

# Segmentierung



# Segmentierung

- Zusammenfassung inhaltlich zusammengehörender *homogener* Regionen, z. B.
  - ☞ Gewebe (Haut, Knochen, . . .)
  - ☞ Organe (Leber, Herz, . . .)
  - ☞ Bereiche (Hirnareale, malign vs. benign, . . .)
- Zuordnung dieser Regionen zu Klassen: *Klassifikation* und *Mustererkennung*
- Grenzen zwischen *Segmentierung* und *Klassifikation* fließend

Homogeniekriterien, z. B.:

- Punktbasiert
- Kantenbasiert
- Regionbasiert
- Texturbasiert
- Modellbasiert

# Segmentierung in der Medizinischen Bildverarbeitung

- Herauslösen von *Objekten* aus dem Bildzusammenhang
- Trennung von wichtigen (Vordergrund) und unwichtigen Bereichen (Hintergrund)
- Wichtiger Schritt zur Weiterverarbeitung von Bildern

konkrete Anwendungen:

- Segmentierung von Hirnbereichen aus MRT-Datensätzen
- Segmentierung von Melanomen aus Videodaten
- Segmentierung verschiedener Gewebe aus Tomographiedaten zur Operationsplanung und Gewebesimulation
- Segmentierung von Organen im Visible Human Projekt
- Nerv-, Wirbeldetektion

Bemerkung:

- Segmentierungsalgorithmen müssen speziell auf BV-anwendungen zugeschnitten werden, d. h. keine Algorithmen „von der Stange“
- aktuelles Forschungsthema

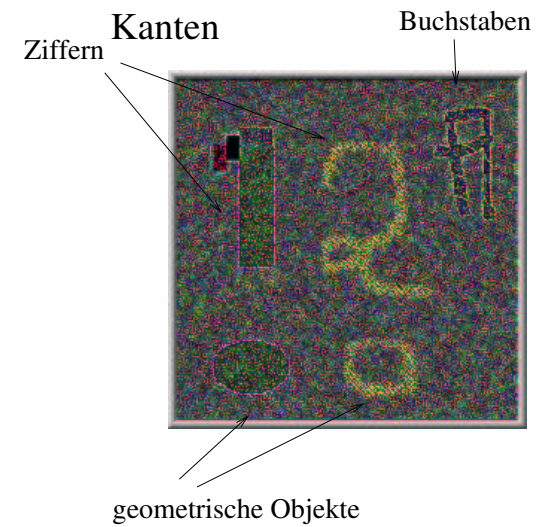
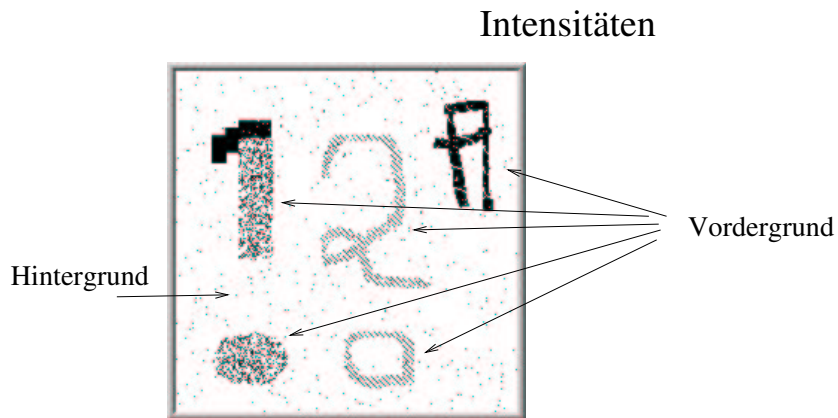
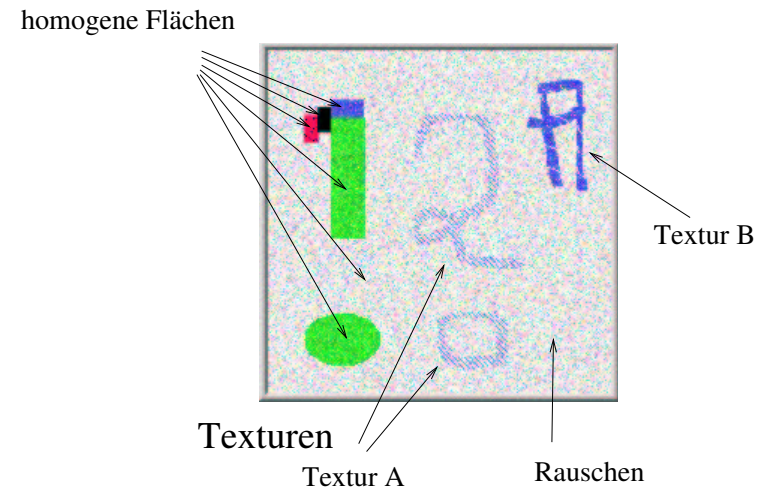
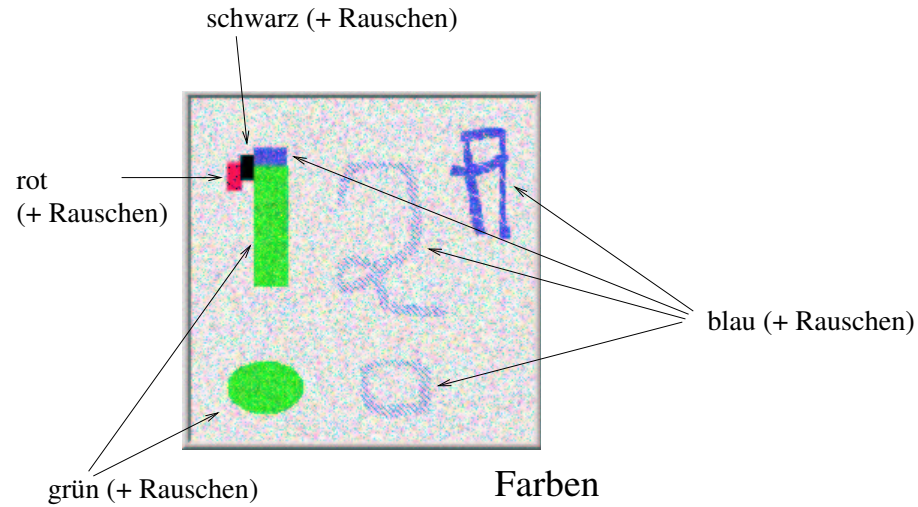


Abbildung 1: Unterschiedliche Homogenitätskriterien

# Probleme bei der Segmentierung

Häufige Probleme bei der Segmentierung von medizinischen Bilddaten:

- Konturen nicht geschlossen (auslaufen bei region-fill)
- Objekte/Texturen nicht homogen
- Bild überlagert von Artefakten, Rauschen
- Anatomische Abnormalitäten (Template-basierte Segmentierung)
- Zuordnung zwischen Grauwertintensität und Gewebe nicht möglich (US, MR)
- Überlagerung und/oder Verdeckung von Objekten (Röntgen, US)
- Bewegung von Organen (Herz, Lunge, Leber, . . .)
- Anatomische Constraints notwendig

# Pixelbasierte Segmentierungsverfahren -

## Schwellwertverfahren (engl. *threshold*)

- Vorteil: effizient
- Nachteil: keine Aussage über Zusammenhang von Pixeln

gegeben:

- Grauwertbild der Größe  $N$
- $G$  Grauwerte  $\in \{0, \dots, G - 1\}$  im Bild  $g$
- $n$  Schwellwerte  $t_i: 0 \leq t_0 \leq \dots \leq t_n \leq G - 1$
- $n$  Klassen  $K_i$
- normiertes Histogramm  $p$  der Grauwerte, mit  $\sum_{g=0}^{G-1} p_g = \sum_{g=0}^{G-1} p(g) = 1.$

Segmentierungsabbildung:

$$g(x, y) \mapsto K_i, \quad \text{falls } t_i \leq g(x, y) \leq t_{i+1}$$

# Schwelwertsegmentierung

Bestimmung der Schwellwerte:

- manuell (Fensterung)
- automatisch
  - ☞ global, lokal oder dynamisch
  - ☞ mit oder ohne Vorwissen über das Bild

Definition:

- Ein Histogramm heißt *bimodal* / *multimodal*, wenn es im Wesentlichen durch 2 / mehrere lokale Maxima bestimmt wird.

# Schwelwertsegmentierung

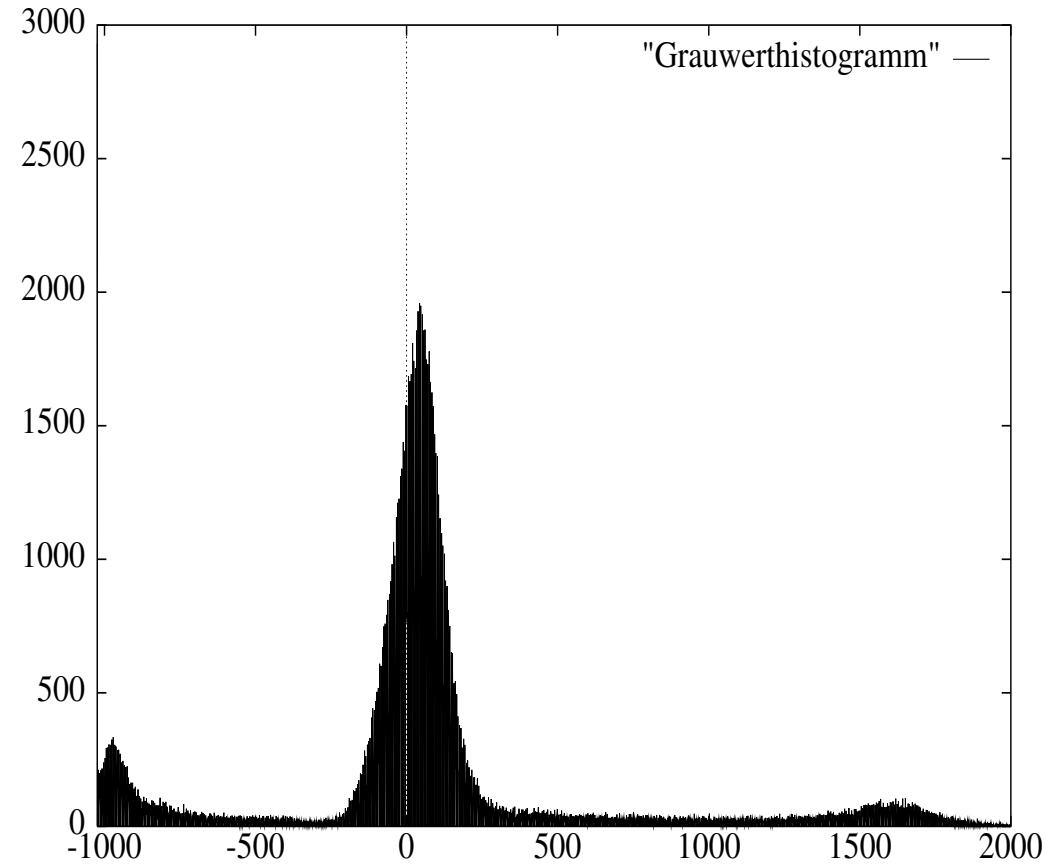


Abbildung 2: CT-Schnittbild mit bimodalem Grauwerthistogramm.

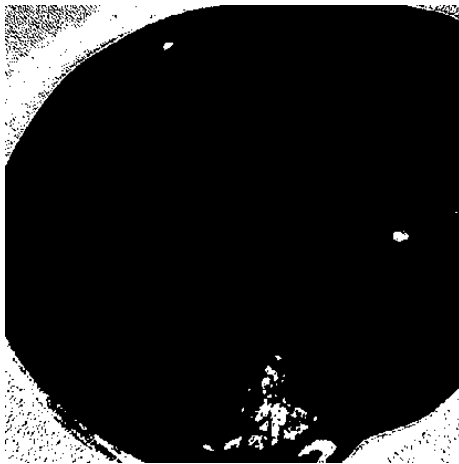
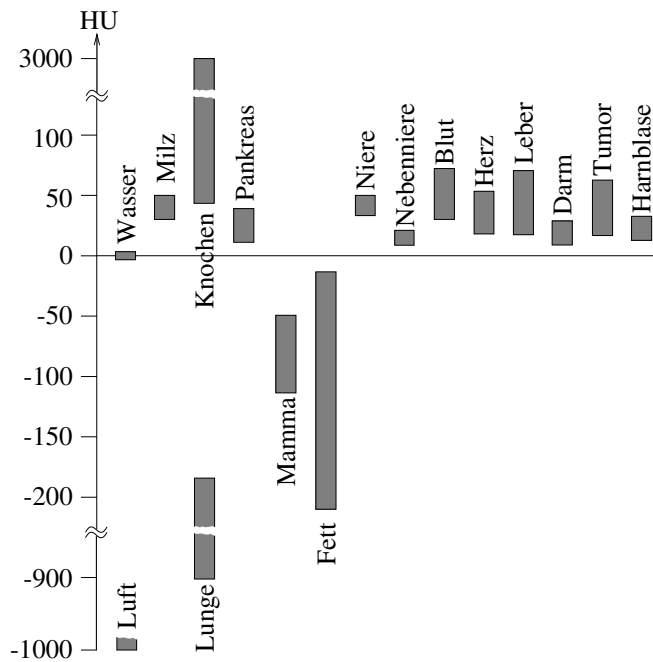


Abbildung 3: Schwellwertsegmentierung: verschiedene „Gewebefenster“  
 „Luft“ ( $-1024 \leq g \leq -500$ ), „Wasser“ ( $-500 \leq g \leq 500$ ), Knochen ( $500 \leq g \leq 1700$ ).

# Gültigkeitsbereiche der Schwellwerte

- globales Schwellwertverfahren: Schwellwerte werden global für alle Pixel im Bild bestimmt. Probleme z. B. bei ungleichmäßig ausgeleuchteten Bildern (Verbesserung durch *Shading-Korrektur* möglich).
- lokales Schwellwertverfahren:
  - ☞ Aufteilung des Bildes in mehrere Teilbereiche. Falls Bereich unter gewähltem Schwellwert homogen ist (keine Bi/Multimodalität im Histogramm vorhanden): interpoliere Schwellwert aus benachbarten Regionen mit bi-/multimodalen Histogrammen
  - ☞ Verwendung lokaladaptiver Schwellwerte, z. B.  $t = t_0 - k\bar{g}$ , mit konstanter Schwelle  $t_0 \in \{0, \dots, G - 1\}$ , Gewichtungsfaktor  $k \in [0.2, 0.5]$  und mittlerem lokalem Grauwert  $\bar{g}$
- dynamisches Schwellwertverfahren: bestimme Schwellwert für jeden Pixel anhand Histogramm seiner lokalen Umgebung

# Vorverarbeitung zur Schwellwertbestimmung

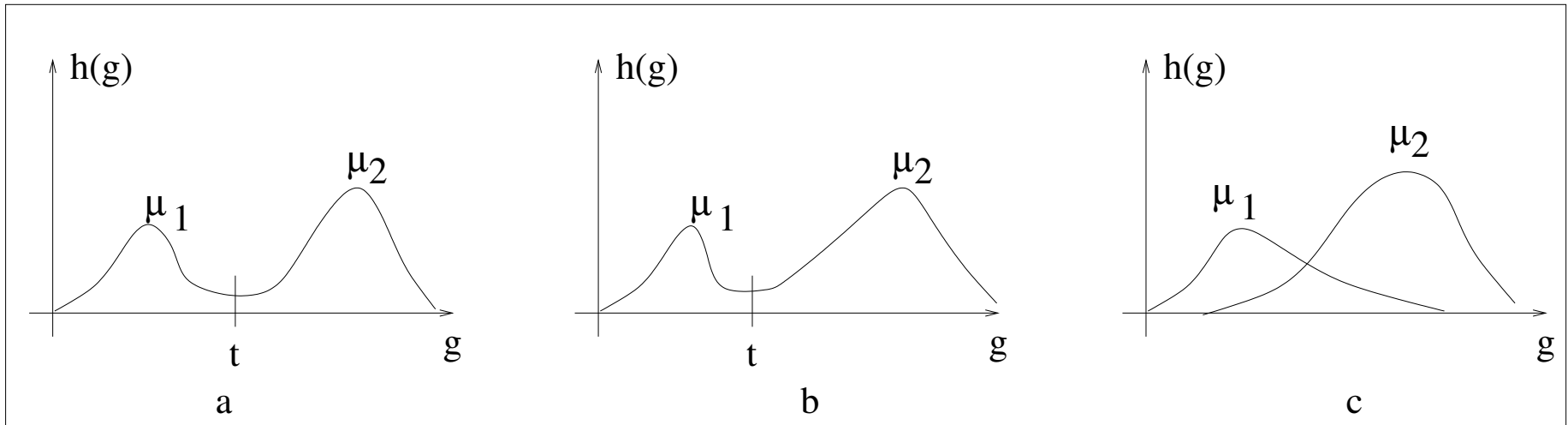


Abbildung 4: Bimodale Histogramme - Probleme bei der Festlegung eines Schwellwertes.

Seien  $\mu_i, \mu_{i+1}$  zwei benachbarte lokale Maxima im Histogramm,  $t_i$  der gesuchte Schwellwert

- Verringerung von Rauscheffekten, z. B. durch Glätten des Histogramms
- Verstärkung des Maximum/Minimum Verhältnis, z. B. durch schwächere Gewichtung

des Einflusses von Pixeln mit hohem Grauwertgradienten im Histogramm, mit

$$|\text{grad}(g(x, y))| = \sqrt{\left(\frac{\partial g}{\partial x}\right)^2 + \left(\frac{\partial g}{\partial y}\right)^2}.$$

Verschiedene Approximationen des Gradienten im Diskreten möglich, z. B.:

$$\frac{\partial g}{\partial x} = \frac{g(x + 1) - g(x - 1)}{2}, \quad \frac{\partial g}{\partial y} = \frac{g(y + 1) - g(y - 1)}{2}.$$

# Verfahren zur Schwellwertbestimmung

Formbasierte Verfahren zur Schwellwertberechnung:

- Bildung des Mittelwerts:  $t_i = \frac{\mu_i + \mu_{i+1}}{2}$
- Schwellwert bei maximaler Histogrammkrümmung

Schwellwertbestimmung mittels Optimierungsverfahren

- ohne Annahmen über Histogramm (nicht-parametrisch):
  - ☞ Verfahren nach Otsu (hier für bimodale Histogramme)
  - ☞ Entropiebasierter Ansatz (hier für bimodale Histogramme)
- mit Annahmen über Histogramm (parametrisch):
  - ☞ Approximation des Histogramms durch überlagerte Gaußfunktionen

# Verfahren nach Otsu für bimodale Histogramme

gegeben:

- normiertes Histogramm  $p(0), \dots, p(G - 1)$
- mittlerer Grauwert  $\bar{g}$  des Bildes

es gilt:

- Schwellwert  $t$  trennt Bildpixel in die beiden Klassen  $K_0$  und  $K_1$

Verfahren:

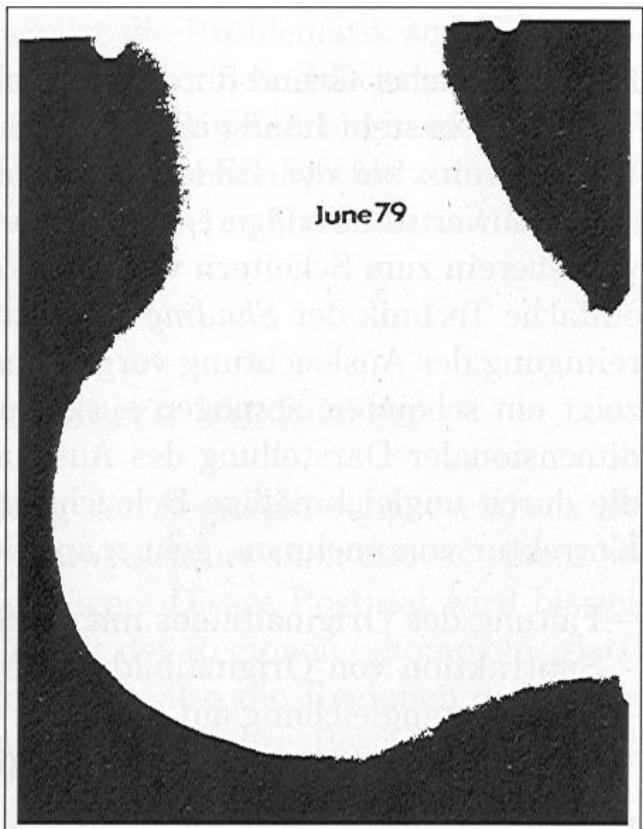
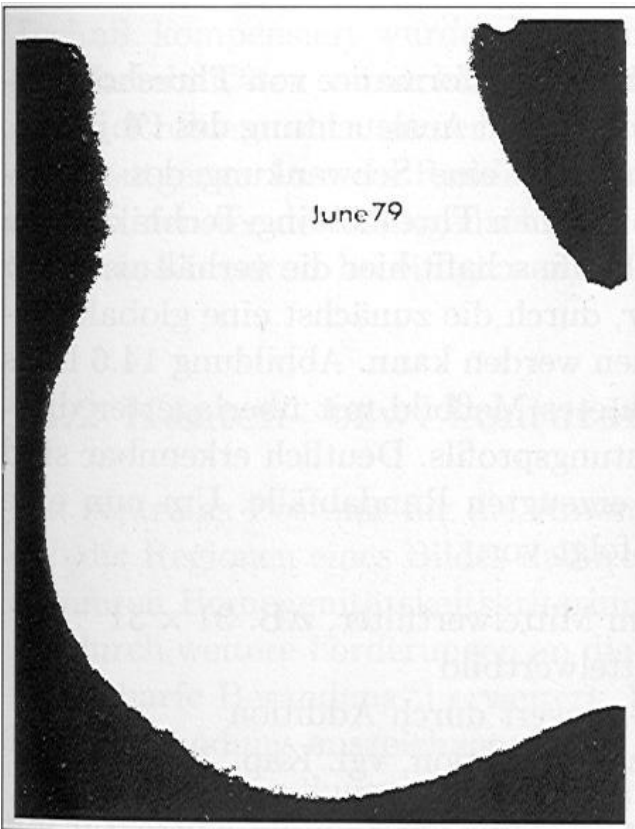
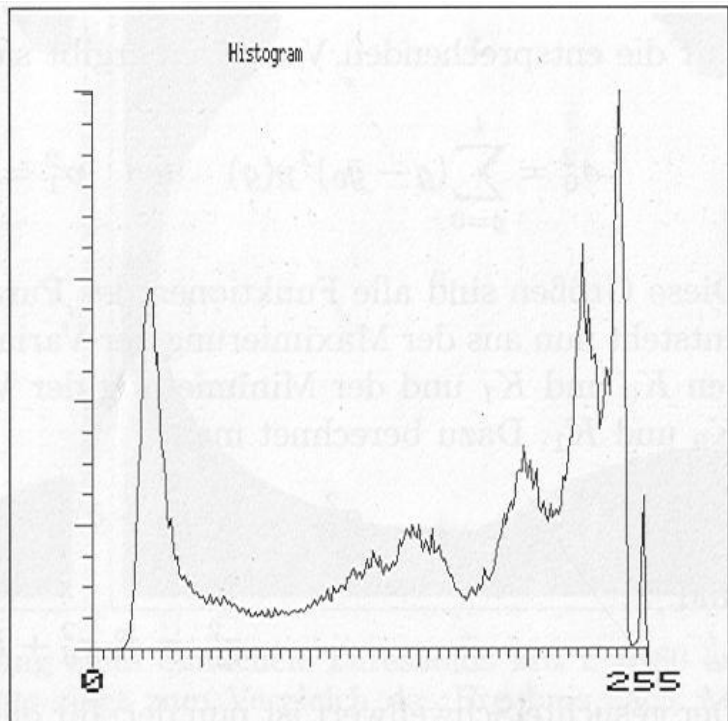
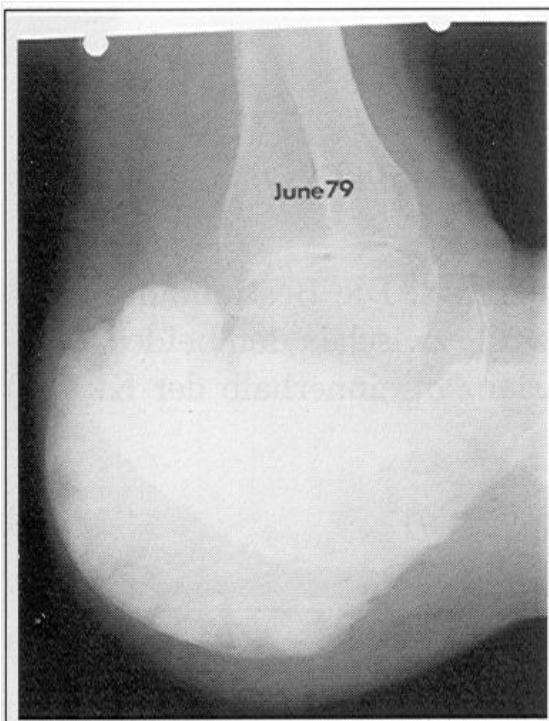
- Bestimme Mittelwerte  $\bar{g}_0, \bar{g}_1$  und Varianzen  $\sigma_0^2, \sigma_1^2$  der Klassen  $K_0$  und  $K_1$
- Bestimme Auftrittswahrscheinlichkeiten  $P_0 = \sum_{g=0}^t p(g), P_1 = \sum_{g=t+1}^{G-1} p(g)$  der beiden Klassen (es gilt:  $\bar{g} = P_0\bar{g}_0 + P_1\bar{g}_1$ )
- Maximiere  $\sigma_{zw}^2 / \sigma_{in}^2$ , mit

$$\sigma_{zw}^2 = P_0(\bar{g}_0 - \bar{g})^2 + P_1(\bar{g}_1 - \bar{g})^2 \quad (\text{Varianz zwischen den Klassen})$$

und

$$\sigma_{in}^2 = P_0\sigma_0^2 + P_1\sigma_1^2 \quad (\text{Varianz innerhalb der Klassen}).$$

# Verfahren nach Otsu



# Entropiebasiertes Verfahren für bimodale Histogramme

Einsatz zur Binarisierung komplexer Grauwertbilder

Idee:

➤ Maximiere Informationsgehalt  $H_b$  von binarisiertem Bild

$$\begin{aligned} \text{➤ } H_b &= -(P_0 \text{ ld } P_0 + P_1 \text{ ld } P_1) = \\ &- \left( \left[ \sum_{g=0}^t p(g) \right] \cdot \text{ld} \left[ \sum_{g=0}^t p(g) \right] + \left[ \sum_{g=t}^{G-1} p(g) \right] \cdot \text{ld} \left[ \sum_{g=t}^{G-1} p(g) \right] \right) \end{aligned}$$

# Approximation durch Gaußfunktionen

Approximation der realen Grauwertverteilung bei  $n$  Maxima durch eine Schar von  $n$  verschobenen und skalierten Gauß-Kurven  $q_k$ :

$$p(g) \approx \tilde{p}(g) = \sum_{k=1}^n a_k e^{(-g - \mu_k)^2 / \sigma_k^2} = \sum_{k=1}^n q_k(g)$$

mit

$$\sum_{g=0}^{G-1} |p(g) - \tilde{p}(g)| \stackrel{!}{=} \min$$

Anwendung bei multimodalen Histogrammen mit unbekanntem  $n$  schwierig. Einsatz hauptsächlich bei bimodalen Histogrammen. Bestimmung des Schwellwerts  $t$  z. B. durch Minimierung der Fehlklassifikation:

$$\frac{d}{dt_k} \left( P_0 \int_0^t q_1(g) dg + P_1 \int_t^{G-1} q_0(g) dg \right) \stackrel{!}{=} 0$$

$\Leftrightarrow$

$$\alpha t^2 + \beta t + \gamma \stackrel{!}{=} 0$$

mit

$$\begin{aligned}\alpha &= \sigma_0^2 - \sigma_1^2, \\ \beta &= 2(\mu_0\sigma_1^2 - \mu_1\sigma_0^2), \\ \gamma &= \mu_1^2\sigma_0^2 + \sigma_0^2\sigma_1^2 \ln\left(\frac{\sigma_0 P_0}{\sigma_1 P_1}\right).\end{aligned}$$

mit  $\sigma_0^2 = \sigma_1^2 = \sigma^2$  gilt:

$$t = \frac{\mu_0 + \mu_1}{2} + \frac{\sigma^2}{\mu_0 - \mu_1} \ln \frac{P_0}{P_1}. \quad (1)$$

Minimierung des Histogrammdifferenz:

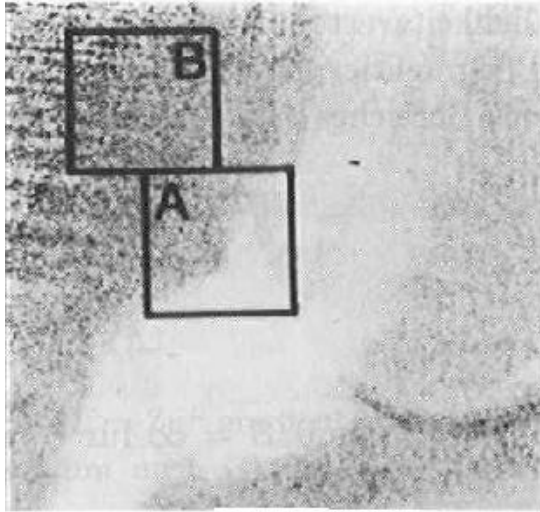
➤ kleinste Quadrate

$$\sum_{g=0}^{G-1} (p(g) - \tilde{p}(g))^2 \stackrel{!}{=} \min,$$

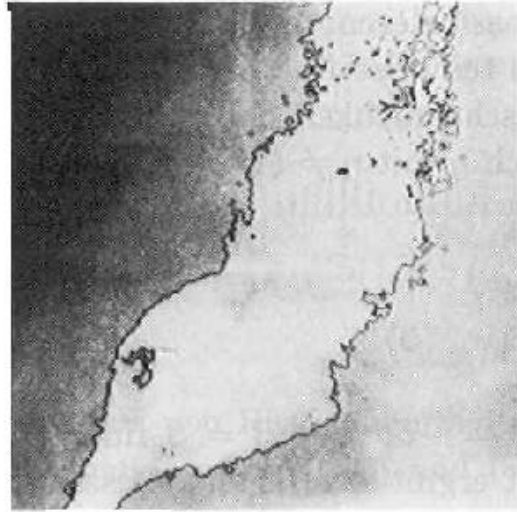
➤ Maximum Likelihood Schätzung mit *Expectation Maximization* Verfahren

Verfahren mehrdimensional für bekannte Anzahl  $n$  erweiterbar

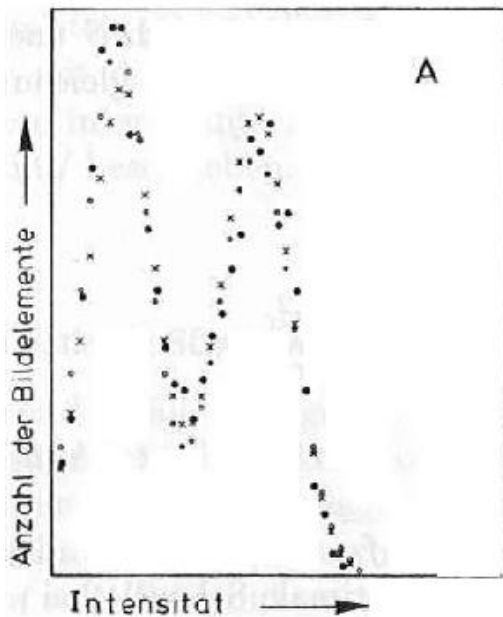
# Anwendungsbeispiel



(a)

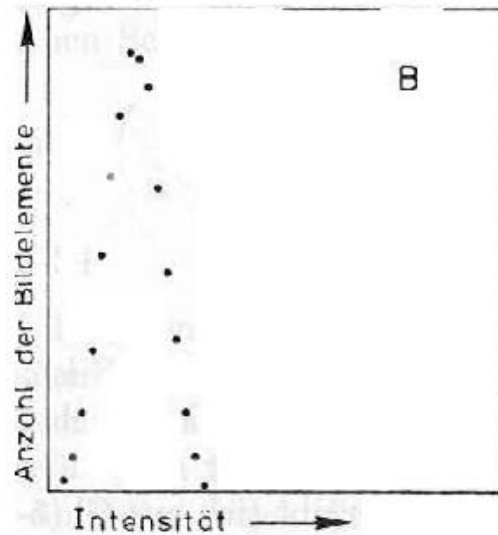


(b)



(c)

Quelle: F.M. Wahl:  
Digitale Bildsignalverarbeitung



(d)

Abbildung 5: Segmentierung des Herzbinnenraums.

## Anwendungsbeispiel

- Eingangsbild  $256 \times 256$  Pixel
- Zerteilung in  $7 \times 7$  Unterbereiche mit jeweils  $64 \times 64$  Bildpunkten
- dadurch 50%-ige Überlappung benachbarter Bereiche
- Histogramm für jeden dieser 49 Bereiche
- Überprüfung auf Bimodalität
- Bestimme optimale Schwellwert für bimodale Bereiche
- Bestimme Schwellwert unimodaler Bereiche durch Interpolation
- Bilde für gesamtes Bild lokale Schwellwertfunktion  $t(x, y)$  und wende sie auf das Ausgangsbild an

# Kantenbasierte Segmentierung

Annahme: Objekt ist durch zusammenhängende Kante vom Hintergrund getrennt gesucht:

1. Kantenpixel (evtl. Wahrscheinlichkeit)
2. Zusammenhang von Kantenpixeln

einige wichtige Kantenoperatoren:

Marr-Hildreth (= LoG), Roberts, Sobel, Prewitt, Canny

Konturverfolgungsalgorithmen:

- Pixelorientiert (parallel)
  - ☞ Operatorbasiert: Bestimme für jedes Pixel Maßzahl anhand Gradient. Evtl. anschließende Skelettierung, bzw. Anwendung morphologischer Operatoren notwendig
  - ☞ Kantenrelaxation (iterativ): Jedes Pixel bekommt Wahrscheinlichkeit zugeordnet (z. B. Pixel liegt auf Kante / liegt nicht auf Kante). Benachbarte Wahrscheinlichkeiten beeinflussen sich gegenseitig
- Linienorientiert (sequentiell)

# Linienorientierte Kantenextraktion (Linienverfolgen)

Probleme:

- Finde geeigneten Startpunkt
  - ☞ manuell (interaktiv)
  - ☞ automatisch
- Finde Nachfolgepunkte

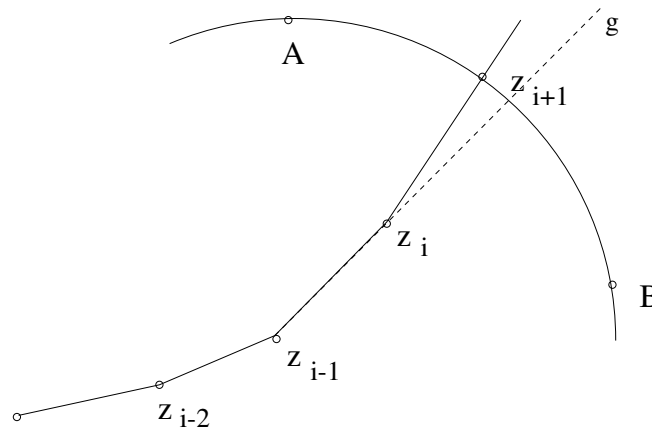


Abbildung 6: Typische Situationen bei der Linienverfolgung:  
Sich kreuzende, parallele unterbrochene, sich berührende Linien

# Linienextraktion: Bestimmung von Nachfolgepixeln

## Suchstrahlverfahren:

- $z_{i-1}, z_i$  bilden Gerade  $g$
- bestimme Kreissegment um  $g$  (maximaler Winkelbereich bis zu Punkten  $A$  bzw.  $B$ )
- für alle Pixel  $k$  auf Kreissegment:
  - ☞ bilde Verbindungslinie  $\updownarrow(k, z_i)$
  - ☞ bestimme mittleren Grauwert auf dieser Linie
- Schnittpunkt von Linie mit optimalem mittleren Grauwert (z. B. Minimum bei dunkler Kante auf hellem Hintergrund) und Kreissegment bestimmt nächsten Konturpunkt  $z_{i+1}$



# Konturverfolgung

gegeben:

- Kantenstärke  $\kappa(x, y)$  (Gradientenbetrag)
- Kantenorientierung  $\phi(x, y)$

gesucht: Pixel  $z_{i+1}$  in Nachbarschaft von  $z_i$  für den gilt:

$$\begin{aligned} |\kappa(z_{i+1}) - \kappa(z_i)| &< T_1, \\ |\phi(z_{i+1}) - \phi(z_i)| \bmod 2\pi &< T_2, \\ |\kappa(z_{i+1})| &> T, \\ |\kappa(z_i)| &> T. \end{aligned}$$

$T_1, T_2, T$  festgelegte Schwellwerte

- bei mehreren Kandidaten, wähle optimalen
- Algorithmusende, falls kein Kandidat in Nachbarschaft
- robuster: *Graphensuche* oder *dynamische Programmierung*

# Graphensuche

Definiere:

- Pixelkosten (z. B. durch Kantenoperator)
- Übergangskosten (z. B. Pixeldistanz, Differenz des Gradientenbetrags, Differenz der Kantenorientierung)

Bestimme optimalen Pfad (Pfad mit geringsten Kosten) durch Graphen. Dafür existieren verschiedene Algorithmen. Weitverbreitet ist der  $A^*$ -Algorithmus.

## $A^*$ -Algorithmus (nach Winston, *Artificial Intelligence*)

Die Kosten auf einem Pfad  $z_1, \dots, z_n$  seien  $C(n) = \sum_{i=2}^n c(z_{i-1}, z_i) + \sum_{i=1}^n c(z_i)$ . Die

Gesamtkosten für die geschlossene Kontur werden mittels einer unteren Schranke  $\tilde{C}(n)$  abgeschätzt:  $C_g(n) = C(n) + \tilde{C}(n)$

1. Initialisierung: Wähle Startknoten (-pixel)  $z_i$  und notiere alle Nachfolger in einer Liste  $\mathcal{L}_o$  der *offenen* Kanten jeweils mit Zeiger auf  $z_i$ . Seien nun  $n$  Knoten im Pfad gewählt
2. Wenn  $\mathcal{L}_o = \emptyset$  stoppe, andernfalls fahre fort

3. Wähle aus  $\mathcal{L}_o$  den Knoten  $z_n$  mit kleinsten Kosten  $C_g(n)$  und verschiebe ihn auf die Liste  $\mathcal{L}_c$  der geschlossenen Kanten
4. Wenn  $n$  der Endknoten ist, verfolge den Pfad zurück bis zum Anfangsknoten, gib diesen optimalen Pfad aus und stoppe
5.  $n \leftarrow n + 1$
6. Erzeuge alle Nachfolger
7. Wenn ein Nachfolger  $z_n \notin \mathcal{L}_o \cup \mathcal{L}_c$  setze  $C(n) = C(n - 1) + c(z_{n-1}, z_n) + c(z_n)$ , füge  $z_n$  der Liste  $\mathcal{L}_o$  hinzu mit einem Zeiger auf  $z_{n-1}$
8. Wenn ein Nachfolger  $\in \mathcal{L}_o \cup \mathcal{L}_c$  wähle das Minimum  $C(z_i) = \min\{C(z_i), C(n - 1) + c(z_{n-1}, z_n) + c(z_n)\}$ . Aktualisiere die Markierung des Knoten (geöffnet) wenn nötig und verzeigere neu. Gehe weiter zu Schritt 2

Algorithmus führt dann zu optimalem Pfad, wenn  $\tilde{C}(n)$  eine korrekte untere Schranke vom Knoten  $z_n$  zum Endknoten ist.

# Dynamische Programmierung

In jedem Schritt wird abhängig von einem Ausgangszustand eine optimaler Folgezustand gesucht. Beispiel: Kante im 2D-Bild, die von oben nach unten verläuft. Zeilenbasierte Suche

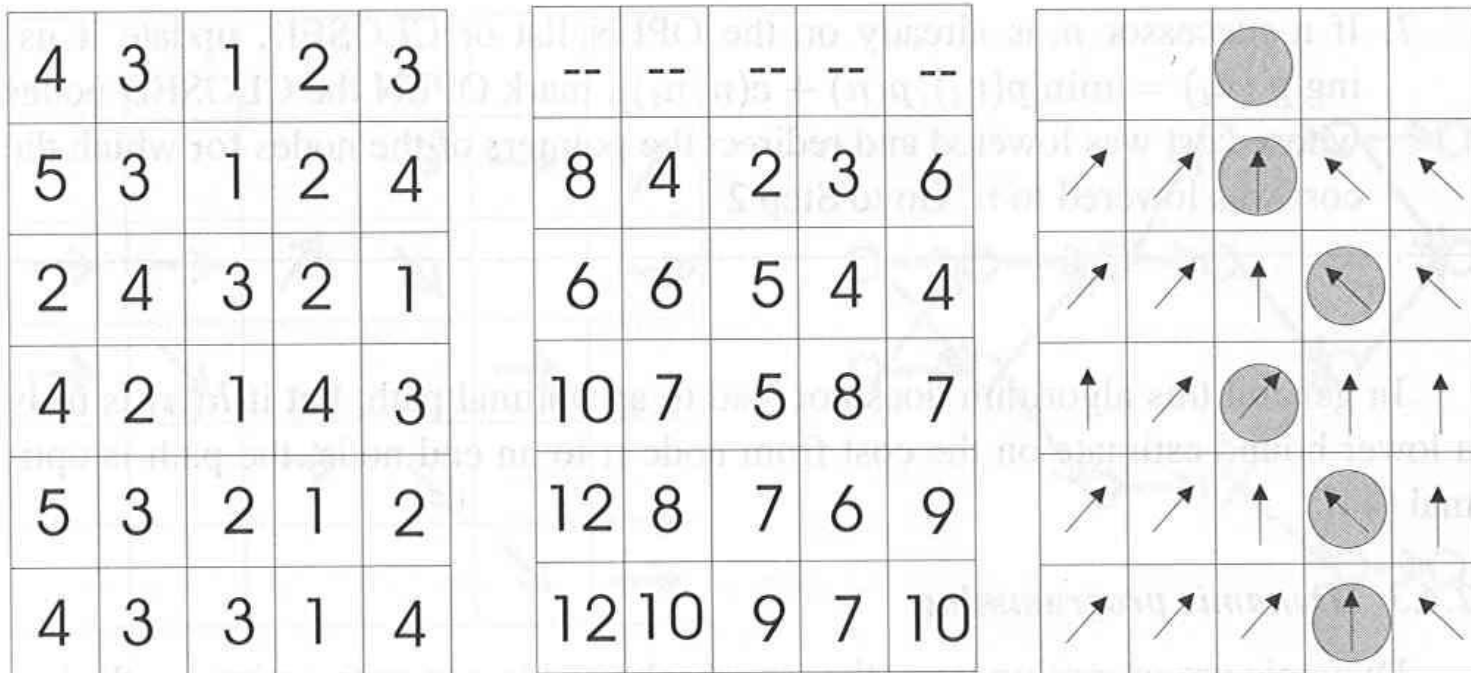


Abbildung 7: Kostenfunktion, akkumulierte optimale Kosten, Richtungsvektoren

# Dynamische Programmierung: Beispiel

Durch geometrische Transformation kann beschriebenes Verfahren auch auf geschlossene Konturen angewandt werden.

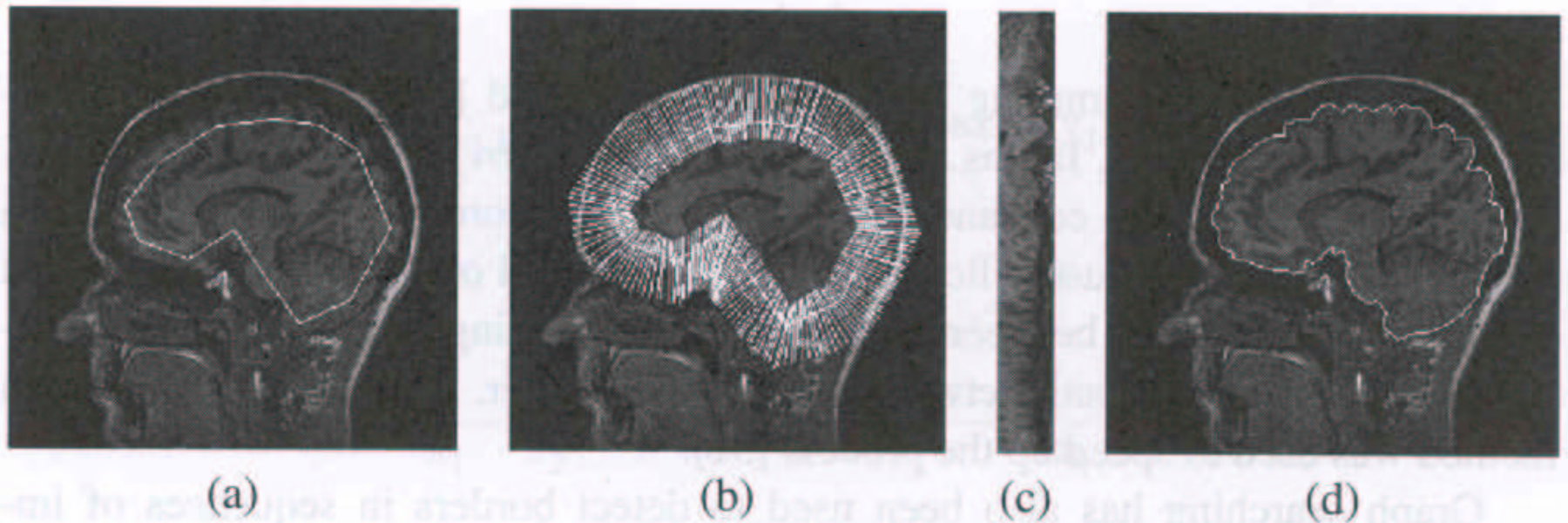


Abbildung 8: Geometrische Transformation: a) Initialkontur b) Linien senkrecht zu Initialkontur c) aus b transformierte 1D-Kante d) optimale Kante rücktransformiert

# Dynamische Programmierung

- i. A. sehr zeitintensiv
- Effizienzsteigerung:
  - ☞ Regions of interest
  - ☞ semi-automatische Algorithmen (z. B. *live-wire*)
  - ☞ suboptimale Verfahren
- Implementierungen in C:  
Pitas, I. *Digital Image Processing Algorithms*. Prentice-Hall, 1993

# Hough-Transformation

- Detektion von parametrisierten Kurven (z. B. Kreise, Geraden, Ellipsen, Kugeln, Ellipsoide)
- Transformation eines Binärbildes (z. B. Schwellwertbild, Schwellwert auf Gradientenbild) in Parameterraum
- Problem wird auf Suche eines Maximums im Parameterraum reduziert
- Hough-Transformation zur Geradendetektion entspricht Radontransformation des binarisierten Bildes
- unempfindlich gegenüber Löchern in Kanten
- unempfindlich gegenüber Rauschen

# Hough-Transformation zur Geradendetektion

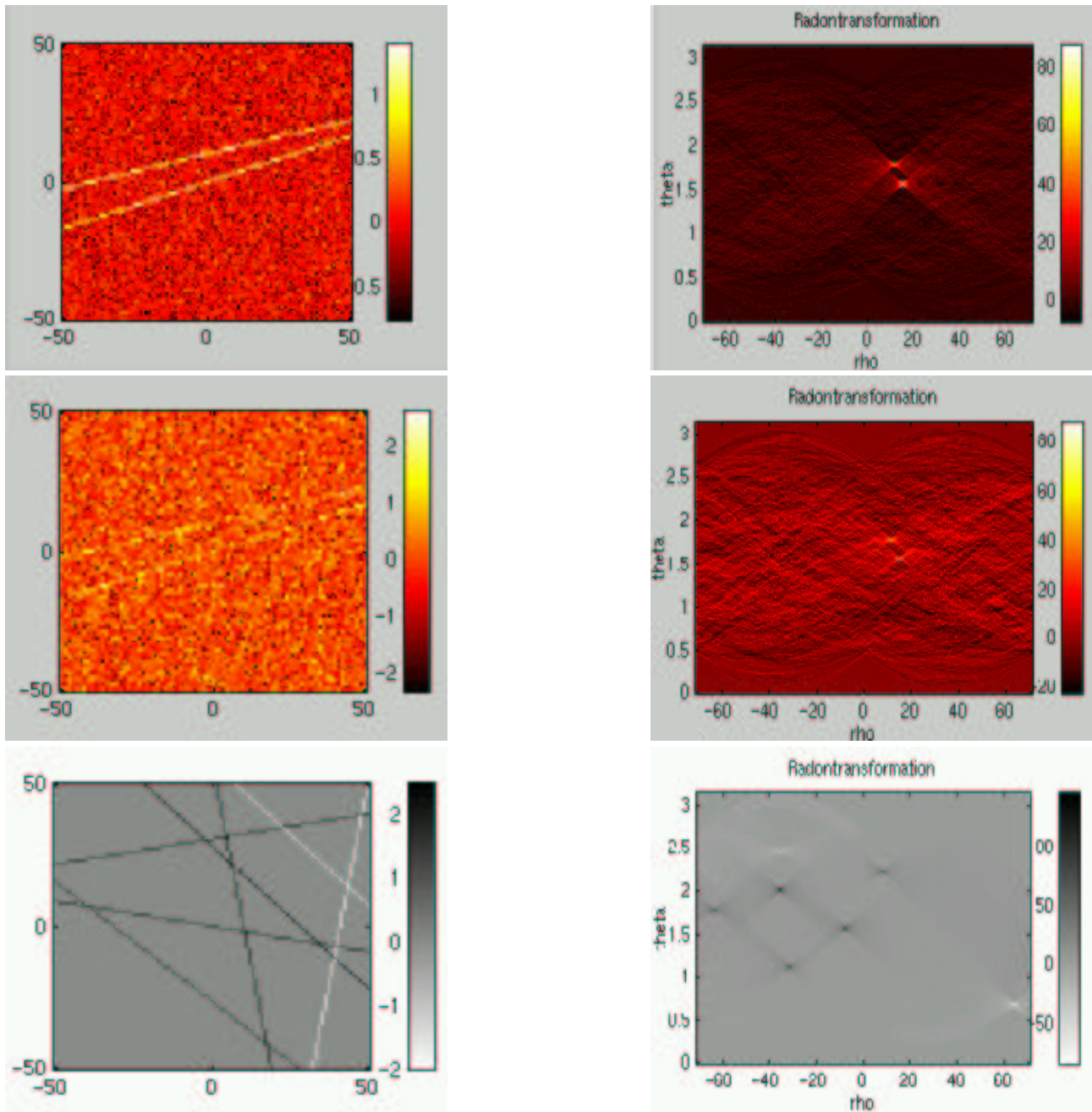


Abbildung 9: Quelle: Homepage P.Toft

# Hough-Transformation zur Kreisdetektion

- Parametrisierung eines Kreises  $k$  mittels  $(x - a)^2 + (y - b)^2 = r^2$
- Reduktion des 3D-Parameterraums durch Vorwissen:
  - ☞ Kenntnis über den ungefähren Radius
  - ☞ vorheriges Kantenfilter mit Kantenorientierungsinformation hilft zur Abschätzung des Radius

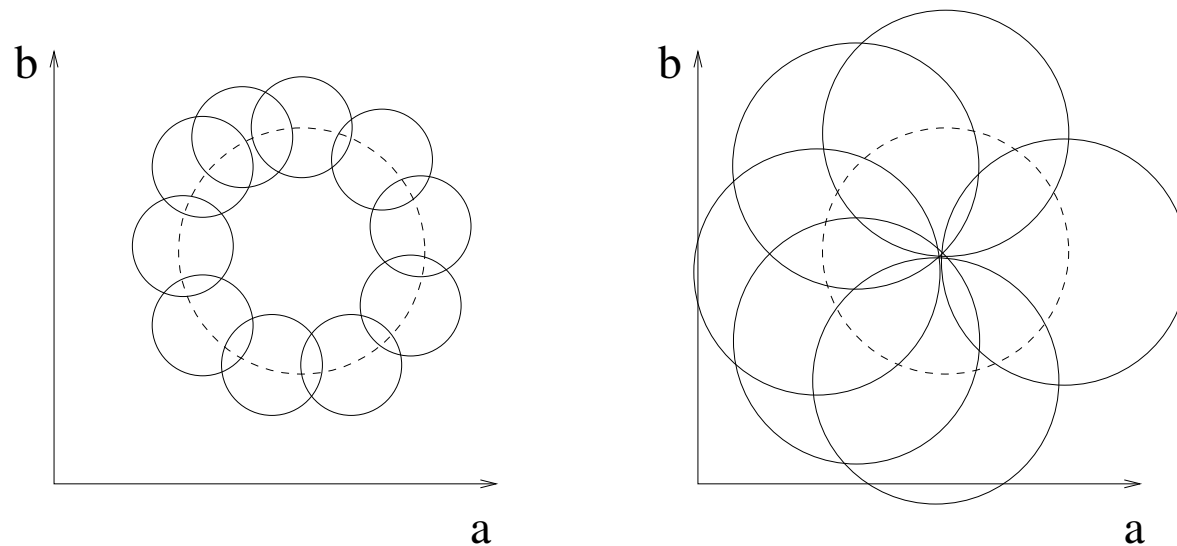


Abbildung 10: Hough-Transformation zur Kreisdetektion

# Regionenorientierte Verfahren

gegeben:

- 2 Regionen  $A, B$  (z. B. jew. 1 Pixel)
- Distanzmaß

$$d(r_1, r_2)$$

(z. B.  $|g_1 - g_2|$ )

gesucht: Verschmelzung beider Regionen unter gewissen Bedingungen

Verbindungs-Strategien:

- Single-Linkage:  $\exists p_A \in A, p_B \in B : d(p_A, p_B) \leq t, p_A, p_B$  benachbart
- Contiguity-Constraint-Complete-Linkage: wie *Single-Linkage*, jedoch alle benachbarten Pixelpaare müssen Kriterium erfüllen
- Contiguity-Constraint-Average-Linkage: Mittelwert aller benachbarten Pixelpaare muss Bedingung erfüllen
- Centroid-Linkage: Differenz der Grauwertmittelwerte der Regionen darf nicht zu groß sein

- Complete-Linkage: zusätzlich zu *Centroid-Linkage* darf die maximale Grauwertdifferenz nicht zu groß sein

Bemerkung: da die Reihenfolge der Verschmelzungen einen erheblichen Einfluss auf das Ergebnis hat, empfiehlt sich eine Optimierung unter globaler Betrachtung aller möglichen Verschmelzungen

# Agglomerative Verfahren

## Regionbildung durch Wachstumsprozess

Bereichswachstumsverfahren (region growing):

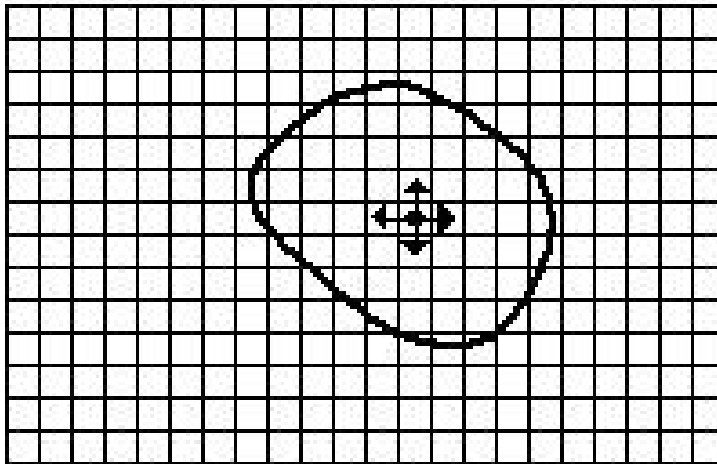
gegeben: geeignetes Distanzmaß  $d$ , Schwelle  $t$

1. Initialisiere aktuellen Pixel  $P$  in Region (beispielsweise manuell)
2. Wenn Kriterium erfüllt und  $P$  noch nicht markiert ist:
  - markiere Pixel
  - für alle Nachbapixel:
    - (a)  $P :=$  aktueller Pixel
    - (b) gehe zu Punkt 2ansonsten stoppe

Probleme:

- „auslaufen“, wenn Rand nicht geschlossen ist
- Ergebnis von Nachbarschaftsbeziehung abhängig
- Hohe Rekursionstiefe bei Standardalgorithmus

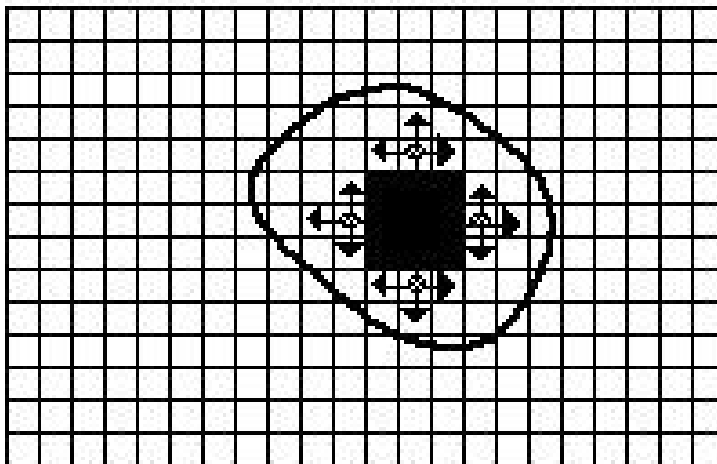
# Region Growing



• Seed Pixel

↑ Direction of Growth

(a) Start of Growing a Region



■ Grown Pixels

○ Pixels Being Considered

(b) Growing Process After a Few Iterations

# Divisive Verfahren

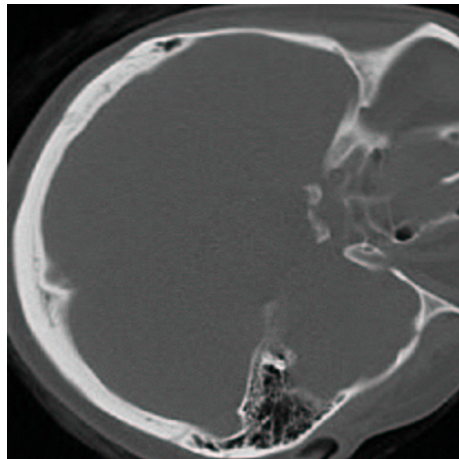
## Regionenbildung durch Zerteilungsprozess

Split (& Merge):

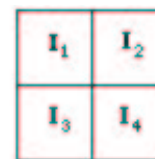
Betrachte zuerst gesamtes Bild als Initialpartition

1. Überprüfe, ob Partition homogen ist (z. B. statistische Merkmale). Wenn ja, stoppe
2. Zerlege die Partition in 4 gleich große Rechtecke.
3. Für alle Partitionen gehe zu Schritt 1

Im Mergingschritt werden benachbarte Partitionen verschmolzen, falls sie homogen sind



(a) Whole Image



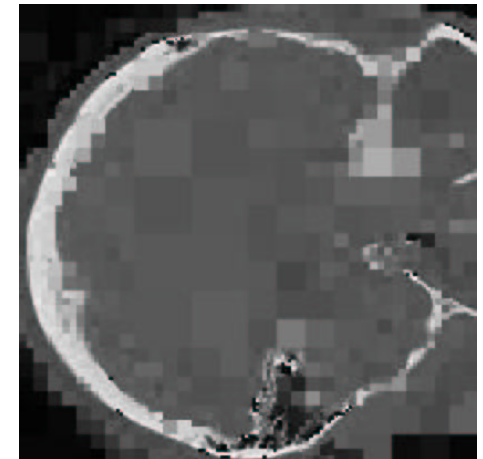
(b) First Split



(c) Second Split



(d) Merge



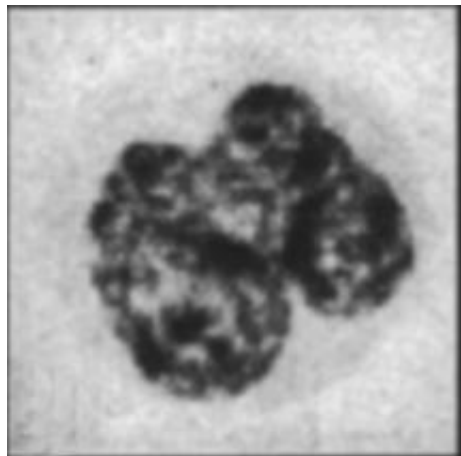
# Pyramid-Linking

gegeben: Gauß-Pyramide des Bildes (Bild in unterschiedlichen Auflösungsstufen, jeweils geglättet und unterabgetastet)

Iteriere nun folgende Schritte bis Stabilisierung erfolgt:

- Segmentierungsschritt: Verbinde Knoten mit „verwandten“ Knoten nächsthöherer Ebene und minimaler Distanz
- Merkmalsneuberechnung: Bilde erneut arithmetisches Mittel anhand neuer Struktur

Bemerkung: Die Konvergenz des Verfahrens kann nicht gezeigt werden. Abbruch sollte nach Konvergenz oder einer maximalen Anzahl von Schritten erfolgen



Pyramid-Linking bei vorauschten Bildern  
Quelle: Jähne, *Digitale Bildverarbeitung*



# Pyramid-Linking

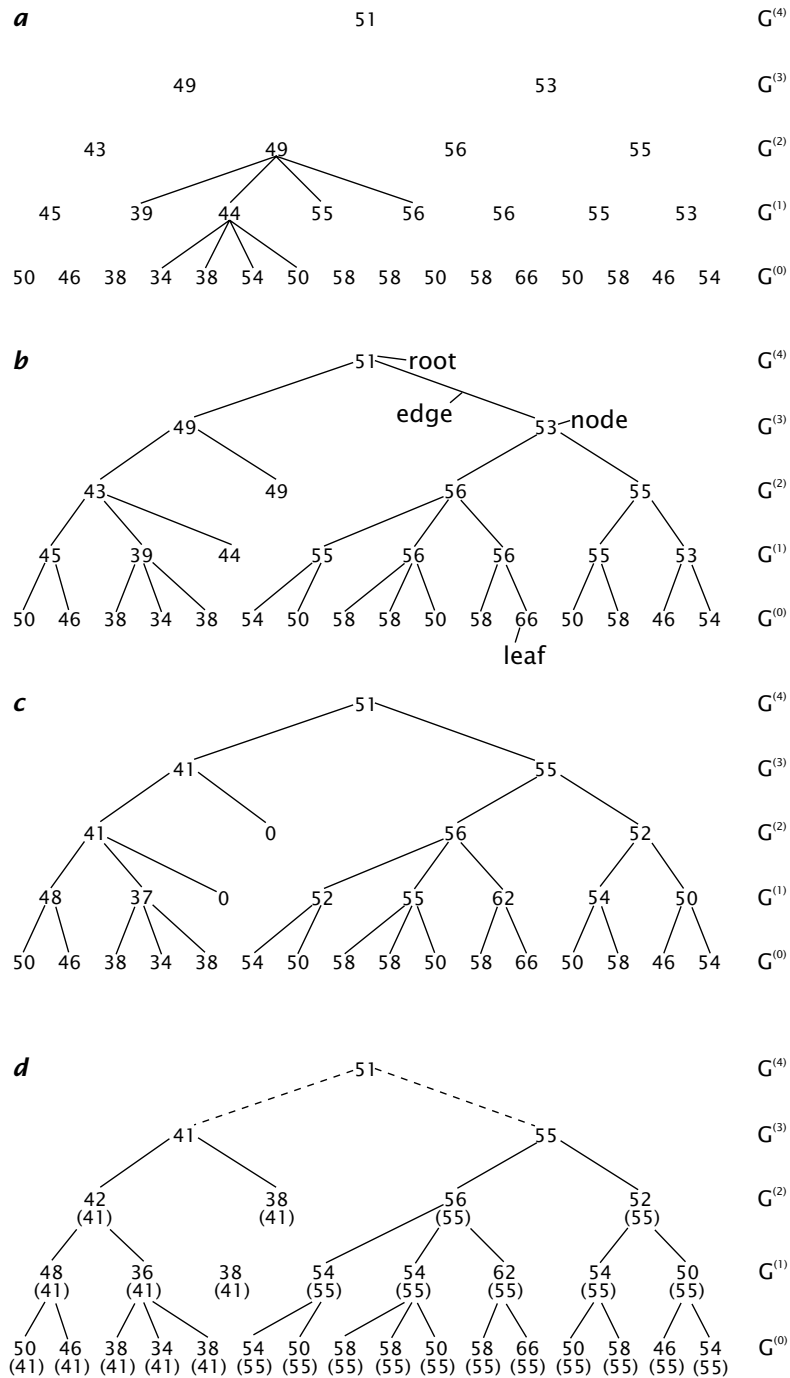


Abbildung 11: Quelle: Jähne, *Digitale Bildverarbeitung*

# Texturbasierte Segmentierung

## Texturbegriff

- Wiederkehrendes Muster
- Gerichtetheit
- Periodizität

## Texturberechnung mittels Haralick'scher Texturmerkmale:

Cooccurrence-Matrix:

vorgegeben: geometrische Anordnung  $\delta$ . Ein Eintrag  $C_\delta(k, l)$  in der *Cooccurrence-Matrix*

$$C_\delta := \begin{bmatrix} C_\delta(0, 0) & \dots & C_\delta(0, g - 1) \\ \vdots & \ddots & \vdots \\ C_\delta(g - 1, 0) & \dots & C_\delta(g - 1, g - 1) \end{bmatrix}$$

ergibt sich durch die Häufigkeit des Auftretens von Pixelpaaren  $(\mathbf{p}, \mathbf{q})$  in der durch  $\delta$  definierten geometrischen Anordnung an, für die gilt

$$(f(\mathbf{p}) = k \wedge f(\mathbf{q}) = l) \quad \vee \quad (f(\mathbf{p}) = l \wedge f(\mathbf{q}) = k).$$

normierte Coocurrance Matrix:

$$P_{\delta} = \frac{1}{\sum_{k=0}^{g-1} \sum_{l=0}^{g-1} C_{\delta}(k, l)} \cdot C_{\delta}.$$

Texturparameter:

➤ **Kontrast**

$$\text{CON}_{\delta} := \sum_{i=0}^{g-1} \sum_{j=0}^{g-1} (i - j)^2 P_{\delta}(i, j)$$

➤ **Entropie**

$$\text{ENT}_{\delta} := - \sum_{i=0}^{g-1} \sum_{j=0}^{g-1} P_{\delta}(i, j) \log(P_{\delta}(i, j))$$

➤ **Inverse Difference Moment oder Local Homogeneity**

$$\text{IDM}_{\delta} := \sum_{i=0}^{g-1} \sum_{j=0}^{g-1} \frac{P_{\delta}(i, j)}{1 + (i - j)^2}$$

## Weitere Segmentierungsverfahren

- Elastizitätsmodelle (Feder-Masse-Modelle)
- Markov Random Fields (Gibbs-Verteilung)
- Diffusionsgleichungen
- Aktive Konturen (Snakes)

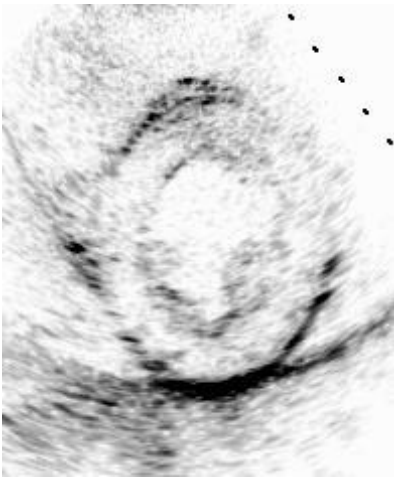


Abbildung 12: Segmentierung von Herz-US Aufnahmen

Quelle: <http://icsl.ee.washington.edu/vikram/research.html>

# Snakes (statische Kräfte)

- Initialkontur

$$\xi(s) = (x(s), y(s)), \quad s \in [0, 1]$$

- Energieminimierung:

$$\mathcal{E}(\xi) = I(\xi) + P(\xi)$$

- Innere Energie (Kurven - inhärent):

$$I(\xi) = \frac{1}{2} \int_0^1 \alpha(s) \left| \frac{\partial \xi}{\partial s} \right|^2 + \beta(s) \left| \frac{\partial^2 \xi}{\partial^2 s} \right| ds$$

- Potential:

$$P(\xi) = \int_0^1 P(\xi(s)) ds$$

- typisches Potential (Gradientenbild):

$$P(x, y) = -w |\nabla [G_\sigma(x, y) * I(x, y)]|^2$$

- $\sigma$  vergrößert den Anziehungsbereich

- Optimierung durch *Euler-Lagrange-Gleichung*

$$\frac{\partial}{\partial s} \left( \alpha \frac{\partial \xi}{\partial s} \right) - \frac{\partial^2}{\partial s^2} \left( \beta \frac{\partial^2 \xi}{\partial s^2} \right) - \nabla P(\xi) = 0$$

- „Kräftegleichgewicht“:

$$F_{\text{int}}(\xi) + F_{\text{pot}}(\xi t) = 0$$

- „innere Kraft“ (Stetigkeit und Glattheit):

$$F_{\text{int}}(\xi) = \frac{\partial}{\partial s} \left( \alpha \frac{\partial \xi}{\partial s} \right) - \frac{\partial^2}{\partial s^2} \left( \beta \frac{\partial^2 \xi}{\partial s^2} \right)$$

- $F_{\text{pot}}(\xi) = -\nabla P(\xi)$  zieht die Kurve  $\xi$  an die Kontur

- Lösung der Euler-Lagrange Gleichung durch zeitliche Veränderung der Kurve  $\xi(s, t)$ :

$$\gamma \frac{\partial \xi}{\partial t} = \frac{\partial}{\partial s} \left( \alpha \frac{\partial \xi}{\partial s} \right) - \frac{\partial^2}{\partial s^2} \left( \beta \frac{\partial^2 \xi}{\partial s^2} \right) - \nabla P(\xi)$$

Die linke Seite verschwindet bei Stabilisierung der Lösung Kurve und man erhält die Lösung der Euler-Lagrange Gleichung

# Snakes (dynamische Kräfte)

- Modellierung durch dynamische Kräfte
- Nach Newton gilt für  $\xi(s, t)$ :

$$\mu \frac{\partial^2 \xi}{\partial t^2} = F_{\text{dämpf}}(\xi) + F_{\text{int}}(\xi) + F_{\text{ext}}(\xi),$$

$\mu$  (Masse),

$$F_{\text{dämpf}}(\xi) = -\gamma \frac{\partial \xi}{\partial t} \quad (\text{Viskosität})$$

- In der Bildverarbeitung oft:  $\mu = 0$  (erhöht Empfindlichkeit gegenüber schwachen Ecken)
- resultierende Gleichung:

$$\gamma \frac{\partial \xi}{\partial t} = F_{\text{int}}(\xi) + F_{\text{ext}}(\xi)$$

- interne Kräfte wie oben
- externe Kräfte

$$F_{\text{ext}}(\xi) = F_1(\xi) + F_2(\xi) + \dots + F_n(\xi)$$

- ☞ Potential
- ☞ Druck
- ☞ Distanz
- ☞ interaktive Kräfte
- Implementierung
  - ☞ Greedy
  - ☞ dynamische Programmierung (Graphenoptimierung)
- Erweiterungen
  - ☞ Ballonmodelle
  - ☞ aktive Oberflächen
  - ☞ topologically adaptable snakes (*T-Snakes*)