



Farbbasiertes Image Retrieval

Bachelorarbeit

vorgelegt von:

Thomas Kluth

Matrikelnummer: 356798

Studiengang: Bachelor Informatik

Thema gestellt von:

Prof. Dr. Achim Clausing

Münster, 15. Februar 2012

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit mit dem Titel „*Farbbasiertes Image Retrieval*“ selbständig verfasst habe, dass ich keine anderen Quellen und Hilfsmittel als die angegebenen benutzt habe und dass die Stellen der Arbeit, die anderen Werken – auch elektronischen Medien – dem Wortlaut oder Sinn nach entnommen wurden, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht worden sind.

Ort, Datum

Thomas Kluth

Inhaltsverzeichnis

1	Einleitung	5
2	Inhaltsbasierte Bildsuche	7
2.1	Evaluierung unterschiedlicher Methoden	8
3	Farbe	10
3.1	RGB-Farbraum	10
3.2	HSV-Farbraum	11
3.2.1	Umrechnung von RGB-Werten zu HSV-Werten	11
3.3	BCR-Farbrepräsentation	13
3.3.1	Approximation der Grundform	15
3.3.2	RGB9-Basisfarbmenge	15
3.3.3	DIN5023(RGB)-Basisfarbmenge	16
3.3.4	DIN5023(HSV)-Basisfarbmenge	19
3.4	Farbhistogramm	19
4	Ähnlichkeit	21
4.1	Minkowski-Distanzen	21
4.2	Histogramm-Schnitt	21
4.3	Distanz zwischen verschiedenen Bins	22
4.3.1	Kumulative Histogramme	22
4.3.2	Quadratische Distanz	23
4.4	Verteilungsmomente	23
5	Bildpartitionierung	25
5.1	Rechteckige Aufteilung	25
5.2	Fuzzy-Aufteilung	26
6	Vergleich	29
6.1	Evaluierung	29
6.2	Bildpartitionierung	30
6.3	Distanzen	33
6.4	Farbe	35
6.5	BCR Farbrepräsentationen	37
6.6	Implementierung	39
7	Fazit	42

Anhang	44
A Delaunay-Zerlegungen	44
B Klassendiagramm	47
Literaturverzeichnis	48

1 Einleitung

Die visuelle Wahrnehmung des Menschen ist ein sehr komplexer Prozess. Über die Hälfte des menschlichen Gehirns ist an der Analyse und der Interpretation visueller Reize beteiligt¹. Dies zeigt die wichtige Funktion des visuellen Systems für den Menschen. Ob bildende Kunst, Fotografie, Filme oder Visualisierungen komplexer Sachverhalte - für einen Menschen ist es in Sekundenschnelle möglich ein Bild zu analysieren und ihm eine Bedeutung zu geben. Nicht umsonst gibt es das Sprichwort „Ein Bild sagt mehr als tausend Worte“.

Das „passende“ Bild zu finden ist allerdings nicht immer einfach. Mit der immer schneller voranschreitenden Entwicklung des Internets sowie medizinischer Visualisierungstechniken wird es zunehmend wichtiger in großen Bilddatenbanken relevante Bilder in kurzer Zeit zu finden. Das *Content-Based Image Retrieval* versucht zu erforschen, wie diese Aufgabe möglichst effizient gelöst werden kann.

Dieses Forschungsgebiet ist zwar noch verhältnismäßig jung, es gibt aber bereits eine Vielzahl unterschiedlicher Forschungsansätze. Zunächst wird in Kapitel 2 das Themenfeld *Content-Based Image Retrieval* kurz zusammengefasst.

Ziel der vorliegenden Arbeit ist es, verschiedene Methoden des farbbasierten Image Retrievals zu vergleichen – d. h. als Merkmal zur Ermittlung der Ähnlichkeit von Bildern kommt lediglich *Farbe* zum Einsatz. Es wird sich zeigen, dass dieses Merkmal teils gute Ergebnisse, teils aber auch unbrauchbare Ergebnisse liefert. Da das Thema Farbe zentral ist, drehen sich die Kapitel 3 und 4 um diesen Themenkomplex: Im ersteren werden unterschiedliche Ansätze zur Beschreibung des Phänomens Farbe besprochen, im letzteren geht es um unterschiedliche Arten die Ähnlichkeit zwischen Farben zu bestimmen

Da das Merkmal Farbe sehr grundlegend ist, kann es leicht vorkommen, dass zwei semantisch völlig unterschiedliche Bilder eine sehr ähnliche Farbgebung besitzen. Um dieses Problem in den Griff zu bekommen, ist es gängige Praxis, die zu untersuchenden Bilder in kleinere Einheiten aufzuteilen. Zwei Methoden dazu werden im Kapitel 5 präsentiert.

Schließlich wird in Kapitel 6 dazu übergegangen, verschiedene Kombinationen der zuvor genannten Ansätze miteinander zu vergleichen, um deren Leistungsfähigkeit einschätzen zu können. Dies geschieht zum einen mit Evaluierungsmethoden der inhaltsbasierten Bildsuche (siehe Abschnitt 2.1), zum anderen werden die Ergebnisse einer Umfrage analysiert und als Maß für die Eignung verschiedener Methoden verwendet. Die Umfrageergebnisse werden den Ergebnissen der *CBIR*-Evaluierungsmethoden gegenübergestellt.

Im Rahmen der Arbeit wurde ein Programm entwickelt, das alle genannten Kon-

¹[Bellebaum u. a., 2012, S. 31]

zepte realisiert. Der Quelltext sowie das ausführbare Programm befindet sich mitsamt Dokumentation auf der beigelegten CD. Auf den Aufbau des Programms wird näher in Abschnitt 6.6 eingegangen.

2 Inhaltsbasierte Bildsuche

Das *Content-Based Image Retrieval* (inhaltsbasierte Bildsuche, kurz *CBIR*) entwickelte sich ungefähr ab 1994¹.

Allgemein bezeichnet *CBIR* den Prozess bestimmte Bilder aufgrund ihrer Merkmale (Farbe, Textur, Form, ...) aus einer größeren Menge von Bildern zu liefern. Dabei werden die Merkmale automatisiert aus den Bildern berechnet – manuelle Verschlagwortung von Bildern und darauffolgende Suche zählt also nicht zum *CBIR*.

Es gibt viele verschiedene Merkmale, die zur Suche im *CBIR* verwendet werden. Diese reichen von sehr elementaren Merkmalen wie der Farbe zu komplexeren Merkmalen wie z. B. der automatischen Erkennung verschiedener Objekte im Bild.

In [Smeulders u. a., 2000] werden zwei Lücken, mit denen sich das *CBIR* beschäftigt, genannt:

- Die *sensorische Lücke* (*sensory gap*) bezeichnet die Lücke zwischen einem Objekt in der realen Welt und der Information aus einer (rechnerischen) Beschreibung, die von einer Aufnahme dieser Szene stammt².
- Die *semantische Lücke* (*semantic gap*) bezeichnet das Fehlen der Koinzidenz zwischen der Information, die man aus den visuellen Daten extrahieren kann und der Interpretation derselben Daten durch einen Menschen in einer bestimmten Situation³.

Die *sensorische Lücke* betrifft den Informationsverlust, der bei der Aufnahme einer Szene entsteht. Besonders bei großen Bilddatenbanken mit wenig Informationen über die Aufnahmebedingungen ist diese Lücke daher relevant. Fragen wie

- Zeigen zwei zweidimensionale Bilder die gleiche dreidimensionale Szene und damit auch das selbe Objekt?
- Sind zwei Objekte zu sehen oder ist es nur ein Objekt, das durch ein anderes Objekt so verdeckt wird, dass es nicht mehr zusammenhängend erscheint?

können nicht problemlos beantwortet werden.

Die *semantische Lücke* ist ungleich wichtiger für das *CBIR*. Sie bezeichnet die Schwierigkeit bestimmten Bildern allein aufgrund bestimmter syntaktischer Merkmale Bedeutung zuzuweisen. Etwas, was ein menschlicher Betrachter innerhalb von Sekunden vermag, gestaltet sich als eine große Herausforderung für einen automatisierten Prozess.

¹[Datta u. a., 2008, S. 5]. Die Entwicklung der Zeit von 1994-2000 wird in [Smeulders u. a., 2000] zusammengefasst, die Entwicklung ab 2000 bis Mitte 2006 wird in [Datta u. a., 2008] dargestellt.

²[Smeulders u. a., 2000, S. 1352]

³[Smeulders u. a., 2000, S. 1353]

Die Formulierung einer Anfrage an ein *CBIR*-System lässt sich auf verschiedene Arten durchführen⁴:

- **query by example image (QBE):** Der Benutzer gibt dem System ein Beispielbild vor und das System liefert dem Beispielbild ähnliche Bilder zurück. Diese Art von Anfrage wurde in dem Programm, das im Laufe dieser Arbeit entwickelt wurde, implementiert.
- **query by multiple example images (QBME):** Der Benutzer wählt mehrere Beispielbilder aus, deren Gemeinsamkeiten als Basis für die Suche dienen. Außerdem können die verschiedenen Beispielbilder unterschiedlich gewichtet werden.
- **query by visual sketch:** Der Benutzer zeichnet die ihm wichtigen Teile des gesuchten Bildes auf und das System liefert zur Skizze ähnliche Bilder zurück. Die Formulierung der Anfrage hängt bei diesem Ansatz von der künstlerischen Fähigkeit des Benutzers ab.
- **query by keyword:** Der Benutzer gibt Schlagwörter an, die auf das gesuchte Bild zutreffen. Dies funktioniert allerdings nur, wenn die Bilder vorher mit Schlagwörtern versehen wurden.

Viele *CBIR*-Systeme arbeiten auch interaktiv, d. h. der Benutzer bewertet das Ergebnis und das System versucht Ergebnisse zu liefern, die den Präferenzen des Benutzers entsprechen (*relevance feedback*).

2.1 Evaluierung unterschiedlicher Methoden

Die Evaluierung verschiedener Ansätze des *Content-Based Image Retrieval* ist zwar ein wichtiges Thema, wenn es darum geht verschiedene Ansätze objektiv miteinander vergleichen zu können, dennoch sind die Bemühungen, Vergleiche unterschiedlicher Systeme zu standardisieren, bisher noch nicht sehr umfangreich. In [Datta u. a., 2008] werden zwei Projekte genannt, die das Ziel haben, standardisierte Evaluationsstrategien für die inhaltsbasierte Bildsuche zu schaffen^{5,6}. Außerdem wird [Müller u. a., 2001] als eine umfassende Übersicht erwähnt.

In [Müller u. a., 2001] werden objektive Evaluierungsstandards für *CBIR*-Systeme durch Adaption von Evaluationsmethoden des *Information Retrieval* – insbesondere des *Text Retrieval* – vorgeschlagen. Als Teil dieser Standards wird auch eine Standard-Datenbasis vorgeschlagen, da ein System auf unterschiedlichen Datenbasen unterschiedlich gute Ergebnisse liefert.

Als Maß der Leistungsfähigkeit eines *CBIR*-Systems wird in [Müller u. a., 2001] eine Zusammenstellung folgender Größen genannt⁷:

⁴[Mayron, 2008, S. 38 ff.]

⁵ImageCLEF, <http://ir.shef.ac.uk/imageclef/> (15. Februar 2012)

⁶Benchathlon, <http://www.benchathlon.net> (15. Februar 2012)

⁷[Müller u. a., 2001, S. 7]

- $Rank_1$: Der Rang, an dem das erste relevante Bild im Ergebnis auftaucht
- Ein normalisierter, durchschnittlicher Rang: Sei N die Gesamtanzahl der Bilder, die als Datenbasis dienen; N_R die Anzahl an relevanten Bildern zu einer gegebenen Anfrage und R_i der Rang des i -ten relevanten Bildes, dann ist folgendes Maß gleich 0 für eine perfekte Rückgabe und tendiert gegen 1 für eine komplett falsche Rückgabe auf eine gegebene Antwort:

$$\widetilde{Rank} = \frac{1}{NN_R} \left(\sum_{i=1}^{N_R} R_i - \frac{N_R(N_R - 1)}{2} \right) \quad (2.1)$$

- $P(20)$, $P(50)$ und $P(N_R)$: *precision* nachdem 20, 50 und N_R Bilder abgefragt wurden
- $R_P(0,5)$ und $R(100)$: *recall* bei *precision* 0,5 und nachdem 100 Bilder abgefragt wurden
- *precision-recall*-Graph

Die in dieser Liste genannten Maße *precision* und *recall* sind wie folgt definiert⁸:

$$precision = \frac{\text{Anzahl gelieferter relevanter Dokumente}}{\text{Gesamtanzahl gelieferter Dokumente}} \quad (2.2)$$

$$recall = \frac{\text{Anzahl gelieferter relevanter Dokumente}}{\text{Gesamtanzahl relevanter Dokumente der Sammlung}} \quad (2.3)$$

In einem *precision-recall*-Graph (PR-Graph) wird *recall* gegen *precision* aufgetragen.

⁸[Müller u. a., 2001, S. 2]

3 Farbe

Da es in dieser Arbeit um den Vergleich *farbbasierter* Ansätze in der Bildsuche geht, behandelt dieses Kapitel das Thema *Farbe*. Es werden im Folgenden zwei unterschiedliche Farbräume sowie eine alternative Farbrepräsentation, die zur Nutzung in der inhaltsbasierten Bildsuche entwickelt wurde, vorgestellt. Darüber hinaus wird das Farbhistogramm erläutert.

3.1 RGB-Farbraum

Der RGB-Farbraum ist ein additiver Farbraum, d. h. eine Farbe im RGB-Farbraum setzt sich additiv aus drei Komponenten zusammen: R für Rot, G für Grün und B für Blau.

In Abbildung 3.1 ist der RGB-Farbraum als Würfel dargestellt. Die Ecken des Würfels tragen die Namen der entsprechenden Farben.

Der RGB-Farbraum ist vor allem bei digitalen Bildern weit verbreitet. Das liegt daran, dass technische Geräte zur Farbwiedergabe (z. B. Monitore) nach dem RGB-Prinzip arbeiten. Dies ist zum einen ein Vorteil des RGB-Farbraums, da standardmäßig die Farben digitaler Bilddaten in RGB kodiert sind und somit zur Anzeige auf einem Monitor nicht umgerechnet werden müssen. Zum anderen ist dies aber auch ein Nachteil des RGB-Farbraums, da sich die technische Realisierung von Farbe nicht an der menschlichen Wahrnehmung orientiert. Zur inhaltsbasierten Bildsuche ist der RGB-Farbraum deshalb nicht geeignet, da die für den Menschen ähnlichen Farben im RGB-Farbwürfel nicht zwingend nah beieinander liegen.

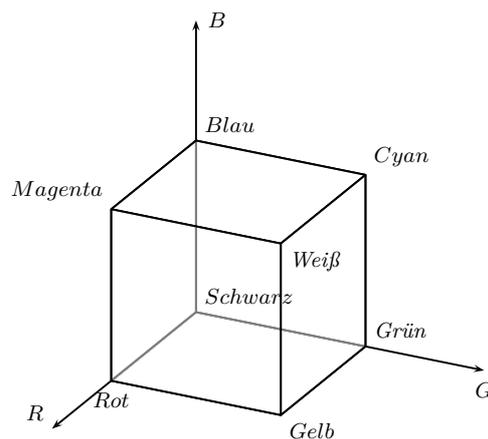


Abbildung 3.1: Der RGB-Farbraum als Würfel

3.2 HSV-Farbraum

Der HSV-Farbraum (auch HSB-Farbraum) wurde primär entwickelt, um Computerbenutzern eine intuitivere Möglichkeit zu geben Farben auszuwählen. So ist heutzutage in fast jedem Computerprogramm, das eine Farbauswahl ermöglicht (z. B. in Bildbearbeitungsprogrammen oder HTML-Editoren), ein auf dem HSV-Farbraum basierender Farbwähler zu finden. In Abbildung 3.2 ist beispielhaft der Farbwähler der GTK+-GUI-Bibliothek¹ abgebildet.

Im HSV-Farbraum wird eine Farbe ebenfalls durch drei Werte spezifiziert: H steht für den Farbton (*engl. hue*), S steht für die Sättigung (*engl. saturation*) und V (bzw. B) steht für den Helligkeitswert (*engl. value, brightness*). Im abgebildeten Farbwähler lässt sich der Farbtonwert anhand des Kreises einstellen. Der Sättigungs- und Helligkeitswert lässt sich innerhalb des Kreises bestimmen.

Es ist üblich, diesen Farbraum durch einen Kegel darzustellen, obgleich der zugrundeliegende Farbraum eigentlich einen Zylinder beschreibt (s. Abbildung 3.3). Der Farbtonwert (H) wird als Winkel um die Rotationsachse des Kegels angegeben, der Sättigungswert (S) liegt zwischen 0 an der Rotationsachse und 1 am Rand des Kegels, der Helligkeitswert (V, B) liegt zwischen 0 am Boden des Kegels und 1 an der Spitze des Kegels.

Der HSV-Farbraum hat den entscheidenden Vorteil, dass er sich mit wenig Aufwand in den RGB-Farbraum umrechnen lässt (siehe Gleichungen 3.1 bis 3.3). Außerdem basiert er auf den Grundlagen der menschlichen Farbwahrnehmung, soweit dies in der Einfachheit des Farbraumes möglich ist. Farben im HSV-Farbraum liegen also dann nahe beieinander, wenn ein Mensch diese Farben auch als ähnlich betrachtet. Deshalb ist der HSV-Farbraum im Kontext der inhaltsbasierten Bildsuche eine beliebte Alternative zum RGB-Farbraum.

3.2.1 Umrechnung von RGB-Werten zu HSV-Werten

Die Umrechnung von RGB-Werten zu HSV-Werten erfolgt durch folgende Gleichungen², wobei $max = \max(R, G, B)$ und $min = \min(R, G, B)$ gilt:

$$hue = \begin{cases} 0 & \text{wenn } max = min \Rightarrow R = G = B \\ \frac{g-b}{max-min} \cdot 60 & \text{wenn } r = max \wedge (g-b) \geq 0 \\ \frac{g-b}{max-min} \cdot 60 + 360 & \text{wenn } r = max \wedge (g-b) < 0 \\ (2 + \frac{b-r}{max-min}) \cdot 60 & \text{wenn } g = max \\ (4 + \frac{r-g}{max-min}) \cdot 60 & \text{wenn } b = max \end{cases} \quad (3.1)$$

$$saturation = \begin{cases} 0 & \text{wenn } max = 0 \\ \frac{max-min}{max} & \text{sonst} \end{cases} \quad (3.2)$$

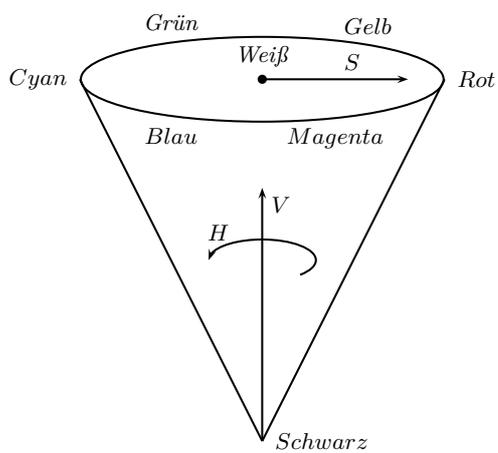
$$value = max \quad (3.3)$$

¹<http://www.gtk.org/> (15. Februar 2012)

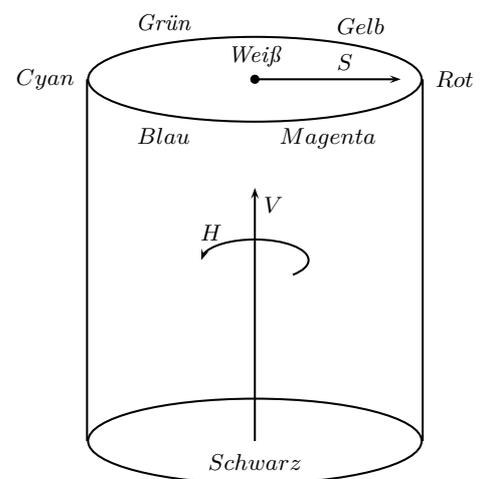
²vgl. [Leicht, 2007, S. 17]



Abbildung 3.2: Farbwähler



(a) HSV-Kegel



(b) HSV-Zylinder

Abbildung 3.3: Der HSV-Farbraum als (a) Kegel und (b) Zylinder

3.3 BCR-Farbrepräsentation

In seiner Dissertation³ entwickelt Volmer eine Farbrepräsentation, die auf der Grundlage menschlicher Farbwahrnehmung basiert:

„Der Mensch beurteilt die Welt der Farben anhand ihm bekannter prototypischer Muster. Das menschliche Wahrnehmungsvermögen unterteilt den Farbraum in sich teilweise überlappende Farbkategorien, deren Grenzen nur unscharf definiert sind.“⁴

Weiter zitiert Volmer eine Studie, die zeigt,

„dass dem Mensch[en] – unabhängig von einer bestimmten Sprache – maximal elf grundsätzlich verschiedene Begriffe zur Bezeichnung von Farben zur Verfügung stehen: ‚Schwarz‘, ‚Weiß‘, ‚Rot‘, ‚Grün‘, ‚Gelb‘, ‚Blau‘, ‚Braun‘, ‚Orange‘, ‚Pink‘, ‚Violett‘ und ‚Grau‘.“⁵

Deshalb entwickelt Volmer im Rahmen seiner Dissertation die sogenannte BCR-Farbrepräsentation (*engl. base color ratio*). In dieser Farbrepräsentation wird eine Menge von Basisfarben definiert. Relativ zu diesen Basisfarben werden alle zu repräsentierenden Farben als Mischung der Basisfarben dargestellt. Diese Darstellung von Farben ähnelt dem Mischen von (neuen) Farben aus den Grundfarben eines Wasserfarbkastens.

Definition 3.1 (BCR-Farbrepräsentation) *Gegeben sei eine geordnete, endliche Menge*

$$C = \{c_1, c_2, \dots, c_K\} \subset \mathbb{R}^3 \quad (3.4)$$

von $K \in \mathbb{N}$ Basisfarben in allgemeiner Lage, die in einem trichromatischen Bezugssystem definiert sind. Jede beliebige Farbe c lässt sich als eine Linearkombination dieser Basisfarben gemäß

$$c = \chi_1 c_1 + \chi_2 c_2 + \dots + \chi_K c_K \quad (3.5)$$

mit $\chi_K \in \mathbb{R}$ und der Normierung

$$\chi_1 + \chi_2 + \dots + \chi_K = 1 \quad (3.6)$$

darstellen. Das geordnete K -Tupel

$$\chi(c, C) = (\chi_1, \chi_2, \dots, \chi_K) \quad (3.7)$$

wird eine BCR-Farbrepräsentation der Farbe c bezüglich der Basisfarbmenge C genannt.⁶

³vgl. [Volmer, 2006, S. 61 - 88]

⁴[Volmer, 2006, S. 61]. Volmer verweist auf: E. Rosch. Natural Categories. *Cognitive Psychology*, 4(3):328–350, 1974.

⁵[Volmer, 2006, S. 62]. Die Studie: B. Berlin and P. Kay. Basic Colour Terms: Their Universality and Evolution. University of California Press, 1969.

⁶[Volmer, 2006, S. 63]

Eine BCR-Farbrepräsentation $\chi(\mathbf{c}, C)$ stellt die baryzentrischen Koordinaten der Farbe \mathbf{c} dar (d. h. relativ zu den Koordinaten der Basisfarben). Dabei steht jedes χ_K für den Anteil, den man von der Basisfarbe \mathbf{c}_K in der Mischung braucht, um die Farbe \mathbf{c} zu erzeugen.

Die durch 3.1 definierte BCR-Farbrepräsentation hat den Nachteil, dass sie für eine Farbe \mathbf{c} mehrere mögliche BCR-Farbrepräsentationen zulässt (unter anderem auch deshalb, weil sie negative Werte für χ_k zulässt). Um zu einer eindeutigen Darstellung zu gelangen, formuliert Volmer zwei weitere Kriterien:

- „eine beliebige Farbe sollte als eine Linearkombination möglichst weniger Basisfarben dargestellt werden“⁷
- „eine beliebige Farbe sollte als eine Linearkombination ausnahmslos nicht-negativer Anteile von Basisfarben dargestellt werden“⁷

Diese Kriterien werden dadurch erfüllt, dass jede beliebige Farbe durch „genau vier nicht-koplanare Basisfarben $\mathbf{c}_i, \mathbf{c}_j, \mathbf{c}_k$ und \mathbf{c}_l “⁸ beschrieben wird. Dies entspricht der Definition eines dreidimensionalen Simplex (d. h. eines unregelmäßigen Tetraeders). Wenn eine Farbe innerhalb der konvexen Hülle dieses Simplex liegt, sind auch – wie gefordert – χ_i, χ_j, χ_k und χ_l nicht-negativ.

Leider gibt es bei der Auswahl der vier Basisfarben, die zur Beschreibung eines Simplex dienen, in welchem die zu beschreibende Farbe liegt, wieder mehrere Möglichkeiten. Volmer diskutiert deshalb die Aufteilung des Farbraums in eine Menge von Simplexen durch die sogenannte Delaunay-Zerlegung ([Volmer, 2006, S. 68-70]). Auf dieses Konzept soll hier nicht weiter eingegangen werden. Im Anhang sind die Delaunay-Zerlegungen für die in 3.3.2 und 3.3.3 besprochenen Basisfarbmengen aufgeführt.

Mittels der Delaunay-Zerlegung definiert Volmer schließlich die BCR-Grundform:

Definition 3.2 (BCR-Grundform) Die BCR-Farbrepräsentation $\chi(\mathbf{c}, C)$ einer beliebigen Farbe

$$\mathbf{c} \in \text{conv}(C) \tag{3.8}$$

innerhalb der konvexen Hülle der Basisfarbmengens C ist in Grundform, falls die folgenden Eigenschaften erfüllt sind:

(i) Alle Koeffizienten von $\chi(\mathbf{c}, C)$ sind nicht-negativ.

$$\chi_i \geq 0, \forall i \in \{1, 2, \dots, K\} \tag{3.9}$$

(ii) Es existiert ein Simplex

$$S = (\mathbf{c}_i, \mathbf{c}_j, \mathbf{c}_k, \mathbf{c}_l) \in \text{DT}(C) \tag{3.10}$$

mit

$$\mathbf{c} \in \text{conv}(S), \tag{3.11}$$

⁷[Volmer, 2006, S. 66]

⁸[Volmer, 2006, S. 67]

so dass sich (3.5) zu

$$\mathbf{c} = \chi_i \mathbf{c}_i + \chi_j \mathbf{c}_j + \chi_k \mathbf{c}_k + \chi_l \mathbf{c}_l \quad (3.12)$$

reduziert.⁹

Anmerkung: $\text{conv}(C)$ bezeichnet die konvexe Hülle der Basisfarbmenge C . $\text{DT}(C)$ bezeichnet die Delaunay-Zerlegung der Basisfarbmenge C .

3.3.1 Approximation der Grundform

Je nachdem welche Basisfarbmenge man wählt, kann es sein, dass die konvexe Hülle der Basisfarbmenge nicht den gesamten Bezugsfarbraum überdeckt. Um jedoch auch einer Farbe \mathbf{c} außerhalb dieser konvexen Hülle eine BCR-Grundform zuordnen zu können, wird eine zu \mathbf{c} ähnliche Farbe \mathbf{c}' , die in der konvexen Hülle der Basisfarbmenge liegt, approximiert. Dazu wird der Schnittpunkt der Geraden zwischen der Farbe \mathbf{c} und der neutralen Farbe \mathbf{c}_N und der Oberfläche der konvexen Hülle der Basisfarbmenge gewählt:

Definition 3.3 Gegeben sei eine geordnete, endliche Menge $C = \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_K\} \subset \mathbb{R}^3$ von $K \in \mathbb{N}$ Basisfarben in allgemeiner Lage sowie die im Inneren von $\text{conv}(C)$ liegende neutrale Farbe \mathbf{c}_N . Für alle Farben $\mathbf{c} \in \mathbb{R}^3 \setminus \text{conv}(C)$ sei \mathbf{c}' der \mathbf{c} am nächsten liegende Schnittpunkt der Geraden durch \mathbf{c} und \mathbf{c}_N mit der konvexen Hülle $\text{conv}(C)$. Für \mathbf{c}' existiert ein Simplex

$$S = (\mathbf{c}_i, \mathbf{c}_j, \mathbf{c}_k) \in \text{conv}(C) \quad (3.13)$$

mit $\mathbf{c}' \in S$, so dass sich die Linearkombination

$$\mathbf{c} \approx \mathbf{c}' = \sum_{i=1}^K \chi_i \mathbf{c}_i \quad (3.14)$$

auf

$$\mathbf{c} \approx \mathbf{c}' = \chi_i \mathbf{c}_i + \chi_j \mathbf{c}_j + \chi_k \mathbf{c}_k \quad (3.15)$$

reduziert. Dann ist $\chi(\mathbf{c}', C)$ die Approximation der BCR-Grundform von \mathbf{c} bezüglich der Basisfarbmenge C .¹⁰

3.3.2 RGB9-Basisfarbmenge

In Tabelle 3.1 sind die im Folgenden RGB9-Basisfarben genannten neun Farben aufgeführt.

Die als RGB9-Basisfarben bezeichneten Farben decken den kompletten RGB-Farbraum mit ihrer konvexen Hülle ab. Dies ist vorteilhaft, da dadurch alle Farben eine eindeutige BCR-Farbrepräsentation besitzen und man keinerlei Informationsverlust durch die Approximation (s. 3.3.1) riskiert. Der Grund für die Überdeckung des gesamten Farbraums ist einfach: Die neun Basisfarben der RGB9-Basisfarbmenge entsprechen den

⁹[Volmer, 2006, S. 71]

¹⁰[Volmer, 2006, S. 76]

Tabelle 3.1: Normalisierte Werte der RGB9-Basisfarben.¹¹

Farbname	R	G	B
Neutralgrau	0.5	0.5	0.5
RGB Schwarz	0.0	0.0	0.0
RGB Rot	1.0	0.0	0.0
RGB Gelb	1.0	1.0	0.0
RGB Grün	0.0	1.0	0.0
RGB Cyan	0.0	1.0	1.0
RGB Blau	0.0	0.0	1.0
RGB Magenta	1.0	0.0	1.0
RGB Weiß	1.0	1.0	1.0

Ecken des RGB-Würfels (s. 3.1). Desweiteren wurde die Farbe Neutralgrau (der Mittelpunkt des RGB-Würfels) zur Basisfarbmenge hinzugenommen.

Zwar ist die Delaunay-Zerlegung des RGB-Farbraums mit den RGB9-Basisfarben nicht eindeutig, sie kann jedoch durch eine Nebenbedingung festgelegt werden: „Aus intuitiver Sicht ist es sinnvoll, dass jeder Delaunay-Tetraeder genau zwei ‚unbunte‘ und zwei ‚bunte‘ Farben als Ecken hat, d. h. neben ‚Neutralgrau‘ muss zusätzlich entweder ‚RGB Schwarz‘ oder ‚RGB Weiß‘ eine Ecke des Delaunay-Tetraeders sein.“¹² Im Anhang (Tabelle A.1, Seite 44) ist diese Zerlegung aufgeführt; Abbildung 3.4 zeigt beispielhaft zwei der insgesamt zwölf Tetraeder.

Die RGB9-Basisfarbmenge ist jedoch im Kontext der inhaltsbasierten Bildsuche nicht die beste Wahl:

„Bilddatenbanken mit uneingeschränkter Domäne bestehen in der Regel vorwiegend aus Abbildern realer Szenen mit fotografischem Ursprung. Statistisch gesehen enthalten solche Bilder vorwiegend ungesättigte Farben, die in der weiteren Umgebung der unbunten Grauwert-Diagonale des RGB-Würfels liegen. [...] Die bunten Basisfarben (R, Y, G, C, B, M) machen statistisch gesehen lediglich 23% des Anteils an der Grundgesamtheit aller Farben aus. Anders ausgedrückt: Aus der Sicht einer CBIR-Anwendung ist die Wahl der RGB-Eckfarben als Basisfarben daher wenig sinnvoll.“¹³

3.3.3 DIN5023(RGB)-Basisfarbmenge

Aus dem Grund, dass sich die RGB9-Basisfarben nur bedingt für die inhaltsbasierte Bildsuche eignen, stellt Volmer eine weitere – geeignete – Basisfarbmenge vor, die er

¹¹vgl. [Volmer, 2006, S. 78]

¹²[Volmer, 2006, S. 78]

¹³[Volmer, 2006, S. 79f.]

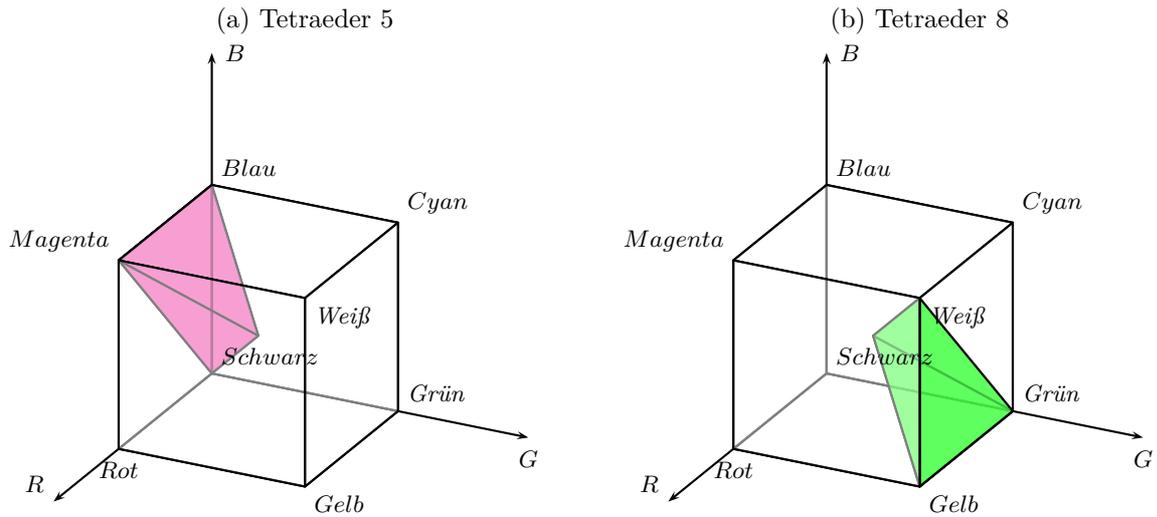


Abbildung 3.4: Zwei Tetraeder der Delaunay-Zerlegung des RGB-Würfels mit den RGB9-Basisfarben (s. Tabelle A.1, Seite 44)

DIN5023-Basisfarbmenge nennt¹⁴. Diese Basisfarbmenge basiert – wie der Name schon sagt – auf der DIN-Norm 5023 von 1989. Diese Norm legt fest, welche Farben sich in einem Wassermal-Farbkasten befinden müssen. Tabelle 3.2 zeigt die DIN5023-Basisfarben als normalisierte RGB-Werte.

Da die konvexe Hülle der DIN5023(RGB)-Basisfarbmenge den Einheitswürfel nicht komplett überdeckt, ist es notwendig die BCR-Grundform von Farben, die außerhalb der konvexen Hülle liegen, mittels des unter 3.3.1 beschriebenen Verfahrens zu approximieren. Dazu muss die neutrale Farbe (Neutralgrau) in der konvexen Hülle der Basisfarbmenge liegen. Wenn man sie als Basisfarbe hinzufügt, erreicht man dies am einfachsten.

¹⁴Da in dieser Arbeit noch eine weitere auf den gleichen Farben basierende Basisfarbmenge vorgestellt wird, wird die von Volmer beschriebene Basisfarbmenge im Folgenden als DIN5023(RGB)-Basisfarbmenge bezeichnet.

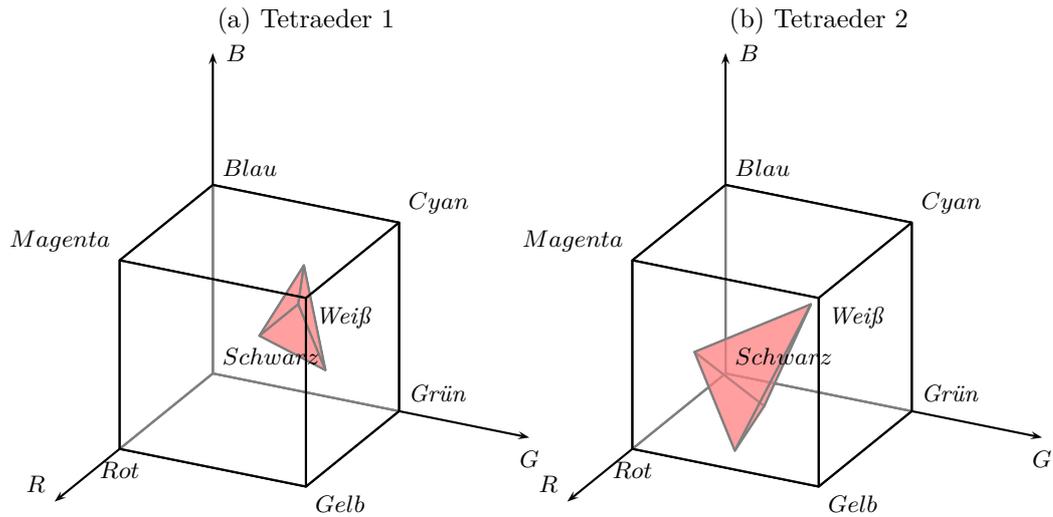


Abbildung 3.5: Zwei Tetraeder der Delaunay-Zerlegung des RGB-Würfels mit den DIN5023-Basisfarben (s. Tabelle A.2, Seite 45)

Tabelle 3.2: Normalisierte RGB-Werte der DIN5023(RGB)-Basisfarbmenge.¹⁵

Farbname	Abkürzung	R	G	B
Neutralgrau	N	0,5000	0,5000	0,5000
Gelb	Y _L	0,9932	0,8476	0,0000
Orange	O _R	0,9971	0,5483	0,0990
Zinnoberrot	C _R	0,8176	0,2679	0,2023
Magentarot	M _R	0,7617	0,2141	0,4615
Violett	V _T	0,5131	0,3083	0,6176
Ultramarinblau	U _B	0,1610	0,2887	0,7420
Cyanblau	C _B	0,0000	0,4889	0,6712
Blaugrün	B _G	0,0000	0,5556	0,4302
Gelbgrün	Y _G	0,4221	0,8157	0,3496
Ockergelb	O _C	0,7963	0,6059	0,2668
Gebrannte Sienna	B _S	0,6383	0,3394	0,2234
Schwarz	B _K	0,2154	0,2150	0,2150
Weiß	W _T	0,9205	0,9189	0,9190

Die Delaunay-Zerlegung der DIN5023(RGB)-Basisfarbmenge findet sich im Anhang (Tabelle A.2, S. 45). In Abbildung 3.5 werden beispielhaft zwei der insgesamt 33 Tetraeder dargestellt.

¹⁵[Volmer, 2006, S. 83]

Tabelle 3.3: Normalisierte HSV-Werte der DIN5023(HSV)-Basisfarbmenge

Farbname	Abkürzung	H	S	V
Neutralgrau	N	0,0000	0,0000	0,5000
Gelb	Y _L	51,2042	1,0000	0,9932
Orange	O _R	30,0167	0,9007	0,9971
Zinnoberrot	C _R	6,397	0,7527	0,8176
Magentarot	M _R	332,893	0,7189	0,7617
Violett	V _T	279,728	0,5008	0,6176
Ultramarinblau	U _B	226,8124	0,7830	0,7420
Cyanblau	C _B	196,2961	1,0000	0,6712
Blaugrün	B _G	166,4579	1,0000	0,5556
Gelbgrün	Y _G	110,6672	0,5714	0,8157
Ockergelb	O _C	38,4249	0,6650	0,7963
Gebrannte Sienna	B _S	16,7751	0,6500	0,6383
Schwarz	B _K	0,0000	0,0019	0,2154
Weiß	W _T	356,25	0,0017	0,9205

3.3.4 DIN5023(HSV)-Basisfarbmenge

Da die BCR-Farbrepräsentation kein eigenständiger Farbraum ist, sondern auf einem trichromatischen Farbraum basiert, ist es auch möglich die BCR-Farbrepräsentation nicht auf dem RGB-Farbraum, sondern auf dem HSV-Farbraum basieren zu lassen. Dies ist als Überlegung sinnvoll, da der HSV-Farbraum nach der menschlichen Wahrnehmung konstruiert wurde und deshalb besser als der RGB-Farbraum für die inhaltsbasierte Bildsuche geeignet sein sollte.

Als Wahl der Basisfarbmenge soll wiederum die Farbmenge der DIN-Norm 5023 gewählt werden. Die HSV-Werte der Farben wurden – ausgehend von den RGB-Werten der DIN5023(RGB)-Basisfarbmenge – mit der in Abschnitt 3.1 beschriebenen Umrechnung bestimmt und sind in Tabelle 3.3 zu finden.

3.4 Farbhistogramm

Durch ein Farbhistogramm wird die Häufigkeit verschiedener Farbwerte eines Bildes ausgedrückt. Dazu wird der Farbraum in n Farbklassen aufgeteilt (quantisiert) und jedes Bildpixel einem dieser n Farbklassen (einem sogenannten Histogramm-Bin) zugeordnet.

Für den RGB-Farbraum ist es üblich die Farbwerte in 256 Farbklassen zu unterteilen, dadurch lässt sich eine RGB-Farbe mit einem Byte pro Farbkanal speichern. Da dies allerdings immer noch einen recht hohen Speicheraufwand bedeutet, wurde der RGB-Farbraum für diese Arbeit in 15 Farbklassen pro Farbkanal eingeteilt.

Der HSV-Farbraum wurde für diese Arbeit nach [Smith u. Chang, 1996a] quantisiert: 18 Farbklassen im *hue*-Farbkanal, jeweils 3 Farbklassen in den *saturation*- und *value*-Farbkanälen.

Mathematisch ist ein Farbhistogrammbin wie folgt definiert:

Definition 3.4 (Farbhistogrammbin) Gegeben sei ein Bild der Breite N und der Höhe M sowie der Farbwert $c_{x,y}$ an der Stelle (x, y) . Dann ist der Farbhistogrammbin der Farbklasse c im Farbkanal i :

$$h_i(c) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f(x, y, c) \quad \text{mit } f(x, y, c) = \begin{cases} 1 & \text{wenn } c_{x,y} = c \\ 0 & \text{sonst.} \end{cases} \quad (3.16)$$

Das Farbhistogramm $H_i = (h_i(1), h_i(2), \dots, h_i(n))$ des Farbkanals i besteht dann aus n Farbhistogrammbins; ein komplettes Farbhistogramm besteht demnach insgesamt aus $i \cdot n$ Farbhistogrammbins.

Die BCR-Farbrepräsentation (s. Abschnitt 3.3) ist ein Versuch den Farbraum in Farbklassen zu unterteilen, die der menschlichen Farbwahrnehmung nahekommen. Ein BCR-Farbhistogramm besteht also nur aus einem Farbkanal und die Anzahl der Farbklassen entspricht der Anzahl der Basisfarben.

Desweiteren ändert sich die Definition eines Farbhistogrammbins für Farbhistogramme, die auf der BCR-Farbrepräsentation basieren, geringfügig:

Definition 3.5 (BCR-Farbhistogrammbin) Gegeben sei ein Bild der Breite N und der Höhe M sowie die BCR-Farbrepräsentation $\chi(\mathbf{c}_{x,y}, C) = (\chi_1, \chi_2, \dots, \chi_K)$ der Farbe $\mathbf{c}_{x,y}$ an der Stelle (x, y) in BCR-Grundform zur Basisfarbmenge $C = \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_K\}$. Dann ist der BCR-Farbhistogrammbin der Farbklasse k :

$$h(k) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \chi_k \quad (3.17)$$

4 Ähnlichkeit

In diesem Kapitel werden unterschiedliche Distanzfunktionen vorgestellt. Die meisten dieser Funktionen arbeiten dabei auf dem in Abschnitt 3.4 erläuterten Farbhistogramm. Gegen Ende des Kapitels wird noch eine Distanzfunktion gezeigt, die nicht auf einem Histogramm operiert.

4.1 Minkowski-Distanzen

Die sogenannte Minkowski-Distanz ist wie folgt definiert:

$$d_p(x, y) = (|x - y|^p)^{\frac{1}{p}} \quad (4.1)$$

Für $p = 1$ ergibt sich daraus die Manhattan-Distanz (auch Cityblock-, Taxi- oder L_1 -Distanz):

$$d_1(x, y) = |x - y| \quad (4.2)$$

Für $p = 2$ ergibt sich die bekannte Euklidische Distanz (auch L_2 genannt):

$$d_2(x, y) = \sqrt{|x - y|^2} \quad (4.3)$$

Auf Histogramme angewendet lässt sich folgende Distanzfunktion zweier Histogramme H und I mit jeweils B Bins und k Farbkanälen formulieren:

$$\delta_p(H, I) = \sum_{j=0}^k \sum_{b=0}^B d_p(h_j(b), i_j(b)) \quad (4.4)$$

4.2 Histogramm-Schnitt

In [Swain u. Ballard, 1991] wurde folgende Distanzfunktion für zwei Histogramme H und I mit jeweils B Bins vorgeschlagen:

$$\delta_{\cap}(H_i, I_i) = \frac{\sum_{b=0}^B \min(h(b), i(b))}{|H|} \quad (4.5)$$

Der Ausdruck im Nenner dient dabei der Normierung. Falls beide Histogramme bereits normiert wurden, ist der Histogramm-Schnitt gleich der Manhattan-Distanz.

Volmer schlägt für Histogramme, die auf der BCR-Farbpräsentation basieren, diese Distanzfunktion vor¹.

¹[Volmer, 2006, S. 102]

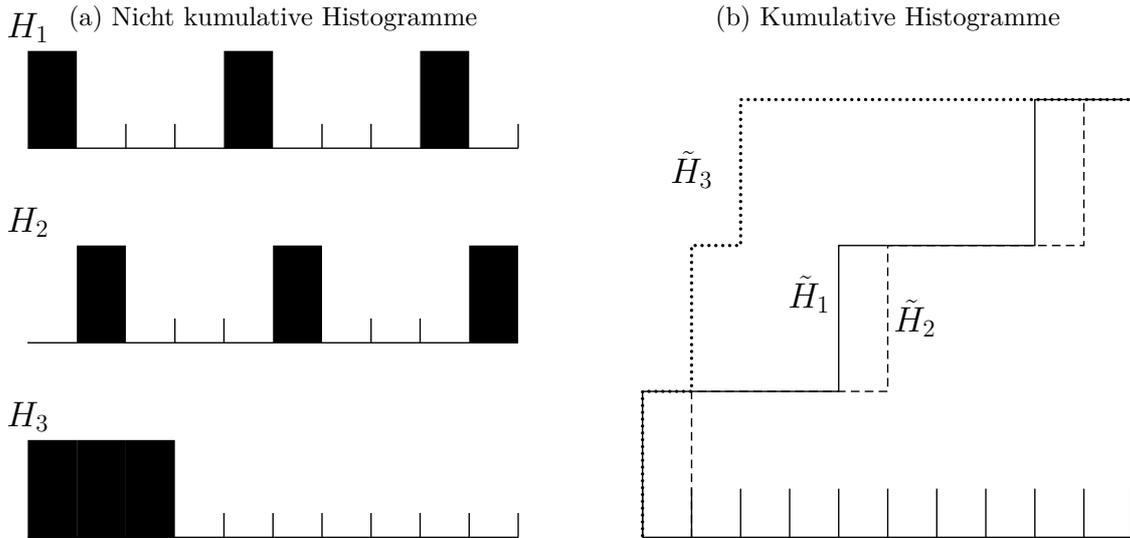


Abbildung 4.1: Vergleich von herkömmlichen mit kumulativen Histogrammen²

4.3 Distanz zwischen verschiedenen Bins

Alle bisher erwähnten Distanzfunktionen haben den Nachteil, dass nur Bins der gleichen Farbe miteinander verglichen werden, aber keine benachbarten Bins. Das führt dazu, dass ähnliche – nicht exakt gleiche – Farben bei dem Ähnlichkeitsvergleich nicht berücksichtigt werden.

4.3.1 Kumulative Histogramme

Insbesondere kann dies dazu führen, dass die Histogramme zweier Bilder, die farblich ähnlich sind, eine große Distanz haben – drei einfache Histogramme zu dieser Problematik sind in Abbildung 4.1a dargestellt. Obwohl die Histogramme H_1 und H_2 auf den ersten Blick relativ ähnlich zueinander wirken und das Histogramm H_3 aus der Reihe fällt, sind die Abstände mit den bisher genannten Distanzfunktionen genau anders verteilt³:

$$\delta_1(H_1, H_2) = 6 > \delta_1(H_1, H_3) = \delta_1(H_2, H_3) = 4$$

Die Idee der kumulativen Histogramme, die in [Stricker u. Orengo, 1995] formuliert wurde, löst dieses Problem, indem die Histogrammbins von links nach rechts aufaddiert werden⁴:

$$\tilde{h}_{c_j} = \sum_{c_l < c_j} h_{c_l} \quad (4.6)$$

²[Stricker u. Orengo, 1995, S. 3, S. 6]

³Ein Bin in der Grafik sei eine Einheit hoch. Als Beispiel ist hier nur die L_1 -Distanz gezeigt, die anderen genannten Distanzen verhalten sich aber genauso.

⁴[Stricker u. Orengo, 1995, S. 5]

Abbildung 4.1b veranschaulicht diese Idee am Beispiel der bereits betrachteten Beispielhistogramme.

4.3.2 Quadratische Distanz

Eine weitere Möglichkeit benachbarte Bins (d. h. ähnliche Farben) in den Vergleich mit einzubeziehen wurde in [Niblack u. a., 1993] für das QBIC-Projekt vorgeschlagen und genutzt – die sogenannte *Quadratic-distance*:

$$\delta_q(H_i, I_i) = (h - i)^T A (h - i) \quad (4.7)$$

Hierbei sind h und i Vektoren, die alle Einträge der Histogramme H und I auflisten. A ist eine Matrix, die – je nach verwendetem Farbraum – genauer spezifiziert, welche der benachbarten Bins miteinander zu vergleichen sind.

Für den RGB-Farbraum wurde in [Niblack u. a., 1993] folgende Matrix vorgeschlagen, wobei mit $d_2(i, j)$ die oben beschriebene Euklidische Distanz gemeint ist:

$$a_{i,j} = 1 - \frac{d_2(i, j)}{\max(d_2(i, j))} \quad (4.8)$$

Für den HSV-Farbraum kann man folgende Matrix verwenden [Smith u. Chang, 1996b]:

$$a_{i,j} = 1 - \frac{1}{\sqrt{5}} \left[(v_i - v_j)^2 + (s_i \cos(h_i) - s_j \cos(h_j))^2 + (s_i \sin(h_i) - s_j \sin(h_j))^2 \right]^{\frac{1}{2}} \quad (4.9)$$

4.4 Verteilungsmomente

In [Stricker u. Orengo, 1995] wird vorgeschlagen anstatt jeweils ein komplettes Histogramm zu berechnen nur die ersten drei Verteilungsmomente der Farbverteilung des Bildes zum Vergleich zu nutzen. Die ersten drei zentralen Momente einer Wahrscheinlichkeitsverteilung sind der Mittelwert, die Varianz und die Schiefe. Aus Gründen der Vergleichbarkeit, wird die zweite bzw. dritte Wurzel der Varianz bzw. Schiefe gespeichert.

Sei der Farbwert des Pixels j im i -ten Farbkanal p_{ij} und es gebe insgesamt N Pixel. Dann sind die drei zentralen Momente⁵:

$$E_i = \frac{1}{N} \sum_{j=1}^N p_{ij} \quad (4.10)$$

$$\sigma_i = \left(\frac{1}{N} \sum_{j=1}^N (p_{ij} - E_i)^2 \right)^{\frac{1}{2}} \quad (4.11)$$

$$s_i = \left(\frac{1}{N} \sum_{j=1}^N (p_{ij} - E_i)^3 \right)^{\frac{1}{3}} \quad (4.12)$$

⁵[Stricker u. Orengo, 1995, S. 7]

Tabelle 4.1: Mögliche Werte der Gewichtsparameter w_{kl} .⁶

	E	σ	s
H	3	3	1
S	2	1	1
V	1	1	1

Die Distanz zweier Farbbilder mit den Farbverteilungen H und I und r Farbkanälen sowie den zentralen Momenten E_i bzw. F_i , σ_i bzw. ς_i und s_i bzw. t_i ist dann wie folgt definiert⁷:

$$\delta_{mom}(H, I) = \sum_{i=1}^r w_{i1}|E_i - F_i| + w_{i2}|\sigma_i - \varsigma_i| + w_{i3}|s_i - t_i| \quad (4.13)$$

Die Werte $w_{kl} \geq 0$ ($1 \leq l, k \leq 3$) sind dabei vom Benutzer zu spezifizierende Parameter. [Stricker u. Dimai, 1996] verwenden für den HSV-Farbraum die in Tabelle 4.1 angegebenen Werte.

⁶[Stricker u. Dimai, 1996, S. 7]

⁷[Stricker u. Orengo, 1995, S. 7]

5 Bildpartitionierung

Die Idee, ein gegebenes Bild in Bereiche aufzuteilen, um auf diese Weise Informationen über die räumliche Lage von Farbverteilungen zu gewinnen, ist schon vielfach angewandt worden¹. In dieser Arbeit werden zwei verschiedene Ausprägungen dieser Idee miteinander verglichen. Der erste Ansatz teilt ein Bild in mehrere Rechtecke auf, der zweite Ansatz teilt ein Bild in eine Ellipse und vier rechteckige Regionen um diese Ellipse herum auf. Darüber hinaus ist die Aufteilung nach dem zweiten Ansatz nicht „scharf“, sondern „unscharf“ mittels Fuzzy-Funktionen festgelegt.

5.1 Rechteckige Aufteilung

Bei diesem Ansatz wird jedes Bild in zwei übereinanderliegende rechteckige Raster aufgeteilt. Dabei teilt das untere Raster das komplette Bild in gleichmäßig große Rechtecke auf, das obere Raster wird dann „versetzt“ über das untere Raster gelegt, d.h. die Eckpunkte der oberen Rechtecke liegen jeweils in der Mitte der unteren Rechtecke. Die Größe dieser Raster kann dabei vom Benutzer festgelegt werden, standardmäßig ist das untere Raster 2×2 groß – und damit das obere Raster 1×1 . In Abbildung 5.1 wird dieses Prinzip gezeigt.

Von jedem dieser so erzeugten Regionen des Bildes wird nun ein Histogramm erstellt. Diese lokalen Histogramme werden dann für den Vergleich mit anderen Bildern genutzt. Dabei können verschiedene Distanzmaße verwendet werden. Im beiliegenden Programm sind alle Distanzen, die in Kapitel 4 besprochen werden, implementiert.

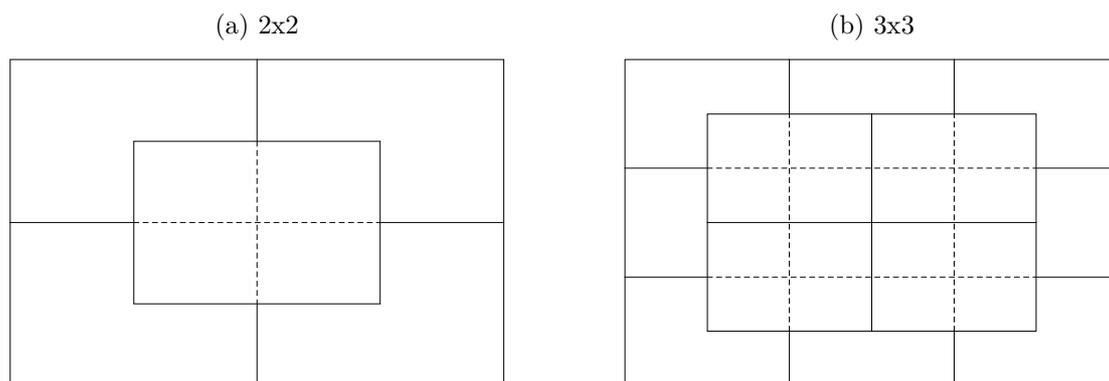


Abbildung 5.1: Rechteckige Aufteilung

¹[Smeulders u. a., 2000, S. 1356f.]

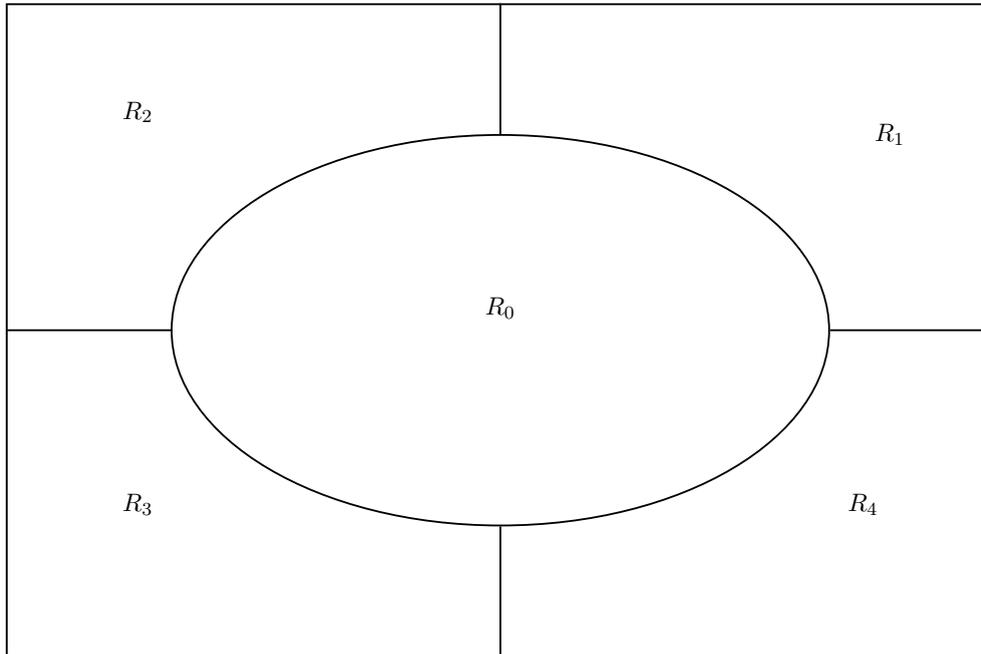


Abbildung 5.2: Aufteilung nach [Stricker u. Dimai, 1997]

5.2 Fuzzy-Aufteilung

Eine etwas komplexere Art und Weise der Aufteilung eines Bildes in verschiedene Regionen beschreiben [Stricker u. Dimai, 1996] (leicht verändert nochmals in [Stricker u. Dimai, 1997]). Ein wesentlicher Unterschied zur Methode der rechteckigen lokalen Histogramme besteht darin, dass sie Fuzzy-Regionen verwenden – also unscharfe Grenzen zwischen den einzelnen Regionen. Dafür definieren sie fünf Regionen und entsprechende Fuzzy-Zugehörigkeitsfunktionen. Abbildung 5.2 zeigt die Einteilung des Bildes in die entsprechenden Kernregionen.

Unter der Annahme, dass sich der wichtigste Teil eines Bildes in der Mitte befindet, wird zunächst eine Ellipse $E_{a,b}$ mit den beiden Halbachsen a und b definiert. Diese Ellipse kennzeichnet den Kernbereich der Region R_0 . Die Fuzzy-Zugehörigkeitsfunktion zur Region R_0 für ein Pixel $P = (p_x, p_y)$ wird folgendermaßen definiert²:

$$R_0(P) = \begin{cases} 1 & \text{wenn } P \text{ innerhalb } E_{a,b} \\ \frac{1}{2}(\cos(d_{\text{ellipse}}(P, E_{a,b})\frac{\pi}{r}) + 1) & \text{wenn } P \text{ außerhalb } E_{a,b} \end{cases} \quad (5.1)$$

Dabei bezeichnet $d_{\text{ellipse}}(P, E_{a,b})$ eine Distanzfunktion, die den Abstand eines Punktes P zu einer Ellipse $E_{a,b}$ angibt. In [Stricker u. Dimai, 1997] wird dafür die Funktion, die in [Stricker, 1994] beschrieben wird, genutzt. Da es aber nicht unbedingt auf die Genauigkeit der Distanz ankommt, lassen sich theoretisch auch andere Distanzfunktionen

²[Stricker u. Dimai, 1996, S. 3]

dort verwenden. Der Parameter r bezeichnet die Breite der Fuzzy-Regionen, d.h den Bereich wo ein Pixel einen Zugehörigkeitswert zwischen 1 und 0 hat.

Die vier weiteren Regionen, die sich rings um die Ellipse verteilen, werden beispielhaft an der Region R_1 (oben rechts) definiert, für die anderen Regionen ergibt sich die Definition entsprechend. Es wird angenommen, dass sich der Koordinatenursprung links unten befindet und das Bild die Breite m und die Höhe n hat. Für die Region werden nun folgende beide Linien definiert: $l_1 : x = \frac{1}{2}(m+r)$ und $l_2 : y = \frac{1}{2}(n+r)$. Die Zugehörigkeit eines Pixels P zur Region R_1 wird wie folgt definiert³:

$$R_1(P) = \begin{cases} 0 & \text{wenn } p_x \leq \frac{1}{2}(m-r) \vee p_y \leq \frac{1}{2}(n-r) \\ 1 - R_0(P) & \text{wenn } p_x \geq \frac{1}{2}(m+r) \wedge p_y \geq \frac{1}{2}(n+r) \\ \frac{1}{2}(\cos(d_{surround}(P, l_1, l_2) \frac{\pi}{r}) \cdot (1 - R_0(P))) & \text{sonst} \end{cases} \quad (5.2)$$

Dabei beschreibt $d_{surround}(P, l_1, l_2) = \min(d(P, l_1), d(P, l_2))$ den Abstand eines Punktes zur Region, indem der kleinste Abstand zu einer der beiden definierten Linien betrachtet wird.

Als Merkmale einer Region wird nicht das lokale Histogramm einer Region gewählt, sondern die ersten drei zentralen Momente der jeweiligen Farbverteilung. Diese Methode wurde in [Stricker u. Orengo, 1995] als robuster und effizienter als die übliche Histogrammbasierte Methode vorgestellt. Bei der Aufteilung des Bildes handelt es sich jedoch um Fuzzy-Regionen, weshalb die Formeln für die ersten drei Momente ausgehend von [Stricker u. Orengo, 1995] (s. Abschnitt 4.4 auf Seite 23) wie folgt modifiziert worden sind⁴:

$$E_{l,i} = \frac{1}{NR_l^{tot}} \sum_P R_l(P) C_i(P) \quad (5.3)$$

$$\sigma_{l,i} = \left(\frac{1}{NR_l^{tot}} \sum_P R_l(P) (C_i(P) - E_{l,i})^2 \right)^{\frac{1}{2}} \quad (5.4)$$

$$s_{l,i} = \left(\frac{1}{NR_l^{tot}} \sum_P R_l(P) (C_i(P) - E_{l,i})^3 \right)^{\frac{1}{3}} \quad (5.5)$$

Dabei bezeichnet $C_i(P)$ den Farbwert des Pixels P im Farbkanal i , l die Region, N die Anzahl an Pixeln im gesamten Bild und $R_l^{tot} = \sum_{j=1}^N R_l(j)$. $E_{l,i}$ bezeichnet die gewichtete Durchschnittsfarbe, $\sigma_{l,i}$ bezeichnet die gewichtete Varianz und $s_{l,i}$ bezeichnet die gewichtete Schiefe der jeweiligen Region l und den jeweiligen Farbkanal $1 \leq i \leq 3$. Gewichtet werden diese Werte aufgrund der Fuzzy-Aufteilung des Bildes. Dadurch werden Pixel, die im Randbereich der jeweiligen Region liegen, nicht so stark berücksichtigt, wie Pixel, die in der Mitte einer Region liegen.

Die in [Stricker u. Dimai, 1996] definierte Distanzfunktion führt verschiedene Gewichte ein. Zunächst gibt es Gewichte w_{km} mit $1 \leq k, m \leq 3$, die für die Distanzberechnung

³[Stricker u. Dimai, 1997, S. 68]

⁴[Stricker u. Dimai, 1996, S. 5]

einzelner Regionen verwendet werden. [Stricker u. Dimai, 1996] haben dafür die in Tabelle 4.1 auf Seite 24 aufgeführten Werte verwendet. Seien H und I zwei Bilder, $E_{l_1,i}$, $\sigma_{l_1,i}$, $s_{l_1,i}$ die oben beschriebenen Momente für H und $F_{l_2,i}$, $\varsigma_{l_2,i}$, $t_{l_2,i}$ die oben beschriebenen Momente für I . Die Distanz zwischen der Region l_1 von H und der Region l_2 von I wird dann wie folgt definiert⁵:

$$d_{l_1,l_2}(H, I) = \sum_{i=1}^3 w_{i1}|E_{l_1,i} - F_{l_2,i}| + w_{i2}|\sigma_{l_1,i} - \varsigma_{l_2,i}| + w_{i3}|s_{l_1,i} - t_{l_2,i}| \quad (5.6)$$

Die gesamte Distanzfunktion⁵ $d(H, I)$ berücksichtigt durch die Funktion f zusätzlich Rotationen des Bildes um ein Vielfaches von 90° . Außerdem lässt sich mit den Gewichten S_i ($0 \leq i \leq 4$) der Beitrag der einzelnen Regionen zum Gesamtabstand beeinflussen:

$$d(H, I) = S_0 R_0^{tot} d_{0,0}(H, I) + \min_{f \in T_{90}} \sum_{l_1=l_2=1}^4 S_{l_1} R_{l_1}^{tot} d_{l_1,f(l_2)}(H, I) \quad (5.7)$$

⁵[Stricker u. Dimai, 1996, S. 6]

6 Vergleich

In diesem Kapitel geht es um die Dokumentation der Anwendung unterschiedlicher Kombinationen aller bisher beschriebenen Ansätze, um damit eine Aussage über die Fähigkeiten der unterschiedlichen Verfahren machen zu können.

Es wurden durchweg Anfragen im Sinne des *query-by-example*-Paradigmas (s. Kapitel 2) an das Programm gestellt. Die Eingabe des Programms ist demnach ein Beispiel-Bild zu dem man ähnliche Bilder präsentiert bekommen möchte; die Ausgabe des Programms sind die 17 zum Anfragebild ähnlichsten Bilder – je nach verwendetem Verfahren variieren diese natürlich. Insgesamt stehen 59.895 Bilder zur Verfügung. Die Bilder wurden freundlicherweise von Prof. James Z. Wang, Pennsylvania State University, USA, zur Verfügung gestellt. Seine Arbeitsgruppe hat diese knapp 60.000 Bilder in den folgenden drei Publikationen: [Wang u. a., 2001], [Li u. Wang, 2003], [Li u. Wang, 2008] verwendet.

In jedem der nun folgenden Vergleiche werden die gleichen vier (zufällig ausgewählten) Bilder als Anfragebild verwendet¹, um die ermittelten Messgrößen sinnvoll miteinander vergleichen zu können.

6.1 Evaluierung

Die Bewertung der einzelnen Testanfragen erfolgt durch die Berechnung der im Abschnitt 2.1 genannten Messgrößen. Diese Größen verlangen jedoch alle die Definition von zum Anfragebild *relevanten* Bildern. Allerdings ist es keine leichte Aufgabe festzulegen, welche Bilder in einer Datenbank *relevant* sind – je nachdem, wie man *Relevanz* definiert, sind mal mehr, mal weniger Bilder *relevant*.

Wenn man beispielsweise als Anfragebild ein Bild von Elefanten hat, stellt sich die Frage, ob nur Bilder auf denen ebenfalls ein Elefant abgebildet ist, alle anderen Bilder auf denen ein Tier abgebildet ist oder gar alle Naturbilder relevant sind.

Ich habe mich für die Berechnung der Evaluierungsgrößen darauf beschränkt alle Bilder innerhalb einer Gruppe als relevant zu betrachten – wobei eine Gruppe aus 100 Bildern² besteht. Da 100 Bilder im Verhältnis zur großen Gesamtanzahl der Bilder jedoch sehr wenig sind, werden in der Regel alle Messgrößen 0 – bis auf \widetilde{Rank} und $Rank_1$. Doch selbst diese Werte sind mit Vorsicht zu genießen, da ich nicht glaube, alle relevanten Bilder mit der Beschränkung auf diese eine Gruppe abgedeckt zu haben. Jede andere Methode relevante Bilder zu gruppieren ist jedoch aufgrund der großen Datenmenge nicht sinnvoll.

¹24145.jpg, 35817.jpg, 33639.jpg, 45396.jpg

²bis auf drei Ausnahmen, wo ein bis drei Bilder fehlen

Um dennoch eine aussagekräftige Bewertung der unterschiedlichen Verfahren zu bekommen, habe ich eine Umfrage erstellt, bei denen die unten beschriebenen Rückgaben unterschiedlicher Methoden (ausgehend vom gleichen Ausgangsbild) ihrer Qualität nach in eine Reihenfolge gebracht werden sollten. Da ich mit vier Anfragebildern gearbeitet habe, gibt es pro Kategorie jeweils vier Fragen bzw. Antwortreihenfolgen.

Es gibt einen wichtigen Unterschied zwischen den berechneten Messgrößen und den von den Umfrageteilnehmern bewerteten Ergebnissen: Die bei der Umfrage bewerteten Rückgaben bestehen aus den 17 ähnlichsten Bildern (zu einem Anfragebild). Die Messgrößen hingegen beziehen sich auf die nach Ähnlichkeit sortierte Gesamtreihenfolge aller 60.000 Bilder (zu einem Anfragebild), die das Programm zurückgeliefert hat. Dies ist auch nicht anders möglich, da bei fast allen Abfragen das erste *relevante* Bild nicht unter den ersten 17 zurückgelieferten Bildern war – wie man an der Spalte $Rank_1$ in den Ergebnistabellen sehen kann – und somit die Berechnung für die ersten 17 Bilder immer 0 ergibt.

Die Auswertung der Umfrage verlief folgendermaßen: Um Systematiken bei der Platzvergabe (immer die gleiche Antwortreihenfolge 1-2-3-4-5 oder 5-4-3-2-1) vorzubeugen, wurden alle Datenreihen manuell durchgesehen und fragwürdige Antworten nicht in die Auswertung mit einbezogen. Für die verschiedenen Plätze wurden in 2er-Schritten Punkte vergeben: Für den letzten Platz gab es 0 Punkte, für den vorletzten Platz 2 Punkte, usw. Je mehr Punkte eine Methode bekam, desto bessere Ergebnisse lieferte sie. Die Ergebnisse der Umfrage werden in den folgenden Abschnitten jeweils durch ein Diagramm dargestellt, welches im Text interpretiert wird. Die Diagramme zeigen die Gesamtpunktzahl (d. h. über jeweils vier Antwortreihenfolgen summiert), die die verschiedenen Methoden bekommen haben. Insgesamt gab es 42 vollständig ausgefüllte Umfragen, bei einigen Fragen wurden jedoch aufgrund von systematischen Antworten nur bis zu 37 Rückmeldungen ausgewertet.

6.2 Bildpartitionierung

In einem ersten Vergleich werden die beiden Partitionierungsmethoden erprobt. Als Farbraum kommt HSV zum Einsatz, als Distanz werden die *Verteilungsmomente* (wie von [Stricker u. Dimai \[1996\]](#) für ihre Methode vorgeschlagen) und L_2 eingesetzt. Desweiteren wird die Verwendung kumulativer Histogramme mit der L_2 -Distanz untersucht.

In Tabelle [6.1](#) sind die Werte der Evaluationsgrößen für die nicht-kumulativen Histogramme, in Tabelle [6.2](#) die Werte für die kumulativen Histogramme zu finden – für jede Anfrage gesondert und zusätzlich als Durchschnitt. In [Abbildung 6.1](#) sind die entsprechenden Umfrageergebnisse zu sehen.

Bei lediglicher Betrachtung der Durchschnittswerte von \widetilde{Rank} fällt auf, dass sie alle recht nahe beieinander liegen, es scheint also keine großen Qualitätsunterschiede zu geben. Stellt man dennoch eine Rangfolge der Methoden auf, so würde diese wie folgt aussehen:

Tabelle 6.1: Evaluationswerte Bildpartitionierung, nicht-kumulative Histogramme

Bild	Rechtecke				Fuzzy			
	L2		Momente		L2		Momente	
	$Rank_1$	\widetilde{Rank}	$Rank_1$	\widetilde{Rank}	$Rank_1$	\widetilde{Rank}	$Rank_1$	\widetilde{Rank}
24145.jpg	609	0,309	59	0,371	374	0,235	776	0,375
35817.jpg	597	0,354	37	0,230	166	0,367	193	0,322
33639.jpg	1	0,257	4	0,223	2	0,216	111	0,461
45396.jpg	324	0,306	1478	0,474	1.207	0,269	58	0,378
Durchschnitt	382,8	0,307	394,5	0,325	437,3	0,272	284,5	0,384

1. Fuzzy, nicht-kumulativ, Distanz: L2 (Durchschnitt \widetilde{Rank} : 0,272)
2. Rechtecke, nicht-kumulativ, Distanz: L2 (Durchschnitt \widetilde{Rank} : 0,307)
3. Rechtecke, Distanz: Momente (Durchschnitt \widetilde{Rank} : 0,325)
4. Fuzzy, kumulativ, Distanz: L2 (Durchschnitt \widetilde{Rank} : 0,361)
5. Fuzzy, Distanz: Momente (Durchschnitt \widetilde{Rank} : 0,384)
6. Rechtecke, kumulativ, Distanz: L2 (Durchschnitt \widetilde{Rank} : 0,399)

Die Durchschnittswerte von $Rank_1$ liegen – bis auf den Wert der momentbasierten, unscharfen Partitionierungsmethode – alle ebenfalls in diesem Trend.

Interessanterweise schneidet das Verfahren von [Stricker u. Dimai, 1996] fast am schlechtesten ab. Wenn aber anstatt der geforderten Vergleichsmethode *Momente* die L2-Distanz gewählt wird, liefert diese Art der Aufteilung das beste Ergebnis.

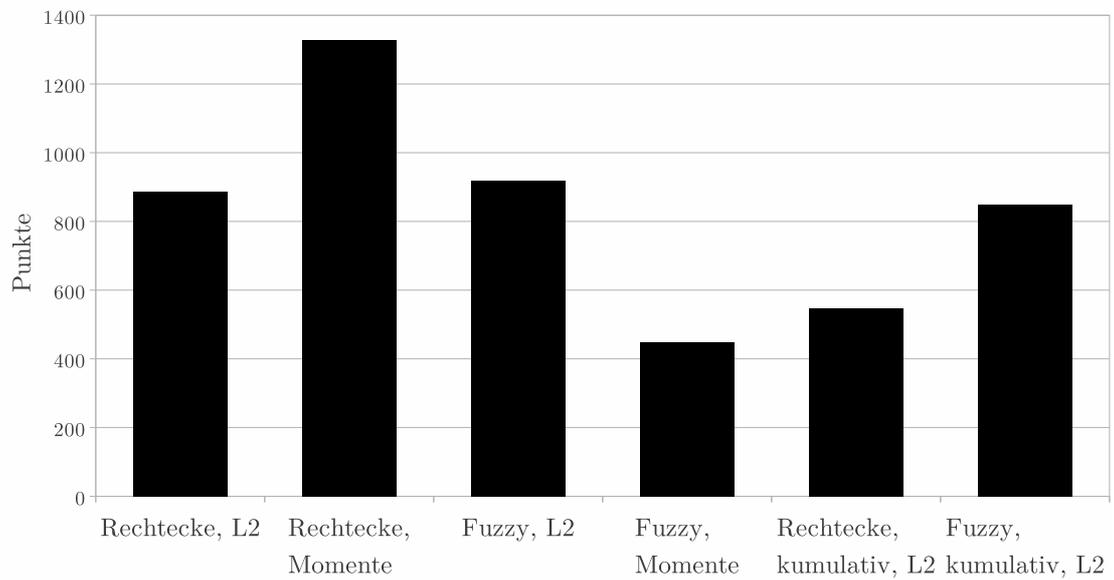
Die Einschätzung der Umfrageteilnehmer fällt geringfügig anders aus. Als beste Methode wurde die rechteckige Partitionierung mit der Distanz *Momente* gewählt. Den zweiten Platz mit ähnlichen Punktwerten teilen sich die Fuzzy- und rechteckige Partitionierung, jeweils mit nicht-kumulativen Histogrammen sowie die Fuzzy-Partitionierung mit kumulativen Histogramme und jeweils der Distanz *L2*. Die Fuzzy-Methode mit *Verteilungsmomenten* als Distanzfunktion schneidet auch bei der Umfrage schlecht ab.

Im Vergleich zwischen den Umfrageergebnissen und den Messgrößen ändert sich also vor allem die Bewertung der rechteckigen Partition mit *Verteilungsmomenten* als Distanz. Die restliche Methodeneinschätzung ist ähnlich.

Tabelle 6.2: Evaluationswerte Bildpartitionierung, kumulative Histogramme

Bild	Rechtecke		Fuzzy	
	$Rank_1$	\widetilde{Rank}	$Rank_1$	\widetilde{Rank}
24145.jpg	927	0,458	338	0,453
35817.jpg	91	0,344	42	0,319
33639.jpg	108	0,295	29	0,294
45396.jpg	5.548	0,497	3.512	0,379
Durchschnitt	1.668,5	0,399	980,3	0,361

Abbildung 6.1: Umfrageergebnisse Bildpartitionierung



6.3 Distanzen

Im folgenden Vergleich werden die unterschiedlichen Distanzen gegenübergestellt. Als Farbraum wird HSV gewählt, die Bilder werden rechteckig partitioniert. In Tabelle 6.3 werden die Messgrößen aufgeführt. Abbildung 6.2 zeigt die Umfrageergebnisse.

Die Durchschnittswerte für \widetilde{Rank} liegen bei diesem Vergleich noch näher beieinander, als beim ersten Vergleich. So lassen sich keine signifikanten Qualitätsunterschiede feststellen. Dennoch sei hier die Rangfolge der verschiedenen Distanzen genannt:

1. L2 (Durchschnitt \widetilde{Rank} : 0,307)
2. L1 (Durchschnitt \widetilde{Rank} : 0,312)
3. Momente (Durchschnitt \widetilde{Rank} : 0,325)
4. Minimum (Durchschnitt \widetilde{Rank} : 0,331)
5. Quadric (Durchschnitt \widetilde{Rank} : 0,332)

Die Durchschnittswerte von $Rank_1$ zeigen eine zu dieser Rangfolge ähnliche Tendenz. Entgegen der Erwartung, dass die *Quadric*-Distanz sinnvollere Ergebnisse als die beiden *Minkowski*-Distanzen $L1$ und $L2$ liefert, da sie zusätzlich ähnliche Farben im Histogramm miteinander vergleicht, landet sie auf dem letzten Platz.

Die Umfrageteilnehmer stufte das *Quadric*-Verfahren ebenfalls schlecht ein, für noch schlechter wurde allerdings die *Minimum*-Distanz empfunden. Als bestes Distanzverfahren rangiert das Vergleichen der *Verteilungsmomente* – dies ist eine Bestätigung der im vorigen Vergleich gewonnenen Resultate. Das Verhältnis der *momentbasierten*- zur *L2-basierten* Methode (jeweils rechteckig partitioniert), ist ebenfalls bei beiden Vergleichen ähnlich³:

$$\begin{aligned} \text{Partitionen: } & 886 \div 1328 = 0,667 \\ \text{Distanzen: } & 732 \div 976 = 0,75 \end{aligned}$$

Die beiden *Minkowski*-Distanzen werden von den Umfrageteilnehmern sehr ähnlich bewertet.

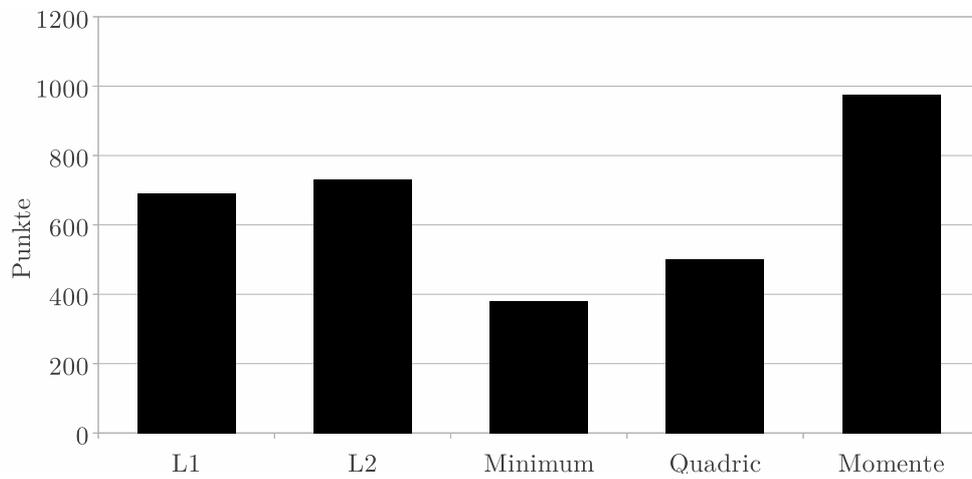
Insgesamt zeigt sich bei dem Vergleich der Distanzen wiederum eine leichte Neuverteilung der Bewertung gegenüber der reinen Betrachtung der Evaluationsgrößen: Die *momentbasierte* Vergleichsmethode wird über den *Minkowski*-Distanzen eingeordnet.

³siehe Abbildungen 6.1 und 6.2

Tabelle 6.3: Evaluationswerte Distanzen

Bild	L1		L2		Minimum		Quadric		Momente	
	$Rank_1$	\widetilde{Rank}								
24145.jpg	495	0,291	609	0,309	117	0,367	634	0,269	59	0,371
35817.jpg	328	0,376	597	0,354	300	0,283	974	0,372	37	0,230
33639.jpg	1	0,265	1	0,257	37	0,346	6	0,301	4	0,223
45396.jpg	174	0,315	324	0,306	4.409	0,326	785	0,387	1478	0,474
Durchschnitt	249,5	0,312	382,8	0,307	1.215,8	0,331	599,8	0,332	394,5	0,325

Abbildung 6.2: Umfrageergebnisse Distanzen



6.4 Farbe

In diesem Abschnitt werden die verschiedenen Farbräume (RGB und HSV) sowie die alternative Farbrepräsentation BCR miteinander verglichen. Als Distanz kommt dabei $L2$ zum Einsatz und die Bilder werden abermals in Rechtecke partitioniert. Tabelle 6.4 gibt einen Überblick über die Messgrößen, Abbildung 6.3 visualisiert die entsprechenden Ergebnisse der Umfrage.

Die Durchschnittswerte von \widetilde{Rank} liegen bei dieser Messreihe zwar etwas weiter auseinander als bei der vorherigen, jedoch immer noch näher beieinander als bei der ersten Reihe. Die nun folgende Rangfolge ist also wiederum mit Vorsicht zu genießen:

1. BCR-RGB9 (Durchschnitt \widetilde{Rank} : 0,298)
2. BCR-RGB-DIN (Durchschnitt \widetilde{Rank} : 0,304)
3. HSV (Durchschnitt \widetilde{Rank} : 0,307)
4. BCR-HSV-DIN (Durchschnitt \widetilde{Rank} : 0,336)
5. RGB (Durchschnitt \widetilde{Rank} : 0,365)

Der RGB-Farbmodus liegt hier verhältnismäßig deutlich hinten. Der BCR-HSV-DIN-Modus belegt auch recht eindeutig den vorletzten Platz, liegen die Werte von \widetilde{Rank} bei den ersten drei Plätzen doch deutlich näher beieinander.

Besonders im Fall des BCR-HSV-DIN-Modus ist das schlechte Abschneiden verwunderlich. Falls die Überlegungen, die im Kapitel 3.3 getätigt wurden, Hand und Fuß haben, kommt der BCR-HSV-DIN-Modus der menschlichen Farbwahrnehmung am nächsten.

Dass der reine RGB-Farbraum den letzten Platz belegt, scheint hingegen plausibel – schließlich ist der RGB-Farbraum nicht nach der menschlichen Farbwahrnehmung konstruiert.

Betrachtet man die Durchschnittswerte von $Rank_1$ belegen die beiden DIN-Basisfarben sogar die letzten beiden Plätze.

Die Ergebnisse der Umfrage bestätigen dann doch eher die zu Anfang gemachten Überlegungen. So bekommt der BCR-HSV-DIN-Modus zweifelsfrei den ersten Platz zugewiesen, die anderen beiden BCR-Modi folgen mit ähnlicher Punktzahl auf dem zweiten Platz. Anscheinend funktioniert der BCR-Ansatz besser als die klassische Farbrepräsentation.

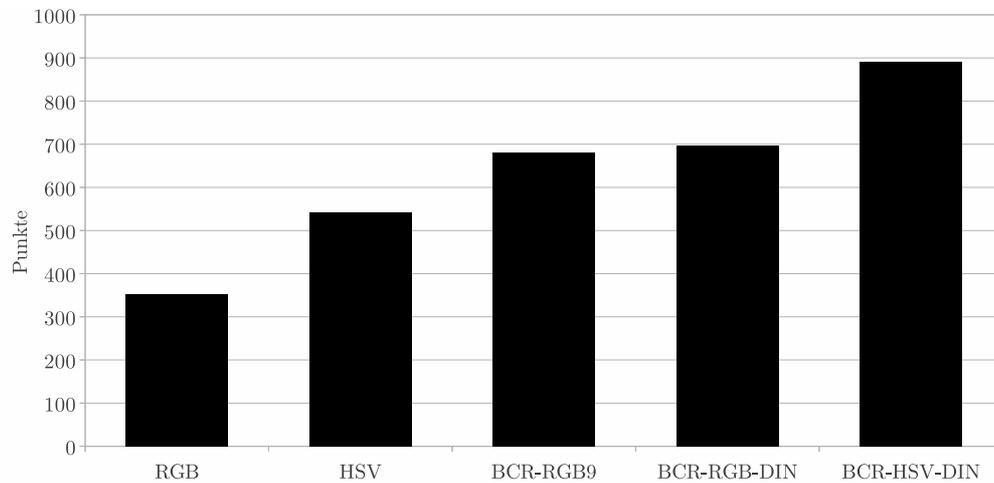
Der RGB-Farbraum kommt auch bei der Umfrage schlecht weg, es zeigt sich also auch hier, dass dieser Farbraum ungeeignet im Bereich der inhaltsbasierten Bildsuche ist. Deutlich besser schneidet der HSV-Raum ab, dennoch nicht so gut wie die BCR-Farbrepräsentationen.

Hier zeigt sich ebenfalls eine Umverteilung der Bewertung der einzelnen Methoden, wenn man sich die Rangfolge der berechneten Größen gegenüber den Umfrageergebnissen anschaut. Waren es bei den ersten Vergleichen noch Methoden die bei Betrachtung von \widetilde{Rank} im „Mittelfeld“ lagen und von Menschen auf den ersten Platz gewählt wurden, so hat es hier sogar eine Methode vom vorletzten Platz auf den ersten Platz geschafft.

Tabelle 6.4: Evaluationswerte Farbe

Bild	RGB		HSV		BCR-RGB9		BCR-RGB-DIN		BCR-HSV-DIN	
	$Rank_1$	\widetilde{Rank}	$Rank_1$	\widetilde{Rank}	$Rank_1$	\widetilde{Rank}	$Rank_1$	\widetilde{Rank}	$Rank_1$	\widetilde{Rank}
24145.jpg	299	0,239	609	0,309	64	0,215	23	0,270	33	0,291
35817.jpg	804	0,403	597	0,354	35	0,233	165	0,236	372	0,280
33639.jpg	48	0,358	1	0,257	207	0,297	15	0,245	1	0,187
45396.jpg	4150	0,460	324	0,306	6904	0,445	12.460	0,465	11.078	0,585
Durchschnitt	1.325,3	0,365	382,8	0,307	1.802,5	0,298	3.165,8	0,304	2.871	0,336

Abbildung 6.3: Umfrageergebnisse Farbe



6.5 BCR Farbrepräsentationen

Als letzte Versuchsreihe werden in diesem Abschnitt die unterschiedlichen BCR-Farbrepräsentationen miteinander verglichen. Das Bild wird dabei jeweils in Rechtecke unterteilt und als Distanzfunktion kommt die von Volmer vorgeschlagene Minimumsfunktion zum Einsatz. Außerdem werden herkömmliche – d. h. nicht-kumulative Histogramme – miteinander verglichen. In Tabelle 6.5 sind die Ergebnisse notiert. Die dazu passende Umfrageauswertung befindet sich in Abbildung 6.4.

Die Rangfolge der Durchschnittswerte von \widetilde{Rank} ist folgende:

1. BCR-RGB9 (Durchschnitt \widetilde{Rank} : 0,280)
2. BCR-RGB-DIN (Durchschnitt \widetilde{Rank} : 0,297)
3. BCR-HSV-DIN (Durchschnitt \widetilde{Rank} : 0,331)

Diese Rangfolge widerspricht der erwarteten Rangfolge komplett. Volmer selbst beschreibt, dass die Ecken des RGB-Würfels als Basisfarben nicht sinnvoll sind und führt deshalb die RGB-DIN-Basisfarben ein. Da auch diese Basisfarben in dem für die inhaltsbasierte Bildsuche nicht geeignetem RGB-Farbraum liegen, habe ich in dieser Arbeit die HSV-DIN-Basisfarben eingeführt in der Erwartung, dadurch eine weitere Verbesserung des Konzeptes zu erreichen. Wenigstens bei den Durchschnittswerten von $Rank_1$ wird die erwartete Reihenfolge erreicht.

Die Ergebnisse der Umfrage sprechen hier eine ganz andere Sprache: So liegen dort die Basisfarben RGB9 eindeutig auf dem letzten Platz und die HSV-DIN-Basisfarben auf dem ersten Platz, wenn auch mit nur geringem Vorsprung vor den RGB-DIN-Basisfarben.

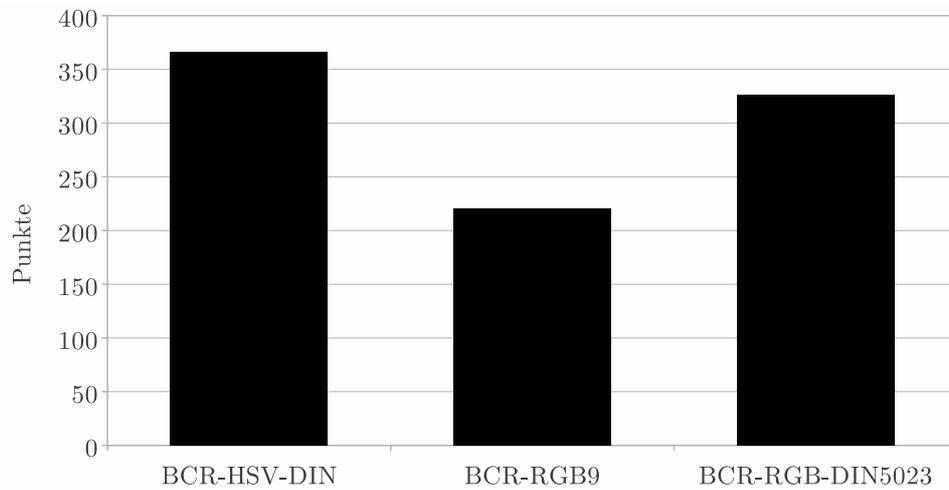
Im Vergleich der Farben miteinander wurde die Distanz L_2 gewählt, in diesem Vergleich die Distanz *Minimum*. Anscheinend dient das Nutzen der *Minimum*-Distanz vor allem den RGB-DIN-Basisfarben. Waren sie im vorherigen Vergleich noch etwa gleichauf mit den RGB9-Basisfarben, sind sie im jetzigen Vergleich fast auf einer Höhe mit den HSV-DIN-Basisfarben.

Bei dieser Messreihe zeigt sich zum einzigen Mal eine totale Diskrepanz zwischen den rein rechnerischen Werten und der menschlichen Bewertung.

Tabelle 6.5: Evaluationswerte BCR-Farbrepräsentation

Bild	BCR-RGB9		BCR-RGB-DIN		BCR-HSV-DIN	
	$Rank_1$	\widetilde{Rank}	$Rank_1$	\widetilde{Rank}	$Rank_1$	\widetilde{Rank}
24145.jpg	1.181	0,340	297	0,292	283	0,314
35817.jpg	932	0,242	180	0,233	380	0,226
33639.jpg	285	0,347	1	0,318	46	0,292
45396.jpg	406	0,191	988	0,345	718	0,493
Durchschnitt	701	0,280	366,5	0,297	356,8	0,331

Abbildung 6.4: Umfrageergebnisse BCR



6.6 Implementierung

In diesem Abschnitt werden Implementierungsdetails des im Rahmen dieser Arbeit entwickelten Programms erläutert, welches zur Berechnung der vorhergehenden Beispiele diente.

Als Programmiersprache wurde Java gewählt. Dies hat u. a. den Vorteil, dass das Programm plattformübergreifend lauffähig ist.

Zu Bildbearbeitungszwecken fand die ImageJ-Bibliothek ([[Rasband](#)]) Verwendung. Diese Bibliothek ist quelloffen (*public domain*) und wurde bereits für viele wissenschaftlichen Publikationen genutzt.

Auf der beigelegten CD befindet sich, neben der javadoc-Dokumentation, der Quelltext sowie das ausführbare Programm. Um dieses zu starten, öffnet man die `programm.jar`-Datei mit der Java Virtual Machine per Mausklick oder per Kommandozeile⁴.

Nach Start des Programms wählt man in der Menüleiste die gewünschten Methodenkombination, die zur Merkmalsberechnung und Distanzbildung eingesetzt werden soll und betätigt den Button **Vergleich**. Dort wählt man im ersten Schritt ein Referenzbild und im zweiten Schritt die Menge der zu vergleichenden Bilder aus. Bei großen Bilddateien kann die Berechnung der Merkmale viel Zeit in Anspruch nehmen.

Um das Errechnen der Histogramme und Verteilungsmomente nicht jedes Mal erneut durchführen zu müssen, legt das Programm im Bildverzeichnis einen neuen Ordner mit dem Namen `index` an und speichert dort – sofern noch nicht vorhanden – zu jedem Bild die berechneten Werte, aufgeteilt nach Farbmodus, Partitionierungsmodus und kumulativen bzw. nicht-kumulativen Histogrammen. Falls diese Dateien bereits existieren, werden sie beim Vergleich geladen – so ist es nicht nötig bereits berechnete Merkmale erneut zu berechnen und die Beantwortung der Anfrage geht wesentlich schneller vonstatten.

Das Programm wurde nach dem Entwurfsmuster *Model-View-Controller* entworfen. Im *Model*-Paket befinden sich reine Datenhaltungsklassen, im *View*-Paket befindet sich die graphische Benutzeroberfläche und im *Controller*-Paket befinden sich die Klassen, in denen die Berechnung abläuft. Im Anhang (Abbildung [B.1](#) auf Seite [47](#)) befindet sich ein vereinfachtes UML-Klassendiagramm des Programms.

Die Klasse `Controller` ist das Herzstück des Programms, sie steuert den Gesamtprogrammablauf.

Für jedes Bild, dessen Merkmale berechnet werden sollen, wird ein `PartitionedImage`-Objekt angelegt – konkret ist dieses je nach Partitionsmethode entweder ein `RectangularPartitionedImage` oder ein `FuzzyPartitionedImage`. Alle diese Klassen erben von der ImageJ-Klasse `ImagePlus`, die einige nützliche Methoden bietet – insbesondere zur Arbeit mit verschiedenen Farbräumen.

Da die `PartitionedImage`-Objekte ein Bild darstellen, kann es bei einer großen Anzahl von Bildern schnell passieren, dass Java der Speicher ausgeht. Um dies zu verhindern, wird nach erfolgter Berechnung des Ähnlichkeitswertes lediglich dieser Wert sowie der

⁴Programmiert wurde mit dem JDK 7, deshalb sollte das Programm mit der JVM 7 lauffähig sein. Eventuell funktionieren auch ältere Versionen der JVM.

Pfad zum Bild (aber eben nicht das Bild selber) in einem Objekt der Klasse `Similarity` gespeichert.

Zu einem `PartitionedImage` gehört ein `PartitionedImageController`, der die nötigen Methoden zur Berechnung und zum Vergleich der gewählten Merkmale kapselt. Partitionenspezifische Methoden sind in den beiden Klassen `RectangularImageController` bzw. `FuzzyImageController` implementiert, partitionsunabhängige Methoden finden sich bereits in deren Oberklasse.

Wird als Farbmodus eine der drei BCR-Farbrepräsentationen gewählt, kommen zusätzlich die Klassen `BCRColor` (*model*) und `BCRColorController` (*controller*) zum Einsatz. Diese dienen der Berechnung und Speicherung von BCR-Farbrepräsentationen ausgehend von einer klassischen RGB- oder HSV-Farbrepräsentation.

Dabei wird ein gegebener Punkt im RGB- bzw. HSV-Farbraum einem der Tetraeder der entsprechenden Delaunay-Zerlegung zugeordnet (je nach BCR-Basisfarben), indem die Tetraeder der Reihe nach durchlaufen werden und jeweils geprüft wird, ob sie den Punkt enthalten.

Die dazu verwendete Methode wird in [Yamaguchi, 2002, S. 215ff.] erläutert: Sei ein Tetraeder T mit den Eckpunkten

$$\begin{aligned} e_1 &= (x_1, y_1, z_1) \\ e_2 &= (x_2, y_2, z_2) \\ e_3 &= (x_3, y_3, z_3) \\ e_4 &= (x_4, y_4, z_4) \end{aligned}$$

gegeben. Ein Punkt $P = (p_x, p_y, p_z)$ befindet sich genau dann im Tetraeder T , wenn die folgenden fünf Determinanten alle das gleiche Vorzeichen haben:

$$D_0 = \begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix}$$

$$D_1 = \begin{vmatrix} p_x & p_y & p_z & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix}$$

$$D_2 = \begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ p_x & p_y & p_z & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix}$$

$$D_3 = \begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ p_x & p_y & p_z & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix}$$

$$D_4 = \begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ p_x & p_y & p_z & 1 \end{vmatrix}$$

Die um die restlichen Basisfarben erweiterten vier baryzentrischen (d. h. relativ zu den Tetraeder-Eckpunkten gelegenen) Koordinaten $b_i = D_i \div D_0$, $1 \leq i \leq 4$ ergeben dann die gesuchte BCR-Farbrepräsentation. Falls sich der gegebene Punkt in keinem Tetraeder befindet, wird gemäß Abschnitt 3.3.1 eine BCR-Grundform approximiert und zurückgegeben.

7 Fazit

Generell lässt sich sagen, dass das Programm keine optimalen Ergebnisse geliefert hat. Je nach Methode gab es teilweise sehr schlechte und teilweise halbwegs akzeptable Rückgaben. Dies zeigt sich nicht zuletzt an den häufigen Kommentaren zur Umfrage, die mangelnde Ähnlichkeit bei den zu bewertenden Ergebnissen kritisierten. In Abbildung 7.1 wird eines der besseren Ergebnisse gezeigt¹.

Es hat sich also gezeigt, dass das Merkmal *Farbe* für die Anwendung im Rahmen des *Content-Based Image Retrieval* nicht ausreichend ist. Zwar werden durchaus semantisch ähnliche Bilder gefunden, allerdings gibt es auch viele Bilder, die mit dem Anfragebild nichts zu tun haben. So wird bei dem Ergebnis in Abbildung 7.1 bereits an dritter Stelle ein Stuhl gefunden, obwohl das Anfragebild ein Schiff in einer Eiswüste zeigt. Dennoch kann das Merkmal *Farbe* für eine erste Einschätzung brauchbare Ergebnisse liefern.

Beim Vergleich der unterschiedlichen Methoden hat sich gezeigt, dass die Evaluationsmethoden, wie im Abschnitt 2.1 beschrieben, einen grundlegenden Nachteil haben: Alle genannten Messgrößen verlangen die Definition von *relevanten* Bildern. Dieser Forderung kann jedoch bei einer umfangreichen Datenbank nicht nachgekommen werden – ganz davon abgesehen, dass die Definition von *Relevanz* je nach Anwendung sehr unterschiedlich ausfallen kann.

Die Ergebnisse der Umfrage, die aufgrund der wenig sinnvoll zu nutzenden Evaluationsmessgrößen durchgeführt wurde, haben allerdings eine recht solide Einschätzung der Leistungsfähigkeit der verschiedenen Methoden ermöglicht. Zwar ist die Datenbasis der Umfrage verhältnismäßig klein gewesen, Tendenzen waren aber dennoch gut erkennbar – und das, obwohl auf eine explizite Ähnlichkeitsdefinition bewusst verzichtet wurde.

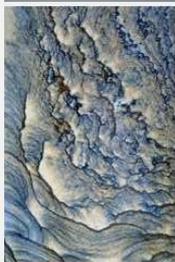
So hat sich gezeigt, dass bei Anwendung in der inhaltsbasierten Bildsuche der HSV-Farbraum dem RGB-Farbraum überlegen ist. Die von Volmer entwickelte BCR-Farbrepräsentation zeigt allerdings noch bessere Ergebnisse. Um für den Menschen ähnliche Bilder zurückzuliefern, ist es wichtig zu untersuchen, wie Menschen Ähnlichkeit wahrnehmen und dieses Wissen in die entsprechenden Systeme – so weit möglich – einzubauen.

Dass die Fuzzy-Methode, die in [Stricker u. Dimai, 1996] vorgeschlagen wurde, am schlechtesten abschneidet, ist überraschend. Dieses Ergebnis zeigt aber, dass ein komplizierterer Bildaufteilungsmechanismus nicht zwingend bessere Ergebnisse liefert.

¹Auf der beigelegten CD befinden sich alle besprochenen Ergebnisse.

Abbildung 7.1: Beispielrückgabe zum Anfragebild 24145.jpg, Farbmodus: BCR-HSV-DIN, rechteckige Partitionierung, Distanz: L2

BCR-HSV-DIN5023 Rechtecke L2 Kumulative Histogramme Vergleich Aus

						
Bild: 24145.jpg Ähnlichkeit: 0,0	Bild: 9698.jpg Ähnlichkeit: 0,14088	Bild: 41775.jpg Ähnlichkeit: 0,14246	Bild: 13177.jpg Ähnlichkeit: 0,14642	Bild: 49636.jpg Ähnlichkeit: 0,15071	Bild: 22724.jpg Ähnlichkeit: 0,15261	
						
Bild: 26789.jpg Ähnlichkeit: 0,15461	Bild: 13175.jpg Ähnlichkeit: 0,16012	Bild: 12996.jpg Ähnlichkeit: 0,16194	Bild: 24966.jpg Ähnlichkeit: 0,16476	Bild: 35769.jpg Ähnlichkeit: 0,16503	Bild: 21512.jpg Ähnlichkeit: 0,17012	Bild: 13350.jpg Ähnlichkeit: 0,1764
						
Bild: 31963.jpg Ähnlichkeit: 0,17283	Bild: 13132.jpg Ähnlichkeit: 0,17319	Bild: 17543.jpg Ähnlichkeit: 0,17319	Bild: 56103.jpg Ähnlichkeit: 0,17558	Bild: 56103.jpg Ähnlichkeit: 0,17558	Bild: 56103.jpg Ähnlichkeit: 0,17558	
						
Bild: 26789.jpg Ähnlichkeit: 0,17081	Bild: 13132.jpg Ähnlichkeit: 0,17319	Bild: 17543.jpg Ähnlichkeit: 0,17319	Bild: 13132.jpg Ähnlichkeit: 0,1735	Bild: 56103.jpg Ähnlichkeit: 0,17558	Bild: 56103.jpg Ähnlichkeit: 0,17558	

Anhang

A Delaunay-Zerlegungen

Tabelle A.1: Delaunay-Zerlegung der RGB9-Basisfarbmenge.²

	Grau	Schwarz	Rot	Gelb	Grün	Cyan	Blau	Magenta	Weiß	Vol.
1	•	•	•	•						8.33 %
2	•	•		•	•					8.33 %
3	•	•			•	•				8.33 %
4	•	•				•	•			8.33 %
5	•	•					•	•		8.33 %
6	•	•	•					•		8.33 %
7	•		•	•					•	8.33 %
8	•			•	•				•	8.33 %
9	•				•	•			•	8.33 %
10	•					•	•		•	8.33 %
11	•						•	•	•	8.33 %
12	•		•					•	•	8.33 %
	12	6	4	4	4	4	4	4	6	100 %

²[Volmer, 2006, S. 151]

Tabelle A.2: Delaunay-Zerlegung der DIN5023(RGB)-Basisfarbmenge.³

	N	Y _L	O _R	C _R	M _R	V _T	U _B	C _B	B _G	Y _G	O _C	B _S	B _K	W _T	Vol.
1	•							•		•				•	2.09 %
2		•								•	•			•	2.03 %
3	•				•						•			•	1.63 %
4	•									•	•			•	1.55 %
5	•					•	•							•	1.39 %
6	•					•		•	•				•		1.27 %
7		•								•		•	•		1.25 %
8			•		•									•	1.21 %
9	•								•	•			•		1.00 %
10	•					•	•							•	0.92%
11	•									•		•	•		0.88 %
12		•	•								•			•	0.87 %
13	•				•	•								•	0.83 %
14		•								•	•	•			0.67 %
15	•					•	•						•		0.63 %
16	•					•						•	•		0.55 %
17	•							•	•	•					0.54 %
18	•									•	•	•			0.50 %
19	•				•						•	•			0.49%
20					•	•						•	•		0.46%
21		•							•	•			•		0.41%
22	•						•	•	•						0,34%
23					•	•	•						•		0,29%
24		•	•								•	•			0,28%
25				•	•						•	•			0,26%
26			•	•	•						•				0,25%
27				•	•							•	•		0,23%
28		•	•									•	•		0,21%
29	•			•	•							•			0,20%
30		•	•								•	•			0,15%
31		•	•									•	•		0,13%
32						•	•	•					•		0,10%
33				•	•	•								•	0,05%
	16	7	7	5	11	8	8	5	6	10	12	14	13	10	23,66%

³[Volmer, 2006, S. 152]

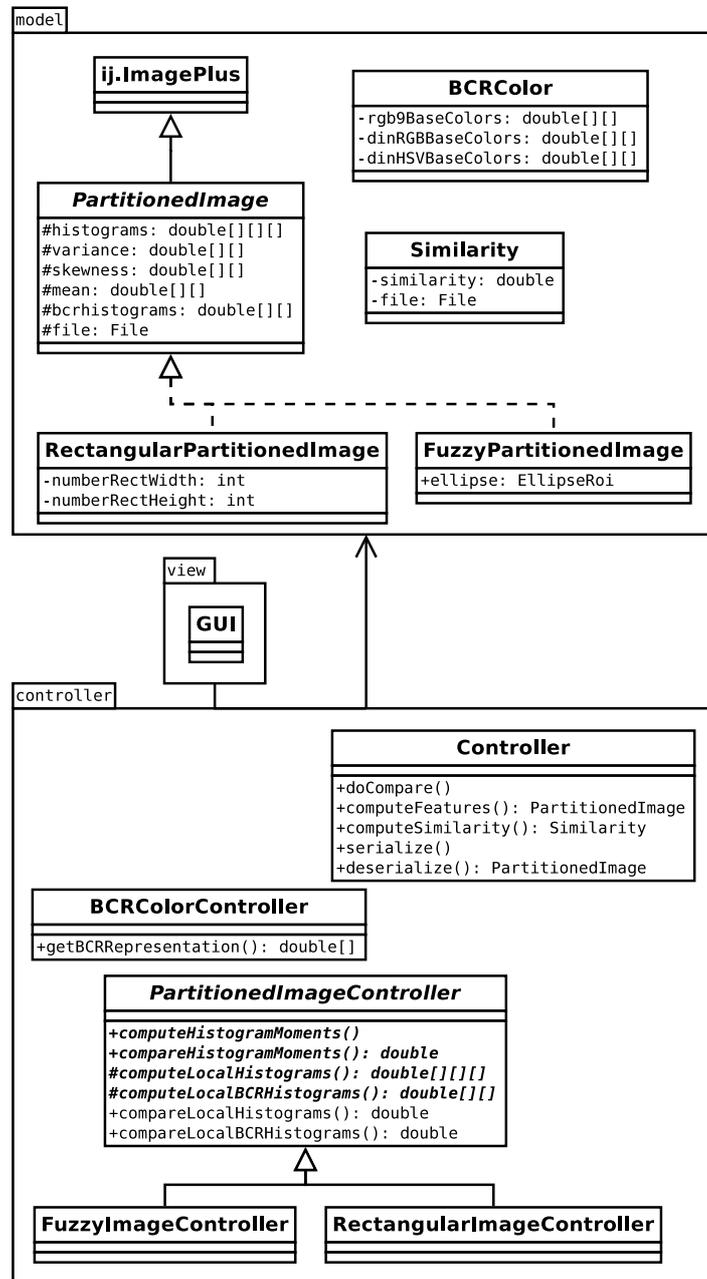
Tabelle A.3: Delaunay-Zerlegung der DIN5023(HSV)-Basisfarbmenge.⁴

	N	Y _L	O _R	C _R	M _R	V _T	U _B	C _B	B _G	Y _G	O _C	B _S	B _K	W _T
1	•										•	•	•	
2	•								•	•			•	
3	•				•							•	•	
4	•				•	•								•
5	•										•	•		•
6	•									•	•			•
7	•					•	•						•	
8	•							•	•					•
9	•							•	•				•	
10	•					•	•							•
11	•									•	•		•	
12		•								•	•			•
13	•								•	•				•
14	•				•	•							•	
15				•							•	•		•
16	•						•	•					•	
17	•				•							•		•
18			•	•							•			•
19					•	•	•							•
20		•								•	•		•	
21								•	•	•				•
22	•						•	•						•
23				•	•							•		•
24		•	•								•			•
25		•	•								•	•		
26		•									•	•	•	
27		•							•	•			•	
28			•	•							•	•		
29			•	•	•									•

⁴erstellt mit TetGen, <http://tetgen.org/> (15. Februar 2012)

B Klassendiagramm

Abbildung B.1: Klassendiagramm



Literaturverzeichnis

- [Bellebaum u. a. 2012] BELLEBAUM, Christian ; THOMA, Patrizia ; DAUM, Irene: Visuelle Wahrnehmung: Was, Wo und Wie. In: *Neuropsychologie*. VS Verlag für Sozialwissenschaften, 2012. – ISBN 978-3-531-94108-0, S. 31–46
- [Datta u. a. 2008] DATTA, Ritendra ; JOSHI, Dhiraj ; LI, Jia ; WANG, J.Z.: Image Retrieval: Ideas, Influences, and Trends of the New Age. In: *ACM Computing Surveys (CSUR)* 40 (2008), April, Nr. 2, S. 5:1–5:60. – ISSN 03600300
- [Leicht 2007] LEICHT, Joachim: *Filterung von Histogrammen in der inhaltsbasierten Bildsuche*, Eberhard Karls Universität Tübingen, Diplomarbeit, 2007. <http://www.home.joachimleicht.de/studium.diplomarbeit.html>, Abruf: 15. Februar 2012
- [Li u. Wang 2008] LI, Jia ; WANG, James Z.: Real-time Computerized Annotation of Pictures. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30 (2008), Juni, Nr. 6, S. 985–1002. – ISSN 0162-8828
- [Li u. Wang 2003] LI, Jia ; WANG, J.Z.: Automatic Linguistic Indexing of Pictures by a Statistical Modeling Approach. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25 (2003), Nr. 9, S. 1075–1088
- [Mayron 2008] MAYRON, L.M.: *Image Retrieval Using Visual Attention*, Florida Atlantic University, Diss., 2008. <http://mayron.net/research/>, Abruf: 15. Februar 2012
- [Müller u. a. 2001] MÜLLER, H. ; MÜLLER, W. ; SQUIRE, D.M.G. ; MARCHAND-MAILLET, S. ; PUN, Thierry: Performance evaluation in content-based image retrieval: Overview and proposals. In: *Pattern Recognition Letters* 22 (2001), Nr. 5, S. 593–601. – ISSN 0167-8655
- [Niblack u. a. 1993] NIBLACK, Wayne ; BARBER, Ron ; EQUITZ, William ; FLICKNER, Myron ; GLASMAN, Eduardo H. ; PETKOVIC, Dragutin ; YANKER, Peter ; FALOUTSOS, Christos ; TAUBIN, Gabriel: The QBIC Project: Querying Images By Content using Color, Texture, and Shape. In: *Proc. Storage and Retrieval for Image and Video Databases (SPIE)* Bd. 1908, 1993, S. 173–187
- [Rasband] RASBAND, W.S.: *ImageJ*. U.S. National Institutes of Health, Bethesda, Maryland, USA. <http://imagej.nih.gov/ij/>, Abruf: 15. Februar 2012. – 1997-2012

- [Smeulders u. a. 2000] SMEULDERS, A.W.M. ; WORRING, Marcel ; SANTINI, Simone ; GUPTA, Amarnath ; JAIN, Ramesh: Content-Based Image Retrieval at the End of the Early Years. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22 (2000), Nr. 12, S. 1349–1380
- [Smith u. Chang 1996a] SMITH, J.R. ; CHANG, S.F.: Tools and Techniques for Color Image Retrieval. In: *Storage and Retrieval for Image and Video Databases (SPIE)* Bd. 2670, 1996, S. 426–437
- [Smith u. Chang 1996b] SMITH, J.R. ; CHANG, S.F.: VisualSEEK: a fully automated content-based image query system. In: *Proceedings of the fourth ACM international conference on Multimedia*, ACM, 1996 (Multimedia '96), S. 87–98
- [Stricker u. Orenge 1995] STRICKER, M. ; ORENGO, M.: Similarity of Color Images. In: *Proc. Storage and Retrieval for Image and Video Databases (SPIE)* Bd. 2420, 1995, S. 381–392
- [Stricker 1994] STRICKER, Markus: A new approach for robust ellipse fitting. In: *Proc. of the Third Intern. Conference on Automation, Robotics and Computer Vision (Singapore)* Bd. 2, 1994, S. 940–945
- [Stricker u. Dimai 1996] STRICKER, Markus ; DIMAI, A.: Color Indexing with Weak Spatial Constraints. In: *Proc. Storage and Retrieval for Image and Video Databases (SPIE)* Bd. 2670, 1996, S. 29–40
- [Stricker u. Dimai 1997] STRICKER, Markus ; DIMAI, Alexander: Spectral Covariance and Fuzzy Regions for Image Indexing. In: *Machine Vision and Applications* 10 (1997), S. 66–73. – ISSN 0932–8092
- [Swain u. Ballard 1991] SWAIN, M.J. ; BALLARD, D.H.: Color Indexing. In: *International Journal of Computer Vision* 7 (1991), Nr. 1, S. 11–32. – ISSN 0920–5691
- [Volmer 2006] VOLMER, Stephan: *Inhaltsbasierte Bildsuche mittels visueller Merkmale*, TU Darmstadt, Diss., November 2006. <http://tuprints.ulb.tu-darmstadt.de/750/>, Abruf: 15. Februar 2012
- [Wang u. a. 2001] WANG, J.Z. ; LI, Jia ; WIEDERHOLD, Gio: SIMPLIcity: Semantics-sensitive integrated matching for picture libraries. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23 (2001), Nr. 9, S. 947–963. – ISSN 0162–8828
- [Yamaguchi 2002] YAMAGUCHI, F.: *Computer-aided geometric design: a totally four-dimensional approach*. Springer, 2002. – ISBN 978–4–431–70340–2