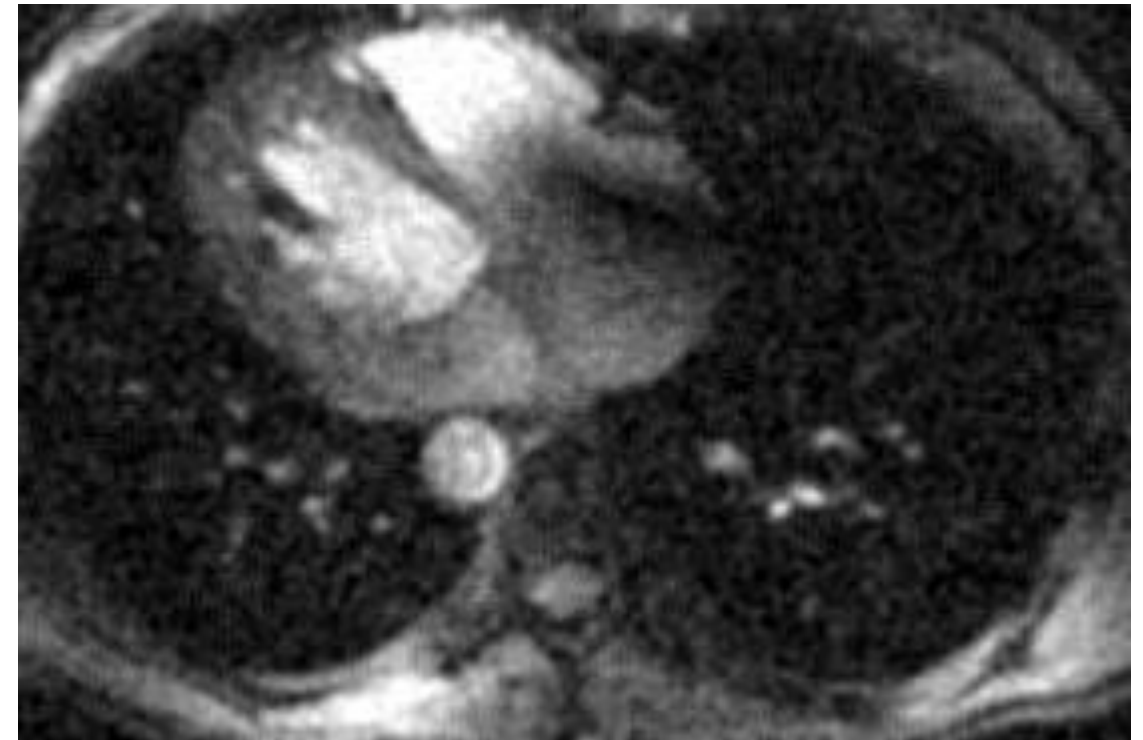




## Famous Last Question

**Wie könnte eine kantenerhaltene Rauschunterdrückung aussehen, die ausnutzt, dass Kanten ihre Richtung nur langsam ändern?**





# Segmentierung

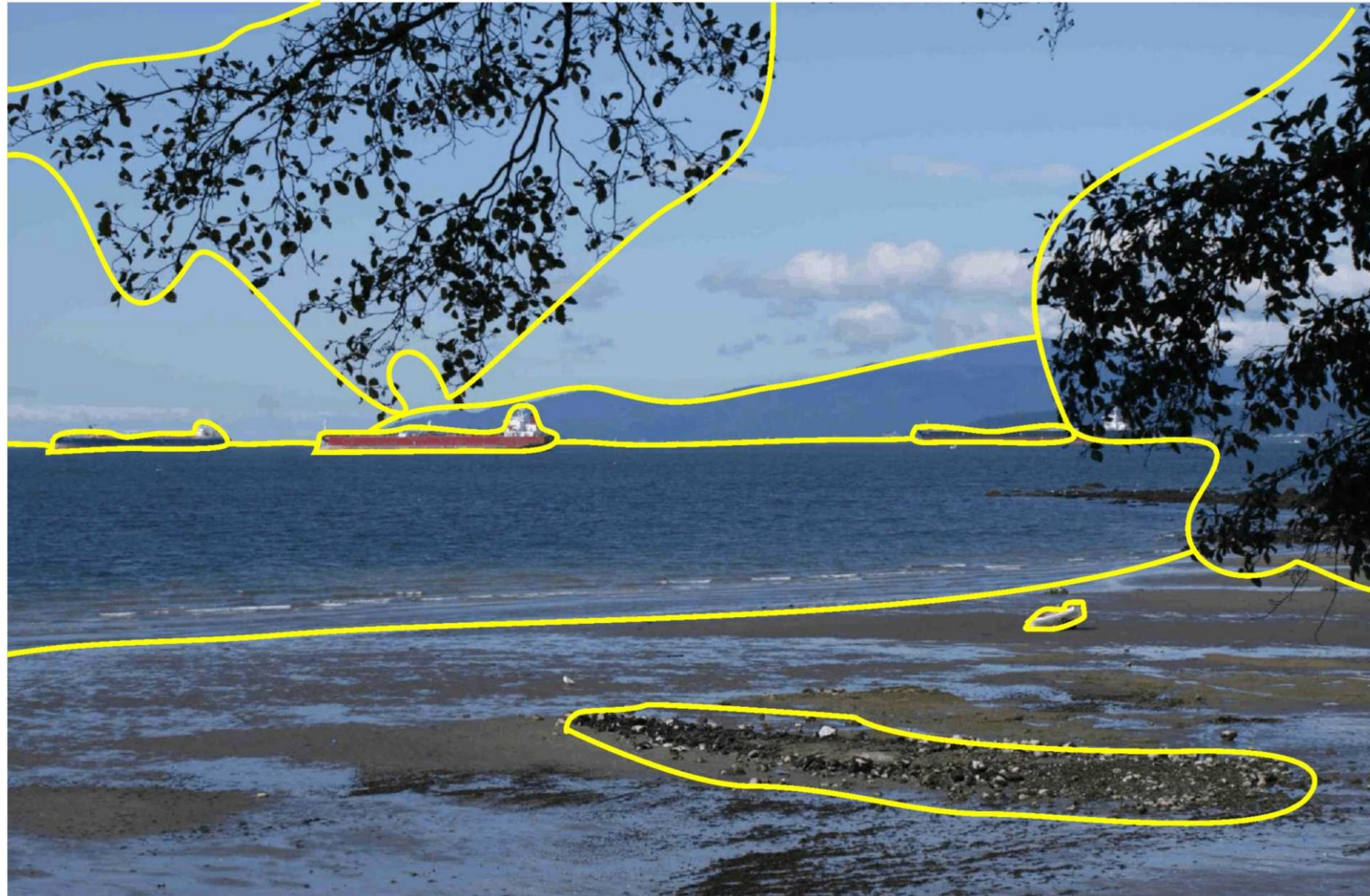


Wieviele Segmente?





# Segmentierung







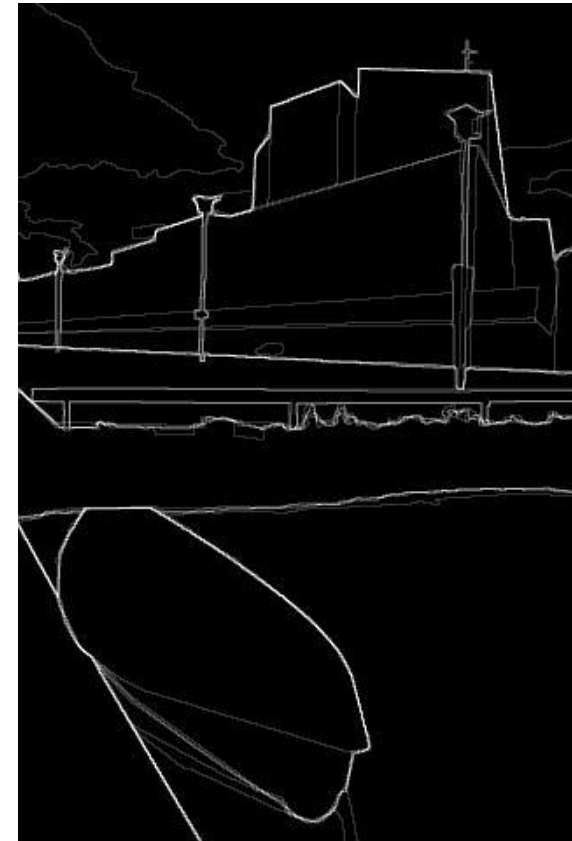
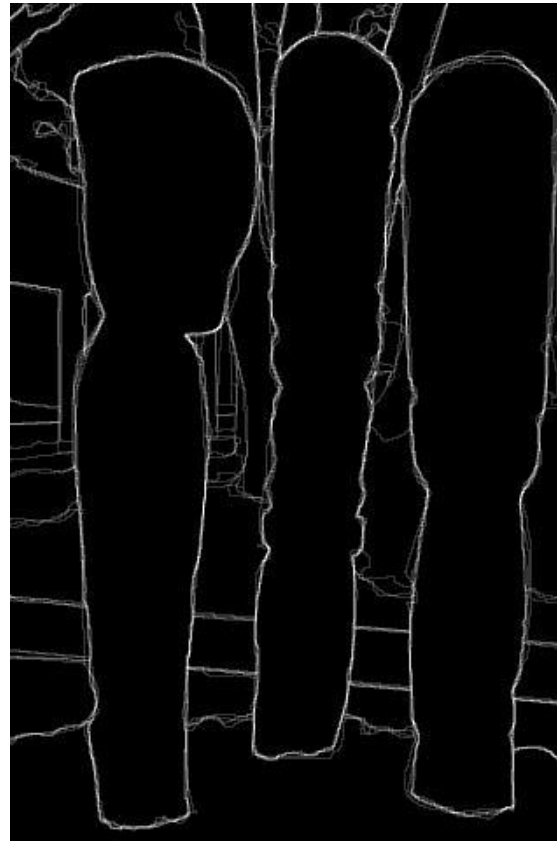
## Gibt es die „ideale Segmentierung“?



Beispielbilder aus dem Berkeley Segmentation Dataset



## Referenzsegmentierungen



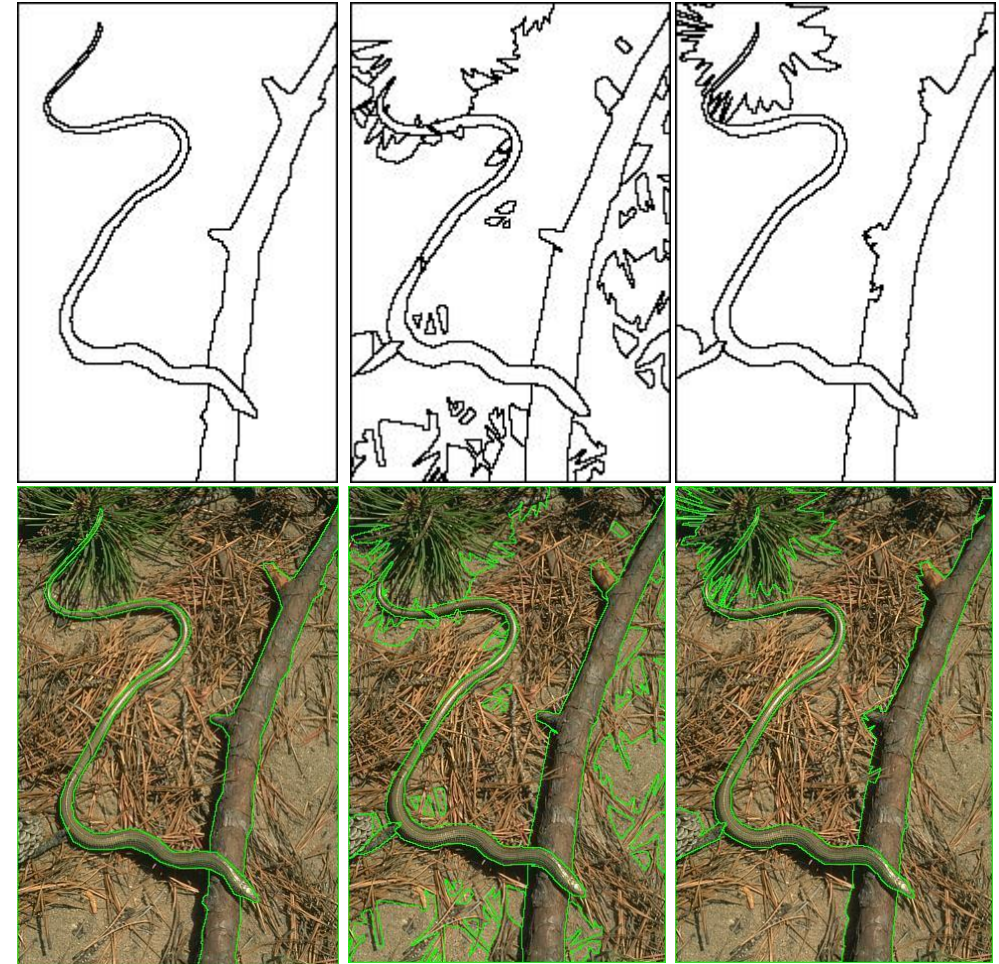
Helligkeit = Häufigkeit, mit der die betreffende Segmentgrenze gewählt wurde





# Referenzsegmentierungen

- Es existieren zwar meist Segmente, die jeder als solche bezeichnet, aber
  - Was Hintergrund ist, wird oft unterschiedlich gesehen
  - Segmente selbst werden unterschiedlich detailliert gesehen
- Segmentierung ist nicht a-priori lösbar





# Segmente

- Segmente und Semantik
  - Unterschiedliche Segmente haben unterschiedliche Bedeutung
  - Pixel innerhalb eines Segments haben die gleiche Bedeutung
- Segmentierung ist ein *subjektiver* Prozess!
- Segmente können aus einem Bild generiert werden, falls
  - Pixel eines Segments andere Eigenschaften haben als die außerhalb eines Segments
  - jedes Pixel nur einem Segment angehört
- Es gibt viele Fälle, in denen diese Annahmen (näherungsweise) zutreffen

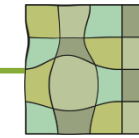


## Segmentierung (abgeschwächt)

- Ziel: Pixelaggregate schaffen, denen Bedeutung zugeordnet werden *kann*
- Pixel gehören zusammen, wenn sie einem zusammenhängenden Gebiet angehören.
- Homogenitätsbedingung ist vom Bildinhalt unabhängig (Segmentierung ist datenbasiert).
- Modellbasierte Segmentierung: Zwitter zwischen Segmentierung und Analyse.

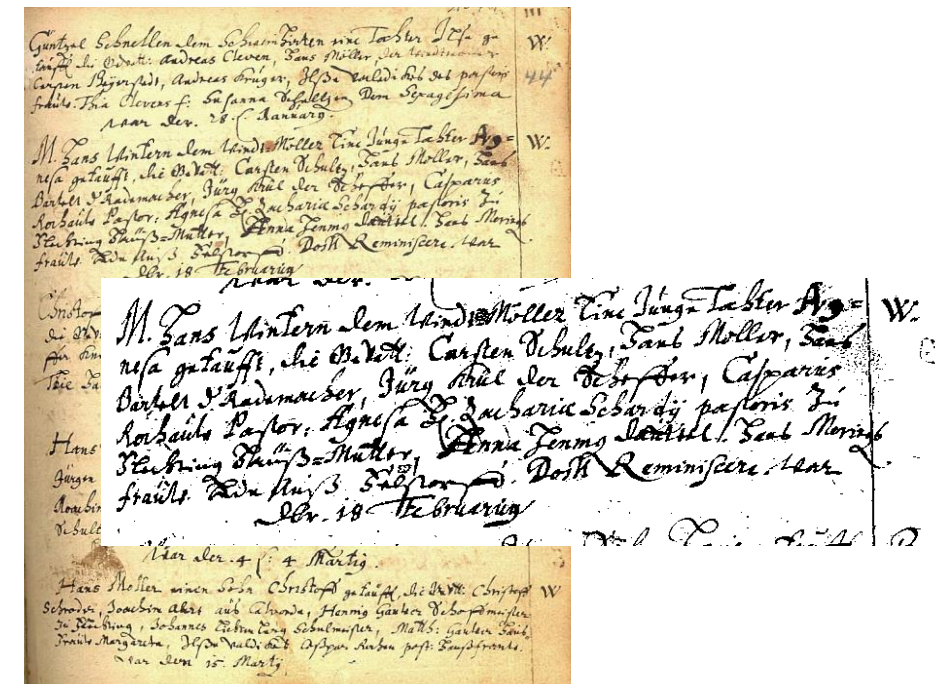




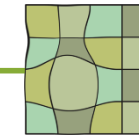


# Segmentierung

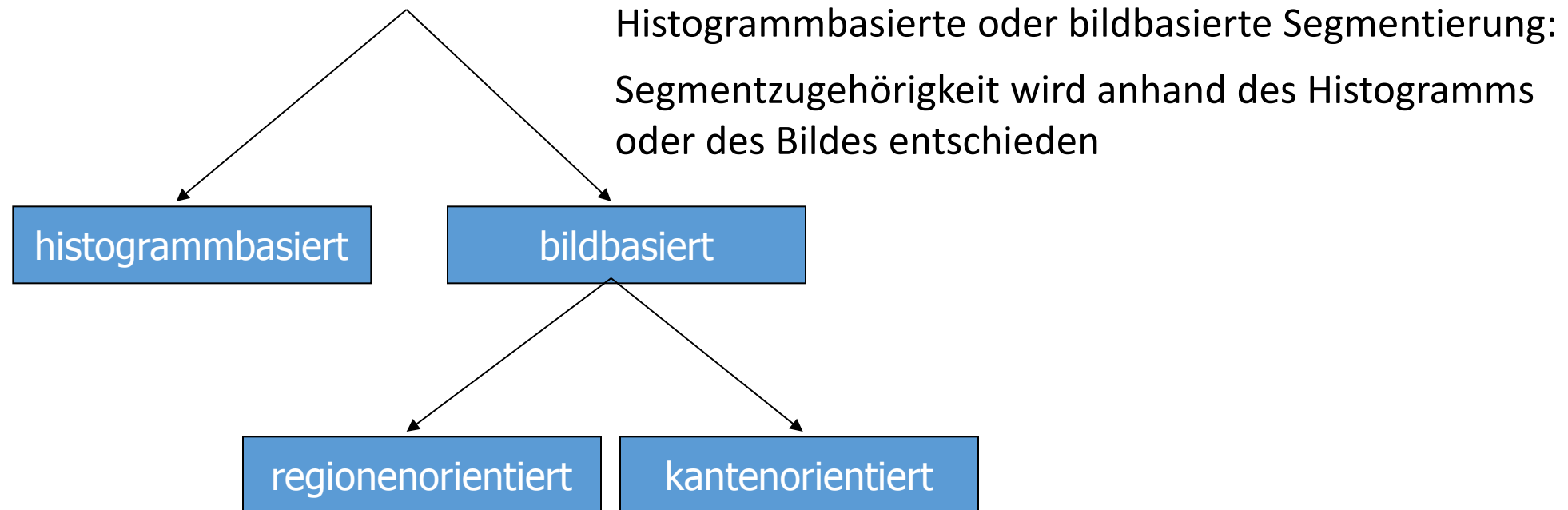
- Zerlegung eines Bildes in semantische Einheiten.
- **Segmente:** Träger der Bedeutung von Strukturen eines Bildes.
- Eigenschaften einer Segmentierung:
  - **vollständig:** jedes Pixel ist einem Segment zugeordnet.
  - **überdeckungsfrei:** ein Pixel ist genau einem Segment zugeordnet.
  - **zusammenhängend:** jedes Segment bildet ein zusammenhängendes Gebiet.



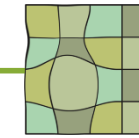
Bsp. Figure-Ground-Segmentation



# Datenbasierte Segmentierungsmethoden

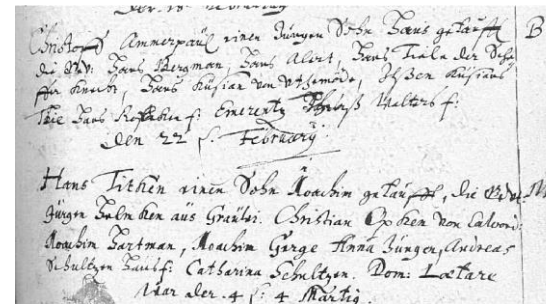


Regionen- oder Kantenorientierung:  
Segmente werden durch ihre Grenzen oder ihr Inneres definiert

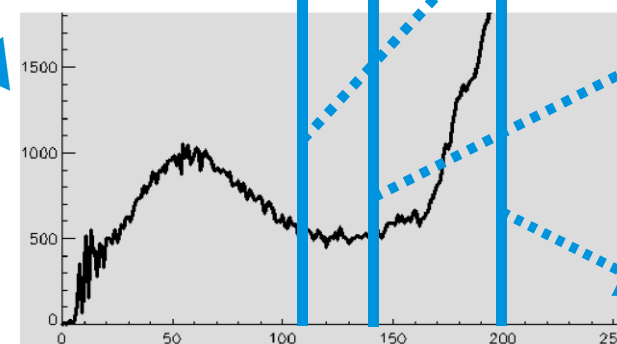


# Histogrammbasierte Segmentierung

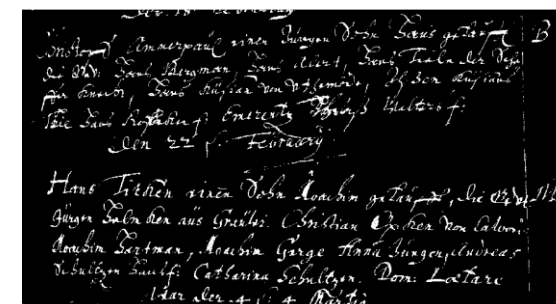
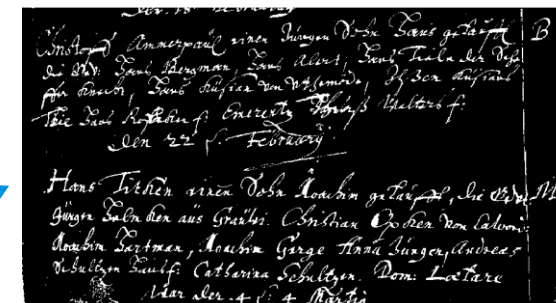
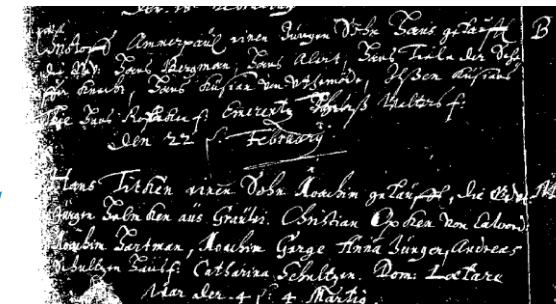
- **Annahme:** Bild besteht aus zwei Anteilen die sich durch ihren Grauwert unterscheiden.
- **Aufgabe:** Schwellenwert zwischen den beiden Grauwerten finden.
- $b(i, j) = \begin{cases} 1, & \text{falls } f(i, j) > T; \\ 0, & \text{sonst} \end{cases}$



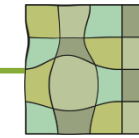
Schwellenwert



Histogramm

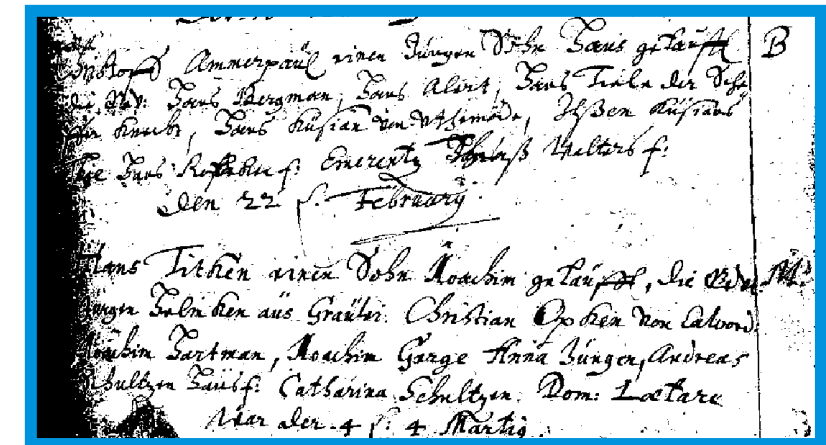
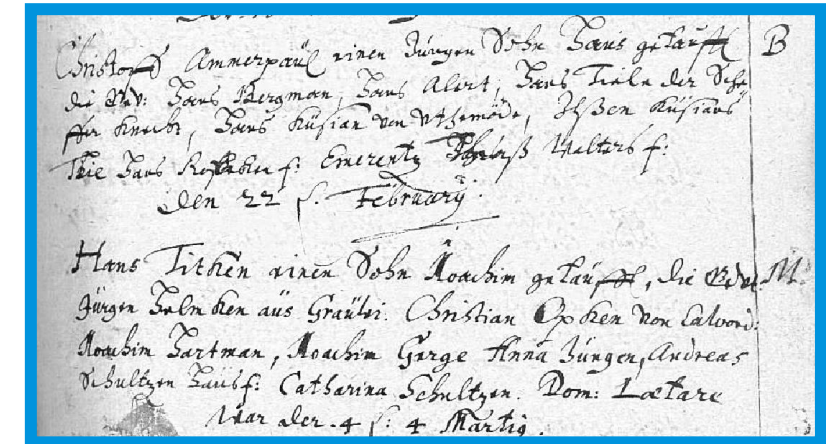






# Shading

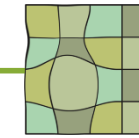
- Helligkeitsvariationen zerstören die bimodale Verteilung der Häufigkeiten.
- ► Schwellenwert ist nicht mehr für das gesamte Bild definierbar.





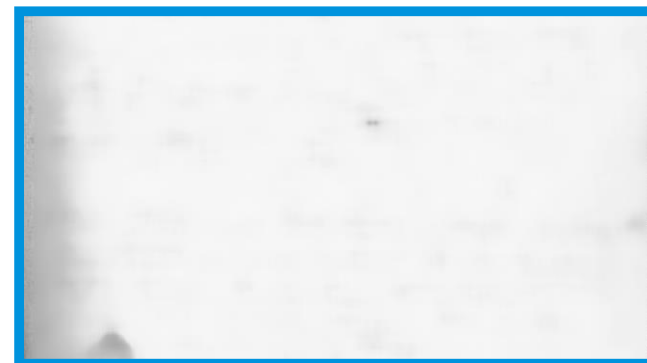
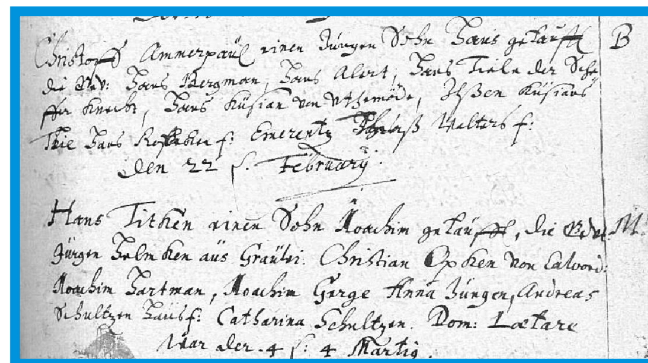
## Berücksichtigung von Shading

- Homogenes Bild unter gleichen Bedingungen aufnehmen.
- Shading-Bild aus dem Bild selbst bestimmen.
- Variierende Schwellenwerte.



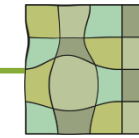
## Bestimmung des Shadingbilds

- Falls die überwiegende Mehrzahl des Bildes aus Vorder- oder Hintergrundpixeln besteht, kann das Shadingbild durch einem Rangordnungsfilter erzeugt werden.
- Das Rangordnungsfilter muss mindestens so groß sein, dass immer mindestens ein Vordergrund- und ein Hintergrundpixel enthalten ist.



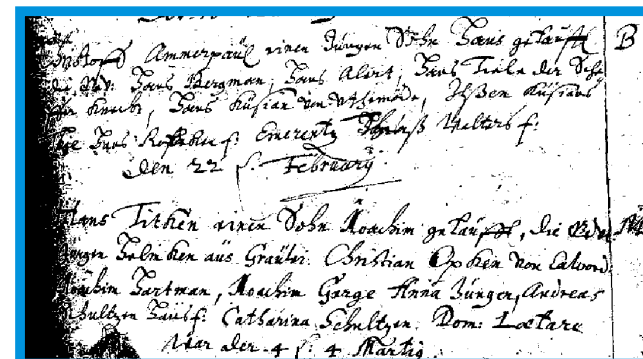
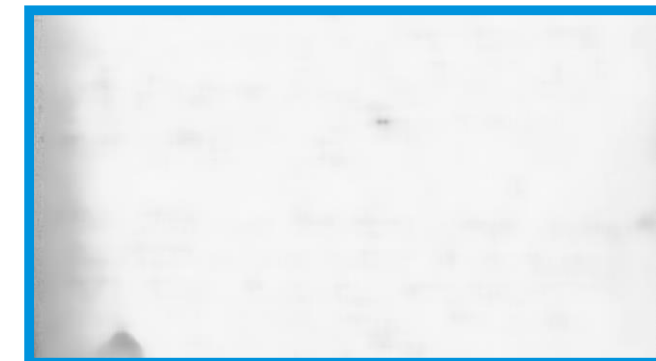
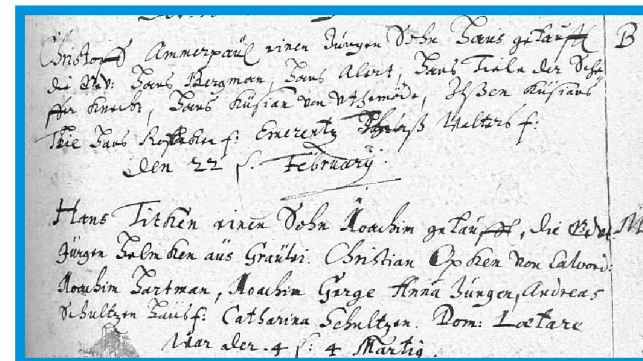
13×13 Maximumsfilter



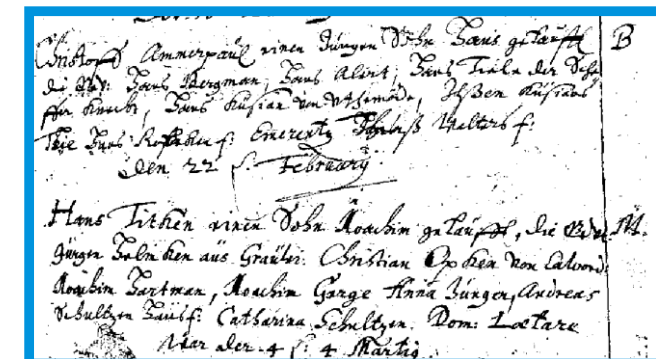


# Shading-Korrektur

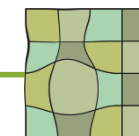
- Berechne aus dem Hintergrundsbild Shadingfunktion  $s(i, j)$
- Korrektur:  $f'(i, j) = f(i, j) \cdot s(i, j)$
- Segmentierung auf dem korrigierten Bild.



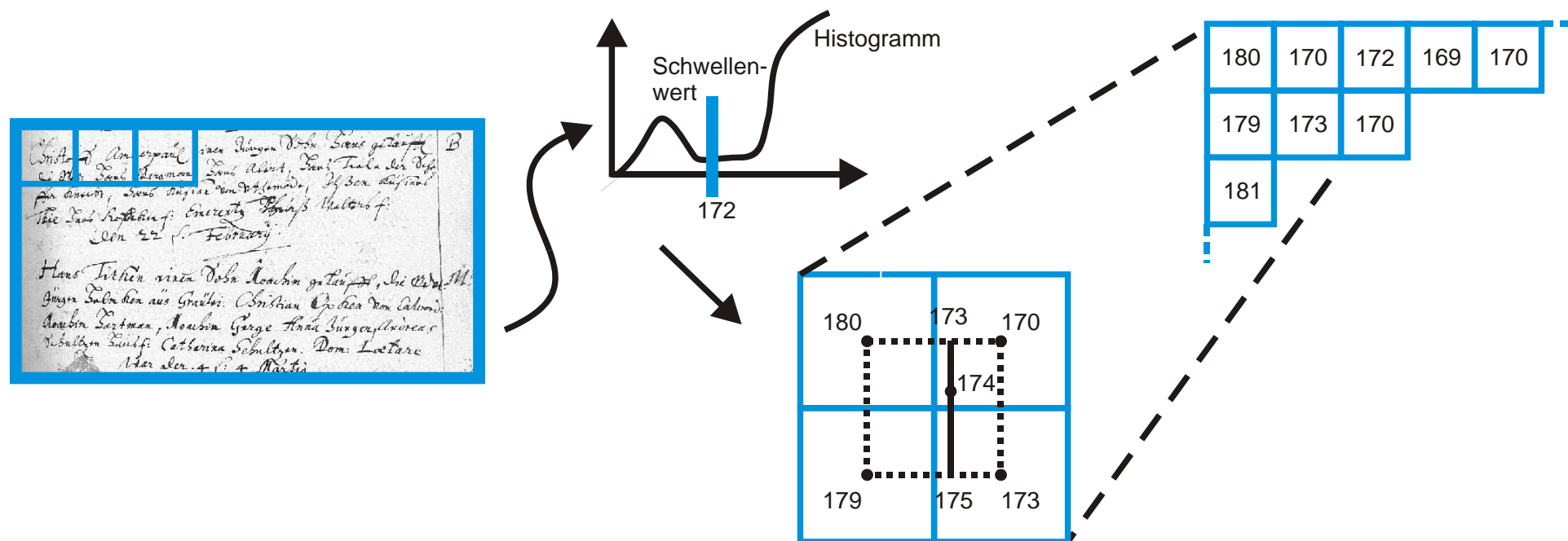
ohne Korrektur



mit Korrektur



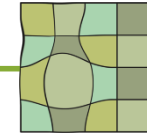
# Variierende Schwelle



Lokale Schwellenwerte aus Histogramme in Teilregionen.

(Lineare) Interpolation von Schwellenwerten  $T(i, j)$  an allen anderen Punkten.

Segmentierung durch  $f(i, j) > T(i, j)$ .

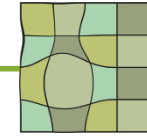


## Region Labeling

- Schwellenwert zerlegt das Bild in Vordergrund und Hintergrundsegmente.
- **Region Labeling** bestimmt Ort und Anzahl aller zusammenhängenden Gebiete im Binärbild  $b$ :

```
region.initialise() // Region der Größe M,N erzeugen und
label=1            // mit Null initialisieren, Startlabel=1
for (i,j) = 0, (M,N) do // Doppelschleife über i und j
    if region.labels(i,j) = 0 then // dieser Ort ist noch nicht
                                    Teil einer Region
        label = label+1            // neues Label vergeben
        region.flood_fill(i,j,label) // zusammenhängendes
                                    Gebiet um (i,j) mit
                                    Label füllen
```





# Flood Fill

```
flood_fill(i,j,label)  // Variablen zur Auswertung der
                        // Zusammenhangsbedingung sind global
                        // verfügbar

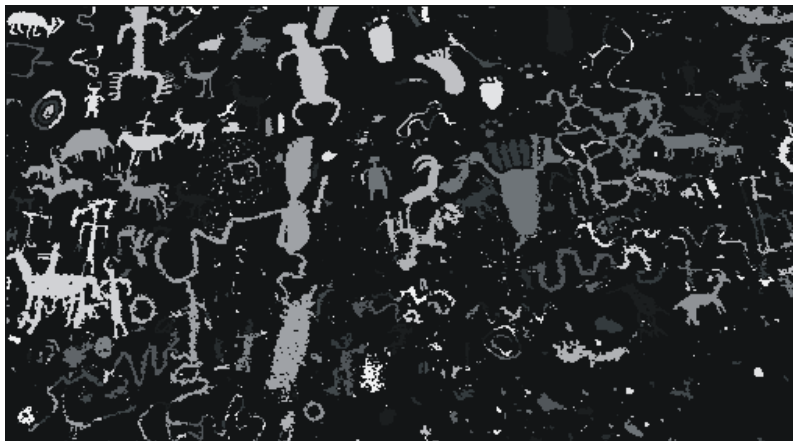
if f(i,j) erfüllt Zusammenhangsbedingung then
    region(i,j) = label    // Region an (i,j) mit Label
                        // versehen
    flood_fill(i-1,j,label) // Nachbarpixel untersuchen
    flood_fill(i,j-1,label)
    flood_fill(i+1,j,label)
    flood_fill(i,j+1,label)
```

Zusammenhangsbedingung:

- hat den gleichen Grauwert wie Saatpunkt  
oder
- ist kein Randpunkt



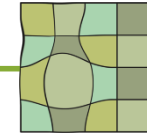
# Resultat



Originalbild

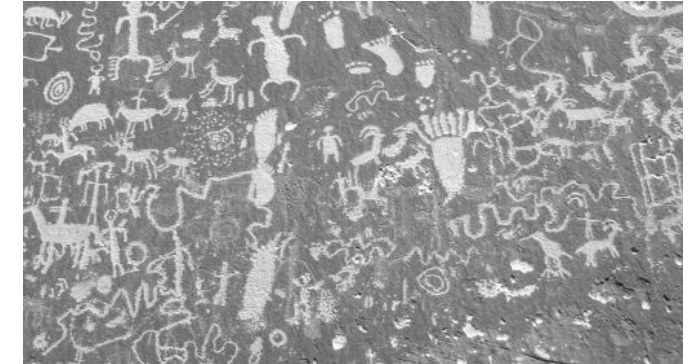
durch Schwellenwert erzeugtes  
Binärbild

Resultat des Region Labeling

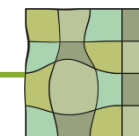


# Nachverarbeitung

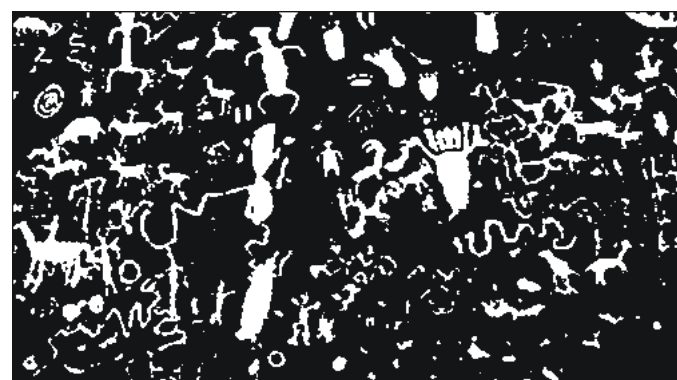
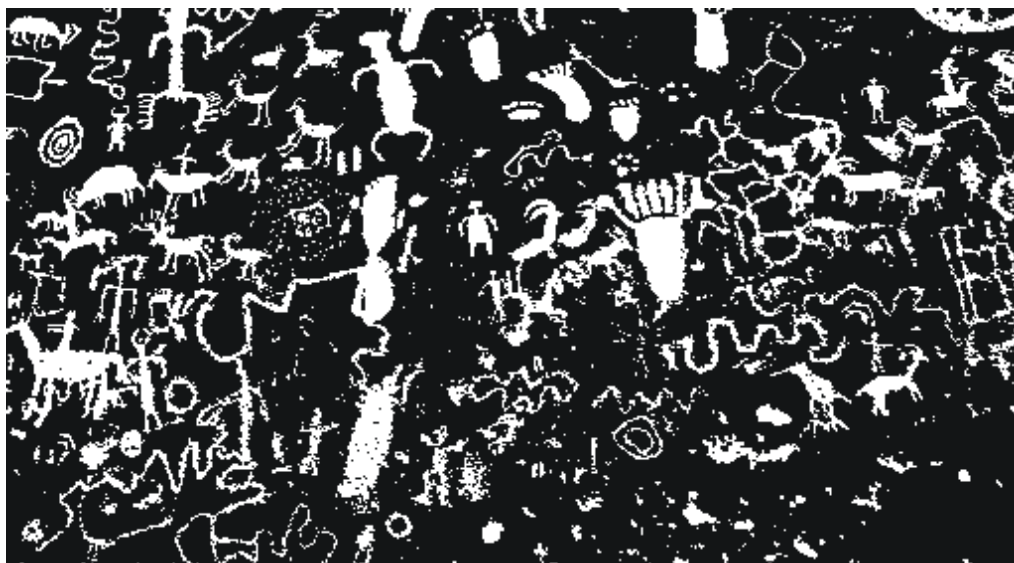
- Trennung nach Grauwerten wird nicht perfekt sein.
- Schwellenwertbild enthält falsche Regionen:
  - kleine fälschlicherweise als Segmente identifizierte Regionen.
  - Störungen am Rand von Regionen.
- Nachverarbeitung
  - Medianfilterung auf den Labeln
  - Entfernung von zu kleinen Regionen
  - Relaxation Labelling





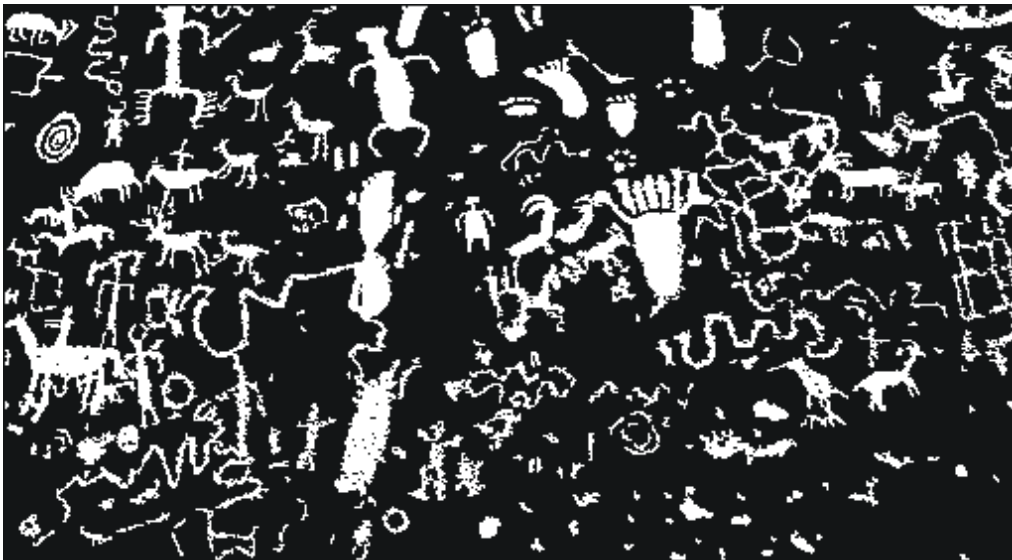


## Medianfilterung auf Labeln





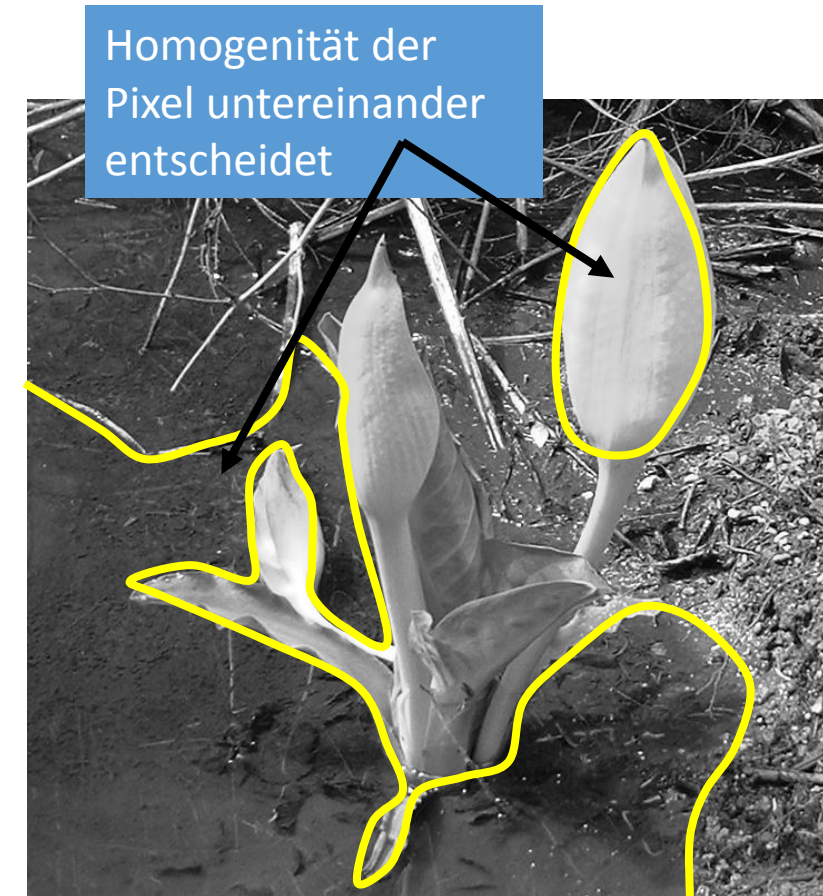
## Entfernung kleiner Gebiete





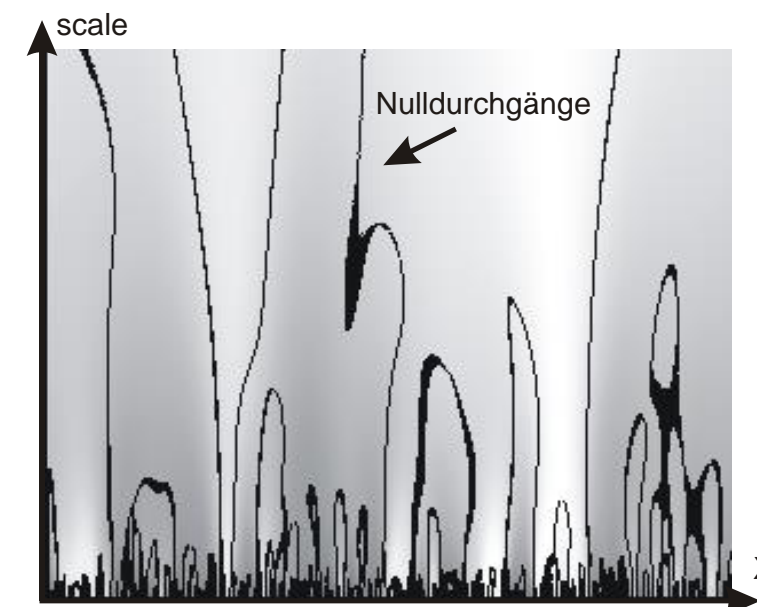
# Regionenbasierte Segmentierung

- **Homogenität** im Inneren des Segments.
- Homogenitätsbedingung wird **bei der Segmentierung** ausgewertet.
- Homogenität ist **relativ** zu den Attributen eines Segments definiert.
- Globale Zusammenhänge über **Multiskalenstrategie**.



# Multiskalenstrategie

- relative Kriterien für Homogenität können über unterschiedliche Entfernungen verschieden wirken.
- Segmentierung nach Multiskalenstrategie wertet Kriterien auf unterschiedlichen Skalierungen aus.
- Multiskalenstrategie
  - explizit auf einer Multiskalenrepräsentation
  - implizit in den Segmentierungsalgorithmus integriert.



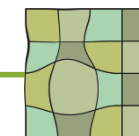




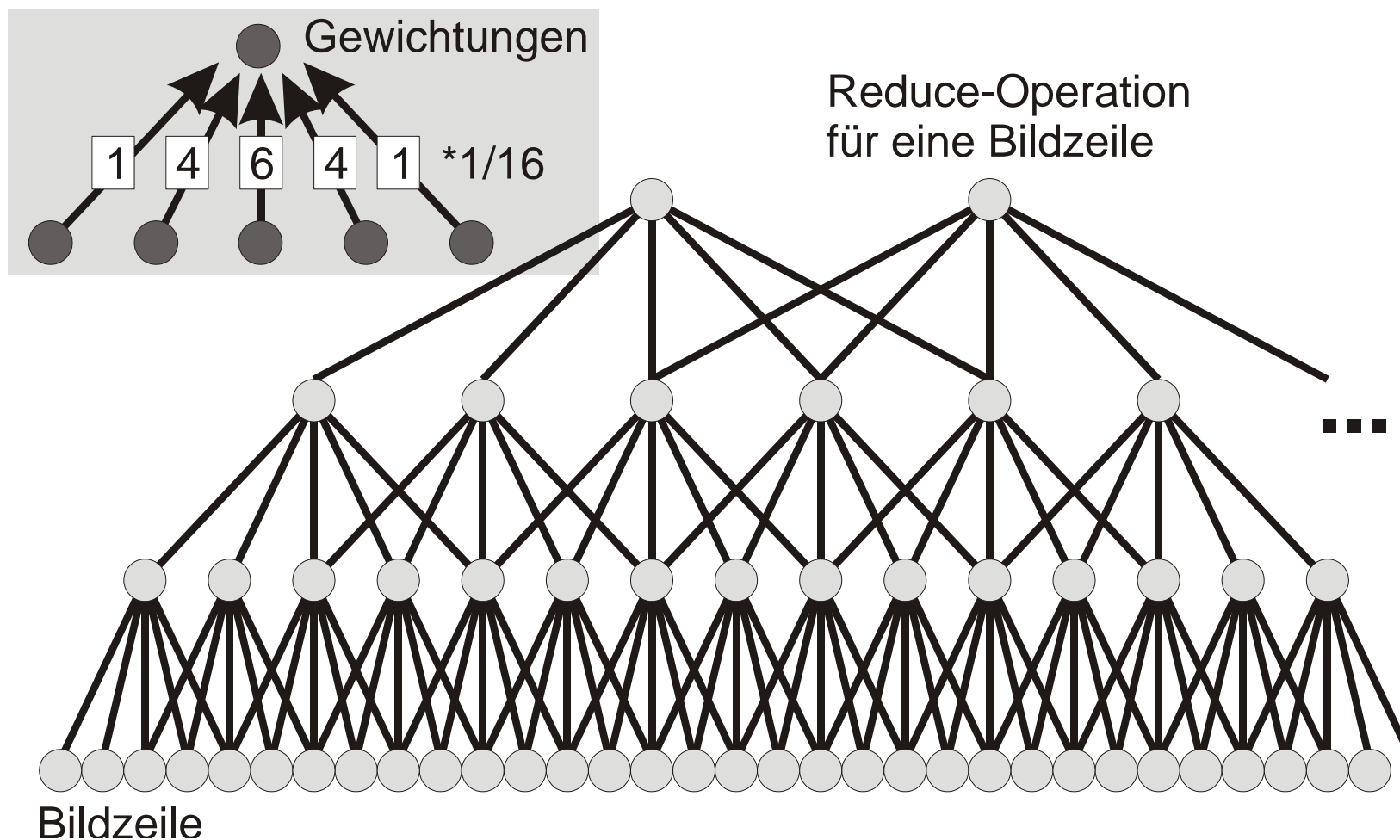
# Gaußpyramide

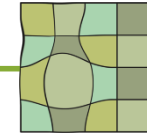
- Das Originalbild wird fortlaufend durch eine „reduce“-Operation skaliert.
- Jedes Pixel der nächsthöheren Skalierungsstufe repräsentiert 4 Pixel der aktuellen Stufe.
- Vor der Reduktion wird der Frequenzumfang durch Filterung vermindert:

- Gaußfilter  $\frac{1}{16} (0.87 \quad 3.91 \quad 6.44 \quad 3.91 \quad 0.87)$
- Binomialfilter  $\frac{1}{16} (1 \quad 4 \quad 6 \quad 4 \quad 1)$



# Gaußpyramide



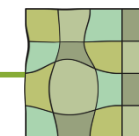


## Expand-Operation

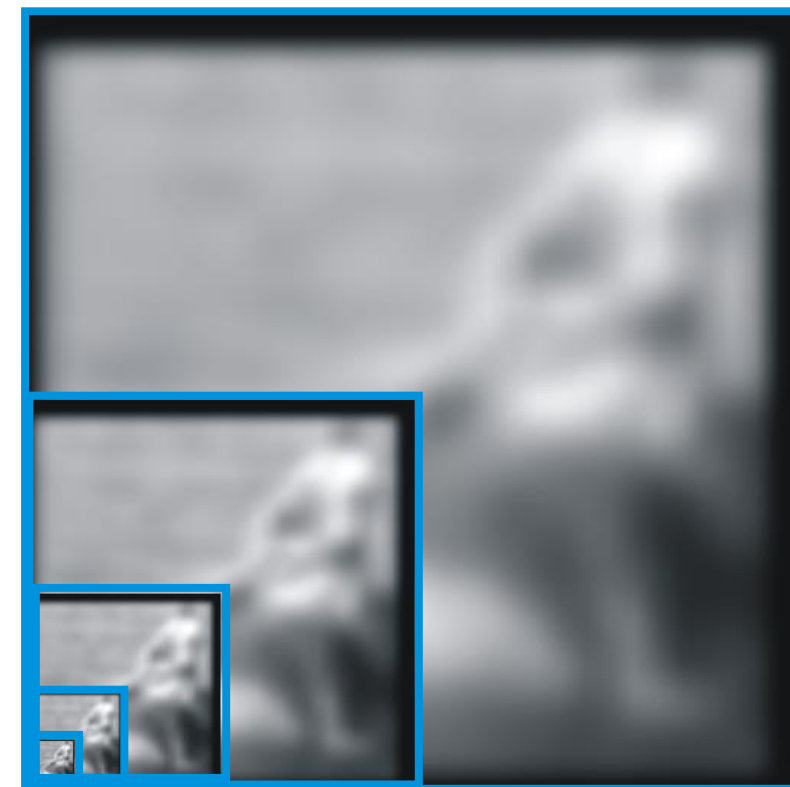
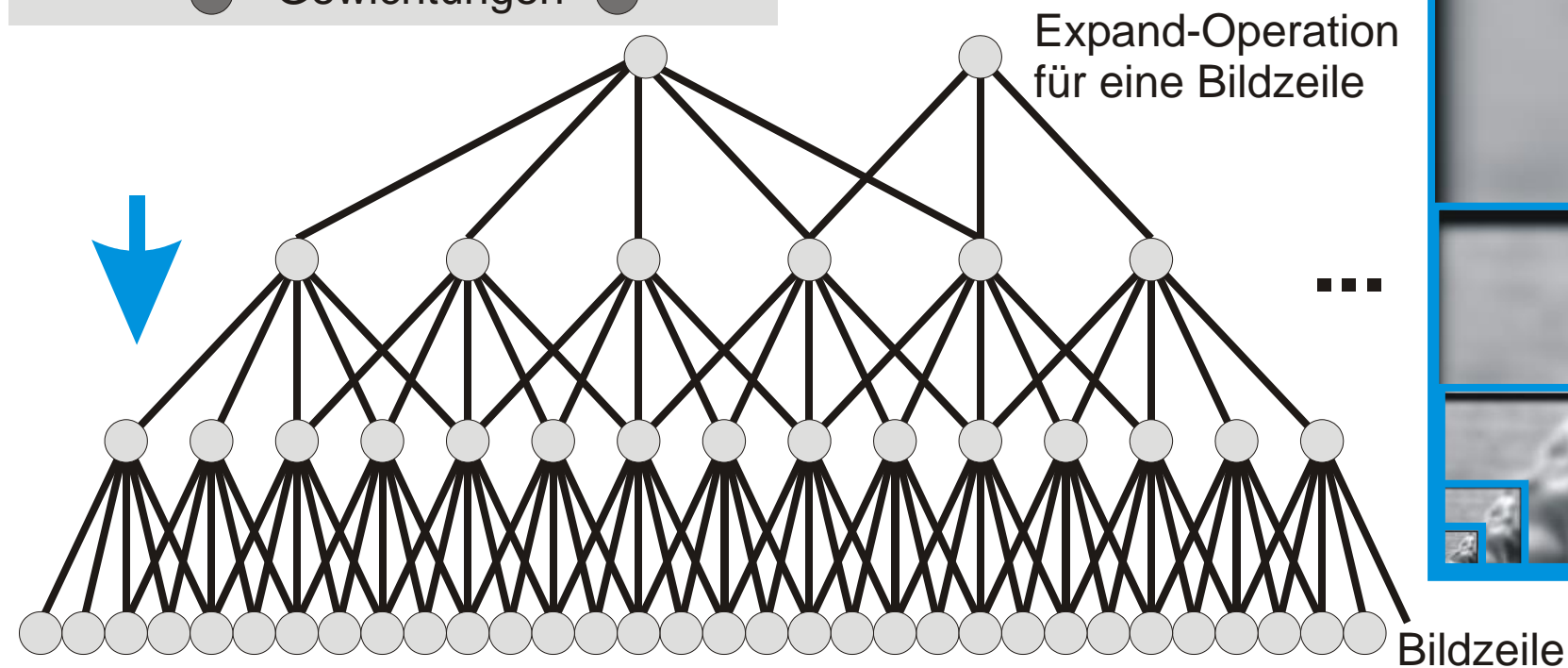
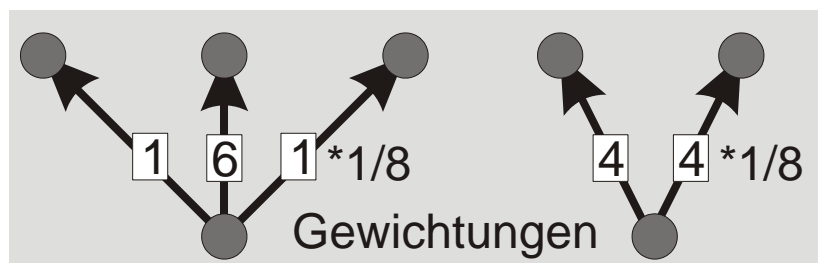
- Um die vorherige Skalierungsstufe zu erzeugen, wird eine „expand“-Operation definiert.
- Pixel der neuen Skalierungsstufe werden durch Interpolation erzeugt:
  - Pixelorte, die auf beiden Skalierungsstufen existieren:
  - Pixelorte, die nur auf der vorherigen Skalierungsstufe existieren:

$$\frac{1}{8.18} (0.87 \quad 6.44 \quad 0.87) \text{ bzw. } \frac{1}{8} (1 \quad 6 \quad 1)$$

$$\frac{1}{7.82} (3.91 \quad 3.91) \text{ bzw. } \frac{1}{8} (4 \quad 4)$$

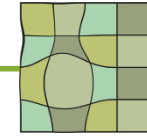


## Expand-Operation



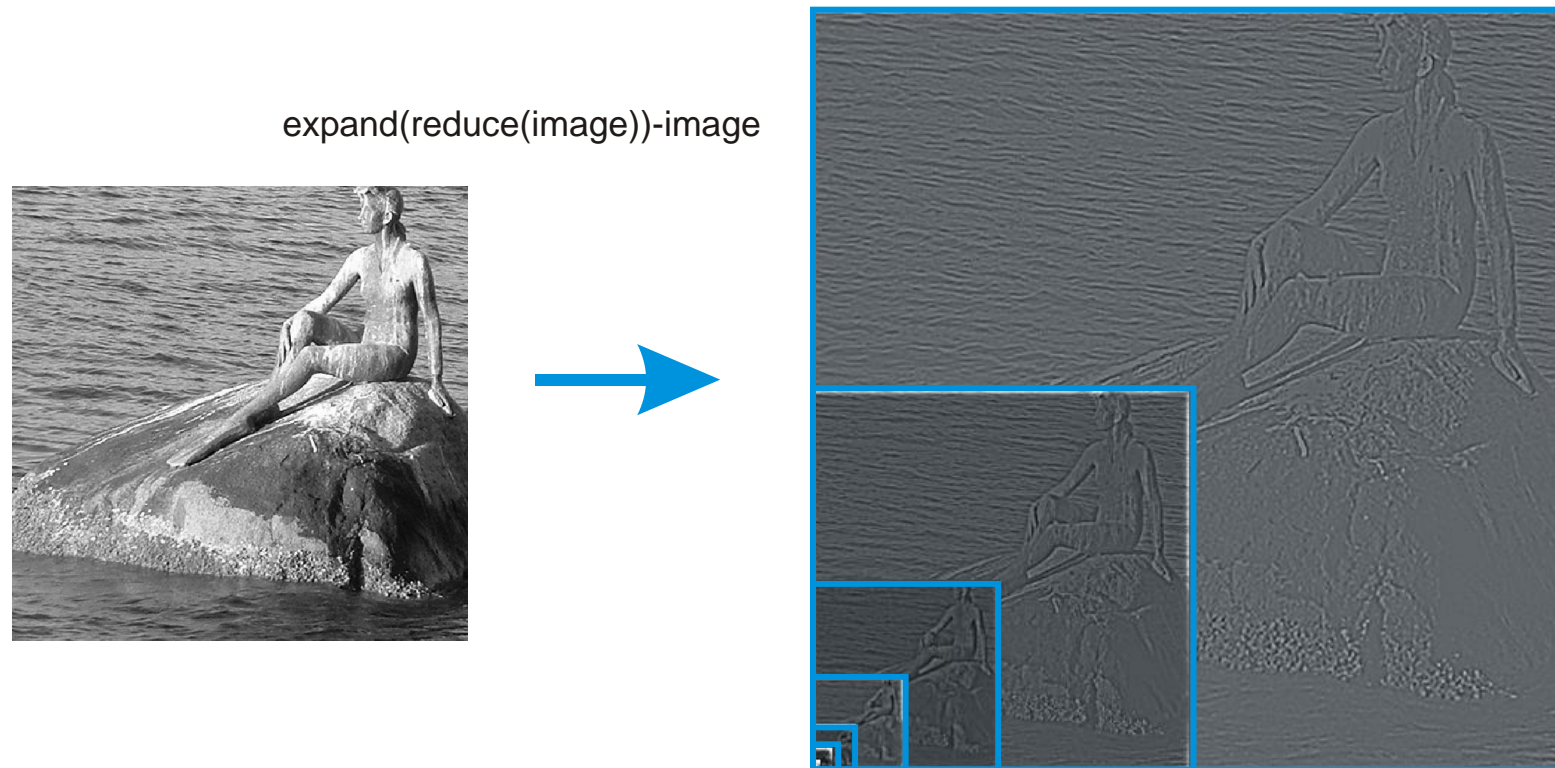
Die Expand-Operation ist nicht verlustfrei

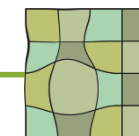




# Laplace-Pyramide

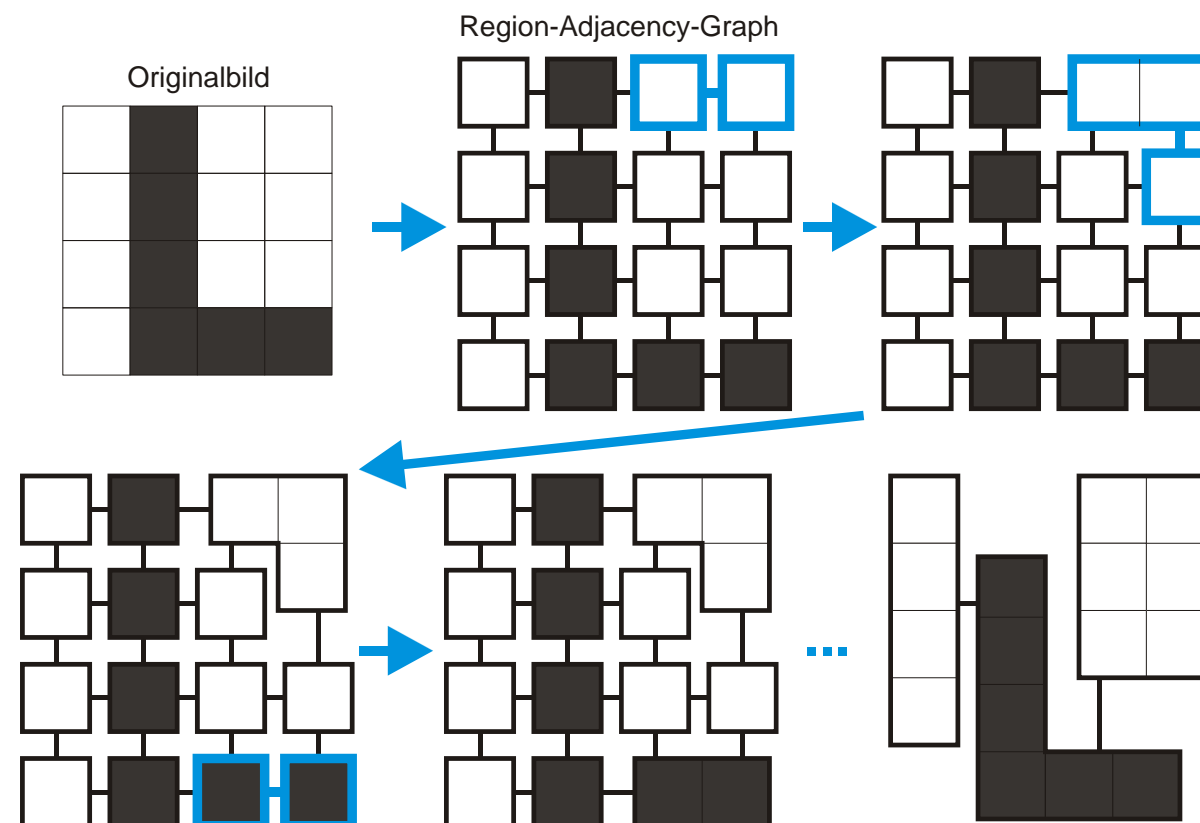
- Jede Skalierungsstufe  $s$  enthält nur den Unterschied  $f_s - \text{expand}(\text{reduce}(f_s))$
- Redundanzfreie Repräsentation

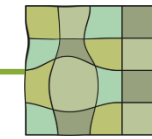




## Region Merging

- Initial wird jedes Pixel zu einem Segment erklärt.
- Zwei benachbarte Regionen werden zusammengefasst, wenn sie auch gemeinsam das Homogenitätskriterium erfüllen.
- Die Segmentierung ist beendet, wenn keine zwei Regionen mehr existieren, die zusammengefasst werden können.
- Zwischenergebnisse werden in einem **Region Adjacency Graph (RAG)** gespeichert.





# Region Merging

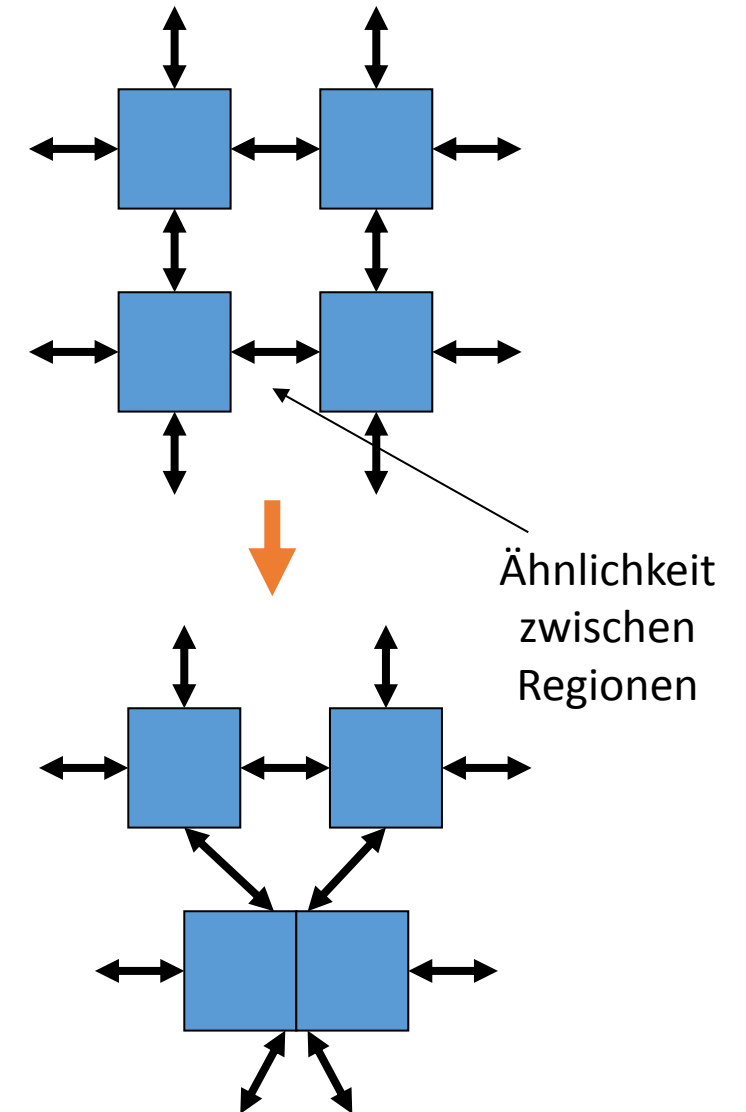
„von Pixeln zu Regionen“:

```
stopMerge = false
while not stopMerge do
  (r1,r2) = MaxSimilarity(region)
  if sim(r1,r2)>T then
    region.merge(r1,r2)
  else
    stopMerge=true
```

Ähnlichkeitskriterium:

z.B., maximaler Grauwertunterschied zwischen Pixeln von r1 und r2.

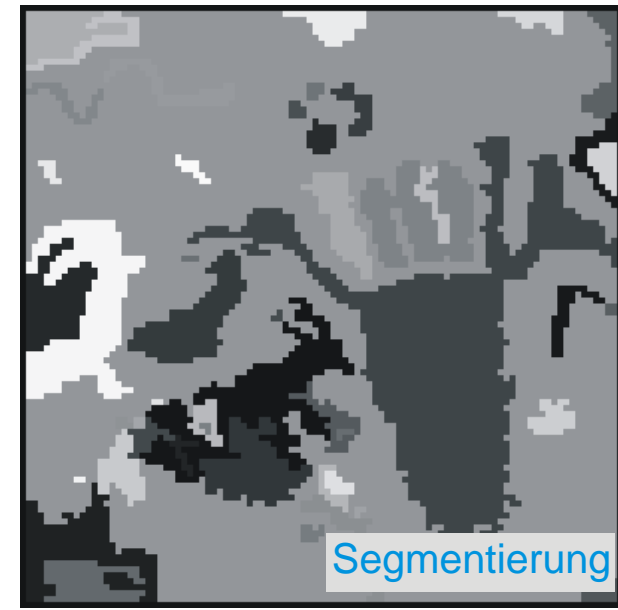
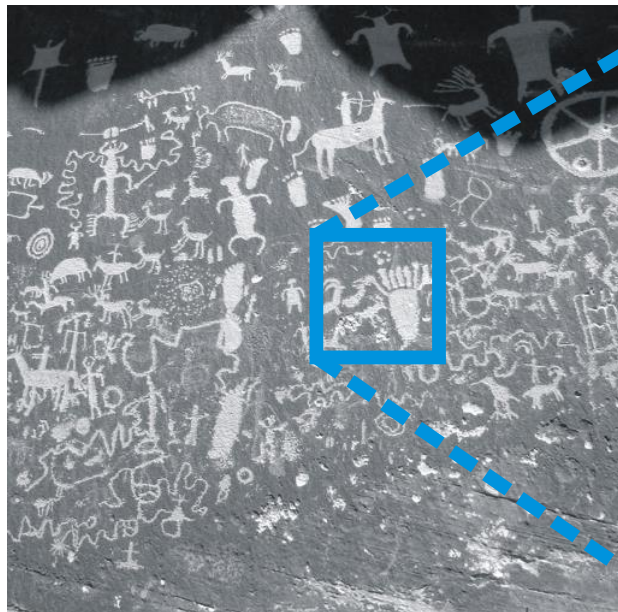
Region Labeling kann integriert werden.





## Region Merging

- Homogenitätskriterium
  - Grauwertdifferenz innerhalb der Region.
  - Wahrscheinlichkeit, dass beide Regionen die gleichen Grauwertverteilungen haben.







# Region Merging und Multiskalenstrategie

Modellannahme:

Die gröbste Skalierungsstufe, auf der sich segmentierungsrelevante Eigenschaften manifestieren, ist bekannt.

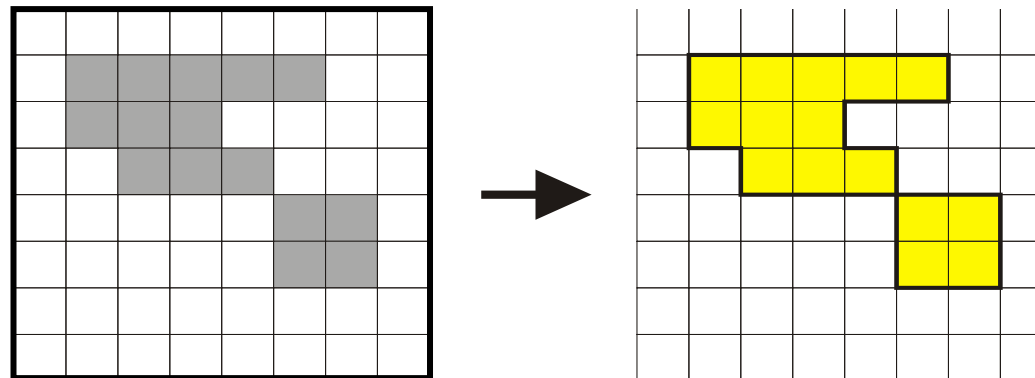
Prozess:

- Region Merging auf grober Skalierung
- Übertragung des Resultats auf die nächstfeinere Stufe.
- Alle Pixel, die zu Pixeln eines anderen Segments benachbart sind, werden nochmals geprüft.
- Verfahren endet, wenn die die feinste Skalierungsstufe erreicht ist



## Split & Merge-Algorithmus (Regionenbasiertes Verfahren)

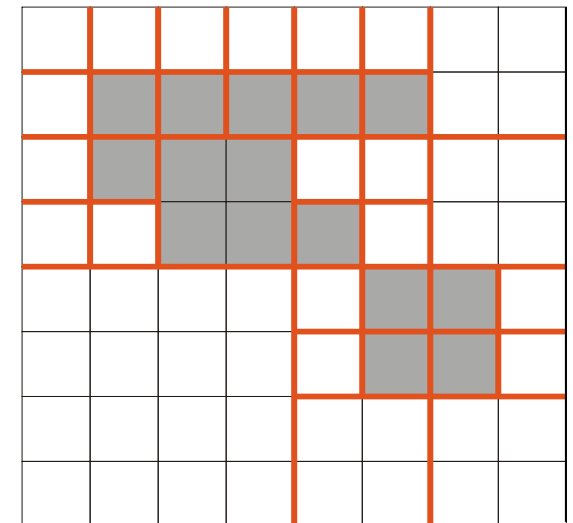
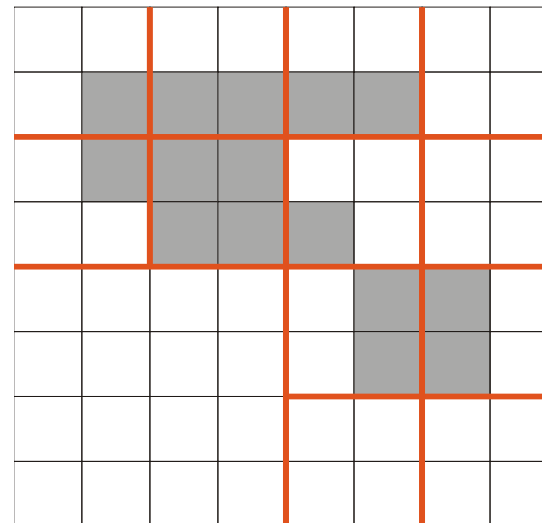
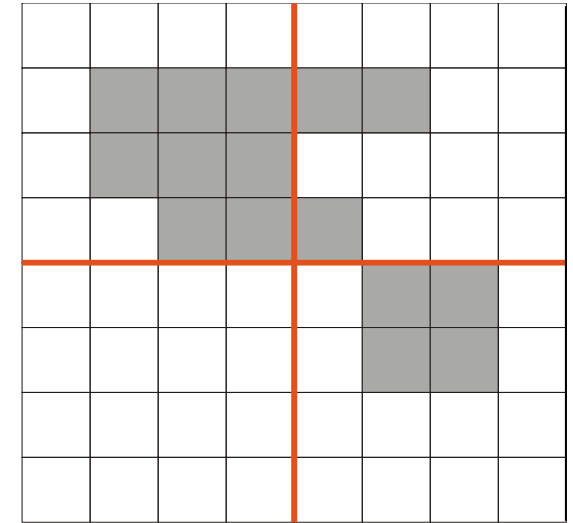
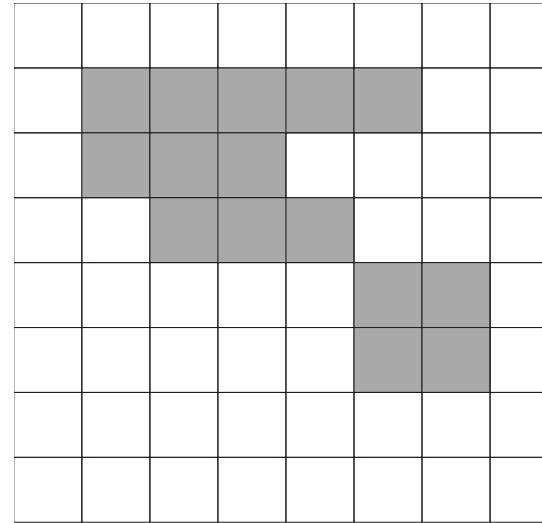
- **Startbedingung:** Das gesamte Bild ist ein Segment.
- Ein Segment wird solange in 4 Untersegmente zerlegt, wie es ein gegebenes Homogenitätskriterium nicht erfüllt
- Benachbarte Segmente werden zusammengefasst, wenn sie auch nach der Zusammenfassung das Homogenitätskriterium erfüllen.
- **Resultat** ist eine vollständige, überdeckungsfreie Zerlegung des Bildes (Segmentierung gemäß Definition)

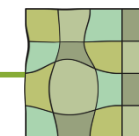




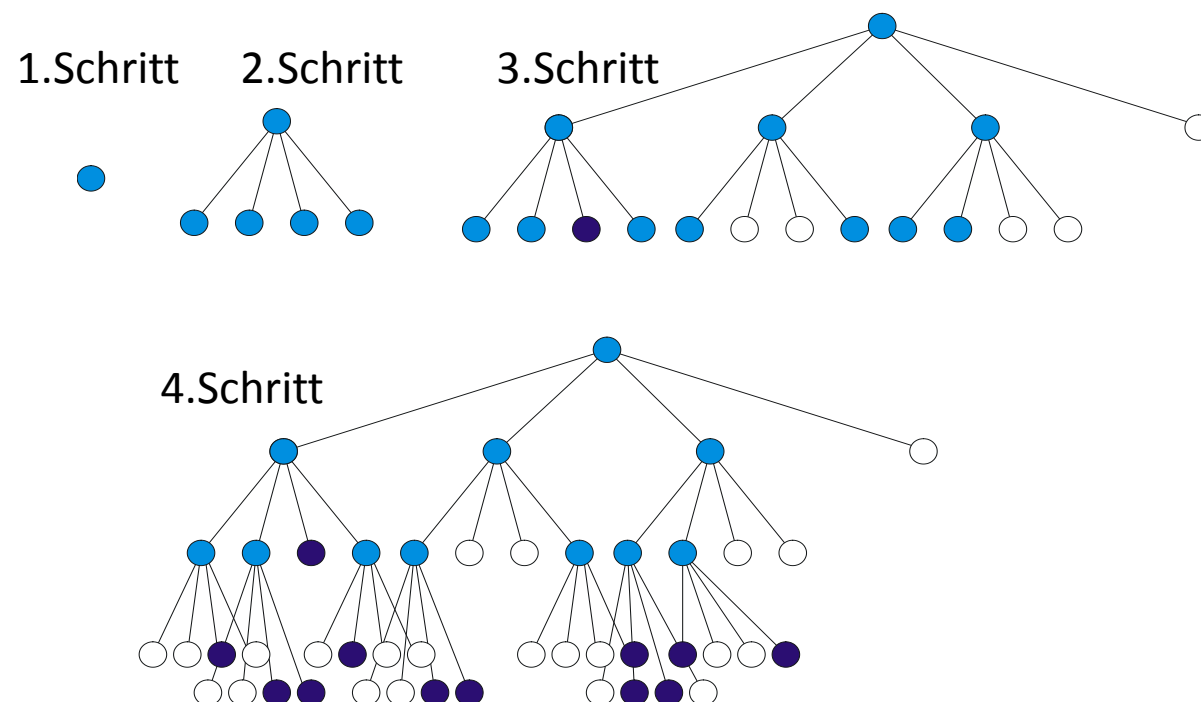
## Zerlegungsschritt

- Zerlegung terminiert spätestens auf Pixelebene.
- Problem: Datenstruktur zur Dokumentation der aktuellen Zerlegung

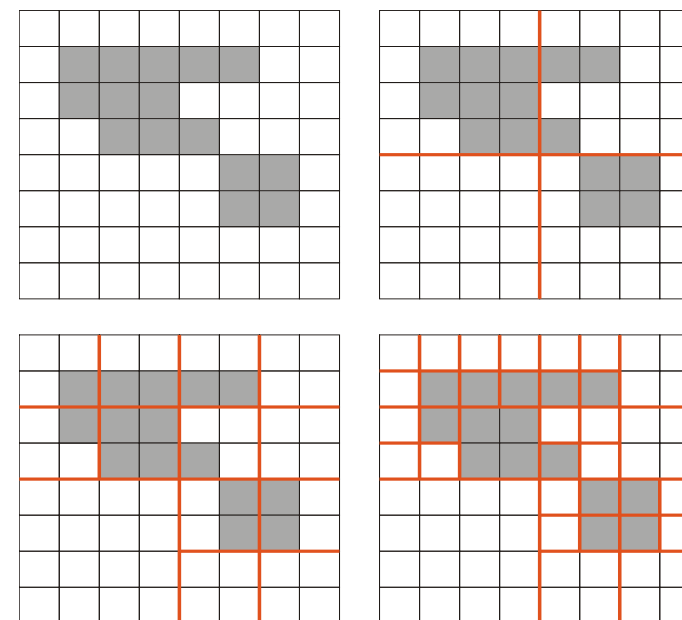




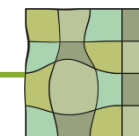
## Zerlegungsschritt (Quad-Tree Repräsentation)



Wert des Homogenitätsmerkmals  
einer Region wird im entsprechen-  
den Blatt des Quad-Tree abgelegt



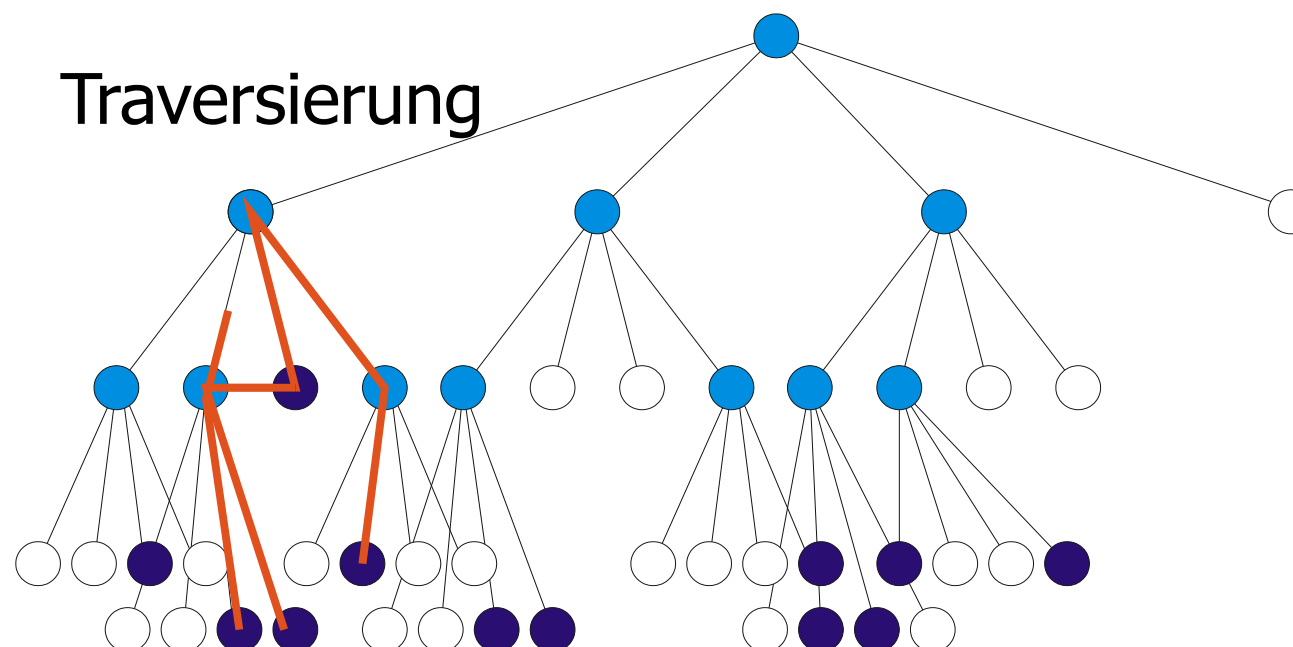
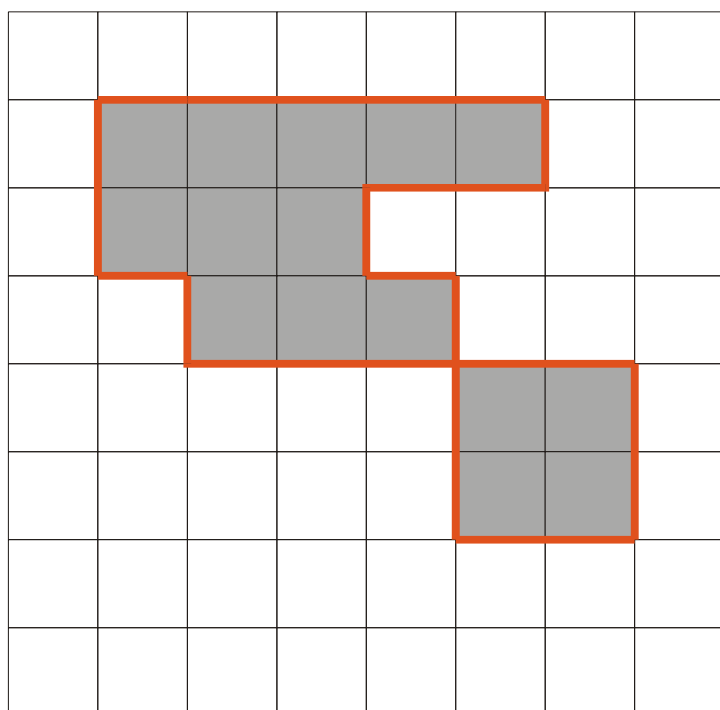




# Merging

Quadtree wird traversiert und *in einen RAG überführt*.

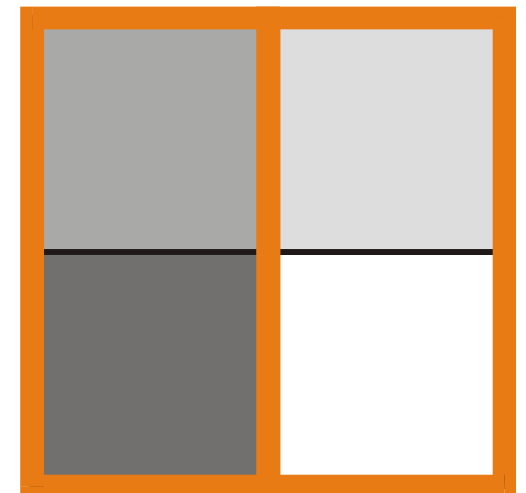
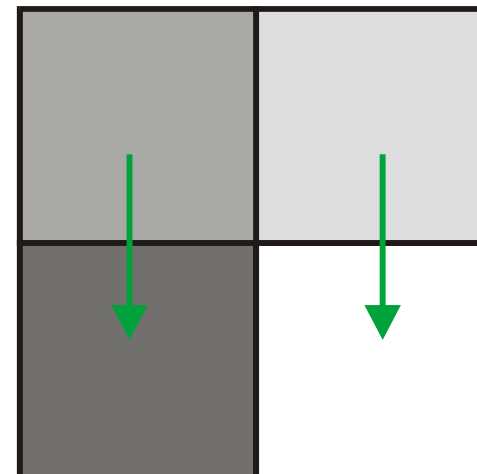
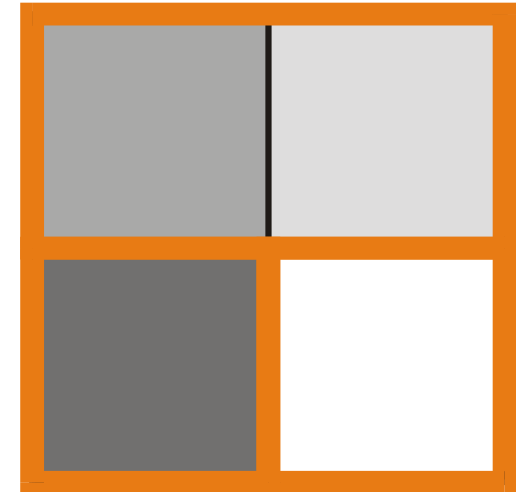
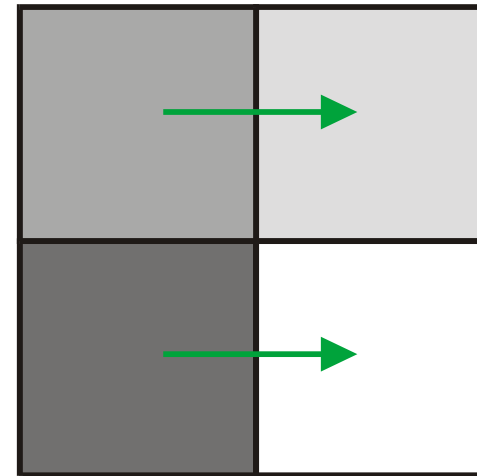
Auf dem RAG wird ein Region Merging durchgeführt.





## Split & Merge

- Resultat: Zerlegung des gesamten Bilds in Regionen.
- Multiskalenstrategie ist integriert
- Homogenitätsmerkmale wie bei Region Merging
- Probleme (Region Merging und Split & Merge):
  - Merge-Schritt ist bei relativen Homogenitätsmerkmal **nicht immer eindeutig**.
  - **Keine absoluten Merkmale** möglich.





## Was Sie heute gelernt haben sollten?

- Schwellenwertsegmentierung
- Region Labelling, Shadingkorrektur und Nachverarbeitung
- Gauß- und Laplacepyramide
- Region Merging
- Split and Merge



## Famous Last Question

**Wie könnte man diese  
Mikroorganismen  
segmentieren?**

