebiger Folge angewandt werden.

**Die Benutzeroberfläche** Die Benutzeroberfläche (Abb. 1) ist in drei Abschnitte gegliedert. Links befinden sich die Darstellungen des Bildraums *Image Space*, in der Mitte die des XY-Farbraums *Color Space* und rechts die des Z-Grauertraums *Intensity Space* sowie ergänzende Bedienelemente. Dem Bild- und dem Farbraum sind unabhängige Menüs zugeordnet. Als XY-Farbraum wird der kartesische, diskrete Farbraum auf der Displayebene aufgefasst. In diesen eingebettet ist der HS-Farbraum, weil dieser mit der menschlichen Farb Wahrnehmung gut korrespondiert und sich daher für die Orientierung des Benutzers im Farbraum gut eignet. Der Z-Grauertraum, ist die diskrete Repräsentation der V-Komponente des HSV-Farbraums [8].

**Farbbildanalyse** Die Trainings- und Testbilder können in beliebigen Verzeichnissen gespeichert sein. Sie sind einzeln über einen Verzeichnisbaum erreichbar (*Load Image*). Die unterstützten Bildformate sind PNG, XPM, JPG, JPEG, BMP, PPM und GIF. Zur Farbbildanalyse werden die Bilder der Reihe nach vom Benutzer geladen und bearbeitet. Das Ergebnis der Analyse sind ein Farb- und ein Grauerthistogramm, welche die



**Abb. 1.** Die CCDT-Benutzeroberfläche bei einer Farbanalyse: Farbbild mit benutzerdefinierten Regionen, Punkten und Menüs des Bildraums (links), Farbbelegung und überlagertes, begrenztes Farbhistogramm (Mitte), kumulatives Grauwert histogramm (rechts)

Grundlage für den nächsten Schritt, die Klassifikatorsynthese bilden. Nach dem Laden eines Bildes (*Load Image*) werden Bildpunkte (*ROI Type: Pixel*) oder Regionen (*ROI Type: Polygon*) oder das ganze Bild (*Click auf Bild*) markiert. Die anschließende Aufforderung *Transform to HSV* bewirkt, dass alle erfassten Pixel des Farbbildes in den XYZ-Farbraum transformiert und die Ergebnisse in ein XY-Farbhistogramm und ein Z-Grauwert histogramm eingetragen werden. Im Modus *Auto transform on* erfolgt die Transformation automatisch nach dem Abschluss jeder Markierung. Die Ergebnisse der Transformation können im Farbbereich auf verschiedene Weise dargestellt werden. In der Grundeinstellung werden alle ermittelten Bildfarben im XY-Farbraum in der zuletzt gefundenen RGB-Originalfarbe markiert. Diese Einstellung vermittelt eine schnelle Übersicht über die Belegung des Farbraums. Eine detailliertere Darstellung für die nachfolgende Klassifikatorauslegung bietet das Farbhistogramm (*View Options: Histogram Options: Histogram on*). Dieses zeigt in Falschfarbendarstellung die relative Farbhäufigkeit im XY-Farbraum, jedoch nur für die Werte, welche über dem einstellbaren Schwellwert *Occurence threshold* liegen.

**Klassifikatorsynthese und -verifikation** Die Synthese des Klassifikators erfolgt durch die unabhängige Festlegung von Grenzen (*LimitVmin*, *LimitVmax*) im Z-Grauwertraum und von Farbwerten im XY-Farbraum. Die Farbwerte können auf unterschiedliche Weise spezifiziert werden:

- (a) alle Farbwerte, die vom begrenzten Farbhistogramm bedeckt werden
- (b) alle Farbwerte die von benutzerdefinierten Polygonen eingeschlossen werden
- (c) alle Farbwerte, die von benutzerdefinierten Punkten getroffen werden

In einfachen Fällen reicht die automatische Auslegung nach (a) aus (*Filter Options: Auto Select on*). Sie kann aber auch ausgeschaltet (*Filter Options: Auto Select off*) und eine manuelle Auslegung nach (b) und (c) durchgeführt werden. Dabei bietet die bloße Anzeige des begrenzten Farbhistogramms (*View Options: Histogram Options: Histogram on*) eine wertvolle Orientierungshilfe. Die benutzerdefinierten Elemente (Polygon

und Punkt) sind wahlweise konstruktiv (*Add to CLUT*) oder destruktiv (*Remove from CLUT*). Damit sind im Laufe der Auslegung immer wieder Korrekturen möglich. In jeder Phase des Entwurfs kann die Wirkung der aktuellen Auslegung auf das jeweils geladene Bild überprüft werden (*Apply Filter*). In der Grundeinstellung werden die erkannten Pixel einfarbig markiert, im *Inverse Filter* Modus alle anderen, nur nicht die erkannten. Ein Beispiel zeigt Abbildung 3.

**Anmerkungen zur Implementierung** Die interne Bilddarstellung erfolgt grundsätzlich im RGB-3x8Bit-Format. Bilder von geringerer Auflösung (z. B. 565-Format) werden automatisch expandiert. Damit ist eine konsistente, formatunabhängige Verarbeitung gewährleistet.

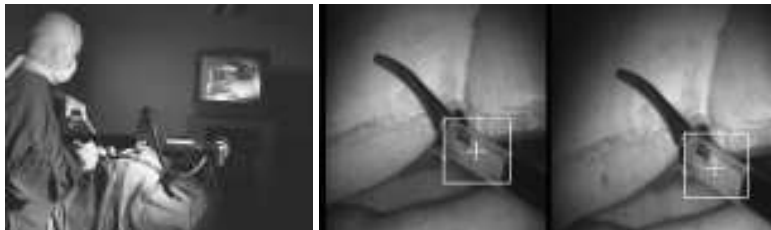
Ein Problem, das besondere Beachtung verdient, sind die Abbildungsfehler in diskreten Räumen. Während die kontinuierliche Abbildung  $RGB \rightarrow HSV$  umkehrbar eindeutig ist, ist es die diskrete Abbildung  $RGB \rightarrow XYZ$  nicht. Zu einem Punkt  $\mathcal{P}$  im XYZ-Raum gibt es im Allgemeinen eine Region  $\mathcal{R}$  im RGB-Raum, deren Abbildung auf  $\mathcal{P}$  führt. Das heißt, dass bei der Auswahl eines Punktes im XY-Raum, und eines Intervalls im Z-Raum alle direkten Nachbarn einer Geraden im diskreten RGB-Raum, sowie alle entfernteren, potenziellen Nachbarn gefunden werden müssen. Die vollständige Suche scheidet aus Effizienzgründen aus. Die direkten Nachbarn der Gerade zu finden, ist nicht schwer. Schwieriger ist es, alle anderen potenziellen Nachbarn, aber nur diese, zu lokalisieren. In der aktuellen CCDT-Version ist dieses Problem noch nicht exakt gelöst, sondern hier werden alle Punkte in einer ausreichend großen Umgebung der direkten Nachbarn zu Nachbarn ernannt. Der daraus resultierende Klassifikator lässt geringfügig mehr Farben zu, als er es exakterweise sollte.

### 3 Anwendungen

**Multisensorielles Robot Servoing** Bei Greif- oder Dockingvorgängen von Robotern ist die Annäherung an eine Nominalposition relativ zum Objekt eine wichtige Grundaufgabe, die bei Objekten mit variabler Lage nur durch den Einsatz von Sensoren gelöst werden kann. In [3] wird ein analytischer Ansatz und in [10] ein künstliches neuronales Netz zur Verarbeitung redundanter Information aus einem Stereo-Kamerapaar und vier Laser-Entfernungsmessern vorgestellt. Beide wurden unter Verwendung des multisensoriellen DLR-Robotergräfers (Abb. 4a) getestet. Als Bezugspunkte für die Bildverarbeitung dienten vier Marken auf dem Objekt, die zuerst als schwarze Punkte auf hellem Grund ausgeführt waren. In der komplexen Experimentalumgebung (Abb. 4b) mit wechselnden Schrägansichten bereitete deren Erkennung gelegentlich Schwierigkeiten, die mit der Verwendung von Farbmarken vollständig behoben werden konnten. Die Abbildung 4c zeigt ein Klassifikationsergebnis der CLUT, in welchem die Restfehler durch eine nachgeschaltete Filterung leicht unterdrückt werden können.

**Kalibrierung eines Datenhandschuhs** Ein sehr aktuelles Thema in der Roboterforschung ist die Entwicklung anthropomorpher Roboterhände und deren Kontrolle zur Bewältigung komplexer Manipulationsaufgaben. Fischer et. al. [4] beschreiben einen Ansatz zur Abbildung der nichtlinearen und verkoppelten Messwerte eines





**Abb. 2.** Solochirurgische Szene mit Assistenzroboter (links) und Stereo-Bildpaar mit umschreibenden Rechtecken der segmentierten Farbmarke (rechts)

Datenhandschuhs auf die Kinematik der DLR-Hand (Abb. 5a) mittels eines künstlichen neuronalen Netzwerks. Zum Training des Netzwerks müssen viele Konfigurationen der Fingerspitzen des Datenhandschuhs zusammen mit den jeweiligen Messwerten erfasst werden. Diese Aufgabe wird mithilfe farbiger Stecknadeln (Abb. 5b) an den Fingerspitzen, deren räumliche Positionen durch Stereobildauswertung ermittelt werden, gelöst. In Abbildung 5c ist die Raumbedeckung durch die Fingerkuppen des Datenhandschuhs dargestellt.

**Roboterassistent für die laparoskopische Chirurgie** Seit einigen Jahren werden minimalinvasive Operationstechniken entwickelt und zunehmend klinisch angewandt. Während der Chirurg beim offenen Eingriff den Situs unmittelbar sieht, blickt er beim minimalinvasiven Eingriff auf einen Monitor, auf dem das Bild eines Laparoscops dargestellt ist. Dazu muss ein Kollege (Assistent) das Laparoskop halten und kontinuierlich am Situs ausrichten. Diese Tätigkeit ist anstrengend und mit vielen Problemen behaftet. Deshalb wurde ein automatisches Laparoskopführungssystem entwickelt [1, 9], bei welchem ein Roboter ein Stereolaparoskop hält (Abb. 2) und selbstständig auf eine Farbmarke (Abb. 6a) an der Instrumentenspitze ausrichtet (Abb. 6b). Der Klassifikator-entwurf erfolgte unter den folgenden, erschwerenden Bedingungen:

- hoher Störpegel (Rauschen) in den Laparoskopbildern
- Material der Marke muss biokompatibel sein
- schwach gesättigtes Farbbild der Marke (Abb. 6f) und
- teilweise überlappende Farbcluster (Abb. 6e und 6f)

Erleichternd wirkte sich aus, dass auf eine Auswertung der Form der Marke in der Vorverarbeitung verzichtet wurde, und daher auch Klassifikationsfehler auf dem Bild der Marke großzügig toleriert werden konnten. Zur Nachbearbeitung (Abb. 6g und 6h) wurden ein Ortsfilter (Closing), ein Zeitfilter (Tiefpass) und eine einfache Blob-Analyse verwendet. Das System wurde ursprünglich unter Verwendung sehr teurer Bildverarbeitungshardware (Datacube, MaxVideo) realisiert und klinisch getestet [7]. Zurzeit ist eine PC-basierte Realisierung in der Entwicklung.

**Visuelle Erfassung der Herzbewegung** Herzoperationen erfolgen üblicherweise am stillgelegten Herzen, wobei der Kreislauf durch eine Herz-Lungen-Maschine

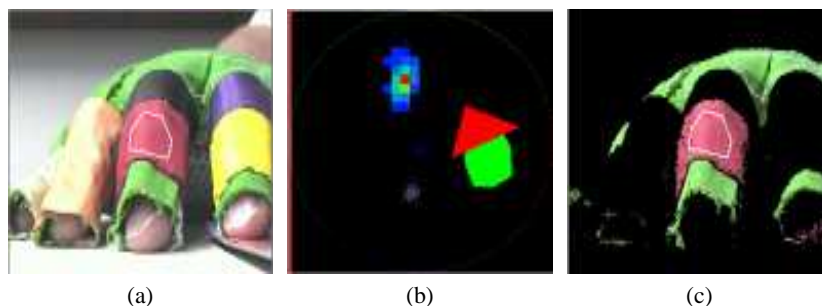
aufrechterhalten wird. Um die Nachteile dieser Maschine zu vermeiden, werden neuerdings Methoden für den robotergestützten Eingriff am schlagenden Herzen untersucht. Eine zentrale Rolle spielt dabei die Erfassung der Restbewegung des mechanisch stabilisierten Herzens, um eine abgestimmte Bildstabilisierung für den Chirurgen und Bewegungskompensation des Roboters zu realisieren. Zur visuellen Bewegungsschätzung wurden von Gröger et. al. lokaler Grauwertstrukturen als natürliche Landmarken und die Eignung farbiger Titan-Clips als künstliche Landmarken untersucht [5]. In den Abbildungen 7a bis 7f sind die Szene und eine Klassifikatorauslegung dargestellt.

#### 4 Zusammenfassung und Ausblick

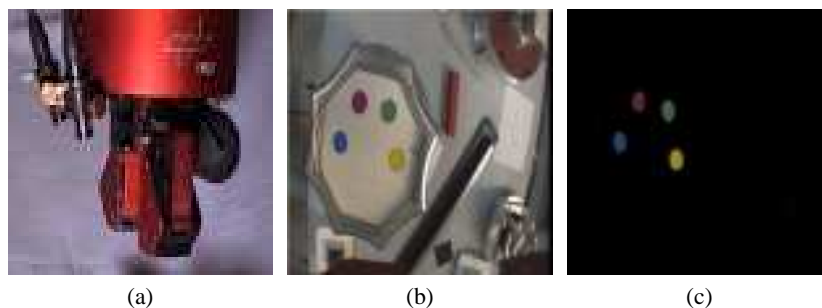
Es wurde ein interaktives Entwurfswerkzeug für Farbklassifikatoren vorgestellt, mit welchem Nichtexperten einfachere Auslegungen bequem und schnell vornehmen können, welches es aber auch Experten erlaubt, ihre Expertise einzubringen. Die Beispiele aus der Robotik zeigen ein breites Einsatzspektrum der Farbklassifikation in Echtzeitanwendung und damit auch den potenziellen Nutzen des vorgestellten Werkzeugs im Labor und in Projekten. In der Folgezeit wird das Werkzeug weiterentwickelt, wobei insbesondere die Erfahrung aus der Praxis in die Entwicklung einfließen soll.

#### Literatur

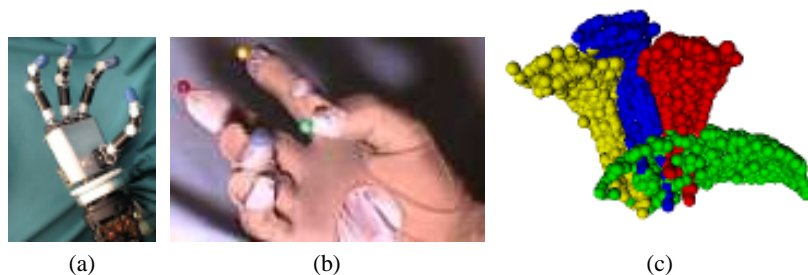
1. K. Arbter and G.Q. Wei. Verfahren zum Nachführen eines Stereo-Laparoskops in der minimalinvasiven Chirurgie. Deutsche Patent Nr. 195 29 950, 14. Nov., 1996.
2. James Bruce, Tucker Balch, and Manuela Veloso. Fast and inexpensive color image degmentation for interactive robots. In *Proc. 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 2, pages 2061–2065. IEEE, 2000.
3. B. Brunner, K. Arbter, and G. Hirzinger. Task directed programming of sensor based robots. In *Proc. IEEE Int. Conf. on Intelligent Robots and Systems (IROS'94), München, Sept. 12 - 16, 1994*.
4. M. Fischer, P. van der Smagt, and G. Hirzinger. Learning techniques in a dataglove based telemanipulation system for the DLR hand. In *Proc. Int. Conf. on Robotics and Automation, ICRA '98, Leuven, Belgium, 1998*.
5. Martin Gröger, Bernhard Kübler, and Gerd Hirzinger. Selection of qualified colour markers for heart surface tracking. In *VDE Proceedings of the 38th Annual Congress of the DGBMT, Ilmenau, Germany, Sept. 2004*.
6. R. Massen, T. Regele, and P. Böttcher. Colour and shape classification with competing paradigms: Neural networks versus trainable table classifiers. In *ECO 3 Int. Congress Optic. Science, Den Haag, 1990*.
7. K. Omote, A. Ungeheuer, H. Feussner, K. Arbter, G.Q. Wei, J. R. Siewert, and G. Hirzinger. Self-guided robotic camera control for laparoscopic surgery compared with human camera control. *American Journal of Surgery*, 177:321–324, April 1999.
8. Alvy Ray Smith. Color gamut transform pairs. In *Proceedings of the 5th annual conference on computer graphics and interactive techniques, SIGGRAPH '78*, pages 12–19. ACM Press, Aug. 1978.
9. G.-Q. Wei, K. Arbter, and G. Hirzinger. Real-time visual servoing for laparoscopic surgery. *IEEE Engineering in Medicine and Biology*, 16(1), January/February 1997.
10. G.Q. Wei and G. Hirzinger. Learning motion from images. In *Proc. The 11th Int. Conf. Pattern Recognition*, pages 189–192, 1992.



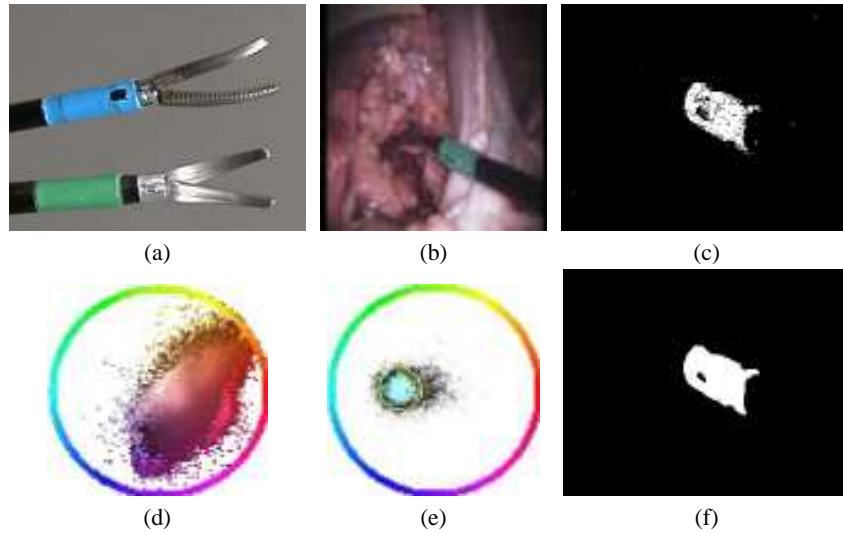
**Abb. 3.** Beispiel für den Klassifikatorentwurf: (a) Farbbild mit eingetragenen Polygonen in den grünen, roten und violetten Bildregionen. (b) Beschränktes Farbhistogramm beim Farbton grün, konstruktive (grüne) Fläche beim Farbton rot, destruktive (rote) Fläche beim Farbton rötlich gelb und Cluster beim Farbton violett. (c) Die roten und grünen Farben werden erkannt (im Modus: *Auto Select On*), die rötlich gelbe Farbe wird explizit unterdrückt (Finger links), und die violette wird ignoriert.



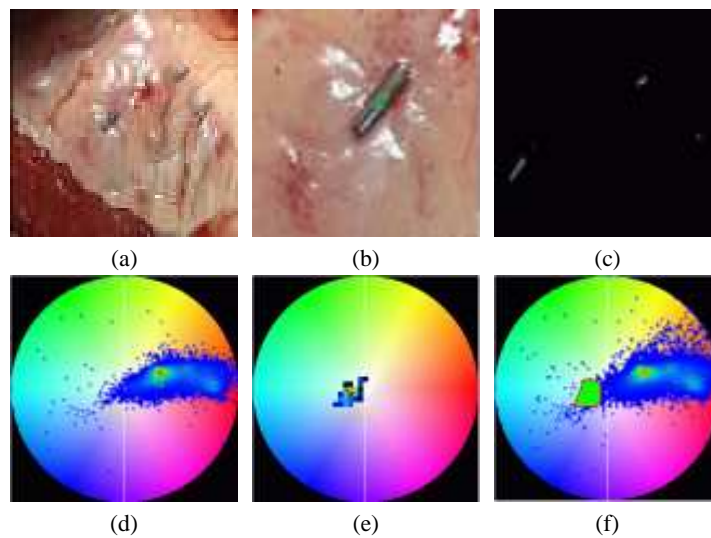
**Abb. 4.** Bildauswertung für den multisensoriell gestützten Greifvorgang eines Roboters: (a) Multisensorieller DLR-Greifer (Stereo-Kamerapaar, vier Laser-Entfernungsmesser in den Greiferbacken), (b) Farbmarkiertes Objekt (ORU: Orbital Removable Unit) mit rötlichen Objekten im Hintergrund der ROTEX-Arbeitszelle, (c) segmentierte Farbmarken des ORU (ohne Filterung)



**Abb. 5.** Farbsegmentierung für die Kalibrierung eines Datenhandschuhs zur Telemanipulation einer Roboterhand: (a) DLR-Hand (erstes Experimentalmodell), (b) Datenhandschuh mit Farbmarken an den Fingerspitzen, (c) aus Stereo-Bildfolge experimentell ermittelter Arbeitsraum der Fingerkuppen während eines Versuchs



**Abb. 6.** Farbsegmentierung für die bildgestützte, automatische Laproskopführung durch einen Chirurgieroboter: (a) farbcodierte Instrumente, (b) Situs mit Instrument und Farbmarke, (c) Klassifikationsergebnis ungefiltert, (d) HS-Histogramm des Hintergrunds, (e) HS-Histogramm der Farbmarke, (f) Klassifikationsergebnis gefiltert



**Abb. 7.** Segmentierung farbiger Clips auf der Herzoberfläche für die Bewegungskompensation bei chirurgischen Eingriffen am schlagenden Herzen: (a) Herzoberfläche mit farbigen Clips, (b) grafisch markierte Region auf einem Clip, (c) segmentierte Clips (vergrößerter Ausschnitt, ohne Filterung), (d) Farbhistogramm der Herzoberfläche, einschließlich Clips, (e) Farbhistogramm der markierten Region in (b), (f) Klassifikationsergebnis: rot umrandete grüne Fläche

## Haustoria Segmentation in Microscope Colour Images of Barley Cells

Alexander Ihlow and Udo Seiffert

Leibniz Institute of Plant Genetics and  
Crop Plant Research (IPK) Gatersleben  
Corrensstr. 3, 06466 Gatersleben, Germany  
Pattern Recognition Group  
{ihlow, seiffert}@ipk-gatersleben.de  
[mue.bic-gh.de](http://mue.bic-gh.de)

**Abstract.** Dyed barley cells in microscope colour images of biological experiments are analysed for the occurrence of haustoria of the powdery mildew fungus by a fully automated screening system. The region of interest in the images is found by applying Canny's edge detector to the hue channel of the HSV colour space. For the segmentation of potential haustoria within the dyed cells, two different methods are considered: A clustering in RGB colour space using the Expectation Maximisation (EM) algorithm, and morphological contrast enhancement of the colour image with subsequent hysteresis thresholding in the saturation channel of the enhanced images. The second approach seems to be more viable because of its robustness and more promising results.

### 1 Introduction

Automating the screening and the analysis of biological experiments is a challenging research area in the field of bioinformatics and engineering. This paper is related to a project studying resistance mechanisms of crop plants against the powdery mildew fungus from the genetical point of view. In the experiments, young barley leaves are bombarded with DNA-coated particles to “switch on or off” desired genes in cells. For analysis purposes, an additional reporter gene<sup>1</sup> is expressed in cells that were hit by a particle. This dyes the affected genetically transformed cells greenish blue and allows their identification by bright field microscopy [7]. The task is to evaluate the susceptibility of the genetically transformed cells to the powdery mildew fungus under the impact of different test genes. A successful penetration of the fungus into the cell is indicated by the development of a haustorium – a dark object consisting of a “waist” with “fingers” that is located between the cell wall and the cell membrane and feeds the fungus by leaching the cell (see Figure 2). These objects have to be counted in an automatic analysis procedure.

Since there are many genes to be considered for a potential resistance of the plant against pathogens, a big number of experiments has to be performed to obtain a sufficient statistical confidence. Therefore, an automated image acquisition system and an

---

<sup>1</sup>  $\beta$ -glucuronidase (GUS) reporter gene

automatic analysis procedure is needed. Manual screening is a tedious, subjective and time-consuming task that cannot be handled by laboratory assistants due to that huge amount of data. For an automatic image acquisition, the microscope slides are mounted on an x-y table which scans a number of preparations fully automatically under the control of a computer, e.g., overnight. Now, finding genetically transformed cells and therein assessing the development status of the haustoria without human interaction is the task and the challenge of the analysis procedure.

This paper deals with the problem of segmenting the scarcely outstanding haustoria from the remaining cell tissue. It is organised as follows: Section 2 introduces the properties of the image material and explains how the regions of interest, i.e., genetically transformed cells, are found in the images. Afterwards, Section 3 reviews the Expectation Maximisation (EM) clustering technique and presents its results on two examples. Section 4 describes a segmentation technique based on thresholding in the saturation channel after morphological contrast enhancement of the colour image and shows its results on four examples, before Section 5 concludes the paper.

## 2 Preprocessing of the image material

Figure 2 shows two typical cutouts of microscope images, both containing one dyed genetically transformed cell with two haustoria of the powdery mildew fungus inside. By default, the microscope camera produces images of  $2600 \times 2060$  pixel in 24-bit colour.

In [3] we have shown that these dyed cells can be reliably detected by applying Canny's edge detector [1] to the hue channel of the HSV colour space. It is advantageous to cut out the cells exactly at its edges first before processing the images by morphological contrast enhancement. This prevents potential distortions from regions outside the cell during operations with large neighborhood-masks. Since Canny's edge detector produces *either* smooth and closed edges at the expense of a significant spatial uncertainty *or* rather disrupted but certainly positioned edge elements at fine scales, an edge linking procedure across multiple scales is considered in tradeoff to get both a *closed* as well as *correctly located* cell boundary. For the EM approach, this step is omitted and the rectangular region of interest, containing a dyed cell, is processed directly.

## 3 Clustering in RGB using the EM algorithm

The Expectation Maximisation (EM) algorithm [2, 6] solves the general problem of classifying a number of  $N$   $d$ -dimensional data vectors  $\mathbf{x}_n \in \mathbb{R}^{d \times 1}$  from the entire data set  $\mathbf{X} \in \mathbb{R}^{d \times N}$  into  $K$  classes. Therefore, a parametric model for the class distributions is chosen whose initial parameters are adapted in an iterative procedure. A popular, powerful and simple model is the multivariate Gaussian distribution, whose equal-probability surfaces describe (hyper)ellipsoids in the  $d$ -dimensional space. The center of an ellipsoid is given by the mean vector  $\boldsymbol{\mu}$ , whereas its shape and orientation

is described by the covariance matrix  $\Sigma$ .

$$p(\mathbf{x}|\boldsymbol{\mu}_k, \Sigma_k) = \frac{1}{\sqrt{\det \Sigma_k (2\pi)^d}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{x}-\boldsymbol{\mu}_k)} \quad (1)$$

Using this model, the EM algorithm is run in the following manner: The probability (at iteration step  $t$ ) of each data vector  $\mathbf{x}_n$  to belong to class  $k$  is calculated (expectation step) by

$$P^t(k|\mathbf{x}_n) = \frac{P^t(k) p(\mathbf{x}_n|\boldsymbol{\mu}_k^t, \Sigma_k^t)}{\sum_{j=1}^K P^t(j) p(\mathbf{x}_n|\boldsymbol{\mu}_j^t, \Sigma_j^t)}. \quad (2)$$

A new parameter set for the iteration step  $t + 1$  containing the prior probabilities, mean vectors and covariance matrices for each class is calculated according to (maximisation step)

$$P^{t+1}(k) = \frac{1}{N} \sum_{n=1}^N P^t(k|\mathbf{x}_n) \quad (3)$$

$$\boldsymbol{\mu}_k^{t+1} = \frac{1}{N P^{t+1}(k)} \sum_{n=1}^N P^t(k|\mathbf{x}_n) \mathbf{x}_n \quad (4)$$

$$\Sigma_k^{t+1} = \frac{1}{N P^{t+1}(k)} \sum_{n=1}^N P^t(k|\mathbf{x}_n) (\mathbf{x}_n - \boldsymbol{\mu}_k^{t+1})(\mathbf{x}_n - \boldsymbol{\mu}_k^{t+1})^T. \quad (5)$$

The algorithm is terminated at some stopping criterion, e.g., when the resulting labelling of the data vectors does not change anymore.

We examined this method in [4] and argued that a direct approach of this technique is not feasible even in case of a very good initialisation parameters. Therefore, a modified approach was introduced: Only data samples that were assigned *reliably* to a class are iterated on. This prevents the EM algorithm from deviating too much from its initial parameter set and segments potential haustoria regions quite well. Figure 3 shows the segmentation results using this constrained version of the EM algorithm. The pixel-labels are depicted in a soft-output manner, i.e., the vector of the posterior probabilities  $[P(k=3|\mathbf{x}), P(k=2|\mathbf{x}), P(k=1|\mathbf{x})]^T$  is assigned to the RGB value of each pixel, making the saturation of the colour follow the reliability of the estimate.

Nevertheless, we have been looking for less fragile methods providing better segmentation results, since there are some interferences in the outputs so far.

## 4 Morphological contrast enhancement

In addition to other applications like texture description, gradient calculation, etc., the powerful techniques of mathematical morphology [8] can be used to gain a significant contrast enhancement of images. This is basically done by *adding enhanced bright features* to the original image  $I$  while *subtracting enhanced dark features*

$$\kappa(I) = I + \underbrace{WTH_B(I)}_{\text{enhanced bright features}} - \underbrace{BTH_B(I)}_{\text{enhanced dark features}}. \quad (6)$$



Both, the enhanced bright and dark features are extracted by the morphological top-hat operations

$$\text{white top-hat (opening top-hat)} \quad WTH_B(I) = I - \gamma_B(I) \quad (7)$$

$$\text{black top-hat (closing top-hat)} \quad BTH_B(I) = \phi_B(I) - I, \quad (8)$$

respectively, using the morphological opening  $\gamma_B$  and closing  $\phi_B$ . In turn, the opening  $\gamma_B(I) = \delta_B[\epsilon_B(I)]$  and closing  $\phi_B(I) = \epsilon_B[\delta_B(I)]$  are based on the fundamental morphological operations dilation  $\delta_B$  and erosion  $\epsilon_B$ , using the structuring element  $B$ . Its shape and size has to be chosen according to the available knowledge about the structures to be filtered. Advantageously, morphological top-hats operate contrary – they extract structures that *cannot* contain the structuring element. Therefore, a simple rectangular or quadratic structuring element, being a little bit larger than the “waist” of a haustorium, is a good choice to enhance the haustoria as the wanted structures.

The theory of mathematical morphology is well established for grayscale images, though the question arises how to deal with colour images. Developing a theory of mathematical morphology for colour images, the step from a scalar-valued function to a vector-valued function needs the definition of a supremum and infimum for vectors instead of scalars, which in turn requires the definition of an order relationship of multivariate samples, making the development of a general theory on “colour-morphology” difficult [5]. Nevertheless, as a straightforward extension to grayscale morphology, a component-wise approach could be deployed which processes each vector component independently. This approach is chosen in the sequel, where the RGB components are treated as independent grayscale images.

The considerable effect of this contrast enhancement method is depicted in Figure 4. In these examples, a (flat) quadratic structuring element of size  $31 \times 31$  pixel was used. While the hue is almost unaffected (this is not shown here), the effect is most expressed in the saturation channel (see the miscoloured images). The saturation of the greenish blue cell tissue stays almost unchanged whereas the saturation of the haustoria is boosted in such a manner that it absolutely reaches its maximum value in some regions of each haustorium.

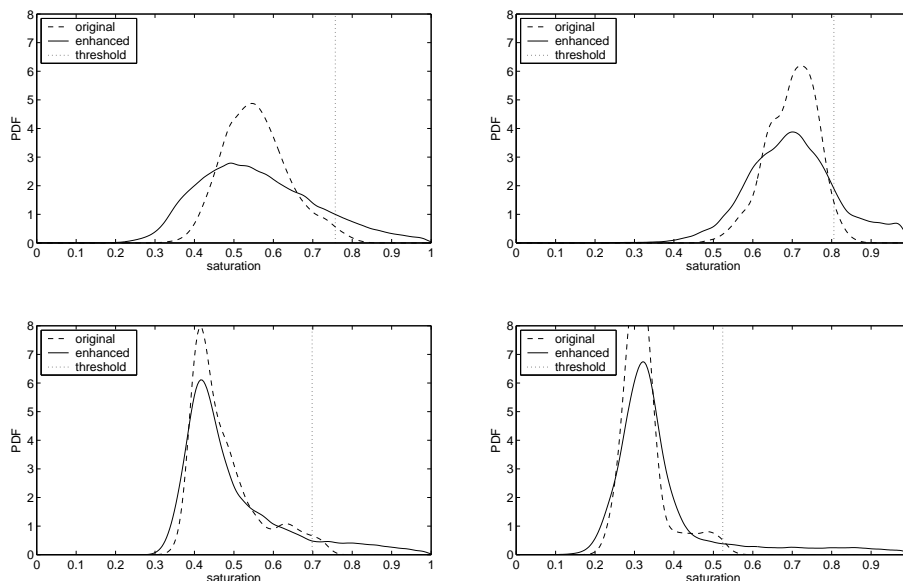
#### 4.1 Thresholding

This effect is exploited by a hysteresis thresholding technique: Since each haustorium reaches maximum saturation somewhere, this maximum value defines an upper threshold, while a lower threshold as a second parameter needs to be tuned on the data. Everything above the lower threshold is considered to be a haustorium if it is *connected* to a region segmented by the upper threshold.

Of course, the segmentation results strongly depend on the selection of the lower threshold. As we mentioned above and as can be clearly seen in the miscoloured images in Figure 4, the saturation of the cell tissue is almost not affected by the contrast enhancement – but the haustoria regions are. Now, the lower threshold is gained directly from the data: Consider the set of pixels belonging to maximum saturation level in the *enhanced* image. Using this set of pixels, we go back into the saturation channel of the *original* image and calculate the mean value of the pixels belonging to the



set. The result gives the lower threshold. Figure 1 depicts the corresponding probability density functions of the examples from Figure 4. Note their variability and the different resulting lower thresholds.



**Fig. 1.** Smoothed probability density functions of the saturation channel of the four exemplary cells of Figure 4. Maximally saturated pixels were excluded for convenience.

- dashed curve: PDF of saturation channel of original image
- solid curve: PDF of saturation channel of contrast enhanced image
- dotted vertical line: lower threshold used in hysteresis thresholding.

## 5 Conclusions

We applied two different segmentation techniques on microscope colour images of barley cells for the identification of small objects – so called haustoria – which stand out scarcely from the cell tissue. As the favorite solution, we propose to enhance the contrast of the cell image by morphological processing first and to perform a hysteresis thresholding in the saturation channel of the enhanced image afterwards. The upper and lower threshold is gained adaptively directly from the image data.

On the other hand, we presented in comparison a pixel-colour clustering technique based on the EM algorithm. Despite a good initialisation, this method needs a constraining mechanism to prevent a defection from the desired segmentation.

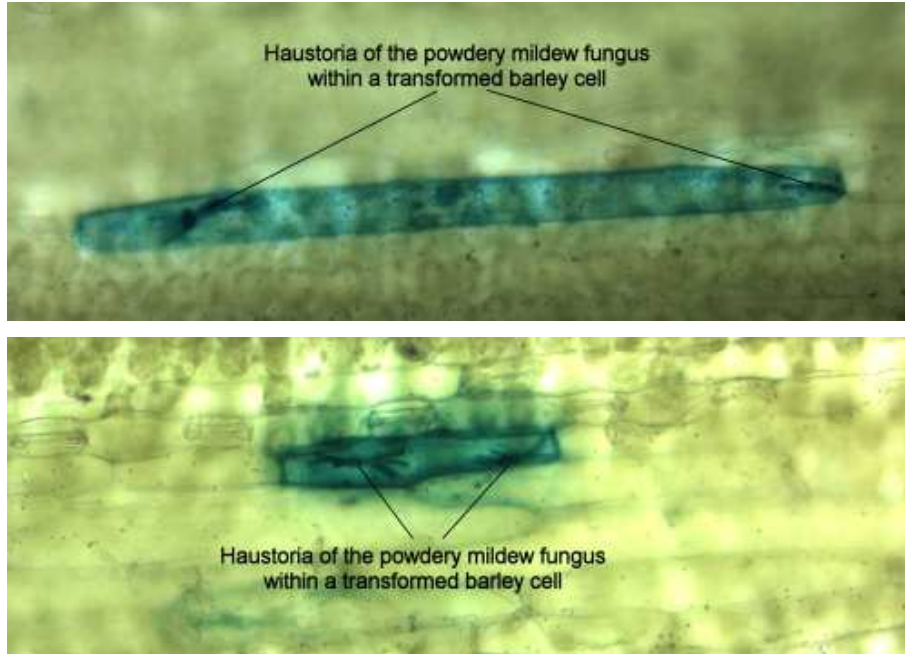
After all, the segmentation via morphological contrast enhancement provides rather good results; Haustoria are segmented properly. Now, a further shape analysis of the

segmented objects is needed to distinguish haustoria from other segmented objects. This is currently in progress.

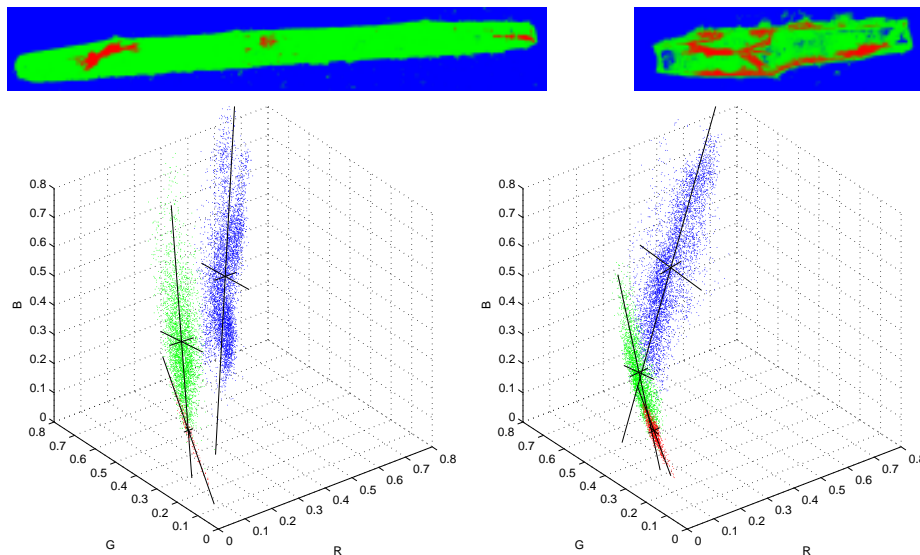
*Acknowledgements.* We thank Patrick Schweizer and the laboratory assistants for their support concerning the biological background. Thanks also to Christian Schulze and Tobias Czauderna for fruitful discussions. This work was supported by the German Ministry of Education and Research (BMBF) under grant 0312706A.

## References

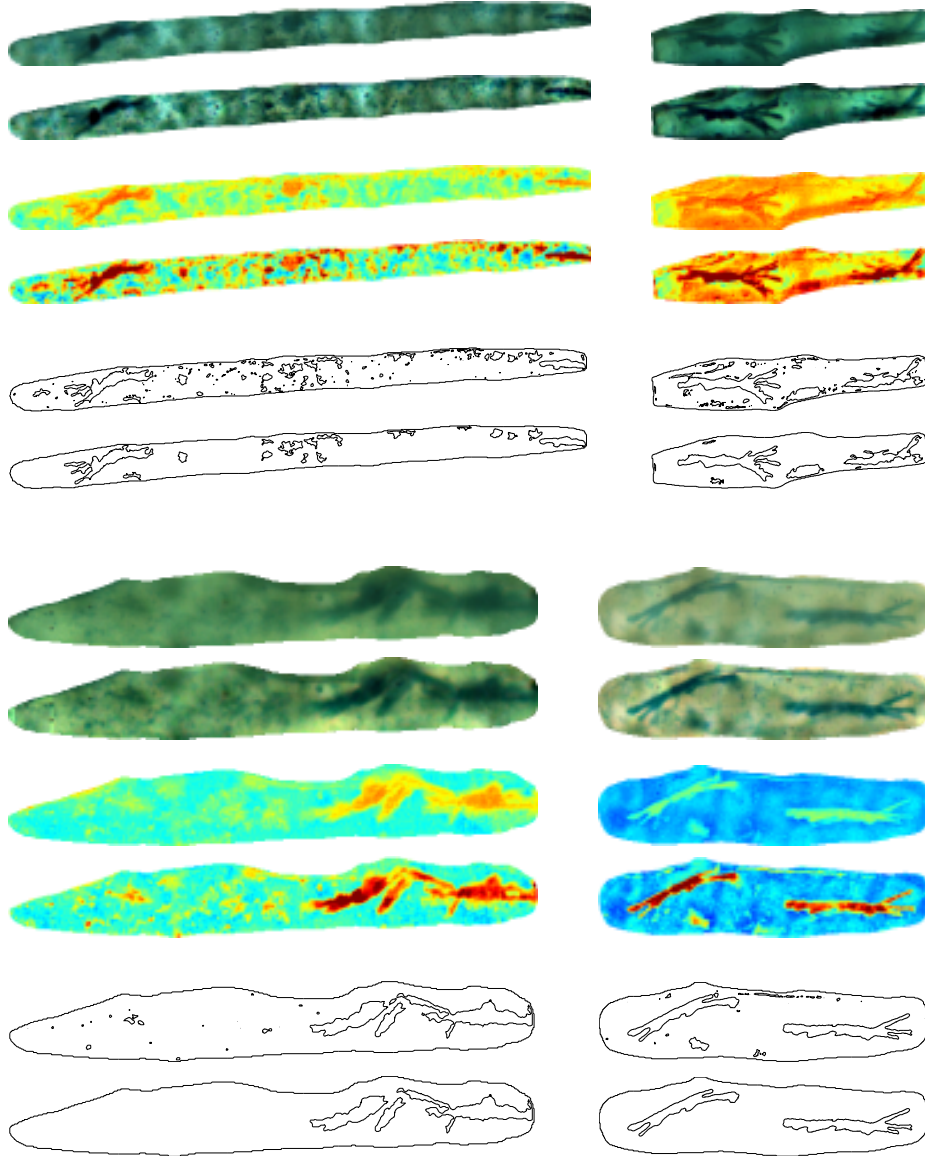
1. John F. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 8(6):679–698, November 1986.
2. Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.
3. Alexander Ihlow and Udo Seiffert. Microscope color image segmentation for resistance analysis of barley cells against powdery mildew. In *9. Workshop "Farbbildverarbeitung"*, ZBS Zentrum für Bild- und Signalverarbeitung e.V. Ilmenau, Report Nr. 3/2003, pages 59–66, Ostfildern-Nellingen, Germany, October 2003.
4. Alexander Ihlow and Udo Seiffert. Automating microscope colour image analysis using the Expectation Maximisation algorithm. In *DAGM'04: 26th Pattern Recognition Symposium*, pages 536–543, Tübingen, 30th Aug – 1st Sept 2004, September 2004. Springer.
5. Mario Köppen, Christoph Nowack, and Gert Rösel. Pareto-morphology for color image processing. In *11th Scandinavian Conf. on Image Analysis (SCIA99)*, pages 195–202, Kangerlussuaq, Greenland, June 1999.
6. Richard A. Redner and Homer F. Walker. Mixture densities, maximum likelihood, and the EM algorithm. *SIAM Review*, 26:195–239, 1984.
7. Patrick Schweizer, Jana Pokorný, Olaf Abderhalden, and Robert Dudler. A transient assay system for the functional assessment of defense-related genes in wheat. *Molecular Plant-Microbe Interactions*, 12(8):647–654, 1999.
8. Pierre Soille. *Morphological Image Analysis*. Springer, 2002.



**Fig. 2.** Two cutouts of typical microscope colour images: Both transformed (greenish blue coloured) barley cells contain two haustoria of the powdery mildew fungus. These objects have to be detected automatically by a high throughput screening system.



**Fig. 3.** Segmentation results of the two cells from Figure 2 via EM-clustering in the RGB colour space into “background” (blue), “cell” (green), and “haustorium” (red).



**Fig. 4.** Four examples for haustoria segmentation via morphological contrast enhancement of the colour image and subsequent thresholding in the saturation channel (for each of the four examples from top to bottom):

- cell image as acquired by the microscope camera
- cell image after contrast enhancement
- saturation channel of original cell image
- saturation channel of contrast enhanced image
- object boundaries (from sat. channel of enhanced image) by *simple* thresholding
- object boundaries (from sat. channel of enhanced image) by *hysteresis* thresholding.

## Farbanalyse von gebackenen oder frittierten Nahrungsmitteln und Vorhersage des Acrylamid-Wertes

Walter Hillen und Felix Fischer

Angewandte Informatik, Fachhochschule Aachen, Abtl. Jülich, 52428 Jülich  
hillen@fh-aachen.de, f.fischer@fh-aachen.de,  
Homepage: [www.medizinischeinformatik.fh-aachen.de](http://www.medizinischeinformatik.fh-aachen.de)

**Zusammenfassung.** Ein System zur automatischen Farbanalyse von gebackenen oder frittierten Nahrungsmitteln wurde realisiert, mit dem der Farbcharakter der Prüfobjekte nach international definierten Standards (Agtron-Klassifizierung) untersucht werden kann. Die Probe wird nach dem flächenmäßigen Anteil der verschiedenen Braunfärbungen (Agtron-Klassen) bewertet. Die Trennung von Objektbereichen und Hintergrund (Targetträger) sowie die Segmentierung entsprechend den Agtron-Farbklassen erfolgen im HSI-Farbraum. Die Untersuchungen zeigen eine eindeutige Korrelation zwischen dem Farbcharakter der Probe und den Acrylamid-(AcA-)Werten einer chemischen Analyse. Dieser Zusammenhang kann im System genutzt werden, um aus den HSI-Werten der Objekte eine Vorhersage über den zu erwartenden AcA-Wert abzuleiten. Das automatisch ablaufende Verfahren wurde in der plattform-unabhängigen Programmiersprache Java realisiert.

### 1 Einleitung

In der Produktion von gebackenen oder frittierten Nahrungsmitteln ist die Bewertung der Braunfärbung ein zentraler Bestandteil der Qualitätskontrolle. Hierbei wird die Prüfmenge nach verschiedenen vordefinierten Farbklassen selektiert (z. B. Agtron-Farbklassen  $Ag_1$  bis  $Ag_5$  in der Farbskala von hellgelb bis dunkelbraun [1]). Die bislang weit verbreitete visuelle Selektion erzeugt erhebliche systematische Probleme, da die objektive Bewertung des Farbcharakters und die Reproduzierbarkeit des Verfahrens schwierig sind. Zum anderen sind die einzelnen Prüfobjekte nicht homogen (dunkle Ränder, Fehlstellen, natürliche Strukturierungen), wodurch eine Auswertung der Farbanteile erschwert wird. Im folgenden wird ein Prüfsystem beschrieben, das den flächenmäßigen Anteil der vordefinierten (Agtron-)Farbklassen automatisch auswertet. Studien im Zusammenhang mit dieser Auswertung deuten darauf hin, dass die genaue Ausmessung des Farbcharakters auch für eine Vorhersage des Acrylamid-Wertes der Probe herangezogen werden kann.

Das Prüfsystem soll unter folgenden Rahmenbedingungen realisiert werden: Angestrebt wird ein universell einsetzbares, hochauflösendes System zur Farbbilderfassung mit einer Targetfläche von ca.  $0,5 \text{ m}^2$ . Es soll in einem Labor der Qualitätskontrolle als eigenständiges Aufnahmegerät eingesetzt werden und einfach zu bedienen und zu warten sein. Die Verfügbarkeit aller verwendeten Komponenten und die Kompatibilität mit zukünftigen Rechner- und Betriebssystemen müssen über einen längeren

Zeitraumen gesichert sein. Die Analyse-Software soll plattform-unabhängig eingesetzt werden. Weiterhin wird eine einfache Erweiterbarkeit verlangt für zukünftige Aufgaben der bildbasierten Qualitätskontrolle.

## 2 Aufnahmesystem

### Kamera und Beleuchtung

Die Kamera (Typ: *Nikon D1x* [2], Auflösung:  $3008 \times 1960$ ) ist auf einem Belichtungsstativ innerhalb eines lichtdichten Gehäuses montiert. Das System nimmt eine gesamte Objektfläche von  $80 \text{ cm} \times 52 \text{ cm}$  auf. Daraus folgt eine Bildpunktgröße von etwa  $0,27 \text{ mm} \times 0,27 \text{ mm}$ . Sämtliche automatischen Korrekturfunktionen der Kamera werden deaktiviert. Die Empfindlichkeit und der Weißabgleich sowie Blendenwert und Belichtung der Kamera sind fest eingestellt. Die Bilddaten werden als „primäre“ RGB-Daten übertragen ohne eine vorgeschaltete Verarbeitung, Korrektur oder Kompression. Hiermit wird eine kontrollierte und reproduzierbare Aufnahmesituation erreicht, was eine grundsätzliche Voraussetzung für die vorliegende Untersuchung ist.

Die Kamera ist über einen IEEE-1394 Anschluss mit einem PC (P4) verbunden. Die Datenübertragung erfolgt mit der *Nikon Capture Software* [3] in einem unkomprimierten RGB-TIFF Format.

Die Objektausleuchtung erfolgt in Auflicht. Für eine diffuse und homogene Beleuchtung dienen zwei Hochfrequenz-Reproleuchten (*Kaiser RB 5004 HF* [4]) mit jeweils 2 Stablampen von 38 cm Länge und einer Reflektorfläche von  $50 \text{ cm} \times 21 \text{ cm}$ . Die Leuchten sind links und rechts in Höhe der Kamera angeordnet.

### Targetträger

Die Untersuchungsobjekte werden auf einem homogen eingefärbten Karton positioniert. Die Hintergrundfarbe (türkis-blau) erlaubt eine sichere und einfache Trennung von Objekt- und Hintergrundbereichen über den Farbwert (siehe Abschnitt 4.2). Der Karton wird bei jeder Messung erneuert, wodurch die Reinigung des Targetträgers auf Grund der fettigen Untersuchungsobjekte entfällt. Mit Hilfe seitlich angebrachter weißer Referenzstreifen wird die Positionierung des Targetträgers, die Homogenität der Ausleuchtung und der Weißabgleich bei jeder Aufnahme überprüft (siehe Abschnitt 4.1).

## 3 Voruntersuchungen

In Voruntersuchungen werden Beleuchtungshomogenität und Weißabgleich optimiert. Ebenso werden die erwarteten Farbwerte für die späteren Untersuchungsobjekte grob charakterisiert und eine geeignete Farbe für den Objektträger (Hintergrund) festgelegt. Die Hintergrundfarbe soll dabei eindeutig von den Farbwerten der Untersuchungsobjekte zu trennen sein. Die Voruntersuchung wird mit dem Bildverarbeitungsprogramm *OPTIMAS 6.5* [5] durchgeführt.

Abbildung 1 zeigt eine Testaufnahme im Rahmen der Voruntersuchung. Als Bildhintergrund wird für die Voruntersuchungen eine rein-weiße Referenzfläche gewählt, um die Beleuchtungshomogenität und den Weißabgleich der Kamera einzustellen. Im unteren Bereich sind die Farbtafeln der Agtron-Klassen 1 (hell-gelb) bis 5 (dunkelbraun) der *Snack Food Association* dargestellt. Die Farbflächen (Abbildung 1, oben) dienen zur Auswahl einer geeigneten Hintergrundfarbe. Die Voruntersuchungen haben zu folgenden Ergebnissen geführt:

**Belichtungshomogenität:** Durch eine genaue Justierung der Hochfrequenz-Reproleuchten wird eine Belichtungshomogenität der gesamten Targetfläche von besser als  $\pm 3\%$  erreicht.

**Weißabgleich:** Für eine manuelle WeißEinstellung stehen der Kamera vordefinierte Einstellbereiche zur Verfügung, die jeweils vom Benutzer feinjustiert werden können. Der Weißabgleich wird im Bereich der weißen Referenzfläche des Test-Objektes durchgeführt, indem die RGB-Farbauszüge auf weitgehend gleiche Größe ( $\pm 2\%$ ) abgeglichen werden.

**Farbwerte der Objekte, Auswahl der Hintergrundfarbe:** Um die Farbwerte der Untersuchungsobjekte vorläufig zu charakterisieren werden die Testaufnahmen in ein HSI-Farbmodell überführt. Hierbei zeigt sich, dass die Hue-Farbwerte der Agtron-Farbtafeln 1 bis 5 zwischen 15 und 31 (Farbwinkelbereich:  $21^\circ$  bis  $47^\circ$ ) liegen. Sucht man für den Hintergrund einen genau „entgegengesetzten“ Farbwert für eine einfache Objekt-/ Hintergrundtrennung über den  $H$ -Wert, so ist die Farbe „Cyan“ mit  $H = 134$  (Farbwinkel:  $189^\circ$ ) von den in Abbildung 1 dargestellten Farben die beste Wahl.

**Langzeit-Stabilität, Reproduzierbarkeit und Kalibrierung:** Die Stabilität und Reproduzierbarkeit der Bilddaten werden über lange Zeiträume (mehrere Monate) beobachtet. Ebenso werden Einflüsse der Lampeneinschaltzeiten sowie Temperatureffekte ausgewertet. Insbesondere eine Intensitätsabnahme und eine leichte Farbwertveränderung der Leuchten im Laufe der Zeit machen eine Bildkalibrierung erforderlich. Sie erfolgt bei der Voruntersuchung und ebenso im endgültigen Prüfsystem, indem die primären RGB-Daten im Bereich der weißen Referenzfelder bei jeder Aufnahme auf vorbestimmte Werte geeicht werden (siehe Abschnitt 4.1). Diese Kalibrierung ergibt im Rahmen der erforderlichen Genauigkeit absolut reproduzierbare und langzeit-stabile Farbwerte der untersuchten Agtron-Farbklassen. Auf eine Farb-Profilierung der Kamera mit eigenen Testmessungen kann im Betrieb verzichtet werden.

## 4 Analyseverfahren

Abbildung 2 zeigt ein typisches Bild des endgültigen Prüfsystems, das von der Analysesoftware (Abschnitt 5) dargestellt wird.

### 4.1 Primärkontrollen und Kalibrierung

Wie oben beschrieben, dienen die sechs weißen Referenzfelder am rechten und linken Bildrand für folgende Primärkontrollen:



**Positionierung des Objektträgers:** An vorbestimmten ROIs, an denen die Referenzfelder bei korrekter Positionierung erwartet werden, werden Mittelwerte und Standardabweichungen der Bildhelligkeit  $I = \frac{1}{3}(I_R + I_G + I_B)$  sowie Mittelwerte der Farbauszüge  $I_R$ ,  $I_G$  und  $I_B$  bestimmt. Eine Fehlpositionierung des Objektträgers sowie eine Beschädigung bzw. Verschmutzung der Referenzfelder wird an einem sprunghaften Anwachsen der Standardabweichungen erkannt und als Fehler signalisiert.

**Homogenität der Ausleuchtung:** Die Mittelwerte der Bildhelligkeit werden für eine Überprüfung der Beleuchtungshomogenität herangezogen.

**Überprüfung des Weißabgleichs:** Der in der Kamera fest eingestellte Weißabgleich wird über einen Vergleich der Mittelwerte von  $I_R$ ,  $I_G$  und  $I_B$  kontrolliert.

Nach den Primärkontrollen erfolgt die Kalibrierung der R-, G- und B-Farbkanäle durch einen R-, G- und B-Eichfaktor in der Weise, dass die Mittelwerte ( $I_R$ ,  $I_G$ ,  $I_B$ ) der kalibrierten Bilddaten auf den Referenzfeldern gleiche vorbestimmte Werte annehmen.

## 4.2 Hintergrundabtrennung

Zur weiteren Analyse wird ein ROI gesetzt, das die gesamte Targetfläche mit den Prüfobjekten erfasst. Der Bildausschnitt wird anschließend in den HSI-Farbraum transformiert. In der Verteilung der Farbwerte  $H$  („Hue“) innerhalb der Targetfläche unterscheiden sich die Farbwerte der Objekte ( $H$  im Bereich von 20 bis 60) deutlich vom Farbwert des Targethintergrundes ( $H \approx 134$ ). Setzt man eine Schwelle in der  $H$ -Verteilung bei  $H = 90$ , so kann man Objekt- und Hintergrundbereiche eindeutig voneinander trennen (auch Schlagschattenbereiche werden erkannt und dem Hintergrund zugeordnet).

Durch die Segmentierung über den  $H$ -Wert wird eine binäre „Bildmaske“ erzeugt. Um in der weiteren Analyse keine Bildpunkte mit einzubeziehen, die im Übergangsbereich von Objekten zu Hintergrund liegen, wird der Objektrand der Maske um eine vorgegebene Zahl von Bildpunkten zu Gunsten des Hintergrundes verschoben. Dies wird durch eine morphologische Filteroperation der Bildmaske erreicht.

## 4.3 Farbwert-Analyse der Agtron-Klassen

Innerhalb der gebackenen bzw. frittierten Objekte werden von Hand repräsentative Proben entsprechend den vorgegebenen Agtron Farbklassen 1 bis 5 ausgewählt. Tabelle 1 zeigt die HSI-Mittelwerte (mit Standardabweichungen) der Agtron Proben. Es ist zu erkennen, dass die HSI-Farbwerte der einzelnen Klassen sich deutlich unterscheiden. Auch wenn diese visuelle Selektion mit höchster Sorgfalt erfolgt, haben die Farbwerte der Stichproben größere Streuungen, da die Objekte naturgemäß keine homogene Färbung besitzen.

Die Farbklassifizierung erfolgt auf der Grundlage der Tabelle 1, wobei die aufgeführten Mittelwerte die „Klassenzentren“ der Agtron Farbklassen  $Ag_1$  bis  $Ag_5$  darstellen. Die Zuordnung eines einzelnen Bildpunktes  $P_{(x,y)}$  zu einer bestimmten Agtron Klasse erfolgt aus dem Minimum der Werte  $D_1$  bis  $D_5$ . Dabei sind  $D_1$  bis  $D_5$  die auf die jeweiligen Standardabweichungen normierten Abstände zu den Klassenzentren 1 bis 5:



**Tabelle 1.** HSI-Farbwerte (mit Standardabweichungen) der Agtron Proben

	Ag. 1	Ag. 2	Ag. 3	Ag. 4	Ag. 5
H (Farbe)	50 ± 4	44 ± 4	39,5 ± 7	30 ± 5	25,5 ± 6
S (Sättigung)	64 ± 13	97 ± 18	120 ± 26	130 ± 20	135 ± 17
I (Intensität)	146 ± 12	126 ± 12	117 ± 17	95 ± 14	75 ± 15

$$D_i^2(x, y) = \frac{(H_{(x,y)} - H_{Ag(i)})^2}{\sigma_{H_{Ag(i)}}^2} + \frac{(S_{(x,y)} - S_{Ag(i)})^2}{\sigma_{S_{Ag(i)}}^2} + \frac{(I_{(x,y)} - I_{Ag(i)})^2}{\sigma_{I_{Ag(i)}}^2}$$

Hierin sind:  $H_{(x,y)}$ ,  $S_{(x,y)}$ ,  $I_{(x,y)}$  der H-, S-, I-Wert des Bildpunktes  $P_{(x,y)}$ ;  $H_{Ag(i)}$  und  $\sigma_{H_{Ag(i)}}^2$  sind Mittelwert und Varianz der  $Ag(i)$ - Klasse (Tabelle 1).

Gemäß dieser Klassifizierung wird jeder Objektpunkt einer Ag-Klasse zugeordnet. Abschließend werden daraus die prozentualen Flächenanteile der Agtron-Klassen über der gesamten Probenfläche ermittelt.

#### 4.4 Vorhersage des Acrylamid-Wertes

In einer begleitenden Untersuchung wird überprüft ob, ein Zusammenhang zwischen dem Farbcharakter und dem Acrylamid (AcA) -Wert einer Probe existiert. Der AcA-Wert wird in einer anschließenden chemischen Analyse bestimmt.

Für die Messungen werden insgesamt sieben Proben einer Produktionscharge visuell mit höchster Sorgfalt selektiert. Jede Probe enthält Objektstücke von genau einem Farbton. Von jeder Probe wird eine HSI-Farbauswertung und anschließend eine chemische Untersuchung durchgeführt. Der Zusammenhang zwischen den mittleren Farbwerten (hier H und S) und dem AcA-Gehalt ist in Abbildung 3 und 4 dargestellt. Ein exponentieller Fit beschreibt den gemessenen Zusammenhang im Rahmen der Fehlerbreiten (Standardabweichungen  $\sigma_H$ ,  $\sigma_S$  der Proben) relativ gut.

Für eine AcA-Vorhersage kann auf der Basis dieser Regressionskurven aus den HSI-Werten jedes Bildpunktes ein zugehöriger AcA-Wert abgeleitet werden. Werden diese Beiträge über die Probenfläche aufsummiert, erhält man eine Vorhersage des zu erwartenden AcA-Gehalts der Probe. Die Ergebnisse dieser Studie sind vorläufig und wurden bislang nur für eine Produktgruppe untersucht (siehe Abschnitt 6).

## 5 Analysesoftware des Prüfsystems

Das Verfahren der Agtron-Farbklassifizierung im HSI-Farbraum wurde als plattform-unabhängige Java-Applikation mit folgenden Merkmalen realisiert:

- Überprüfung des Aufnahmesystems (Kamera und Beleuchtung) vor jeder Analyse.
- Automatische Durchführung und Bearbeitung der Messung.

- Generierung eines Berichtes mit allen wesentlichen Informationen zur Probe, den Kamera-Einstellungen, den Analyse-Ergebnissen sowie des bedienenden Benutzers.
- Chronologische Fortschreibung einer Datenhistorie in Bezug auf Produktsorte, Zeiträume und Ergebnisse.

## 6 Bewertung und Diskussion

Die beschriebene Farbklassifizierung entsprechend den Agtron-Farbklassen von gebackenen oder frittierten Nahrungsmitteln lässt sich im HSI-Farbraum relativ einfach und effizient umsetzen. Das „Basiswissen“ des Verfahrens steckt klar formuliert in den HSI-Klassenzentren der Agtron-Farbwerte. Diese Werte sind leicht in einer Einlernphase festzulegen und gegebenenfalls zu optimieren. Die Agtron-Klassenzentren liegen im HSI-Farbraum deutlich getrennt voneinander; die Objektpunkte können durch eine einfache Abstandsnorm zugeordnet werden. Das Resultat ist eine genaue Analyse des flächenmäßigen Anteils der einzelnen Agtron-Farbklassen. Das Verfahren liefert objektive, reproduzierbare und leicht zu dokumentierende Ergebnisse. Vergleichbare Aussagen lassen sich mit einer visuellen Sortierung der Prüfobjekte in der Praxis nicht erreichen. Aus der Analyse können unmittelbar Parameter zur Optimierung des Back- oder Frittierprozesses gewonnen werden.

Der beobachtete Zusammenhang zwischen dem Farbcharakter einer Probe und dem Acrylamid (AcA)-Wert ist eine vielversprechende Methode, um eine Vorhersage über den zu erwartenden AcA-Gehalt abzuleiten. Hierbei entfallen die zusätzlichen Kosten und längeren Wartezeiten einer chemischen Analyse. Für eine abschließende Bewertung des Verfahrens muss allerdings weitaus mehr Probenmaterial charakterisiert werden und die abschließende Bestimmung des AcA-Wertes aus den gewonnenen Regressionskurven optimiert werden.

## Literatur

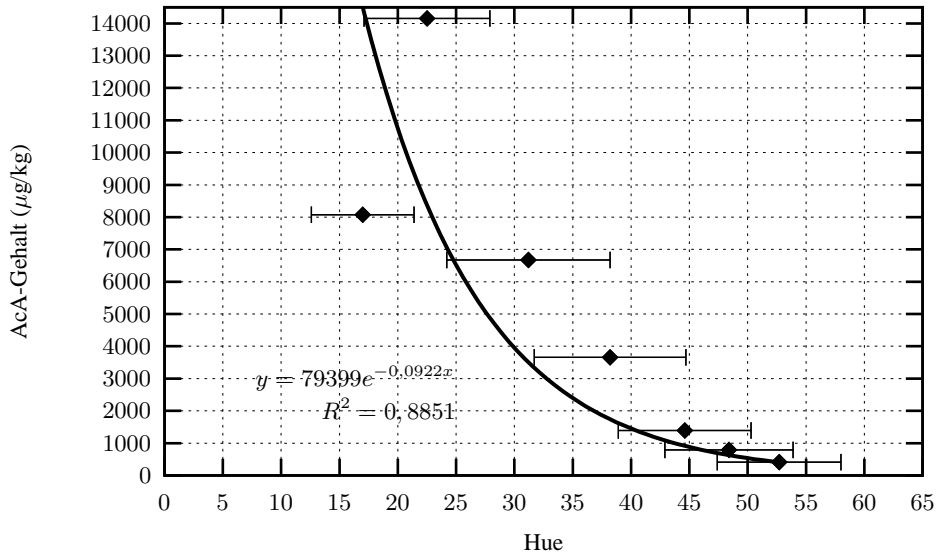
1. Agtron Farbkarten Fry Colours Standards for Potatoes Chipping (PCSFA Agtron), Herausgeber: SNACK FOOD ASSOCIATION, 1711 King Street, Suite One, Alexandria, Virginia 22314, Tel.: (703) 836-4500, Fax: (703) 836-8261.
2. Nikon Imaging. <http://nikonimaging.com/global/products/digitalcamera/slr/d1x/>
3. Nikon Imaging. <http://www.nikon-image.com/eng/software/>
4. Kaiser Fototechnik. <http://www.kaiser-fototechnik.de/en/produkt.asp?artikel=5558>
5. Media Cybernetics, Inc. <http://www.optimas.com>



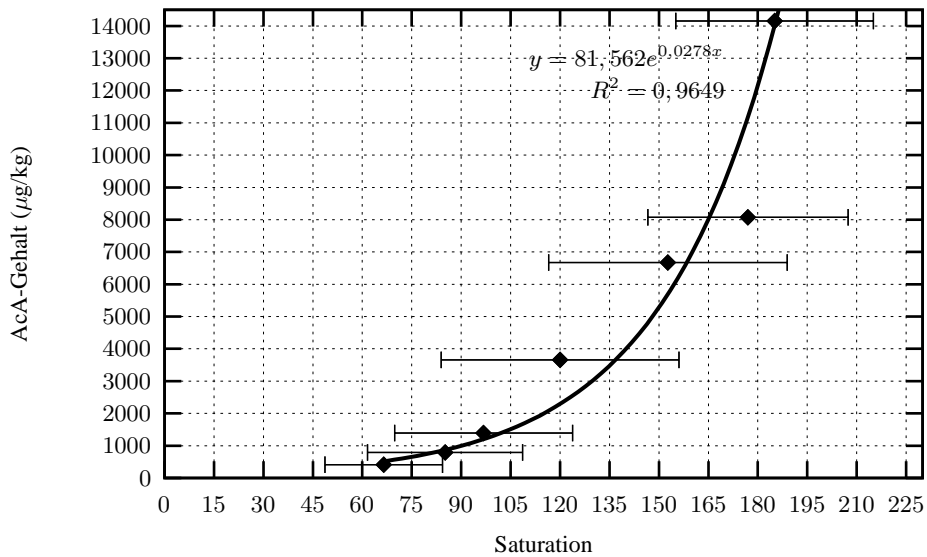
**Abb. 1.** Testaufnahme zu Voruntersuchungen



**Abb. 2.** Aufnahme des Prüfsystems (dargestellt innerhalb der Analysesoftware)



**Abb. 3.** Zusammenhang zwischen Farbcharakter (Hue-Mittelwert) und AcA-Wert der farbselektierten Proben



**Abb. 4.** Zusammenhang zwischen Sättigung und AcA-Wert der farbselektierten Proben

## **Farbauswertung in der Mikroskopie - wo liegen die Herausforderungen der Anwender?**

Peter Schwarzmann

Heerstr. 39, 72555 Metzingen

**Zusammenfassung.** Anwender von Werkzeugen zur Bildauswertung stellen zunehmend erhöhte Anforderungen an die qualitative und quantitative Auswertung zytologischer und histologischer Schnitte. Realistische Modellvorstellungen in Auswerteprogrammen lassen systematische Fehlerquellen qualitativ und quantitativ beurteilen. Der Beitrag beschreibt Fehlerquellen bei der quantitativen Auswertung feiner Strukturen, die Abhängigkeit des Farbtones bei Konzentrationsänderungen von breitbandigen Farbstoffen und schließlich die rechnerische Entmischung von Farbstoffen.

### **1 Einleitung**

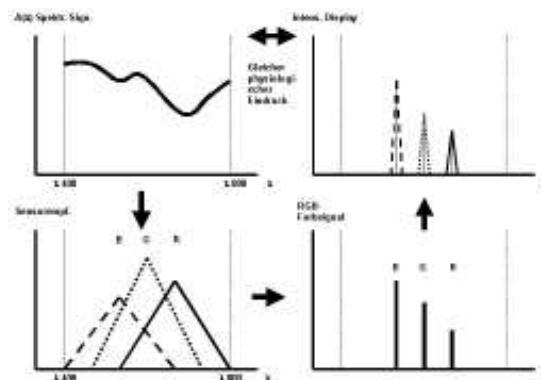
Die klassische Herausforderung der automatischen Bildauswertung in der Mikroskopie mit großem gesundheitspolitischem und wirtschaftlichem Hintergrund war, und ist auch heute noch, das Massenscreening zur Früherkennung des Zervixkarzinoms mit nach Papanicolaou gefärbten Präparaten (PAP-Smears). Trotz großer und langjähriger internationaler Programme ist diese Aufgabe bis heute noch nicht zufriedenstellend gelöst. An diesem Zellbildauswerteproblem lassen sich wesentliche Herausforderungen sehr gut darstellen und auf andere Aufgaben übertragen. So erlauben heute neue Verfahren in der Mikroskopie, zusammen mit neuen Präparationstechniken und Farbmarkern, attraktive, qualitative, und vor allem auch quantitative Einsichten, in die Struktur und Funktionalität von Zellen und Geweben. Zur Illustration zeigt Abb. 7 einen Ausschnitt aus einem PAP-Präparat und Abb. 8 (oben) aus einem histologischen Präparat.

Den Anwender, in diesem Falle den Zytologen, interessieren semantische Aussagen in solchen Präparaten wie: Zellkern, Zytoplasma, Chromatinverteilung, DNA-Gehalt, Kernformen, Populationsdichten usw.; die Technik der Bildverarbeitung operiert dagegen mit: Grauwerten von Pixeln, Schwellwerten, Flächen, RGB-Farbwerten, Objektdetektion, Objektsegmentation usw. Die Verbindung zwischen diesen beiden Denkwelten ist nicht einfach herzustellen und enthält die Herausforderungen für die heutige problemorientierte Bildauswertung. Ein Erfolg versprechender Verbindungsweg führt über genügend realistische Modellvorstellungen. Einige Probleme mit Farbmodellen auf dem Weg dorthin soll dieser Beitrag ansprechen, da die an sich bekannten Probleme immer wieder in Vergessenheit geraten und regelmäßig wieder neu aufgegriffen werden. Angesprochen werden Probleme kleiner Strukturen, der Farbverschiebung bei sich ändernder Farbstoffkonzentration, sowie der Entmischung von Farben bei Überlagerung.

## 2 Benutzte Grundlagen

### 2.1 Farbe und spektrale Signatur

Farbe ist eine Eigenschaft des menschlichen Gesichtssinnes; sie ist das Ergebnis einer Information reduzierenden, bewertenden Integration der spektralen Signatur mit 3 Sensorempfindlichkeiten mit Schwerpunkten im roten, grünen und blauen Bereich. Abb. 1 gibt die Verhältnisse wieder.



**Abb. 1.** Stationen auf dem Wege von der spektralen Signatur eines Signales bis zur Farbdarstellung für das visuelle System des Auges.

### 2.2 Farbspektren in Emission und Absorption

Typischer Vertreter der Mikroskopie in Emission ist die Fluoreszenzmikroskopie. Die spektrale Verteilung der Emission der verschiedenen Fluoreszenzfarbstoffe ist schmalbandig im Vergleich zum sichtbaren Spektrum. Abb. 2 (links) skizziert Emissionsspektren einiger Fluoreszenzfarbstoffe.

Ganz anders dagegen die Situation bei Absorptionsfarbstoffen (normale Hellfeldmikroskopie). Die Farbstoffe zeigen Wellenlänge abhängige Absorptionskennlinien meist über das gesamte sichtbare Spektrum. Abb. 2 (rechts) zeigt beispielhaft die Absorptionskonstanten von 2 gängigen Farbstoffen (Hämatoxylin und Diaminobenzidin DAB). Die einzelnen Farbstoffe können durch Farbfilter nicht auseinandergehalten werden - die Farbinformation ist über das gesamte Spektrum verteilt. Jeder Farbstoff zeigt 'Übersprechen' in die Bereiche des anderen.

### 2.3 Die Quantifizierung des Farbstoffgehaltes

In Emission: Der Farbstoffgehalt  $M_{em}$  der Probe ist proportional zur gemessenen Intensität:

$$M_{em} = b \cdot I \cdot d \cdot k$$

mit  $b$ : Proportionalitätskonstante,  $I$ : gemessene Intensität,  $d$ : Dicke der gefärbten Schicht,  $k$ : Farbkonzentration.

In Absorption: Der Farbstoffgehalt ist proportional zur Extinktion  $M_{abs}$  der Probe:

$$M_{abs} = a(\lambda) \cdot k \cdot d = \ln I_0(\lambda) - \ln I(\lambda)$$

mit  $a(\lambda)$ : Wellenlängen abhängige Absorptionskonstante,  $I_0$ : Einfallende Intensität,  $I$ : Transmittierte Intensität. Dieser Ausdruck geht aus der Absorptionsbeziehung hervor:

$$I(\lambda) = I_0(\lambda) \cdot e^{-a(\lambda) \cdot k \cdot d}$$

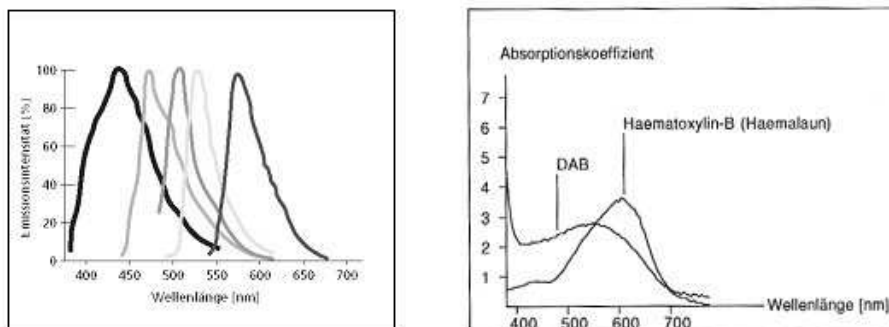
## 2.4 Farbmischung und -Entmischung in Absorption

Am Beispiel von 2 Farbstoffen soll dies erläutert werden (eine Erweiterung auf weitere ist möglich). Modelle für die Farbmischung können sowohl im RGB-Farb-raum als auch direkt aus den spektralen Signaturen der Farbstoffe entwickelt werden. Modelle im RGB-Farb-raum (oder davon abgeleiteten Räumen) erfordern aber die Beachtung von Farbverschiebungen bei Konzentrationsänderungen der Farbstoffe wie in einem der nachfolgenden Abschnitte dargestellt.

Es sind 2 grundsätzlich unterschiedliche Modelle zu unterscheiden: Die 'additive' bzw. 'parallele' Farbmischung innerhalb eines Bildpixels und die 'subtraktive' bzw. 'serielle' Farbmischung innerhalb eines Bildpixels. Abb. 3 verdeutlicht den Sachverhalt. Desweiteren sind natürlich auch Mischungen dieser beiden Modelle denkbar; sie sollen aber im Rahmen dieses Beitrages nicht behandelt werden.

Für den additiven Fall erhält man für die 3 Farbkomponenten eines Bildpixels im RGB-Raum:

$$I_x = \int_{F_1} \int_{\lambda} I_0 \cdot e^{-a_1(\lambda) \cdot d \cdot k_1} \cdot S_x(\lambda) \cdot d\lambda \cdot df + \int_{F_2} \int_{\lambda} I_0 \cdot e^{-a_2(\lambda) \cdot d \cdot k_2} \cdot S_x(\lambda) \cdot d\lambda \cdot df + \int_{F_0} I_0 \cdot df$$



**Abb. 2.** Links: Emissionsspektren einiger Fluoreszenzfarbstoffe (v.l.n.r. BFP, CFP, GFP, YFP, DSRRed); rechts: Absorptionskonstanten von 2 Absorptionsfarbstoffen: Hämatoxylin (bläulich), Diaminobenzidin DAB (braungelblich)

mit:  $x$ : eine der 3 Komponenten  $R$ ,  $G$  oder  $B$ ;  $F_1$ ,  $F_2$ : Flächenanteile der Farbpartikel  $A_1$  und  $A_2$ ;  $F_0$ : Flächenanteil ohne Färbung;  $a_1$ ,  $a_2$ : Absorptionskonstanten von Partikel  $A_1$  und  $A_2$ ;  $k_1$ ,  $k_2$ : Farbkonzentration in Partikel  $A_1$  und  $A_2$ ;  $S_x$ : Sensitivität des Farbkanals  $x$  ( $R, G$  oder  $B$ ).

Das inverse Problem, die Bestimmung von  $\int_{F_1} d \cdot k_1 \cdot df$  bzw.  $\int_{F_2} d \cdot k_2 \cdot df$ , bedeutet dann 'einfach (! - aber meist nicht durchführbar!)' die Auflösung des Integralgleichungssystems nach diesen beiden Werten; bei bekannten Farbstoffen sind alle anderen Größen ja bekannt.

Für den subtraktiven Fall der Überlagerung der beiden Färbungen ergeben sich die folgenden Beziehungen:

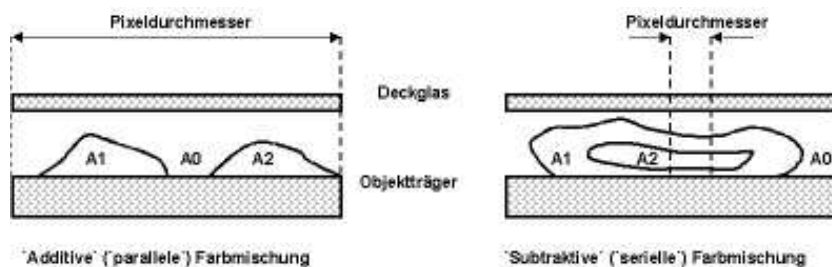
$$I_x = \int_{\lambda} I_0 \cdot e^{-a_1(\lambda) \cdot k_1 \cdot d_1} \cdot e^{-a_2(\lambda) \cdot k_2 \cdot d_2} \cdot S_x(\lambda) \cdot d\lambda$$

Auch in diesem Falle erhalten wir für das inverse Problem der Bestimmung von  $(k_1 \cdot d_1)$  und  $(k_2 \cdot d_2)$  ein Integralgleichungssystem. Es ist allerdings bedeutend einfacher als im additiven Falle. Dies deshalb, weil innerhalb des Bildpixels konstante Werte für die Extinktion  $k \cdot d$  angenommen werden können.

In beiden Fällen vereinfachen sich die Verhältnisse beträchtlich, wenn für die Sensorempfindlichkeit sehr schmalbandige (Idealfall:  $\delta$ -Stoß) Wellenlängenabhängigkeiten durch Verwendung schmalbandiger Farbfilter angenommen werden können. Im subtraktiven Falle erhält man dann zur Lösung ein lineares Gleichungssystem, welches bei Messung in 3 Kanälen sogar überbestimmt ist und damit eine Ausgleichsrechnung zulässt. Gleiches würde gelten, wenn sehr schmalbandig absorbierende Farbstoffe zur Verfügung stehen würden.

### 3 Messung von Farbstoffgehalten feiner Strukturen in Absorption

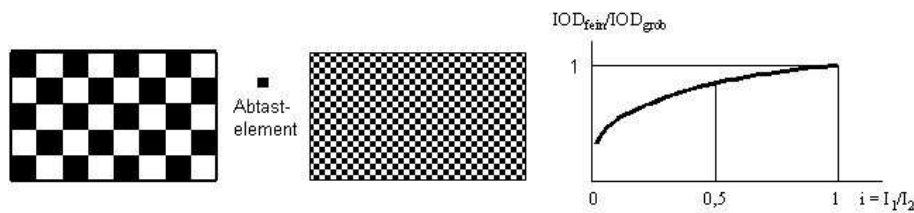
Klassisches Beispiel für eine solche Messung ist die *IOD* (Integrierte Optische Dichte) zur Bestimmung des DNA-Gehaltes eines entsprechend gefärbten Zellkernes. Die Strukturen liegen an der Auflösungsgrenze des Lichtmikroskops.



**Abb. 3.** Das 'additive' ('parallele') und das 'subtraktive' ('serielle') Mischungsmodell für 2 Farbstoffe.  $A_1$  und  $A_2$  sind mit unterschiedlichen Farbstoffen gefärbte Objekte.



Zur Demonstration der Fehlerquelle bei derartigen Messungen seien 2 Präparate mit gleichem Farbstoffgehalt pro Präparatfläche  $F$ , jedoch mit unterschiedlicher Strukturgröße, angenommen (Abb. 4). Die grobe Struktur sei wesentlich größer als ein Bildpixel, die feine dagegen liege darunter, so dass die Bildpixel gemittelte Werte der Struktur darstellen.



**Abb. 4.** Systematischer Fehler bei der Messung der  $IOD$  für Strukturen an der Auflösungsgrenze des Abtaststrahlers. Das Diagramm zeigt den Grenzwert des Messfehlers bezogen auf den wahren Wert als Fkt. des Kontrastes.

Nimmt man als Grenzfall für den Fehler an, dass ein Pixel den Mittelwert der beiden Transmissionswerte darstellt, so erhält man als  $IOD$  d.h. Farbstoffgehalt des Präparates die Werte:

$$IOD_{grob} = 0,5 \cdot F \cdot (\ln I_0 - \ln I_1) + 0,5 \cdot F \cdot (\ln I_0 - \ln I_2)$$

$$IOD_{fein} = F \cdot [\ln I_0 - \ln(0,5 \cdot (I_1 + I_2))]$$

Führt man noch ein Kontrastmaß  $i = I_1/I_2$  ein, so erhält man nach einfachen Umformungen für das Verhältnis der beiden Messwerte:

$$\frac{IOD_{fein}}{IOD_{grob}} = \frac{\ln I_0 - \ln I_2 + \ln 2 - \ln(i + 1)}{\ln I_0 - \ln I_2 - \ln \sqrt{i}}$$

Beachtet man, dass  $I_2 < I_0$  bleiben muss, so erhält man die Messabweichung als Funktion des Kontrastes  $i$  wie exemplarisch in Abb. 4 (rechts) für  $I_2 = I_0$  dargestellt.

Zusätzlich sind Farbfehler zu erwarten, wenn es sich um farbige Strukturen handelt. Ein einfaches Beispiel möge dies veranschaulichen: wie in Abb. 4 sei diesmal ein Muster aus blauen und gelben Flächen gegeben. Erwartungsgemäß geht dieses Muster an der Auflösungsgrenze in ein grünes Moirémuster über.

Der beschriebene Fehler kommt zu den bekannten Fehlern durch Beugungserscheinungen und Glare hinzu.

#### 4 Farbverschiebungen in Absorption bei Konzentrationsänderungen

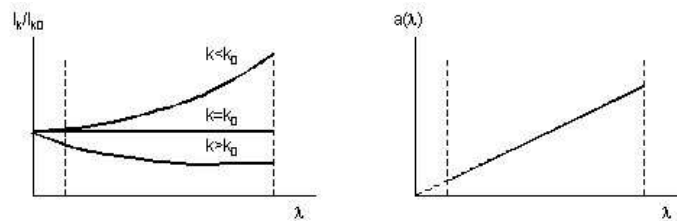
Gegeben sei ein Farbstoff mit der spektralen Absorptionskonstanten  $a(\lambda)$ , der Farbstoffkonzentration  $k_0$  und der Objektstärke  $d$ . Bei einfallendem weißem Licht  $I_0$  erhält man dann als transmittiertes Licht  $I_{k_0}$ :

$$I_{k_0}(\lambda) = I_0 \cdot e^{-a(\lambda) \cdot k_0 \cdot d}$$

Ändert sich die Farbstoffkonzentration auf den Wert  $k$ , so ergibt sich für das Verhältnis der beiden Intensitäten:

$$\frac{I_k(\lambda)}{I_{k_0}(\lambda)} = e^{-a(\lambda) \cdot d \cdot (k - k_0)}$$

Man sieht, obwohl lediglich die Farbkonzentration  $k$  geändert wurde, ist das Intensitätsverhältnis keine Konstante, sondern mit der Absorptionskonstanten  $a(\lambda)$  Wellenlängen abhängig. Ein Beispiel möge wieder die Auswirkungen verdeutlichen (Abb. 5). Angenommen wurde eine mit der Wellenlänge  $\lambda$  linear ansteigende Absorptionskonstante  $a(\lambda)$ . Skizziert sind die Verhältnisse für  $k < k_0$ ,  $k = k_0$  und  $k > k_0$ . Wie man sieht, verschiebt sich der Intensitätsschwerpunkt für  $k < k_0$  zu größeren (rot) und für  $k > k_0$  zu kleineren (blau) Wellenlängen.



**Abb. 5.** Beispiel für eine Farbverschiebung bei Farbstoffkonzentrationsänderung; angenommen ist eine linear mit  $\lambda$  ansteigende Absorptionskonstante  $a(\lambda)$ .

## 5 Entmischung von Farbkomponenten

Ein kritischer Prozessschritt bei der Auswertung zytologischer und histologischer Präparate ist die Segmentation der Komponenten (Zellen, Zytoplasma, Zellkern) für Standardabsorptionsfärbungen (Hämatoxylin, Eosin, Diaminobenzidin). Es zeigt sich, dass hellere Kernbereiche dabei oft dem Zytoplasma zugeordnet werden, oder bei anderer Einstellung des Klassifikators dann dünne Zytoplasmagebiete nicht mehr erkannt werden. Aus Abb. 3 als Prinzipskizze und Abb. 8 (unten rechts) als reale Szene ergibt sich als Grund dafür: Eosin gefärbtes Zytoplasma überlagert die Zellkerne. Deutlich verbesserte Segmentationsergebnisse versprechen daher Bildvorverarbeitungsschritte, welche das Bild in die beiden Färbungen aufteilen. Es macht daher Sinn, aus den im Grundlagenabschnitt beschriebenen Modellvorstellungen nach einer Lösung für das Trennungproblem der beiden Farbstoffe zu suchen.

Ausgangspunkt bleibt, dass breitbandig absorbierende Farbstoffe zugelassen werden müssen, und daher der Lösungsweg über die Integralgleichungsform gehen muss. Eine solche Lösung kann umgangen werden, wenn folgende Näherung akzeptierbar

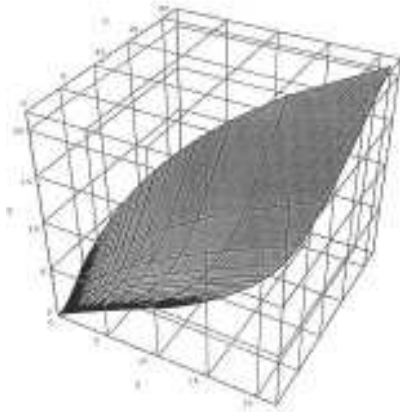
ist: Aus den bekannten Absorptionskennlinien  $a_1(\lambda)$  und  $a_2(\lambda)$ , sowie den Beziehungen zur Berechnung der Farbkanalintensitäten, wird eine Tabelle mit 2 Eingängen ( $k_1 \cdot d$ ) und ( $k_2 \cdot d$ ) und 3 Ausgängen  $I_{rot}$ ,  $I_{grün}$ ,  $I_{blau}$  berechnet, und die Werte im RGB-Farbraum aufgetragen. Abb. 6 gibt solch ein Mischungsdiagramm für 2 Farbstoffe wieder.

Die Menge aller Mischungsverhältnisse stellt sich dort als Fläche dar, welche vom 0-Punkt ausgeht (Schwarzpunkt, keine Transmission) und im Weißpunkt (kein Farbstoff, völlige Transparenz) endet. Auf dieser Fläche lassen sich die jeweiligen Farbanteile als Netz für die Parameter ( $k_1 \cdot d_1$ ) und ( $k_2 \cdot d_2$ ) einzeichnen. Aus den Intensitätsmessungen der 3 Farbkanäle können dann durch Interpolation im Diagramm oder in der Tabelle die Farbstoffkonzentrationen mit ausreichender Genauigkeit entnommen werden. Messfehler bzw. Ungereimtheiten zeigen sich bei diesem Vorgehen dadurch, dass Messwerte nicht genügend nahe auf der Mischungsfläche liegen. Wie alle Absorptionsverfahren, werden die Fehler zunehmend größer, wenn man sich dem 0-Punkt (vollständige Absorption) nähert. Die Auswertungsgrenze kann ein Stück verschoben werden, wenn in einer zweiten Bildaufnahme mit stärkerer Beleuchtung gearbeitet wird.

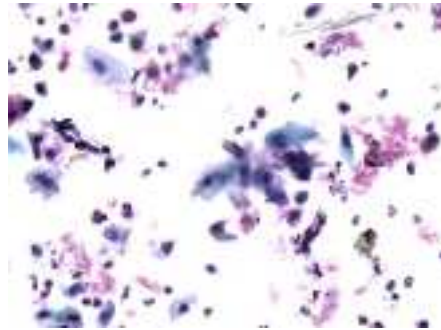
Den Effekt derartiger Farbstoffüberlagerungen zeigt Abb. 8 an einem Beispiel. Das Bild zeigt im oberen Teilbild die vollständig gefärbte Probe also Hämatoxylin als Kernfärbung (blau) und Eosin (purpur) als Zytoplasmafärbung. Zu beachten ist der Rotstich auch in den Kernbereichen. Im Bild unten links wurde nur mit Hämatoxylin gefärbt. Das Bild unten rechts zeigt eine nur mit Eosin gefärbte Probe. Zu beachten ist hier, dass keine von den Kernen herrührenden 'Löcher' sichtbar sind. Dies bedeutet, dass das Zytoplasma auch die Kerne mit gleicher Dicke überdeckt und damit im doppelt gefärbten Präparat helle Stellen der Kerne als Zytoplasma erscheinen lässt und somit die Kernsegmentation deutlich behindert.

## 6 Schlussfolgerungen

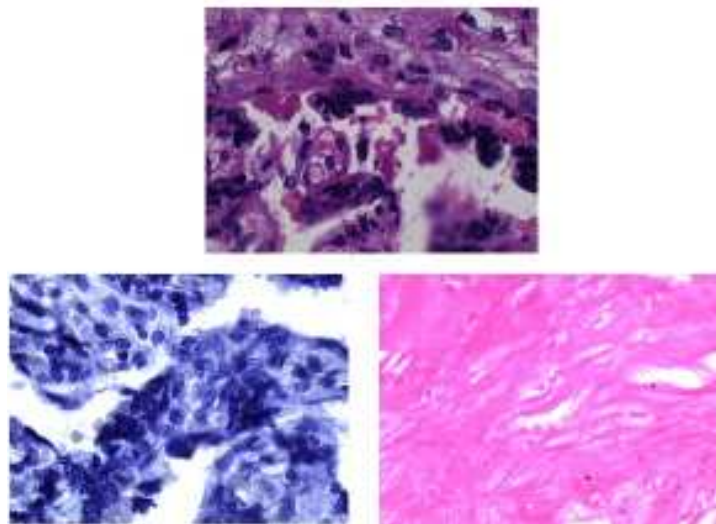
Die Anforderungen der Anwender von Bildauswertewerkzeugen zielen bei zytologischen und histologischen Präparaten zunehmend auf mehr semantische Aussagen. Es wird dabei erwartet, dass eingeführte Präparationsverfahren im Labor nicht verändert werden müssen, d.h. die bisher üblichen Färbungen weiter verwendet werden können. Die Nutzung realistischer Modelle für die Bildentstehung lassen eine Verbesserung der Auswertergebnisse zu. Der Beitrag stellt Modelle vor für die quantitative Auswertung feiner Strukturen, für Farbverschiebungen im RGB-Raum bei Konzentrationsänderungen von breitbandigen Farbstoffen und für die Verbesserung von Segmentationsverfahren für Zellen, Zellzytoplasma und Zellkern durch Verfahren zur Trennung der Farbstoffanteile.



**Abb. 6.** Mischungsdiagramm im RGB-Farbraum für 2 Farbstoffe



**Abb. 7.** Ausschnitt aus einem nach Papanicolaou gefärbten Ausstrichpräparat



**Abb. 8.** Unterschiedlich gefärbte Ausschnitte aus einem histologischen Schnitt; oben: Hämatoxylin und Eosin, links: nur Hämatoxylin (typische Kernfärbung); rechts: nur Eosin (typische Zytoplasmafärbung)