

Computergrafik 2: Segmentierung 2

Prof. Dr. Michael Rohs, Dipl.-Inform. Sven Kratz

michael.rohs@ifi.lmu.de

MHCI Lab, LMU München

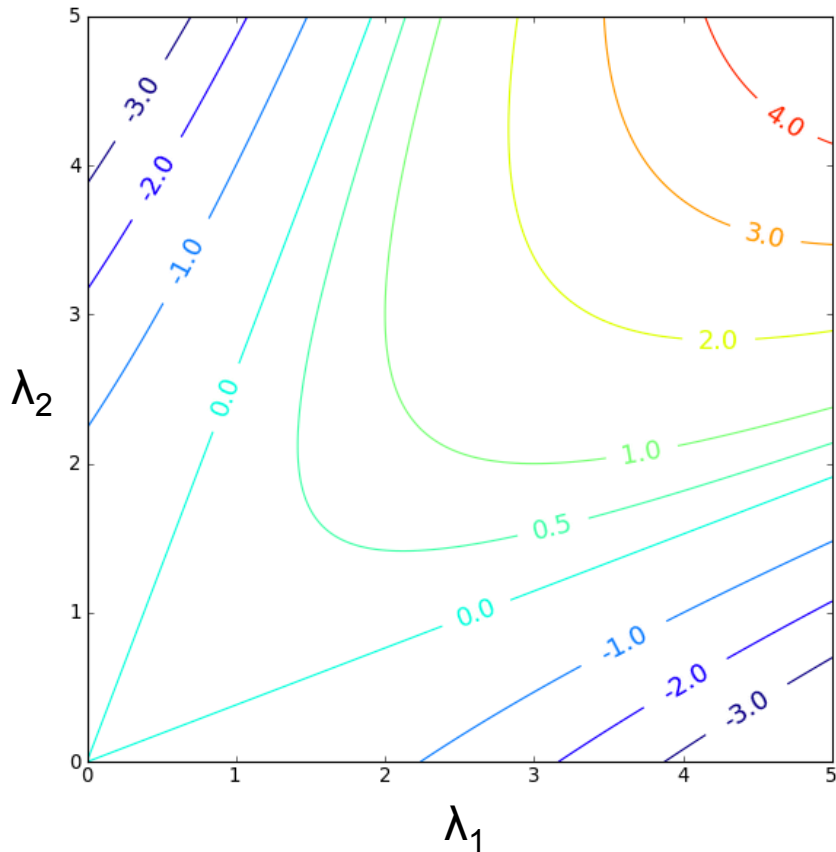
Folien teilweise von Andreas Butz, sowie von Klaus D. Tönnies
(Grundlagen der Bildverarbeitung. Pearson Studium, 2005)

Themen heute

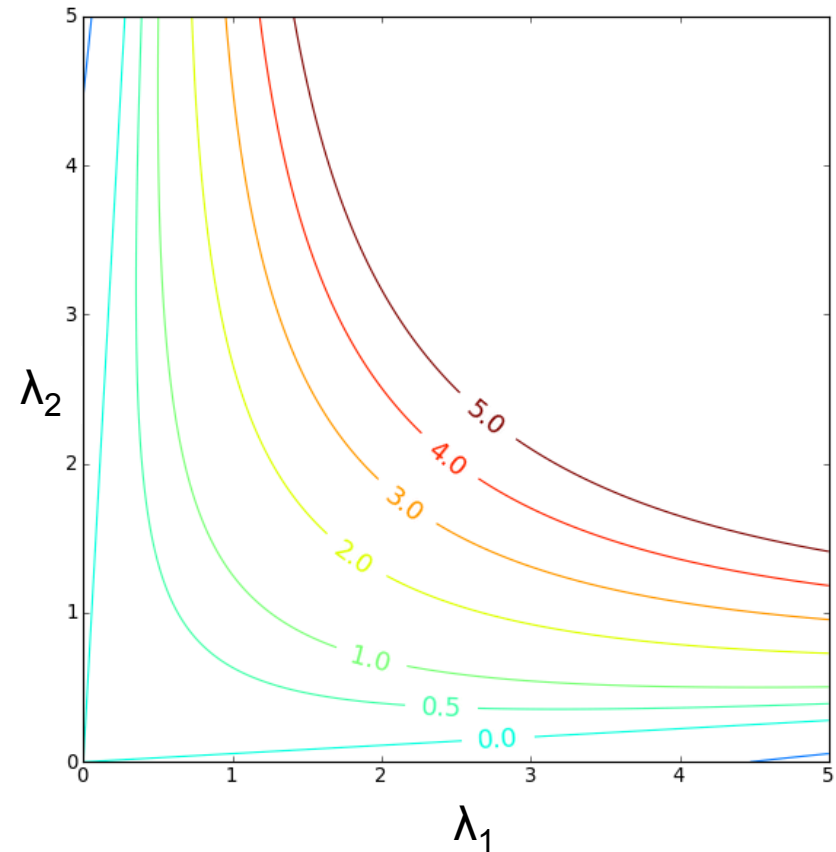
- Ecken (Nachtrag)
- Schwellenwertbasierte Segmentierung
 - Verbesserung durch Kanten
 - Variable Schwellenwerte
- Kantenbasierte Segmentierung
 - Wasserscheidentransformation
 - Verbesserung der Wasserscheidentransformation durch Marker

Harris Corner Detektor: Parameter “k”

k = 0.20

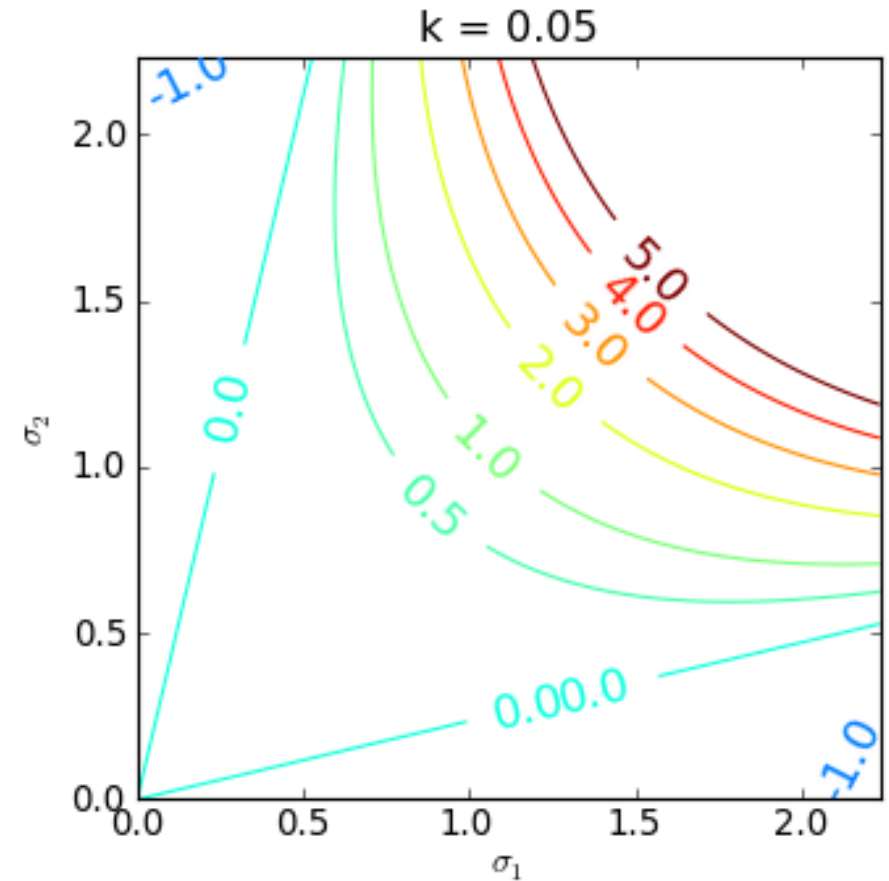
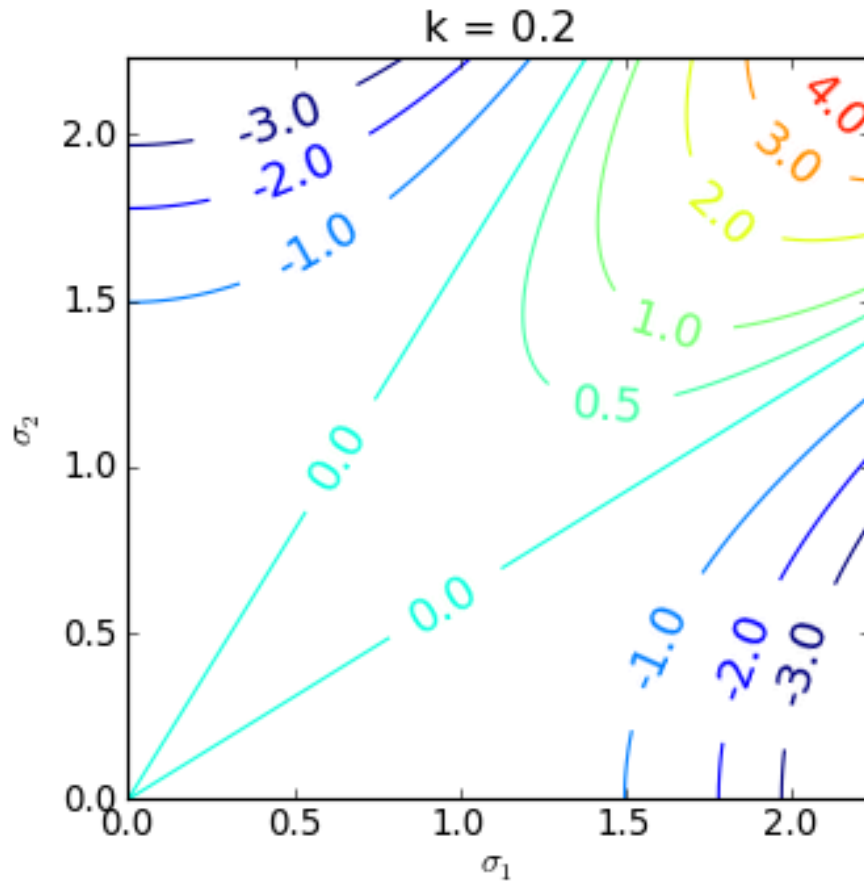


k = 0.05



- $C = \det(M) - k \text{trace}(M)^2 = \lambda_1 \lambda_2 - k (\lambda_1 + \lambda_2)$

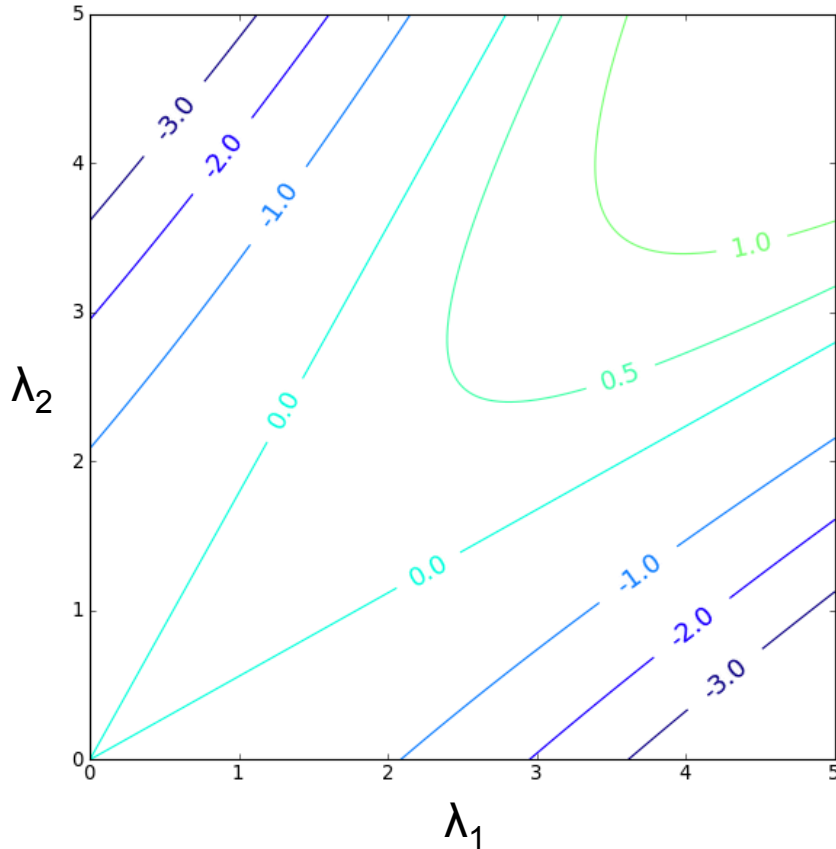
Harris Corner Detektor: Parameter “k”



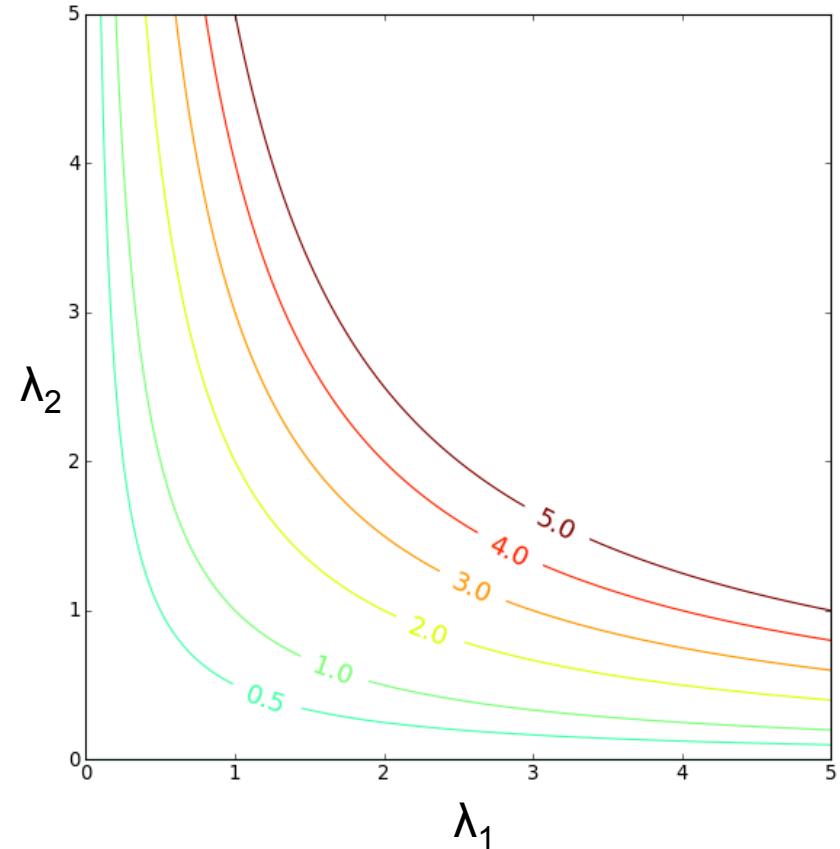
- $C = \det(M) - k \text{trace}(M)^2 = \lambda_1 \lambda_2 - k (\lambda_1 + \lambda_2)$

Harris Corner Detektor: Parameter “k”

k = 0.23

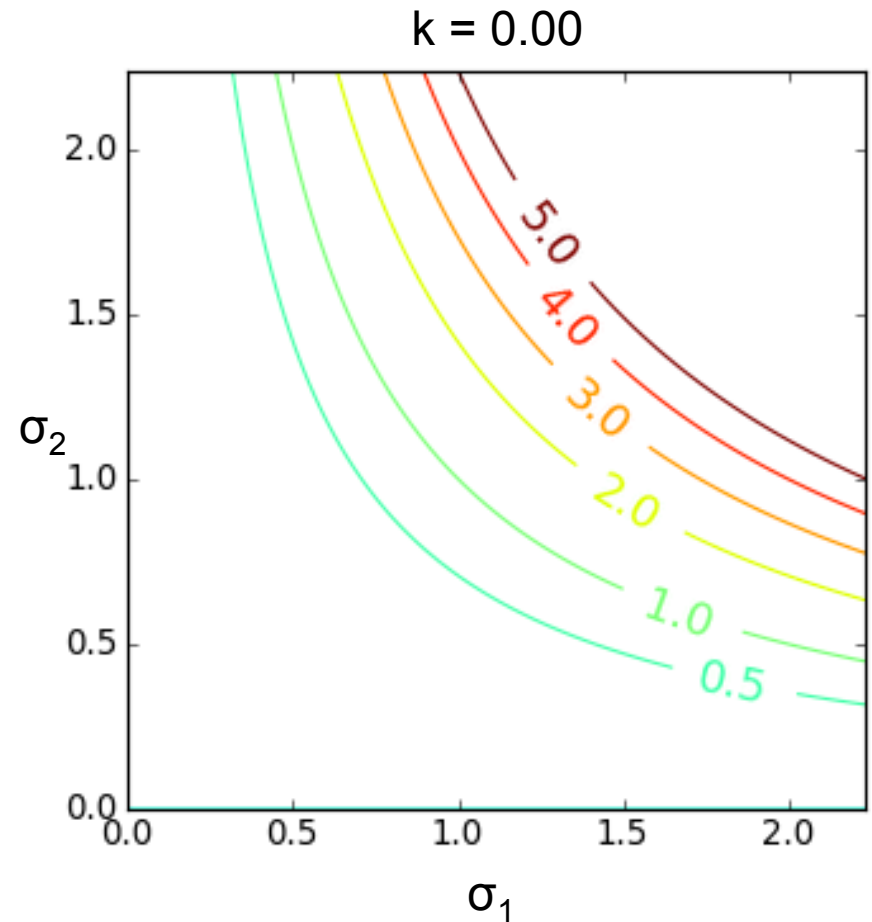
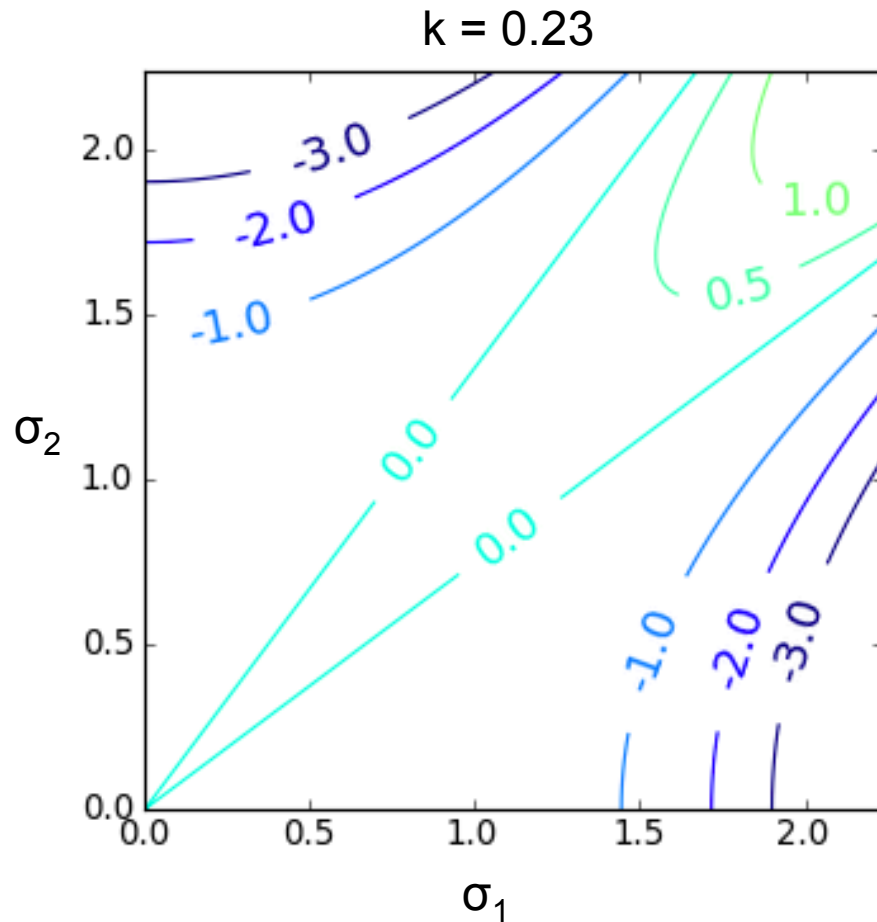


k = 0.00



- $C = \det(M) - k \text{trace}(M)^2 = \lambda_1 \lambda_2 - k (\lambda_1 + \lambda_2)$

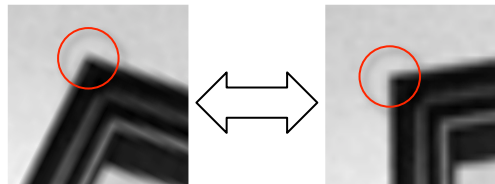
Harris Corner Detektor: Parameter “k”



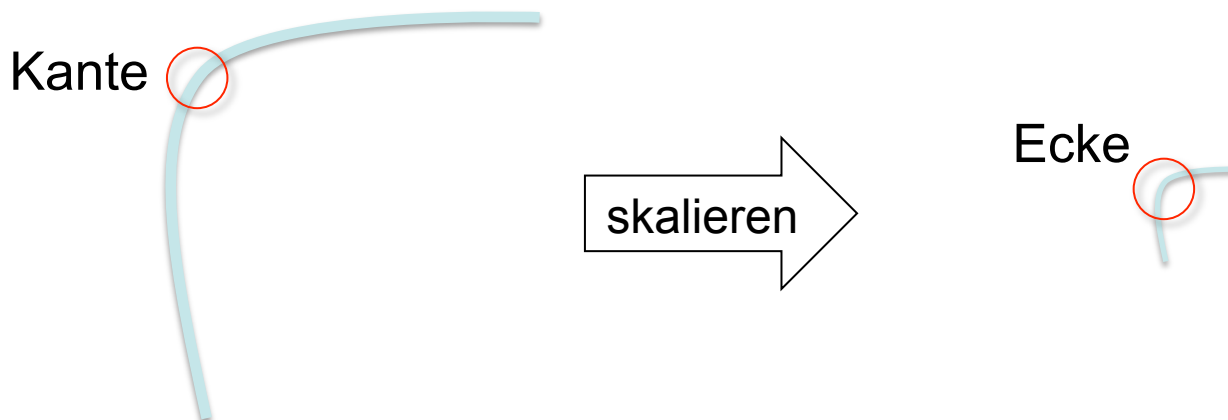
- $C = \det(M) - k \text{trace}(M)^2 = \lambda_1 \lambda_2 - k (\lambda_1 + \lambda_2)$

Robustheit des Harris Corner Detektors

- Invariant gegenüber Helligkeitsänderungen
 - bis zu einem gewissen Grad
- Invariant gegenüber Translation und Rotation

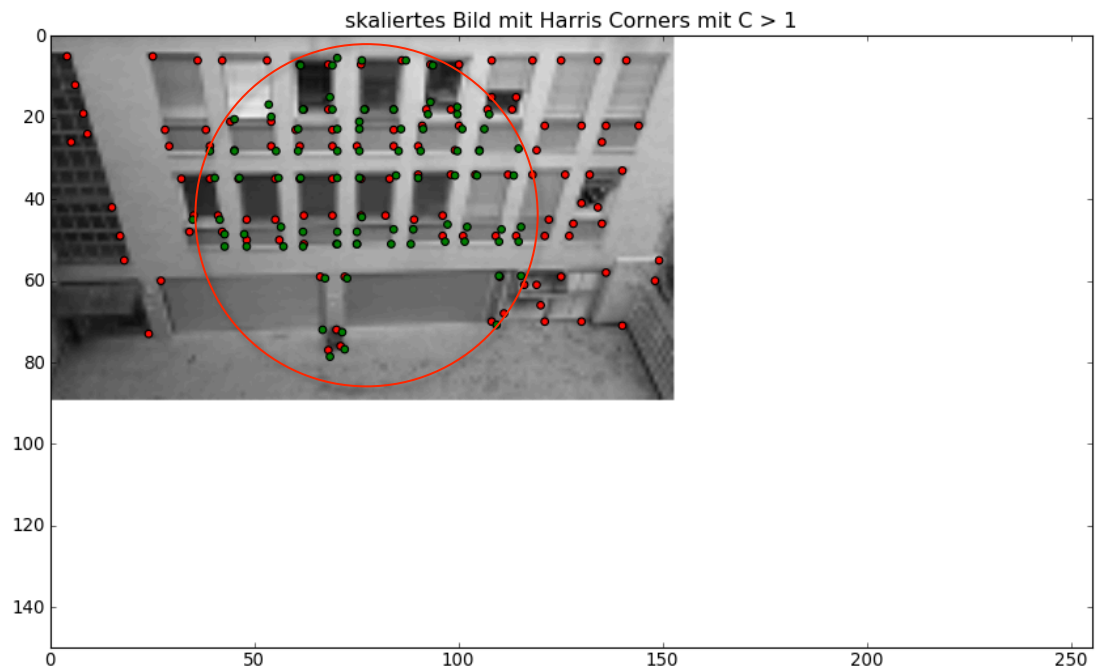
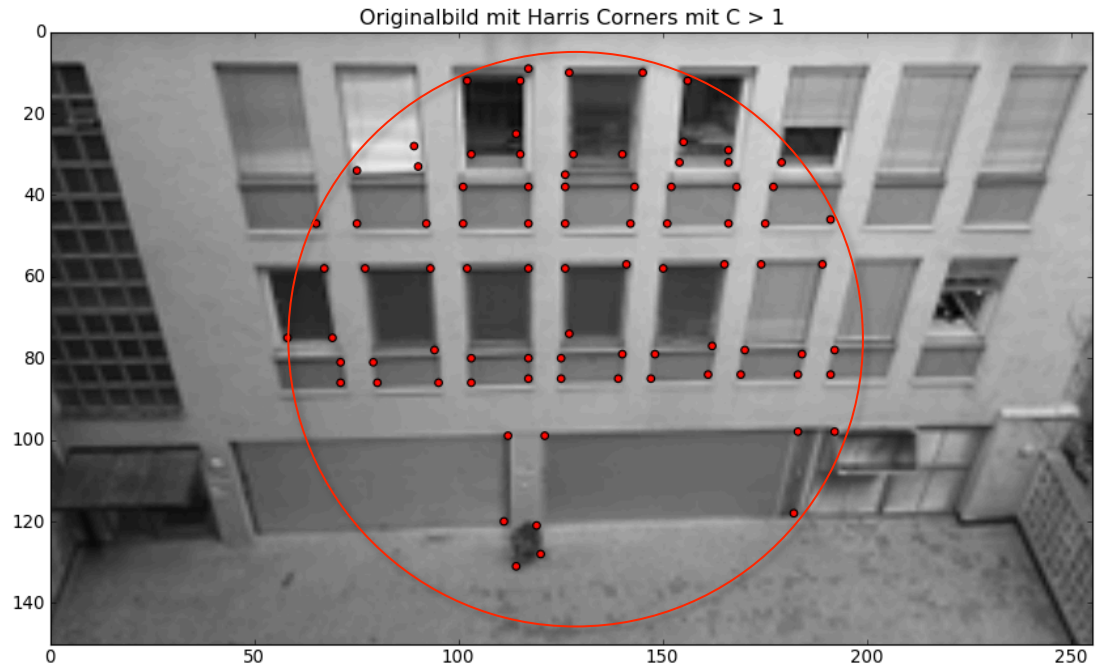
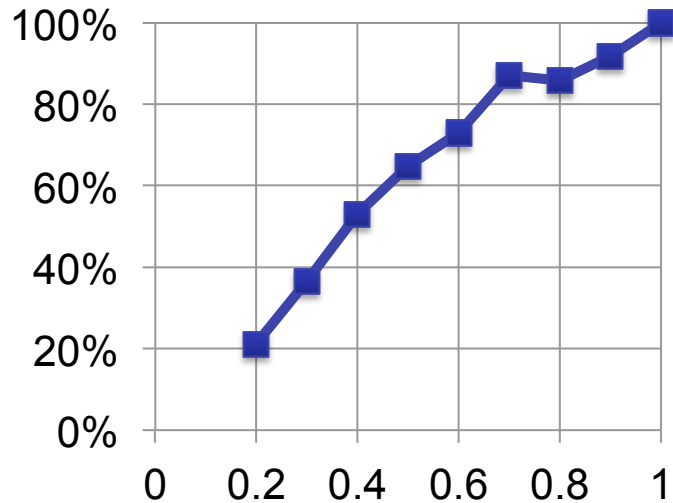


- Nicht invariant gegenüber Skalierung



Repeatability

- Invarianz des Eckenmaßes auf Rotation, Skalierung, Änderung der Perspektive
- hier Skalierung:

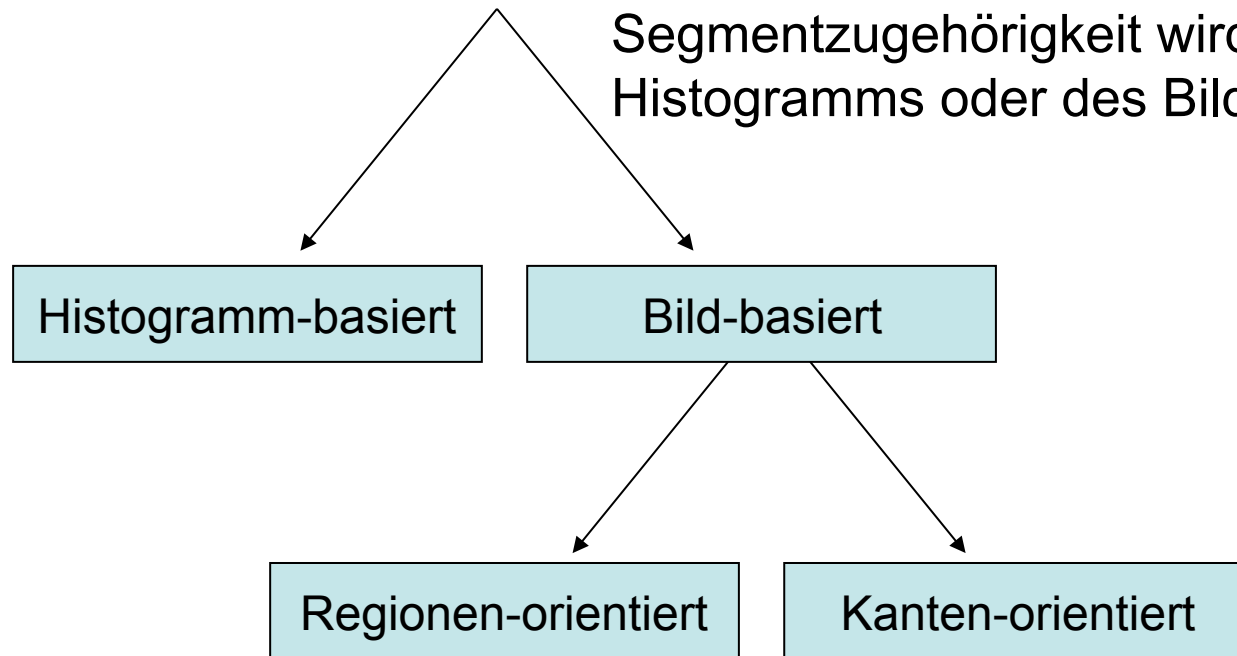


SEGMENTIERUNG

Segmentierungsmethoden

Histogramm-basierte oder
Bild-basierte Segmentierung:

Segmentzugehörigkeit wird anhand des
Histogramms oder des Bildes entschieden



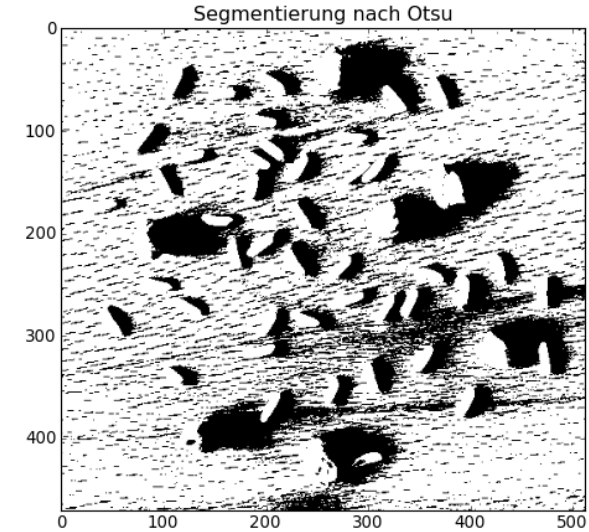
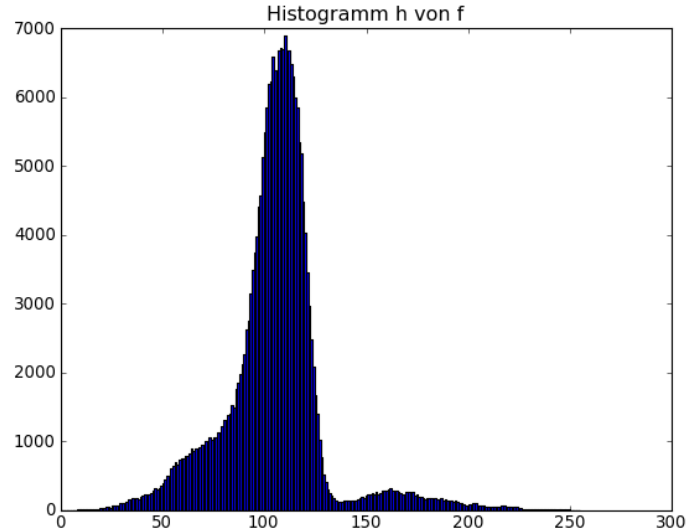
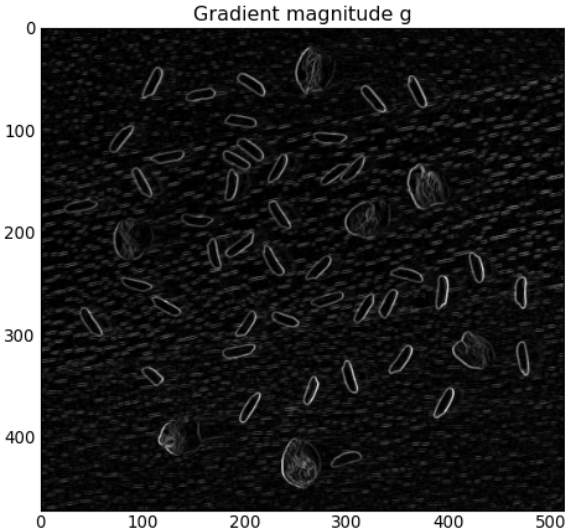
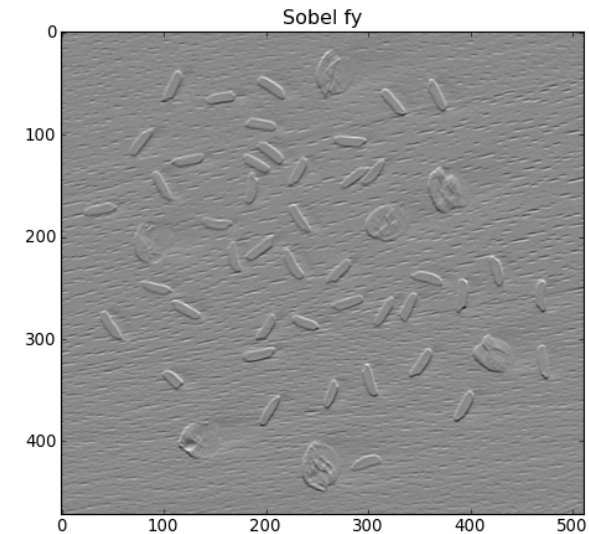
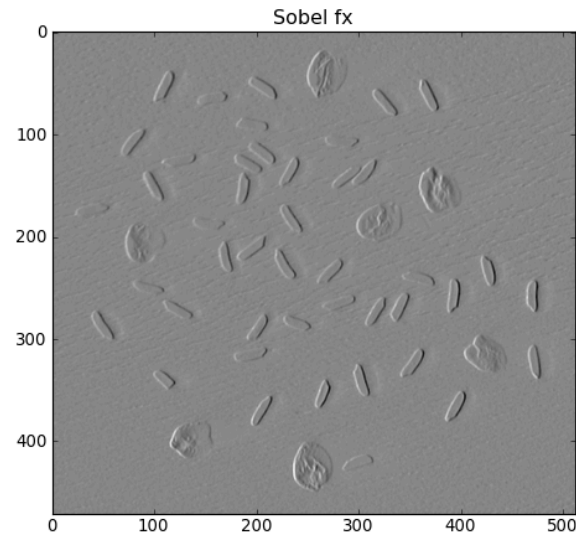
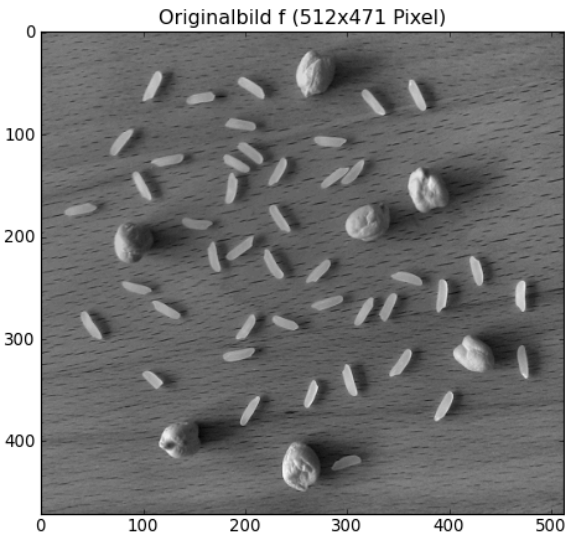
Regionen- oder Kantenorientierung:

Segmente werden durch ihre **Grenzen**
oder ihr **Inneres** definiert

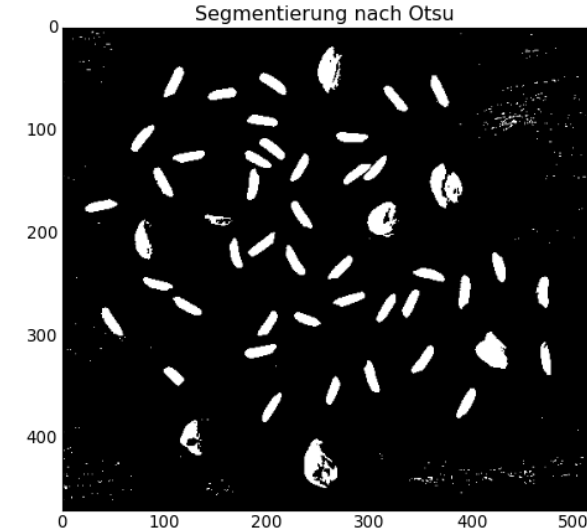
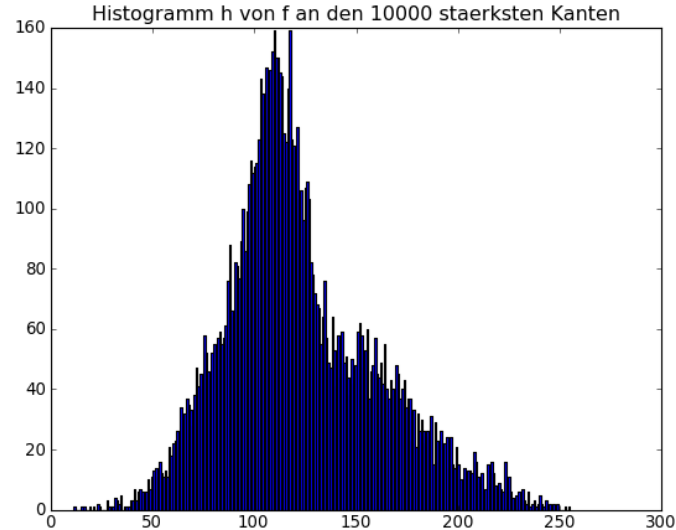
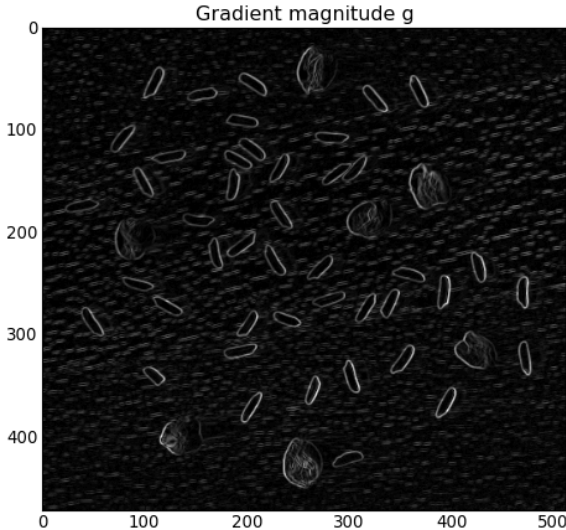
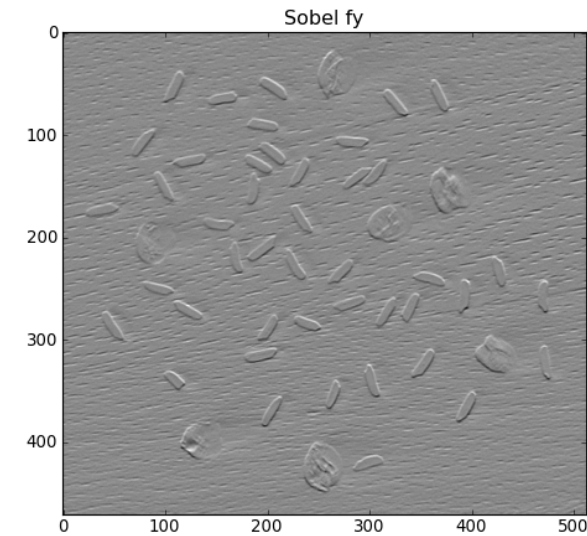
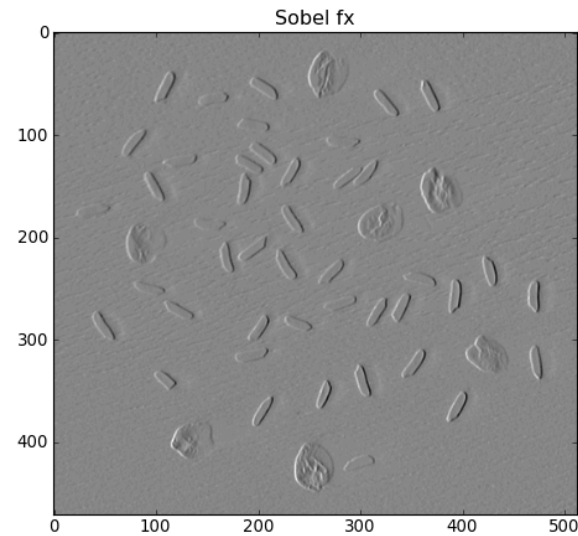
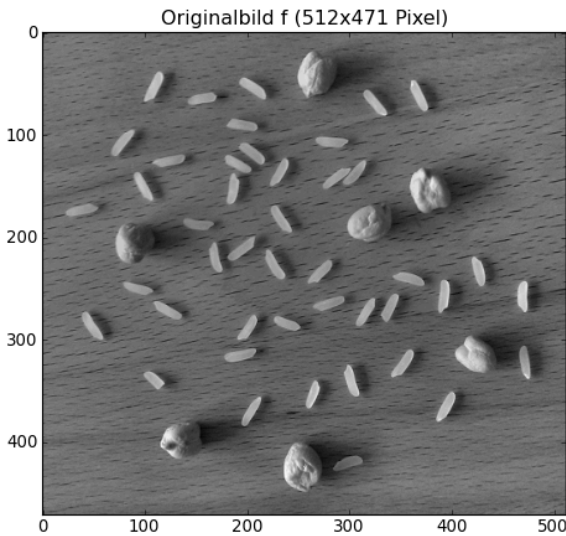
Verbesserung der Schwellenwertbasierte Segmentierung durch Kanten

- Idee: Zur Berechnung des globalen Schwellenwerts nur die Pixel in das Histogramm eintragen, die auf (oder nahe an) Kanten zwischen Vorder- und Hintergrund liegen
 - gleiche Wahrscheinlichkeit, dass solche Pixel auf Vorder- oder Hintergrund liegen
 - Gradienten- oder Laplace-Operatoren zum Finden der Kanten (aber: nicht alle Kanten separieren Vorder- und Hintergrund)
- Algorithmus
 - Kantenbild g von f berechnen (Gradienten-/Laplace-Operatoren)
 - Pixel (x,y) auf Kantenbild g auswählen mit $g(x,y) > T$
 - Histogramm h von f mit ausgewählten Pixeln berechnen
 - Methode von Otsu auf h zur globalen Segmentierung anwenden

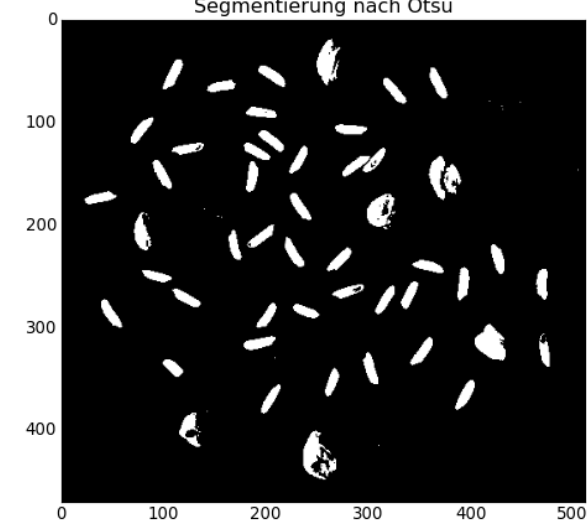
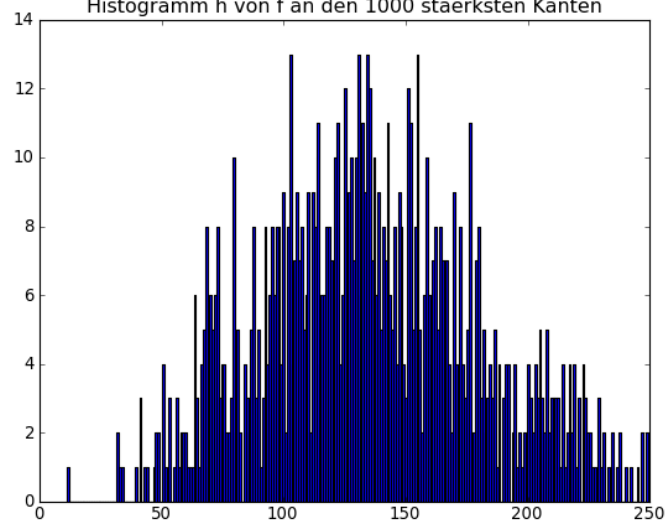
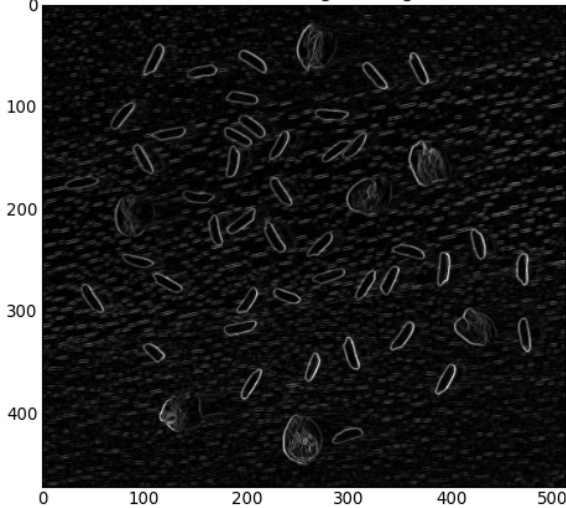
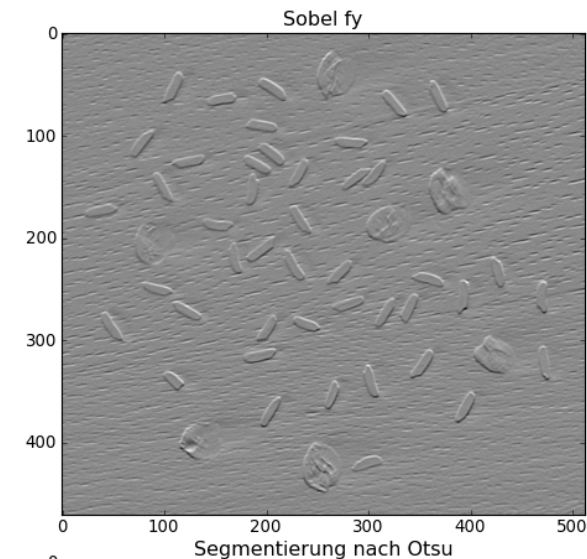
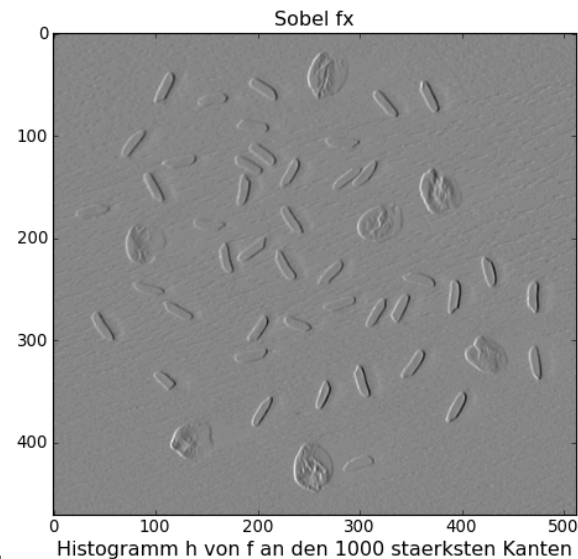
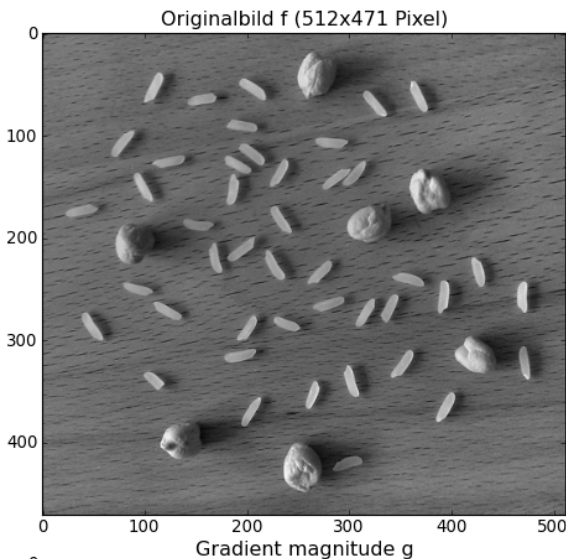
Histogramm aller Pixel



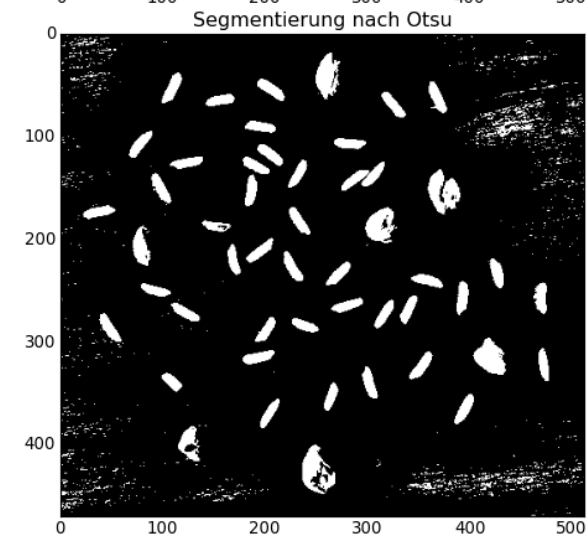
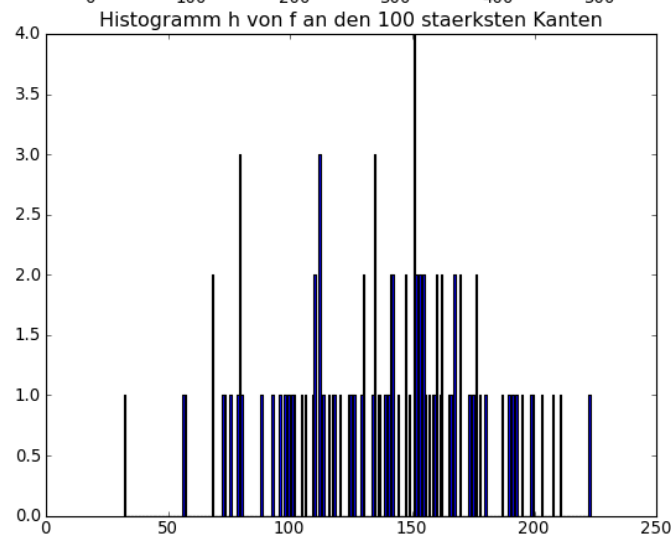
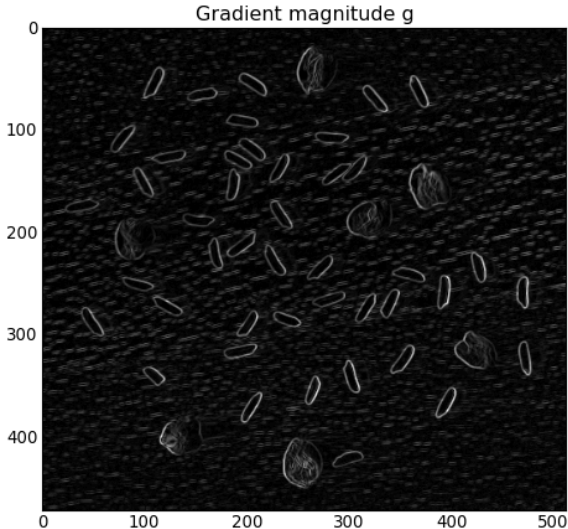
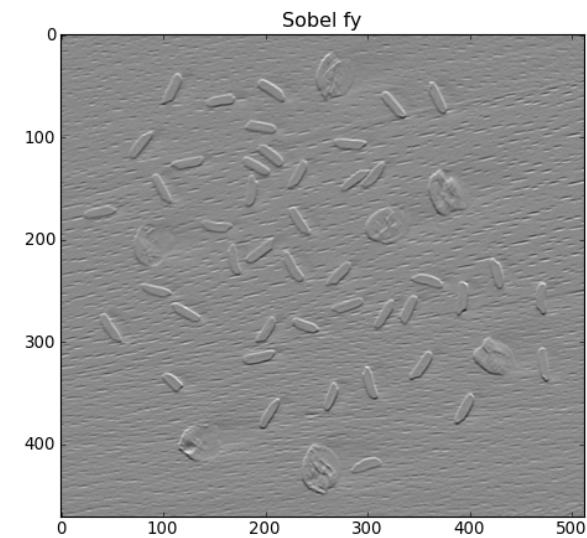
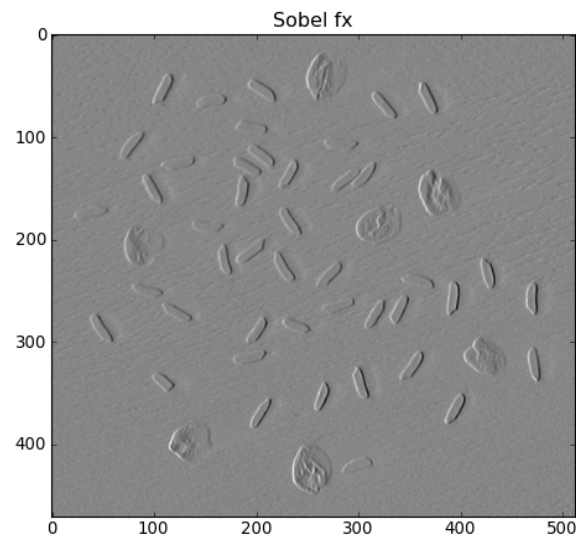
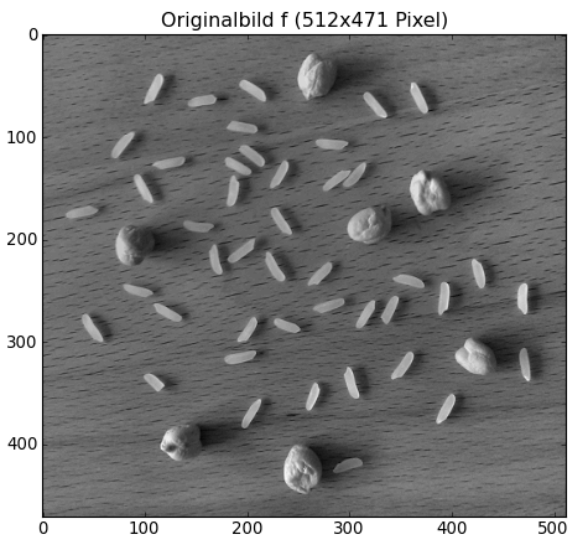
Histogramm der Pixel an den Stellen der 10000 längsten Gradienten



Histogramm der Pixel an den Stellen der 1000 längsten Gradienten

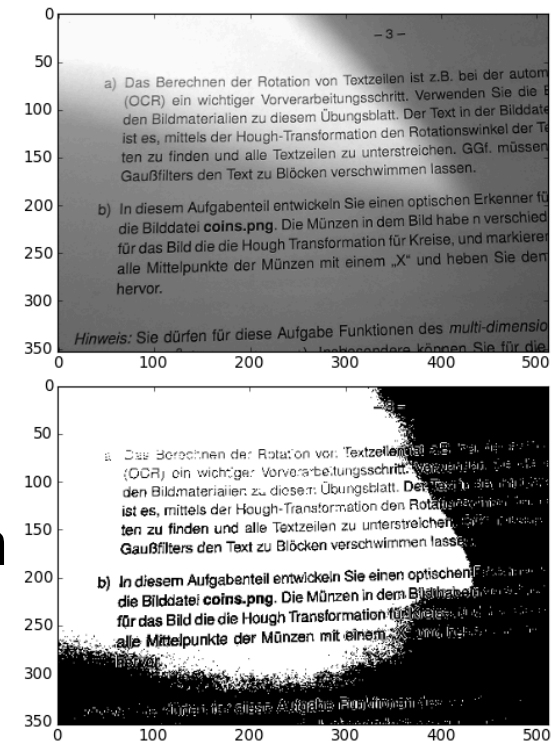


Histogramm der Pixel an den Stellen der 100 längsten Gradienten



Variable Schwellenwerte

- Globale Schwellenwerte problematisch
 - bei Shading nicht oft mehr global (für das gesamte Bild) definierbar
- Aufteilung des Bildes auch problematisch
 - Aufteilung klein genug, dass Beleuchtung (nahezu uniform), aber nicht so klein, dass nur Hintergrund- oder nur Vordergrundpixel
- Variable Schwellenwerte
 - Schwellenwert wird an jedem Punkt im Bild neu berechnet
 - typische Strategie: zeilenweises Durchlaufen im zick-zack, gleitender Durchschnitt der letzten n Grauwerte



Wellner's Adaptive Thresholding

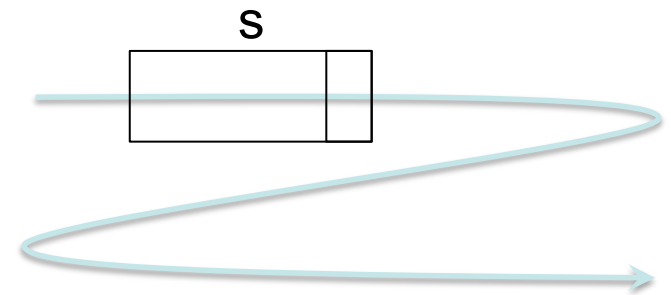
- Zeilenweises Durchlaufen im Zick-zack
- (s-facher) Durchschnittsgrauwert der letzten s Pixel:

$$g_s(n) = g_s(n-1) \cdot \left(1 - \frac{1}{s}\right) + p_n, s = \frac{w}{8}$$

- Initialisierung: $g_s(0) = s \frac{c}{2}$

- Schwellenwert:

$$T(n) = \begin{cases} 0 & \text{falls } p_n < \frac{g_s(n)}{s} \cdot \frac{100-t}{100}, t = 15 \\ 1 & \text{sonst} \end{cases}$$

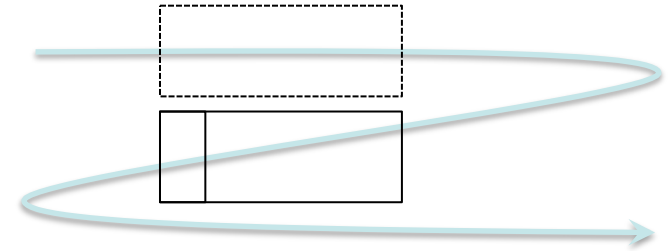


Pierre D. Wellner. Adaptive thresholding for the DigitalDesk. Technical Report EPC-93-110, Rank Xerox Research Centre, Cambridge, UK, 1993.

Wellner's Adaptive Thresholding

- Verbesserung: Berücksichtigung des durchschnittlichen (s-fachen) Grauwerts über dem aktuellen Pixel

$$h_s(n) = \frac{1}{2} (g_s(n) + g_s(n-w))$$



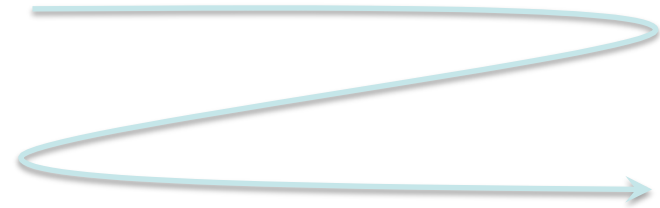
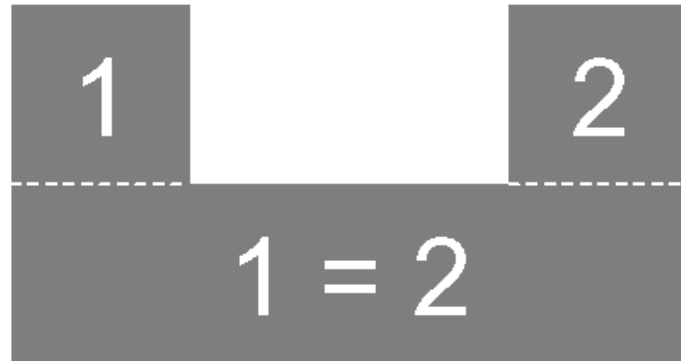
$$T(n) = \begin{cases} 0 & \text{falls } p_n < \frac{h_s(n)}{s} \cdot \frac{100-t}{100} \\ 1 & \text{sonst} \end{cases}, t = 15$$

Pierre D. Wellner. Adaptive thresholding for the DigitalDesk. Technical Report EPC-93-110, Rank Xerox Research Centre, Cambridge, UK, 1993.

Labeling 1. Phase: Vorläufige Label

- Basiert auf 4-Nachbarschaft
- Falls auf Vordergrund-Pixel p gestoßen, betrachte bearbeitete Nachbarpixel darüber und links davon (falls aktuelle Bildzeile v.l.n.r. durchlaufen) bzw. rechts davon (sonst)
- Falls beide Nachbarn (darüber, links/rechts) Vordergrund, dann setze $\text{Label}(p) = \text{Label}(\text{darüber})$
 - falls $\text{Label}(\text{darüber}) \neq \text{Label}(\text{links/rechts})$, dann markiere beide Label als äquivalent (Mergen der Äquivalenzketten)
- sonst, falls einer der Nachbarn (darüber, links/rechts) Vordergrund, dann setze $\text{Label}(p)$ auf dessen Label
- sonst vergebe neues Label für p
 - füge in eigene Äquivalenzkette ein

Warum sind die Label vorläufig?



- Label-Äquivalenzen werden möglicherweise erst später erkannt (→ Durchlauf zeilenweise)
- Labeling 2. Phase: Auflösen der Äquivalenzen
 - Durch Äquivalenzkette laufen und jedem Pixel in der gleichen Äquivalenzkette das gleiche finale Label geben

WASSERSCHEIDEN- TRANSFORMATION

Wasserscheidentransformation

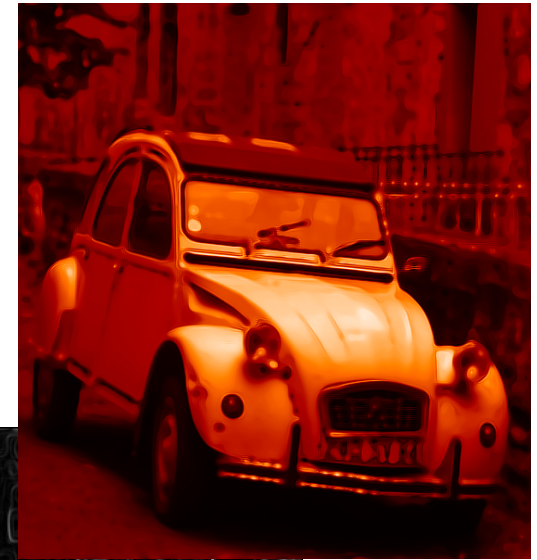
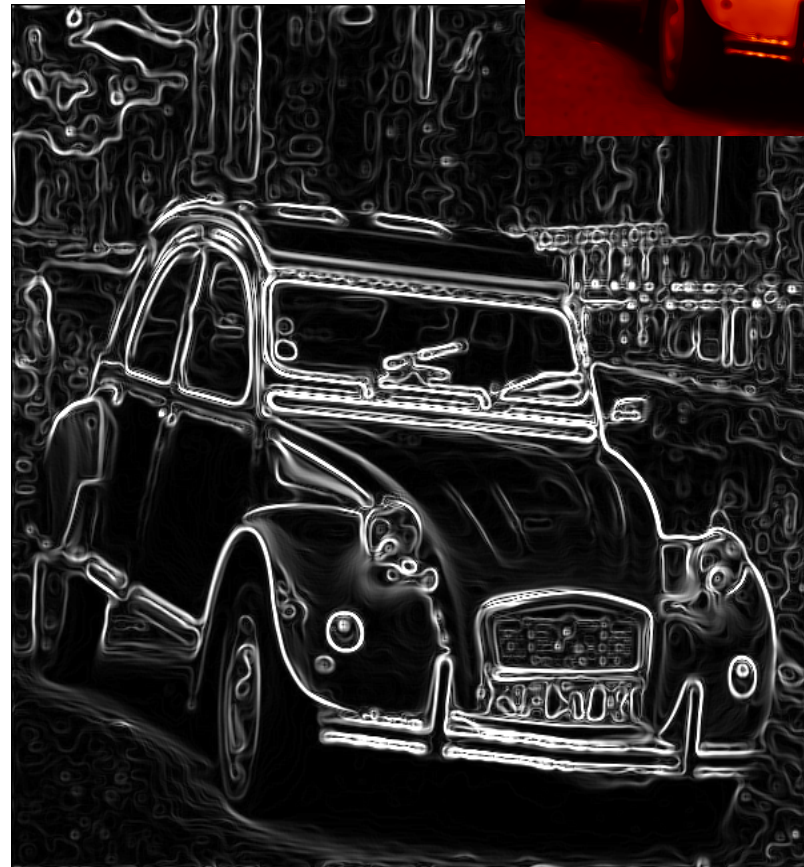
- Wasserscheide: Grenzen der Entwässerung in unterschiedliche Senken
 - Beispiel: Wasserscheide zwischen Nordsee und Mittelmeer entlang des Kamms der Berner Alpen
- Wasserscheide in der Segmentierung: Generiere ein Höhenprofil so, dass die Wasserscheiden die gesuchten Segmentgrenzen sind.





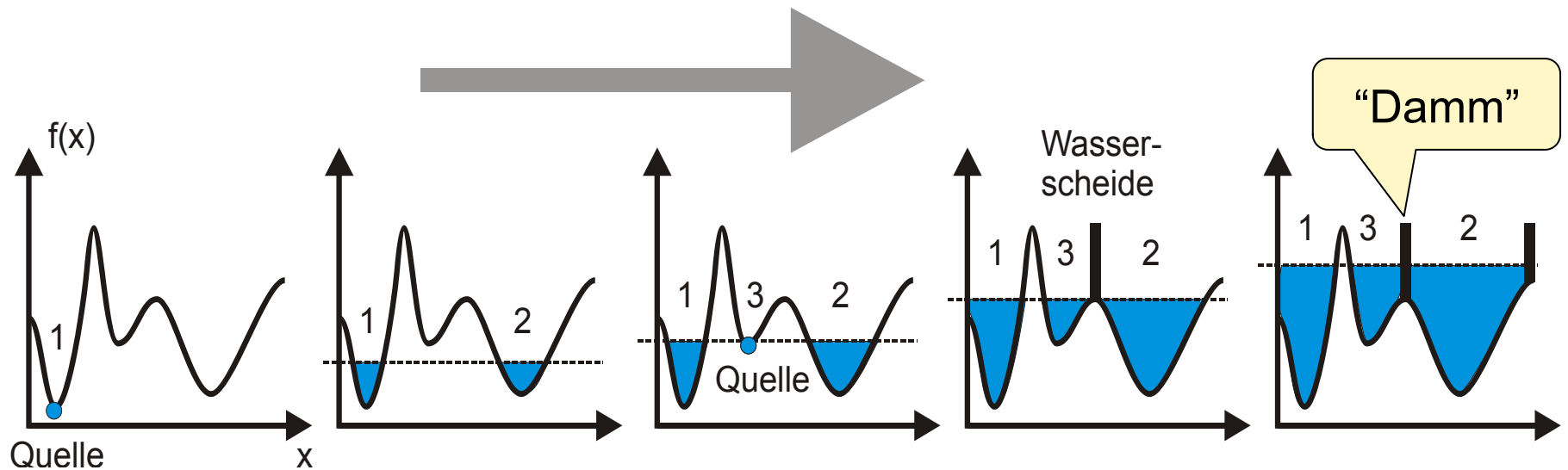
Wasserscheiden

- Wasserscheiden sollen an Kanten verlaufen
- Wasserscheiden sind „Gebirgskämme“
- Wasserscheiden sind die Längen der Grauwertgradienten



Wasserscheidentransformation

- **Beregnung:** Es fällt „Regen“ auf jedes Pixel. Gradienten entschieden, wohin der Regen entwässert wird
- **Flutung:** Die „Welt“ wird von den Senken her geflutet. Wenn Wasser aus zwei Senken zusammen fließt, wird ein Damm gebaut bzw. entsteht eine Wasserscheide



Animation von Serge Beucher

- Partitionierung des Bildes durch Flutung von den Minima aus bei Vermeidung des Zusammenfließens verschiedener Quellen
 - zwei Mengen:
Niederschlagsgebiete und Wasserscheidenlinien
- Transformation auf Gradientenbild:
Niederschlagsgebiete entsprechen homogenen Regionen



Copyright © 2010, Serge Beucher

<http://cmm.ensmp.fr/~beucher/wtshed.html>

Flutungsalgorithmus (Skizze)

Jedes bei Höhe h_{aktuell} neu überflutete Pixel (m_f, n_f) ist

- **in Isolation:**

Es nicht zu anderen überfluteten Pixeln der Höhen $h < h_{\text{aktuell}}$ benachbart.

Isolierte Pixel sind Kerne von neuen Segmenten.

- **Erweiterung:**

Es ist zu anderen überfluteten Pixeln der Höhen $h < h_{\text{aktuell}}$ mit gleichem Label benachbart.

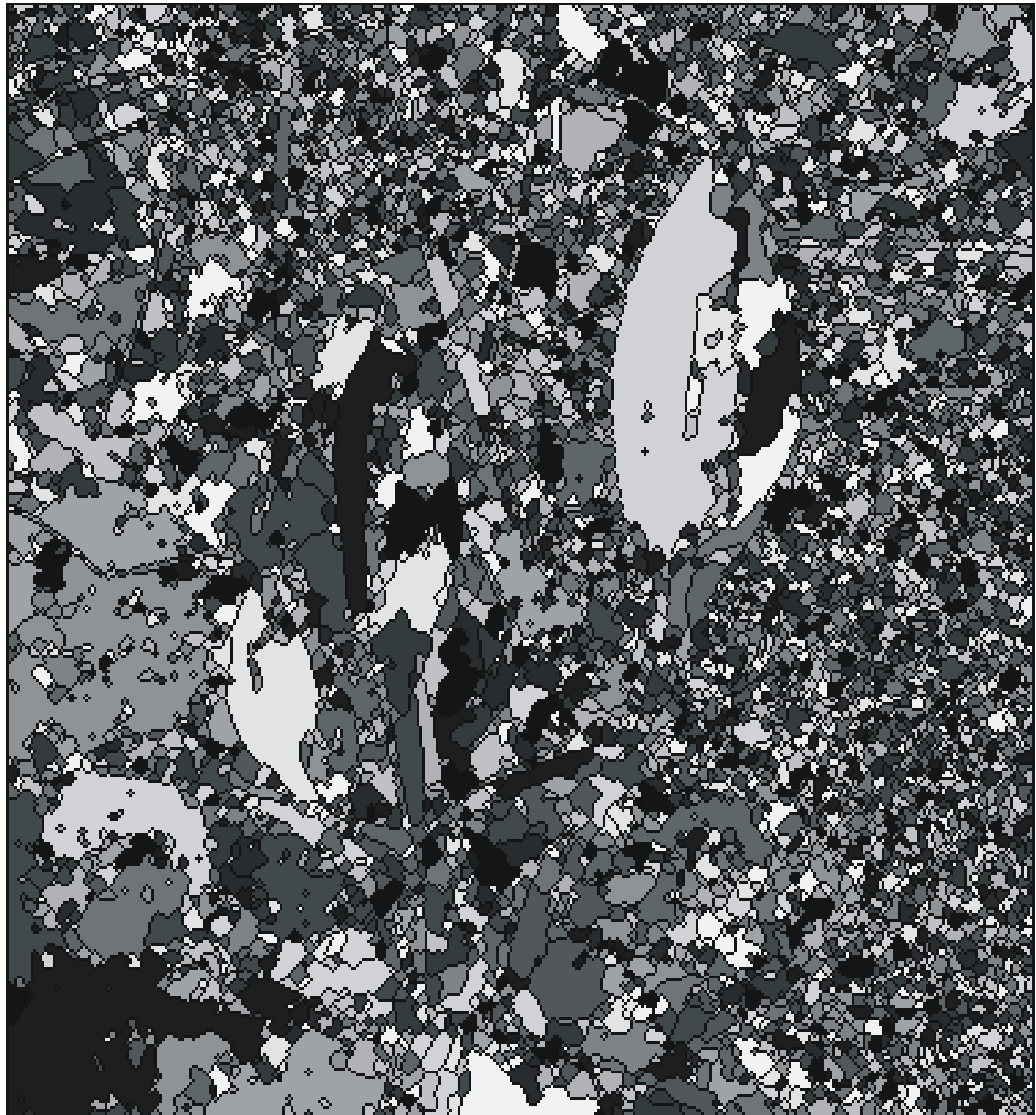
Das Pixel wird dem Segment mit diesem Label zugeordnet.

- **Wasserscheide:**

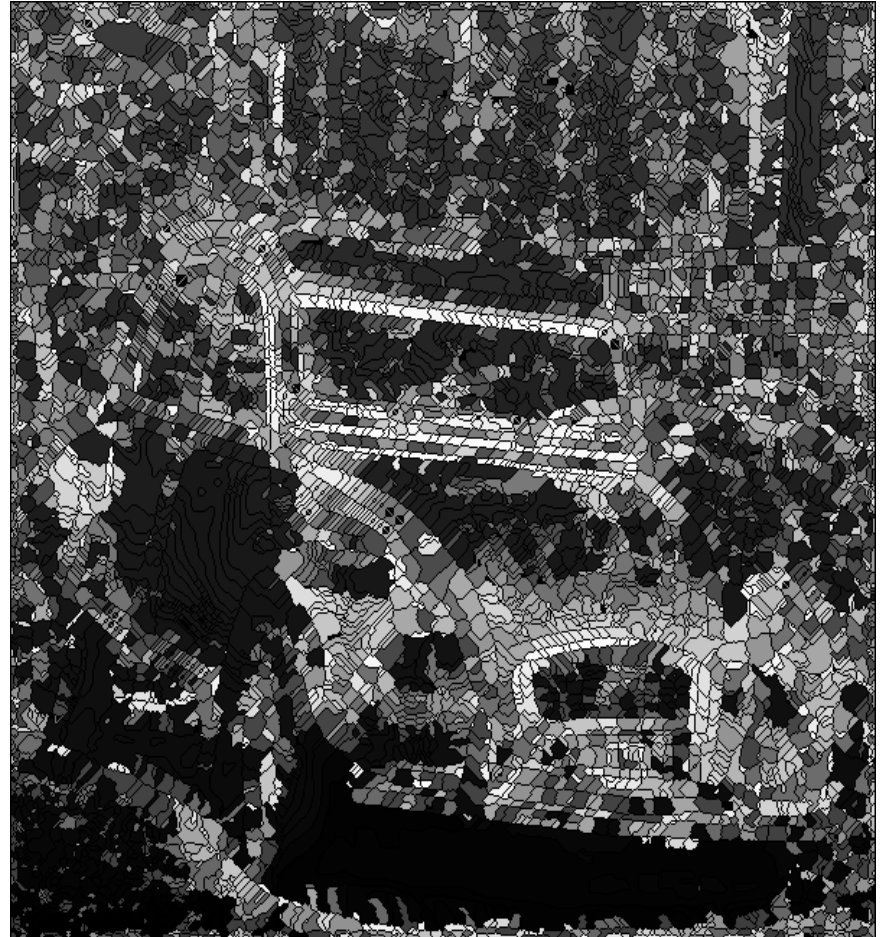
Es ist zu überfluteten Pixeln von mindestens zwei Regionen benachbart.

Dem Pixel wird das Label „Wasserscheide“ zugeordnet.

Resultat der WST

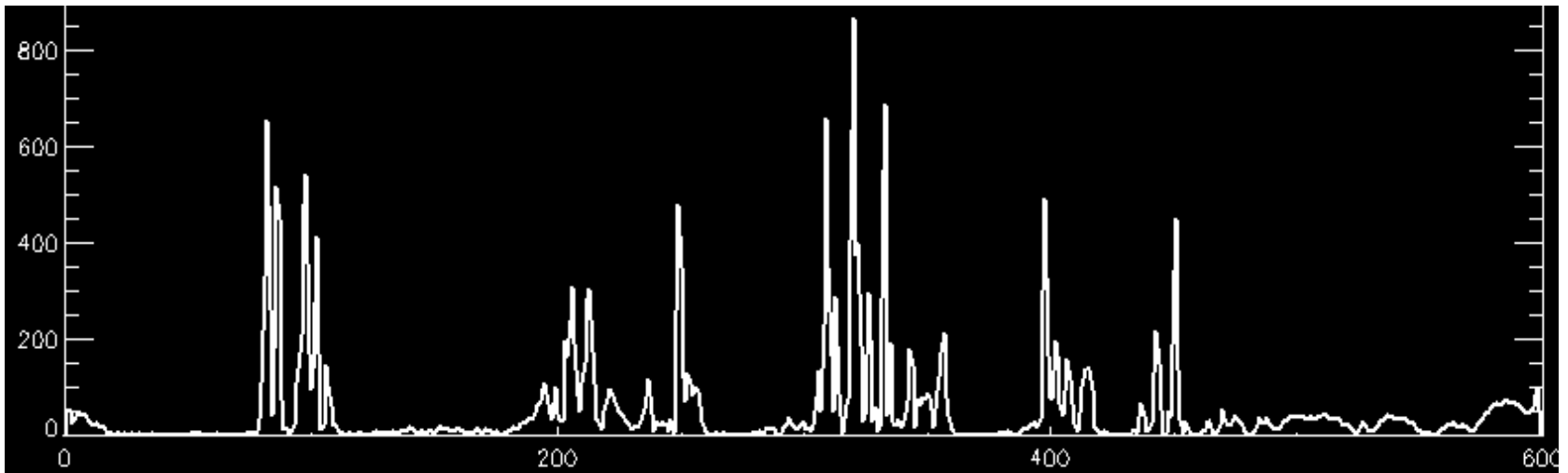


Resultat der WST



Problem Übersegmentierung

- Für die WST ist jedes lokale Minimum eine Senke
- Die meisten Senken werden durch Rauschen verursacht
- Senken durch Rauschen sind weniger tief als die von Kanten

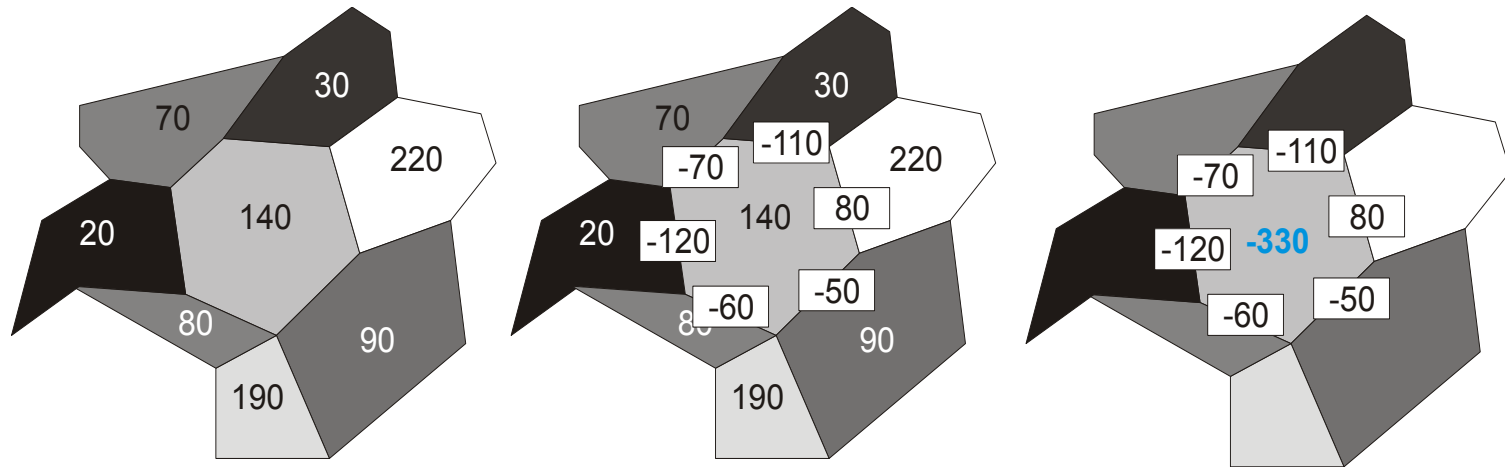


Hierarchische WST

Multiskalenstrategie:

- Wasserscheidentransformation auf dem WST-Resultat
- Jede Region erhält ihren durchschnittlichen Grauwert als Funktionswert
- Die erste WST wird hauptsächlich durch Rauschen verursachte Senken finden
- „Wahre“ Senken sollten über mehrere Stufen der Hierarchie erhalten bleiben

Gradienten für die hierarchische WST



- Differenz zu jeder benachbarten Regionen berechnen
- Gradientenlänge berechnen als durchschnittliche Differenz zu allen Regionen (evtl. mit Fläche gewichten)
- Problem: Kanten werden breiter

Verwendung von Markern bei der WST

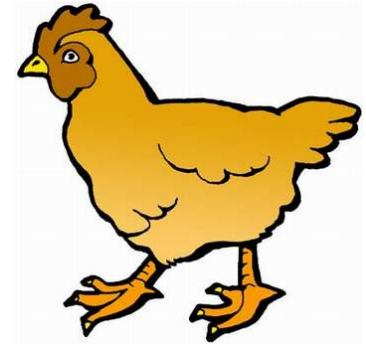
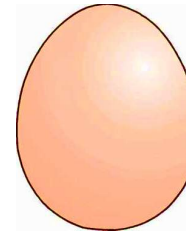
- Marker = Region im Bild
 - interne Marker markieren Vordergrundobjekte
 - externe Marker markieren den Hintergrund
- Strategien zur Erzeugung von Markern
 - Bild zur Vorverarbeitung glätten Marker auf geglättetem Bild
 - interaktives Setzen der Marker
 - Textur
 - Form
- angepasster Algorithmus
 - isoliertes Pixel: Label „unbekannt“
 - Pixel erweitert Region: Label zuordnen
 - Pixel verbindet zwei Regionen: falls eine „unbekannt“ und andere endgültiges Label, dann Label auch für die andere Region

Modellbasierte Segmentierung

- Interaktive Suche
 - Region Growing
 - Kantenverfolgung
 - Markerbasierte WST
- Vollständige Suche
 - Template Matching
 - Hough Transformation

Modellbasierte Segmentierung

- Segmentierung: Generierung von Symbolen (Bedeutungsträgern) aus Pixeln.
- Modell: Erwartete Bedeutung
- ▶ Henne-Ei-Problem??

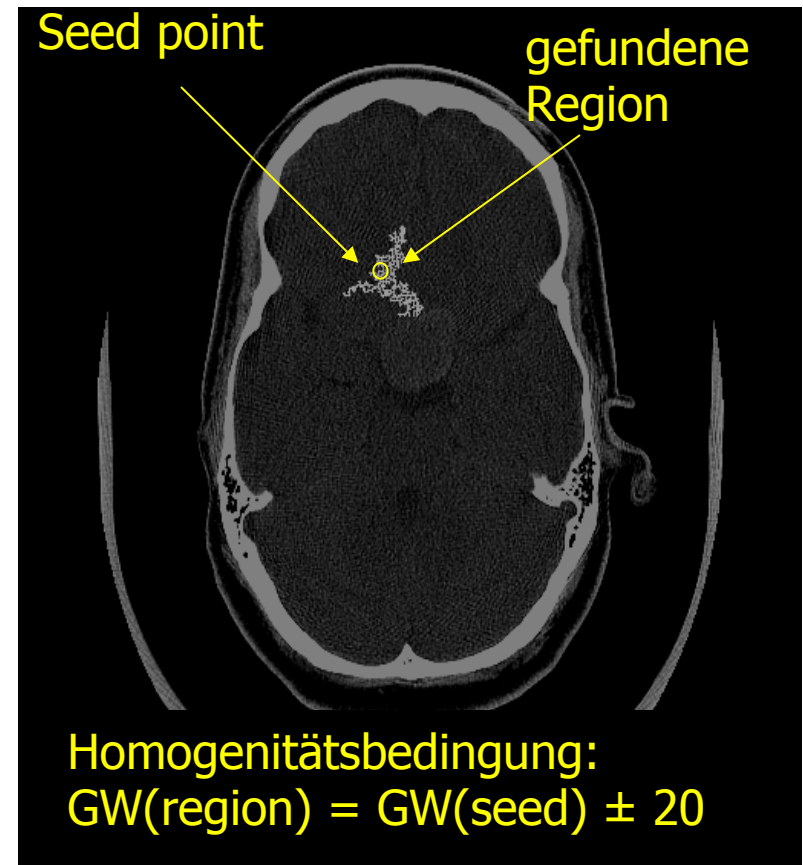


Modellbasierte Segmentierung: Mix aus Segmentierung und Analyse. Mit Vorwissen über ein bestimmtes Objekt wird nach Instanzen gesucht

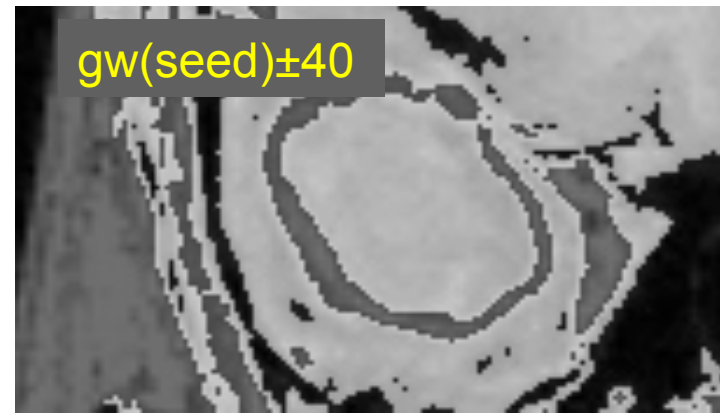
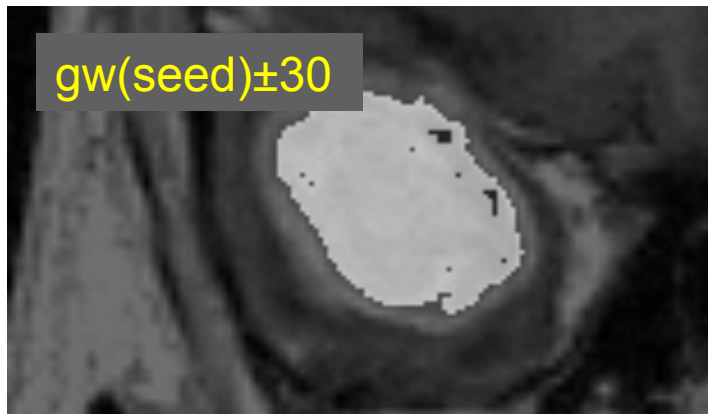
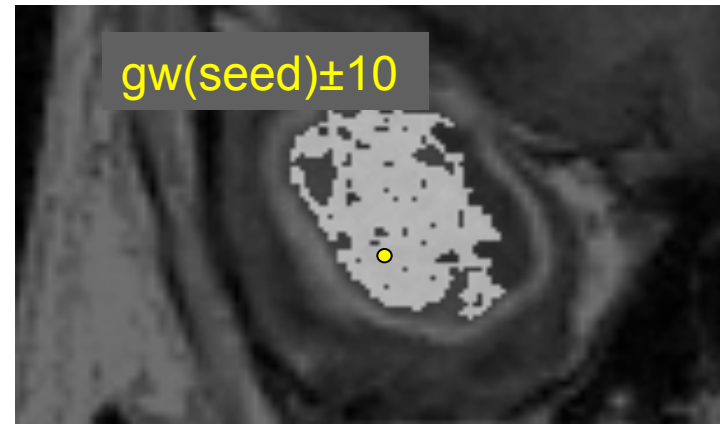
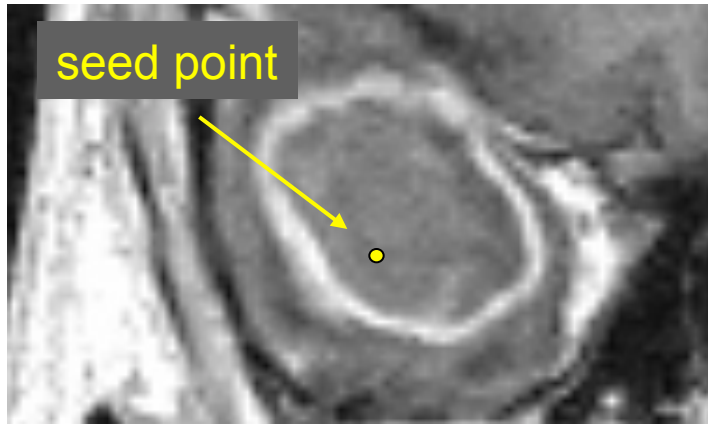
- Interaktive Suche: Benutzer gibt Vorwissen interaktiv ein
- Vollständige Suche: Instanzen eines Modells mit wenigen Parametern werden gesucht
- Iterative Suche: Instanzen eines Modells mit vielen Parametern werden gesucht.

Region Growing

- = Flood Fill auf Grauwertbild für einen einzigen (vorgegebenen) Saatpunkt.
- Homogenitätsbedingungen:
 - Grauwertbereich
 - Grauwertschwankung
- Annahmen und Ziel:
 - Gebiet ist intern von größerer Homogenität als an den Grenzen.
 - Selektion eines einzigen Gebiets.
 - Homogenitätsverhältnisse an anderen Orten interessieren nicht.



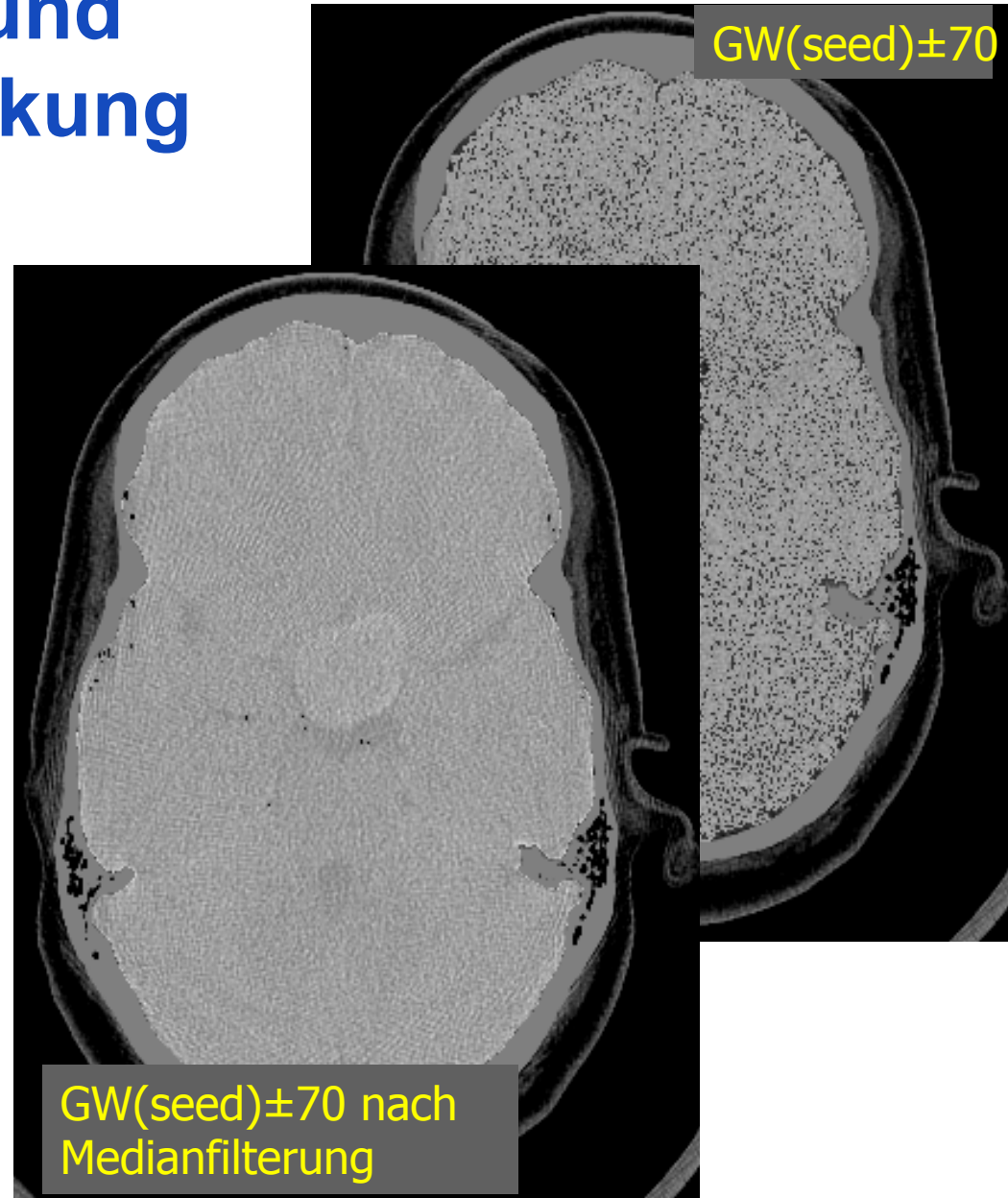
Auslaufen von Regionen



Region Growing und Rauschunterdrückung

Rauschen führt zu Anfälligkeit der Methode

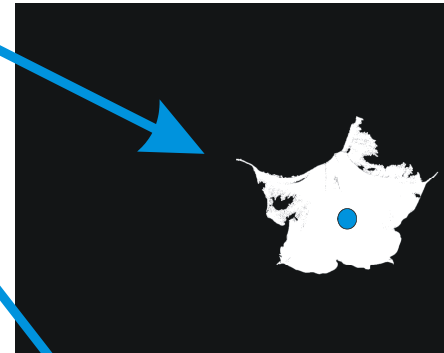
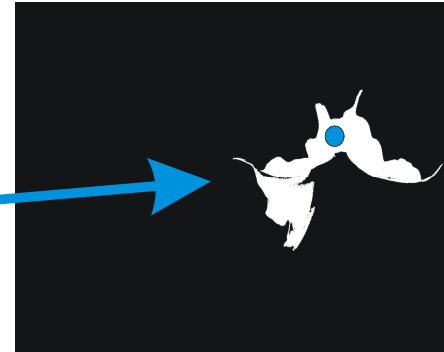
→ vorherige Rauschunterdrückung oder Nachverarbeitung



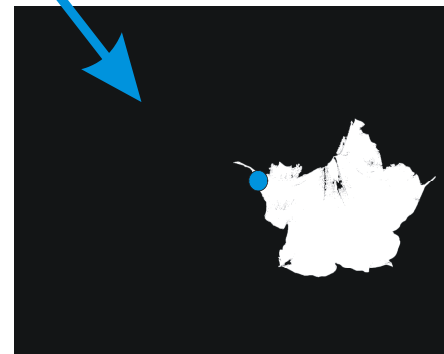
Mehrfaches Region Growing



original



Zusammensetzung aus allen drei Segmenten



Gezielte Kantensuche



Resultat der Kantenfilter:

Liste von Kantenpunktkandidaten

- nicht alle Kandidaten gehören zur gesuchten Regionengrenze.
- die Regionengrenze kann Lücken aufweisen.

Strategien mit Nutzerinteraktion

- Kantenverfolgung
- Optimale Kantenzüge
- Hough Transformation

Optimale Kantenzüge (Graphensuche)

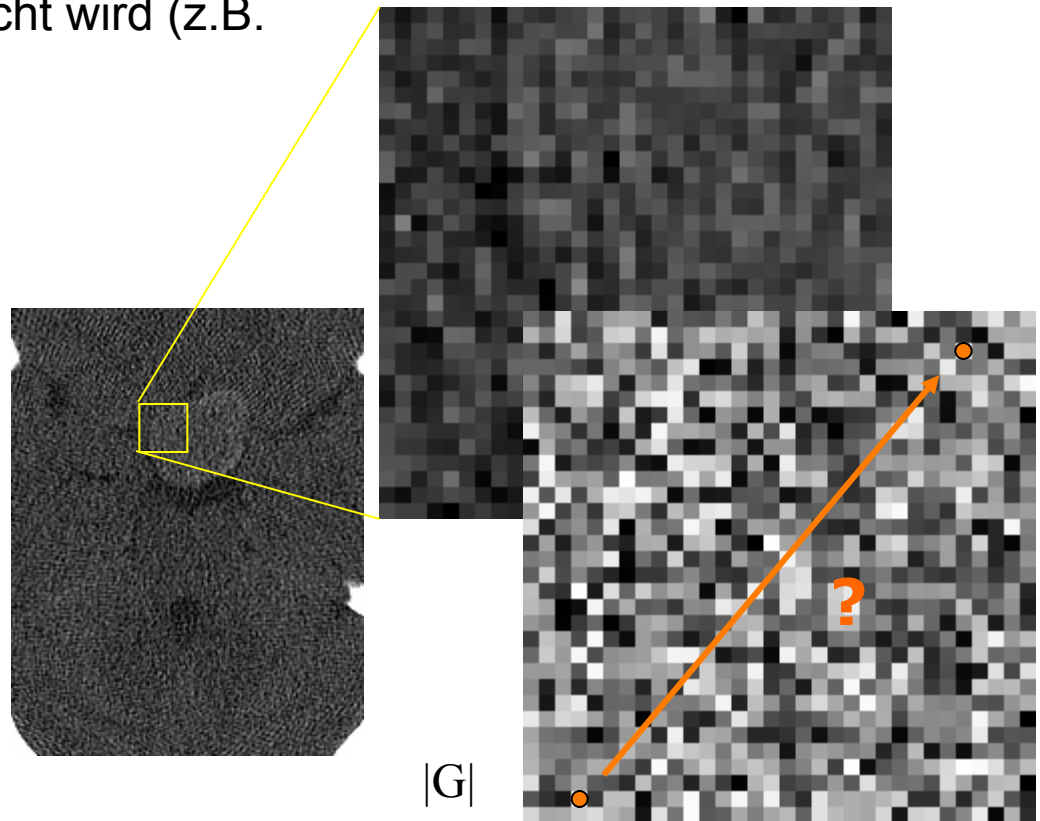
Das Resultat des Gradientenoperators wird als Graph aufgefasst, auf dem zwischen zwei Punkten ein optimaler Pfad gesucht wird (z.B. mittels Dijkstra-Algorithmus).

Vorteil:

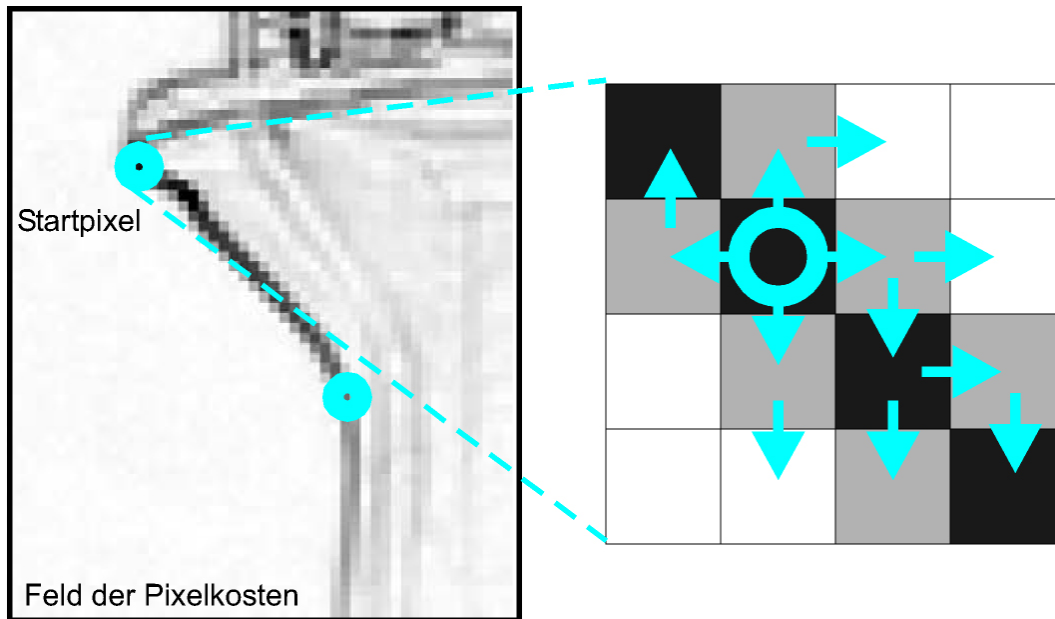
Globale Aspekte des Kantenzugs können eingebracht werden.
(z.B. Glattheitsbedingungen)

Nachteil:

Aufwand ist größer als bei Kantenerfolgung.



Optimale Kantenzüge (Graphensuche)



Optimalitätskriterien:

- Maximierung der (durchschnittlichen) Gradientenlänge
- Minimierung der Pfadlänge
- Minimierung der Richtungsänderungen
- Minimierung der Grauwertänderungen

Resultate



Startpunkt

