

# Digitale Bildverarbeitung

## Einheit 10

### Merkmalsextraktion

---

Lehrauftrag WS 2007/2008

Fachbereich M+I der FH-Offenburg



Dr. Bernard Haasdonk

Albert-Ludwigs-Universität Freiburg

### Ziele der Einheit

---

- Einsehen, dass das Problem des „Vergleichs“ von Bildteilen eine nichttriviale Aufgabe ist
- Verstehen, wie Bildteile/Objekte durch Zahlen, sogenannten **Merkmalen** beschrieben werden können
- **Wünschenswerte Eigenschaften** dieser Attribute kennenlernen
- Den Begriff der **Invarianz** kennenlernen
- Einige **konkrete Beispiele** für Merkmale und deren Berechnung nachvollziehen
- Einige **einfache Anwendungen** vorführen: inhaltsbasierte Bildsuche, Echtzeit-Bildsysteme

## Motivation der Merkmalsextraktion

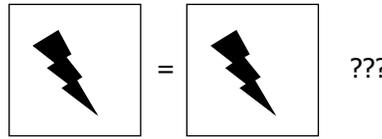
- Bisherige Methoden ermöglichen das Erzeugen von Segmenten, d.h. Bildteilen, die einzelne oder mehrere Objekte enthalten können.



- Wir können aber wenig aussagen über verschiedene Segmente
- Eine wichtige Operation zwischen Segmenten ist das Vergleichen:

### Szenario 1:

Wie können wir hier einen Vergleich durchführen?

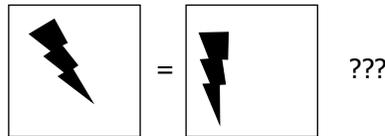


- Einfache Möglichkeit:

- Differenzbild, unterschiedliche Pixel zählen
- bei vielen unterschiedlichen Pixeln => unterschiedliche Segmente
- bei wenig unterschiedlichen Pixeln => ähnliche/gleiche Segmente

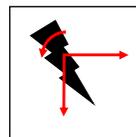
## Motivation der Merkmalsextraktion

- Szenario 2:



- Mögliches Vorgehen: **Template Matching**

- Das erste Segment wird in alle Positionen und Drehungen versetzt und mit dem zweiten verglichen, bis es „passt“.



alle Drehungen müssen mit allen x-Verschiebungen und allen y-Verschiebungen kombiniert werden!

Sehr rechenintensiv also ungeschickt!

- Vergleichen ist also schwierig wegen

- den vielen Pixeln, d.h. hoher Dimensionalität
- den verschiedenen möglichen kombinierten Positionen und Drehungen
- subjektiver oder problemabhängiger Begriff der „Ähnlichkeit“: manchmal ist die Farbe wichtig, manchmal unwichtig, etc.

## Motivation der Merkmalsextraktion

- Menschliches Vorgehen beim Vergleichen/Unterscheiden

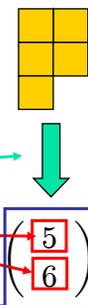
Quadrate:	4	5	4	4	4	4
Ecken:	6	6	8	4	8	4
Umfang:	10	10	10	8	10	10
Farbe:	grün	grün	grün	grün	rot	grün

- ➡ Es werden Charakterisierungen durch „Attribute“ oder Zahlen vorgenommen
- ➡ Vergleich der Attribute oder Zahlen ist einfach:  
Sind diese ungleich so sind die Objekte unterschiedlich
- ➡ Dies ist offensichtlich sehr viel schneller als Template Matching!

4.10.2007 B. Haasdonk, Digitale Bildverarbeitung, FH Offenburg WS 2007/2008, Einheit 10 5

## Motivation der Merkmalsextraktion

- Gegeben: **Segment** eines Bildes
- Ziel: **Charakterisierung** des Segments durch maschinell zu ermittelnde Zahlen, die objektive Vergleiche ermöglichen
- Das Berechnen der Zahlen nennt man **Merkmalsextraktion**
- Diese Zahlen nennt man **Merkmale** (Hier: Anzahl Quadrate und Ecken)
- Werden mehrere Zahlen aus einem Segment extrahiert, nennt man das Ergebnis einen **Merkmalsvektor**
- Der Merkmalsvektor soll **möglichst kurz** sein, damit der Vergleich möglichst schnell ist
- Insgesamt wird damit eine sehr **kompakte Segment-Repräsentation** erzeugt (Art extremer Datenkompression)



4.10.2007 B. Haasdonk, Digitale Bildverarbeitung, FH Offenburg WS 2007/2008, Einheit 10 6

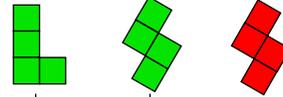
## Merkmalsraum

- Objektvergleich im Merkmalsraum
  - Nach der **Merkmalsextraktion** ist ein Objekt nur noch ein Punkt in einem abstrakten Raum, dem **Merkmalsraum**
  - Seine **Koordinaten** sind genau die Einträge seines Merkmalsvektors
  - Man kann verschiedene Objekte **gleichzeitig** in diesem Raum repräsentieren
  - Man kann verschiedene Objekte **vergleichen**, indem man den **Abstand** zwischen ihren Punkten im Merkmalsraum berechnet (Satz des Pythagoras) und den Abstand als Maß für den Unterschied nimmt.

Beispiel: Abstand 1. & 3. Objekt

$$\sqrt{2^2 + 1^2} = 2.236$$

Objekte



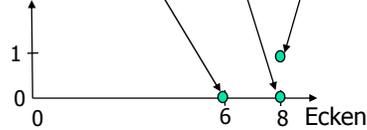
Merkmalsvektoren

(Ecken und Rot-Mittelwert)

$$\begin{pmatrix} 6 \\ 0 \end{pmatrix} \quad \begin{pmatrix} 8 \\ 0 \end{pmatrix} \quad \begin{pmatrix} 8 \\ 1 \end{pmatrix}$$

Merkmalsraum

Rot-Mittelwert



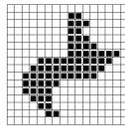
## Einfache Geometrische und Topologische Merkmale

- Viele Merkmale für Segmente sind schön **anschaulich** und einfach zu berechnen

- Die **Fläche**

- Zählen aller Pixel

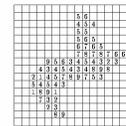
$$\mathcal{F} = 63$$



- Die **Masse**

- Addieren aller Grauwerte

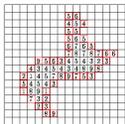
$$\mathcal{M} = 315$$



- Der **Umfang**

- Alle Pixel zählen, die zu einem Hintergrundpixel benachbart sind

$$\mathcal{U} = 37$$

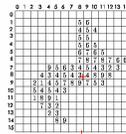


- Der **Schwerpunkt**

- Mittelung über „Position mal Gewicht“

$$x_s = \frac{1}{\sum_{i=1}^M \sum_{k=1}^N m_{ik}} \sum_{i=1}^M \sum_{k=1}^N x_i m_{ik}$$

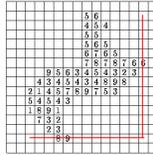
$$y_s = \frac{1}{\sum_{i=1}^M \sum_{k=1}^N m_{ik}} \sum_{i=1}^M \sum_{k=1}^N y_k m_{ik}$$



## Einfache Geometrische und Topologische Merkmale

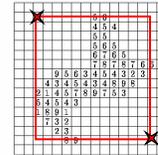
- **Breite und Höhe (FeretXY)**

- Grenzen der integrierten Linienprofile bestimmen



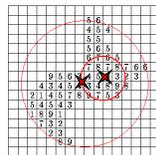
- **Bounding Box**

- Ergibt 4 Werte (2 Eckpunkte)



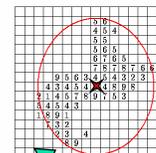
- **Minimaler Umkreis  
Maximaler Inkreis**

- Ergibt 6 Werte: (2 Radien und zwei Mittelpunkte)



- **Weitere (nicht so einfach)**

- Anzahl der Löcher eines Objektes
- Anzahl Ecken
- Rotations-Symmetriegrad
- Ellipsenanpassung



## Anforderungen an Merkmale

- Möglichst **ähnliche Werte** für ähnliche Objekte
- Unterschiedliche Werte für unterschiedliche Objekte
  - Man spricht hier auch von **Separationsfähigkeit**
- Einfache **Berechenbarkeit**
- **Robustheit** bezüglich Störungen
  - bzgl. Rauschen bzw. unwesentlichen Änderungen sollten sich die Merkmale nicht sehr verändern
- Möglichst **kompakter Merkmalsvektor**
  - möglichst wenig Zahlen, keine Abhängigkeiten, keine Redundanz
- Und viele mehr (Invarianz, Stetigkeit, Clustering, ...)

## Invarianz

- In vielen Problemen sind bestimmte Bewegungen der Objekte erlaubt, ohne dass deren Bedeutung sich ändert.
- Es ist also wünschenswert, dass auch die Merkmale sich nicht ändern bei diesen Bewegungen. Diese heißen dann **invariante Merkmale** oder Invarianten
- Merkmale sind
  - **translationsinvariant**, wenn sie unverändert bleiben bei **Verschieben** des Objektes
  - **rotationsinvariant**, wenn sie unverändert bleiben bei **Drehung** des Objektes
  - **skalierungsinvariant**, wenn sie unverändert bleiben, wenn das **Bild gezoomt** wird oder die Kamera näher oder weiter weg fährt
  - **spiegelungsinvariant**, wenn sie unverändert bleiben bei **Spiegelung** des Objektes

4.10.2007 B. Haasdonk, Digitale Bildverarbeitung, FH Offenburg WS 2007/2008, Einheit 10 11

## Konturmerkmale

- Es wird nur der **Rand** des Segmentes verwendet, um Merkmale zu berechnen
- Das **Innere** wird **vernachlässigt**

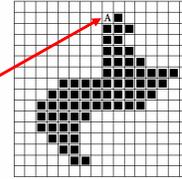


- Es geht hier natürlich viel Information verloren
- Macht nur Sinn, wenn
  - Segmente sehr genau einzelnen Objekten entsprechen: Es darf nicht zu viel Hintergrund im Segment vorhanden sein und es sollten nicht mehrere Objekte verschmolzen sein
  - Objekte sich auch durch äußere Form unterscheiden lassen, ist z.B. sinnlos bei „Sportball-Typen“ oder „Obst“

4.10.2007 B. Haasdonk, Digitale Bildverarbeitung, FH Offenburg WS 2007/2008, Einheit 10 12

## Randcodierung

- Ziel: Der Rand des Segmentes wird in eine **Zahlenfolge übersetzt**
- Vorgehen:
  - Wahl eines Startpunktes und einer **Umlaufrichtung**, z.B. im Uhrzeigersinn
  - Kodierung der Rand-Richtung durch eine von 8 Zahlen, z.B. dem **Freeman-Code**: die Richtung von x aus bekommt die angegebene Zahl, z.B. rechts = 0, runter = 6, hoch = 2
  - Entlanglaufen an dem Rand und Sammeln der Richtungs-Kennzeichner
- Die **Randcodierung** ist eine periodische Zahlenkette
- Sie ist **translationsinvariant**, **nicht rotationsinvariant**, **nicht größeninvariant**.
- **Nachteil**: ist abhängig vom Startpunkt!

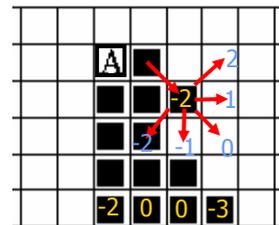


3	2	1
4	X	0
5	6	7

Randcodierung =  
0757770055544445566  
74333221100012222

## Weitere Konturmerkmale

- Die **relative Randcodierung**
  - Es werden nicht die Richtungen, sondern die **Richtungsänderungen** kodiert in Vielfaches von 45°
  - geradeaus = 0  
links = positive Werte  
rechts = negative Werte,
  - Dieser Code ist **translations- und rotationsinvariant**
- Sehr zu empfehlen: **Fourierdeskriptoren** [56]
  - Der Rand kann als komplexe Funktion aufgefasst werden
  - Diese kann in eine **Fourierreihe** entwickelt werden
  - Aus den nicht-invarianten **Fourier-Koeffizienten** können durch Kombination sogenannte **Fourierdeskriptoren** berechnet werden.
  - Diese sind **translations-, rotations- und skalierungsinvariant**

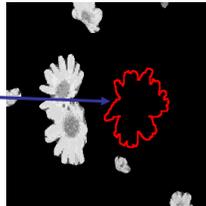


Rel. Randcode =  
-2-1-220-300-2000

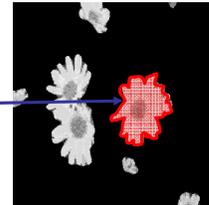
## Regionsmerkmale

- Es werden Merkmale extrahiert, die auf Information des **gesamten Segmentes** basieren, nicht nur des Randes

Konturmerkmale verwenden nur Randinformation



Regionsmerkmale verwenden das gesamte Segment



- Einige bekannte Beispiele sind: **Grauwert- oder Farb-Histogramme** und abgeleitete Größen wie **Mittelwert, Varianz, minimaler, maximaler Grauwert**, etc.
- Wenn man genügend viele Merkmale nimmt, kann man das Segment vollständig ohne Informationsverlust beschreiben.

## Kombinierte Zentrale Momente

- Für ausgeklügeltere Merkmale geht deren **Anschaung** immer weiter **verloren**, sie sind aber für den Rechner aussagekräftig
- Ausgangspunkt sind die **zentralen Momente** (Einheit 05)

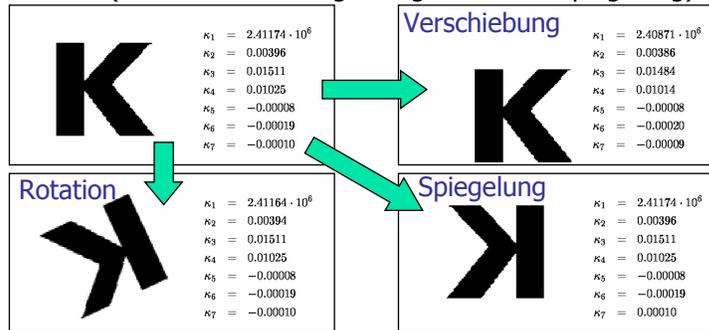
$$\mu_{ik} = \sum_{x=1}^M \sum_{y=1}^N (x - x_s)^i (y - y_s)^k \cdot g(x, y)$$

- Diese werden nun kombiniert und ergeben die **7 Merkmale von Hu**
- Sie sind **translation-, rotations-, skalierungs-** und sogar (außer k7) **spiegelinvariant**

$\begin{aligned} \mu_1 &= \mu_{20} + \mu_{02} \\ \mu_2 &= \frac{1}{M^2} \left[ (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2 \right] \\ \mu_3 &= \frac{1}{M^2 N^2} \left[ (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} + \mu_{03})^2 \right] \\ \mu_4 &= \frac{1}{M^2 N^2} \left[ (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2 \right] \\ \mu_5 &= \frac{1}{M^2} \left[ (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12}) + (\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2 \right] + \\ &\quad + (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03}) + [3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \\ \mu_6 &= \frac{1}{M^2 N^2} \left[ (\mu_{30} - \mu_{03}) \cdot [(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] + \right. \\ &\quad \left. + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03}) \right] \\ \mu_7 &= \frac{1}{M^2} \left[ (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12}) + (\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2 \right] + \\ &\quad + (3\mu_{12} - \mu_{30})(\mu_{21} + \mu_{03}) + [3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \end{aligned}$	  	$\begin{aligned} \kappa_1 &= 2.59069 \cdot 10^6 \\ \kappa_2 &= 0.02263 \\ \kappa_3 &= 0.39967 \\ \kappa_4 &= 0.01108 \\ \kappa_5 &= -0.00063 \\ \kappa_6 &= 0.00163 \\ \kappa_7 &= -0.00016 \end{aligned}$	  	$\begin{aligned} \kappa_1 &= 2.37675 \cdot 10^6 \\ \kappa_2 &= 0.04408 \\ \kappa_3 &= 0.00331 \\ \kappa_4 &= 0.00864 \\ \kappa_5 &= -0.00004 \\ \kappa_6 &= 0.00167 \\ \kappa_7 &= -0.00001 \end{aligned}$  $\begin{aligned} \kappa_1 &= 157742.3 \\ \kappa_2 &= 0.0 \\ \kappa_3 &= 0.0 \\ \kappa_4 &= 0.0 \\ \kappa_5 &= 0.0 \\ \kappa_6 &= 0.0 \\ \kappa_7 &= 0.0 \end{aligned}$
---	----------	--	----------	--

## Kombinierte Zentrale Momente

- Invarianz bezüglich Bewegungen:
  - Die Werte sind **identisch** (bis auf Rechen-Ungenauigkeiten und Spiegelung)



- Webseite mit interaktiver Berechnung:
  - <https://elearning.fh-offenburg.de/moodle/mod/resource/view.php?id=895>

## Anwendungen von Bildmerkmalen

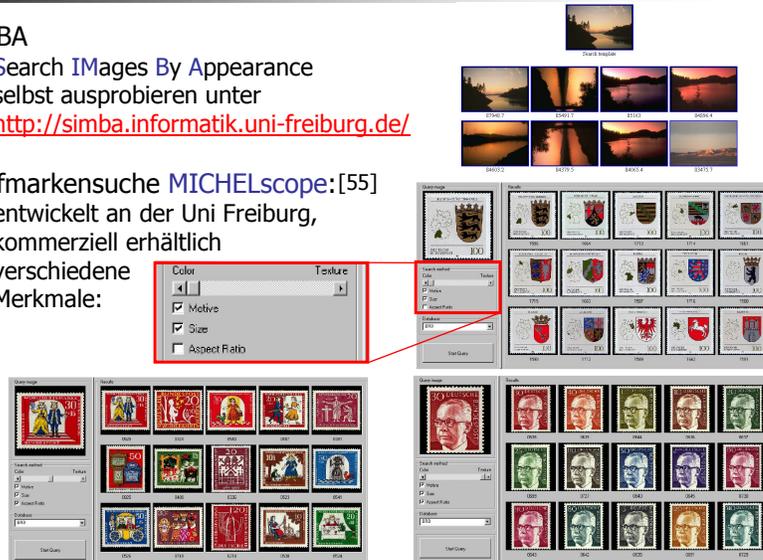
- Inhaltsbasierte Bildsuche
  - Bilder werden aufgrund **inhaltlicher Aspekte** verglichen
  - Gegensatz zu „kontextbasierte“ Bildsuche, z.B. Google sucht nur nach „Wörtern“ in der Umgebung der Bilder.
  - Typisch: „Query by Example“: zu einem Anfragebild des Benutzers werden die besten Treffer in einer Datenbank zurückgegeben



- Bei einer Datenbank von 1000-100000 Bildern und einem Benutzer, der sofort eine Antwort haben will, ist **Geschwindigkeit wichtig**: für Merkmalsvektoren ideal, direkter Bildvergleich unmöglich.
- Schneller Vergleich ist möglich, weil für die Datenbank die **Merkmalsvektoren vorberechnet** werden können.

## Anwendungen von Bildmerkmalen

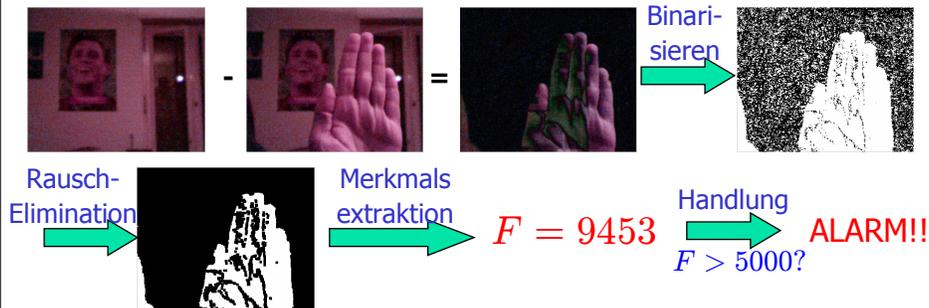
- SIMBA
  - Search IMages By Appearance
  - selbst ausprobieren unter <http://simba.informatik.uni-freiburg.de/>
- Briefmarkensuche MICHELscope:[55]
  - entwickelt an der Uni Freiburg, kommerziell erhältlich
  - verschiedene Merkmale:



4.10.2007 B. Haasdonk, Digitale Bildverarbeitung, FH Offenburg WS 2007/2008, Einheit 10 19

## Anwendungen von Bildmerkmalen

- Überwachungssystem
  - Differenz zu einem Referenzbild liefert Handlungs-Kriterium über Anzahl der „Pixel-Eindringlinge“: Merkmal der Fläche!



- Selbst ausprobieren: MATLAB mit Bibliothek VCAPG2
- Allgemein ist Verwendung von Bildmerkmalen in allen Problemen sinnvoll, die auf vielen und schnellen Vergleichen von Bildsegmenten angewiesen sind
- Insbesondere in nächster Einheit: Klassifikation/Objektdetektion

4.10.2007 B. Haasdonk, Digitale Bildverarbeitung, FH Offenburg WS 2007/2008, Einheit 10 20

## Zusammenfassung

- Alle Objekte, die vom Rechner verarbeitet werden sollen, müssen ihre **Eigenschaften in Zahlen**, sogenannten **Merkmalen ausdrücken**.
- Die Berechnung heißt **Merkmalsextraktion**, mehrere Merkmale eines Objektes werden zu einem **Merkmalsvektor** zusammengefasst.
- Merkmalsextraktion zielt auf eine **kompakte, charakterisierende und unterscheidungsfähige Repräsentation** von Objekten/Bildern durch Zahlen
- Die **Invarianz** von Merkmalen ist oft wünschenswert, wenn bestimmte Objektbewegungen möglich sind, die die Bedeutung der Objekte unverändert lassen.
- Wichtige konkrete Merkmale für Objekte sind **Konturbasierte** und **Regionsbasierte**, z.B. Fläche, Umfang, Bounding Box, Randcodierung, Fourierdeskriptoren, Histogramme, zentrale Momente und hieraus konstruierte Invarianten.
- Die wichtige Operation des **Vergleichs von Bildern** ist effizient möglich: statt teurem Template Matching ist ein schneller Vergleich von Merkmalsvektoren möglich.
- Der **Merkmalsraum** erlaubt eine gemeinsame Darstellung verschiedener Objekte, indem **jedes Objekt** durch **einen Punkt** gegeben ist, dessen Koordinaten die Merkmale sind.

## Weiterführende Literatur

- [53] R.O. Duda, P.E. hart und D.G. Stork: **„Pattern Classification“**. Wiley Interscience, 2nd edition, 2001.
- [54] S. Theodoridis und K. Koutroumbas: **„Pattern Recognition“**. Academic Press Inc.(London) Ltd, 2nd edition, 2003.
- [55] S. Siggelkow. „Feature Histograms for Content-based Image Retrieval“. Dissertation, Universität Freiburg, 2003. Online unter <http://www.freidok.uni-freiburg.de/volltexte/631>
- [56] H. Burkhardt und B. Haasdonk: „Mustererkennung WS 02/03, ein multimedialer Grundlagenkurs im Hauptstudium Informatik“. Institut für Informatik, Universität Freiburg, 2003. (CDs werden verkauft), Online unter <http://imb.informatik.uni-freiburg.de/lectures/mustererkennung/WS0304/material.html>
- [57] K. D. Tönnies, **„Digitale Bildverarbeitung“**, Pearson Studium, 2005

