Vision and Image Processing: Segmentation

François Lauze
Department of Computer Science
University of Copenhagen

Plan for today

- A general introduction of segmentation.
- Marr-Hildreth, Canny.
- Closing gaps with Snakes.
- Segmentation and Clustering.
- Introduce the notion of spatial regularization.

Outline

Image Segmentation

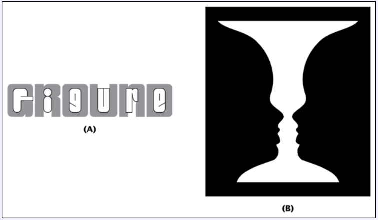
- An intelligible image is not formed of random pixels. There must fundamental consistencies between them.
- Image segmentation is the process of dividing an image into coherent regions of similarity, called segments, by grouping similar pixels:
    - Region: a group of connected pixels that share some common properties.
    - Property: intensity, color, texture, motion (for sequences), boundary (edges)
    - Some of these properties can be defined in a pixelwise manner: intensity, color for instance. Some are related to pixel neighborhoods, texture for instance.
    - Segments may be grouped together into complex objects.

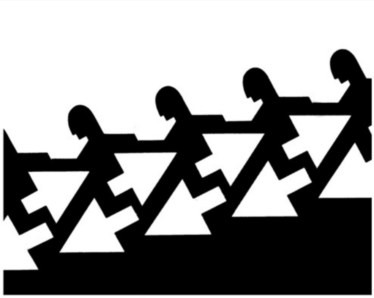
A bit of Gestalt Theory: Grouping

- Figure–Ground organization.
  - Perceptual apparatus picks out some objects to be figures, while other are less relevant in the background.
- Grouping
  - Inherent properties from the stimulus environment lead people to group them together.
  - Grouping principles
    - Proximity: group nearby figures together.
    - Similarity: group figures that are similar.
    - Continuity: perceive continuous patterns
    - Closure: fill-in gaps
    - Connectedness: spots, lines and areas are seen as unit when connected.
    - Synchronicity: occur at the same time
    - Common region: located within some boundary
    - Connectedness: connected by other elements.

Slide adapted from "Sensation and Perception" at `http://college.cengage.com`

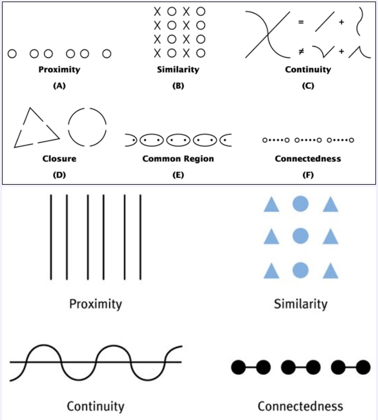
Figure/Ground





Slide adapted from “Sensation and Perception” at `http://college.cengage.com`

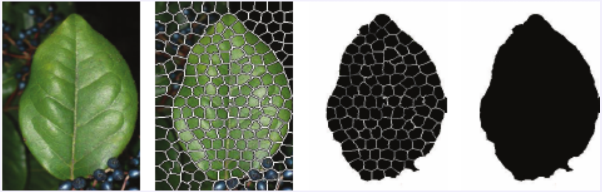
Grouping



Slide adapted from “Sensation and Perception” at `http://college.cengage.com`

Operationalization

- Similarity and Common Regions principles. Detect coherent regions, a.k.a segments. Group them further.



- Two different segments should have dissimilar properties. In particular, boundary between segments should present large variations.
- Two main approaches:
  - Region based segmentation.
  - Edge / contour based segmentations.

Regions and Edges

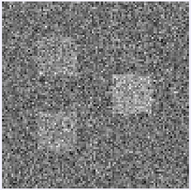
- A region should be bounded by a closed contour: edge detection.
- Regions may be obtained by “boundary filling”.

- However edges are not always well defined in observed images.

Regions and Edges

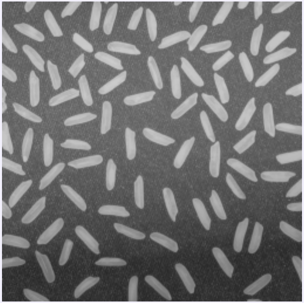
- A region should be bounded by a closed contour: edge detection.
- Regions may be obtained by “boundary filling”.

- However edges are not always well defined in observed images.



Regions

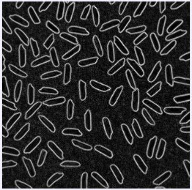
- Regions correspond generally to objects or pieces of objects in a scene.
- Scenes may contain several objects.



Outline

Edge Based Segmentation

- Boundary between region corresponds to sharp intensity / color / texture change.
- Derivatives should indicate this:



- Edges should correspond to local maximum of gradient magnitude.
- Gradient of f : $\nabla f = (f_x, f_y)^T$, Gradient magnitude = $\sqrt{f_x^2 + f_y^2}$.
- Zero-crossing of Laplacian $\Delta f = \nabla^2 f := f_{xx} + f_{yy}$ is also an indicator.
- How to compute image derivatives / gradient?

Image Derivatives - First Order

- Finite difference approximation:
  - Forward differences
    $$\frac{\partial f(x,y)}{\partial x} \approx \frac{f(x+h,y)-f(x,y)}{h}, \quad \frac{\partial f(x,y)}{\partial y} \approx \frac{f(x,y+h)-f(x,y)}{h}$$
  - Backward differences
    $$\frac{\partial f(x,y)}{\partial x} \approx \frac{f(x,y)-f(x-h,y)}{h}, \quad \frac{\partial f(x,y)}{\partial y} \approx \frac{f(x,y)-f(x,y-h)}{h}$$
  - Central differences
    $$\frac{\partial f(x,y)}{\partial x} \approx \frac{f(x+h,y)-f(x-h,y)}{2h}, \frac{\partial f(x,y)}{\partial y} \approx \frac{f(x,y+h)-f(x,y-h)}{2h}$$
- For those familiar with it, they can be implemented by convolution!

Second Order Derivatives

- *xx*-direction

$$\frac{\partial^2 f(x,y)}{\partial x^2} \approx \frac{f(x+h,y) - 2f(x,y) + f(x-h,y)}{h^2},$$

- *yy*-direction

$$\frac{\partial^2 f(x,y)}{\partial y^2} \approx \frac{f(x,y+h) - 2f(x,y) + f(x,y-h)}{h^2},$$

- *xy*-direction

$$\frac{\partial^2 f(x,y)}{\partial x \partial y} \approx \frac{1}{4h^2}\Big(- f(x-h,y+h) + f(x+h,y+h) + f(x-h,y-h) - f(x+h,y-h)\Big)$$

- Here too, convolution!

Derivatives and noise

- Derivation filters are high pass: they amplify high pass signals and noise is high pass.
- Solution: Apply smoothing filter prior to differentiation.
- Better: combine smoothing filter and differentiation!
- Remember Søren's lectures!

$$D(g * f) = Dg * f$$

Derivatives and Gaussian, Again Søren's lectures

- Convolution by Gaussian smooths: noise removal. Standard deviation of Gaussian directly linked to scale of features.
- Derivative of Gaussian:

$$g_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad \frac{\partial g_\sigma}{\partial x} = -\frac{x}{\sigma^2} g_\sigma, \quad \frac{\partial g_\sigma}{\partial y} = -\frac{y}{\sigma^2} g_\sigma$$

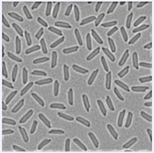
- Laplacian of Gaussian (LoG)

$$\Delta g_\sigma = \frac{\partial^2 g_\sigma}{\partial x^2} + \frac{\partial^2 g_\sigma}{\partial y^2} = \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} g_\sigma$$

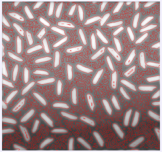
Marr-Hildreth Edge Detector

D. Marr and E. Hildreth, 1980.

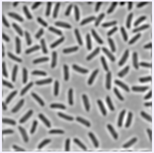
- Observe zero-crossing of Laplacian of Gaussian (the same use for SIFT for instance)
- Convolve image f with Laplacian of Gaussian filter: $\Delta_\sigma f = \Delta g_\sigma * f$
- Detect the zero-crossings of $\Delta_\sigma f$



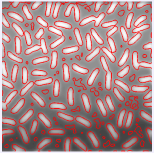
LoG $\sigma = 1.4$



detected edges



LoG $\sigma = 3.0$



detected edges

Canny Edge Detector

J. Canny, 1986.

- Develop an algorithm optimal wrt:
  - Detection: maximize probability of detection of true edges, minimize probability of detection of false ones.
  - Localization: detected edges as close as possible to real ones.
  - Number of responses: 1 real edge should not result in 2 or more detected ones.
- Canny algorithm attempts to answer optimally to these criteria.
- Divided in 5 steps.

Algorithm - I

1. Smoothing. Noise removal by Gaussian smoothing with 2D-Gaussian g_σ.
2. Gradient computation. Can be combined with step 1 by convolution with Gaussian derivatives. Gives $f_{x\sigma}$ and $f_{y\sigma}$. Potential edges marked as gradients with large magnitudes. Directions computed by

$$\theta = \arctan\left(\frac{|f_{y\sigma}|}{|f_{x\sigma}|}\right).$$

3. Non-maximum suppression.
   - Round gradient direction to nearest 45° (connectivity of a 8-points neighborhood).
   - Compare gradient magnitude of current pixel with magnitudes of neighbors in the gradient direction. That is, if $\theta = 90°$ compare with north and south pixel, if $\theta = 135°$ compare with ? (this is a question for you!)
   - If gradient magnitude of the current pixel is largest, keep it, otherwise suppress it (say set to 0).

Algorithm - II

1. Double thresholding. Non suppressed edges marked with their gradient magnitude. Some are true edges, some may be noise. Two thresholds τ_{high} and τ_{low} are used. Edges with magnitude $\geq \tau_{high}$ marked as strong edges. Edges with magnitude between τ_{low} and τ_{high} marked as weak edges. Remaining are suppressed.

2. Hysteresis edge tracking Strong edges are certain and included in the final image. Weak edges included only if connected (via other weak edges) to strong edges. Connectivity means existence of a chain of neighbor weak edges pixels from candidate edge to a strong edge.

Canny Example

$\tau_{low} = 0.08$, $\tau_{high} = 0.2$

source image

$\sigma = 1.4$

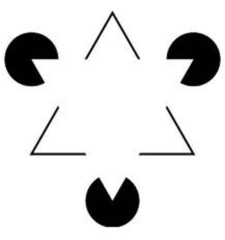
$\sigma = 3.0$

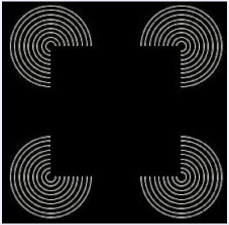
- Edges are not closed: good but might be insufficient for segmentation.

Outline

Contours and Perception: Amodal Completion

A Classical example from Italian psychologist G. Kanizsa[1]



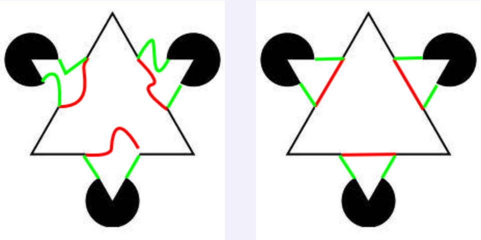


The brain is “closing the gaps” – Gestalt’s closure principle.

[1] Grammatica del vedere / Organization in Vision

But how...

What is the best continuation?



- We reconstruct the gap with an implicit assumption of simplicity / regularity for the contour.
- Variational segmentation algorithms operationalize it.

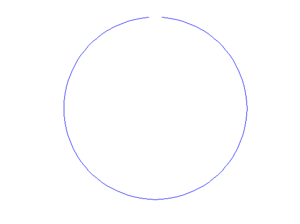
Contour representation

- The simplest way to represent a 2D contour is to use a curve.
- Mathematically, it is a function

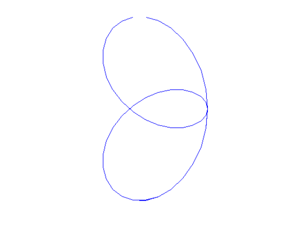
$$C : p \in [a, b] \mapsto C(p) = (x(p), y(p)) \in \mathbb{R}^2$$

- If $C(a) = C(b)$ the curve is closed.

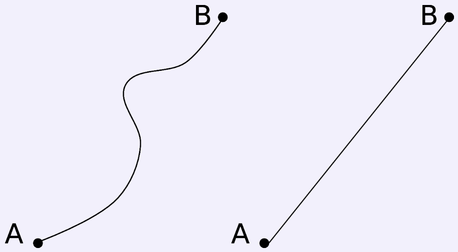
$t \mapsto (\cos(p), \sin(p))$



$p \mapsto (\cos(2p)^2, \sin(p)\cos(3p))$



Simplicity / Regularity



- The first curve is longer and winds more than the second one.
- The second one is simpler.

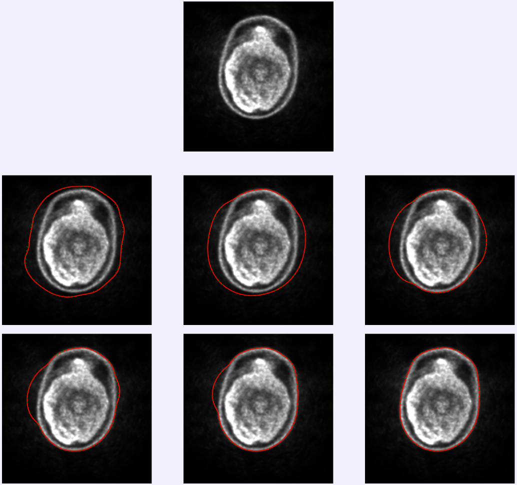
Snakes, Kass, Witkin, Terzopoulos 1988

- Curves evolving (as snakes) to catch an object. Also called Active Contours.
- How to define the"Cost of a Segmentation":
- Cost of a curve lying on an image such that
  - The more a curve follows edges of the image, the "cheaper" it is.
  - The more a curve is bending / winding, the "more expensive" it is.
  - Design an algorithm to compute the "minimal cost", or a cost low enough.
  - Kass, Witkin. Terzopoulos: minimize

$$\mathcal{E}(C) = \int \alpha|\dot{C}|^2 + \beta|\ddot{C}|^2\, dt + \int F_{\text{edge}}(C(t))\, dt$$

  - Solution via Partial Differential Equation (PDE).
  - Start with initial contour. Evolve it so as to decrease cost as fast as possible (steepest descent).
  - Thousands of derived methods!

Active Contour Example

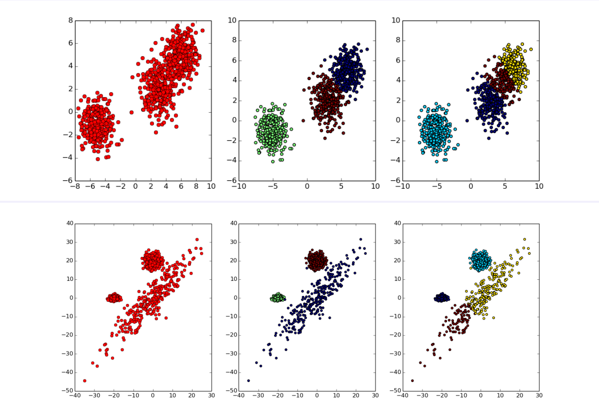


Outline

Clusters of Features

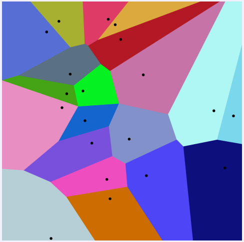
- Idea: extract features for each pixel of the image.
- Group features by similarity.
- Regions with similar features = segments.
- Need to define what features are.
- Need to define what feature similarity is.
- Automatic determination of clusters? Choice of the number of clusters?

Clustering In General



Up: k-means clustering, with $k = 3, 4$. (3 = optimal) Down: Gaussian Mixture models with 3 and 4 (3 optimal too).

The k-Means Algorithm



- Plan divided into cells.
- Proximity to centroid.
- k-means: compute centroids, implicit Voronoï tessellation, from data.
- Remember Søren's lecture from Monday!

- $X = \{x_1, \dots, x_n\}$ set of data points. Find a partition $S_1, \dots, S_k$ of X and points $\mu_1, \dots, \mu_k$ such that

$$\mathcal{D}(S_1 \dots, S_k) = \sum_{j=1}^{k} \sum_{x \in S_j} \|x - \mu_j\|^2 = \text{ minimum.}$$

- The μ_js are the centroids of the S_js.

Lloyd's Algorithm

- Start with k candidate centroids. How to choose them?
- Repeat:
  1. Label pixels / assign them to clusters from their feature distance to centroids.
  2. Recompute centroids of features as average of clusters
- until regions or centroids don't change.

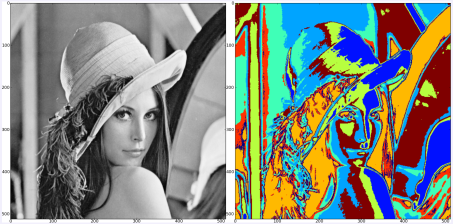
- What are good features for segmentation?
- Depends on image content.

Gaussian Scale Space Features

- 4 scales, $\sigma = 0.5, 1.0, 2.0, 4.0$
- Gaussian derivatives from order 0 (no derivative) to order 3

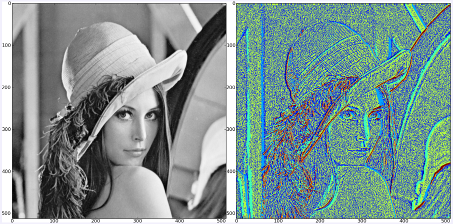
$$\frac{\partial^0}{\partial x^0 \partial y^0}, \frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial^2}{\partial x^2}, \frac{\partial^2}{\partial y^2}, \frac{\partial^2}{\partial x \partial y}, \frac{\partial^3}{\partial x^3}, \frac{\partial^3}{\partial y^3}, \frac{\partial^3}{\partial x^2 \partial y}, \frac{\partial^3}{\partial x \partial y^2}$$

- In all, 40 features per pixel!



Variation

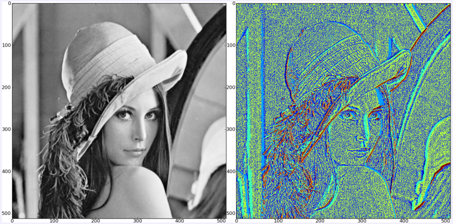
- Same features, but without order 0 (intensity)



- Conclusion: a lot of information in intensity alone!

Variation

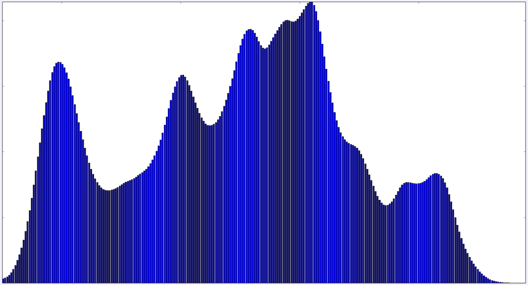
- Same features, but without order 0 (intensity)



- Conclusion: a lot of information in intensity alone!

Choice of *k*

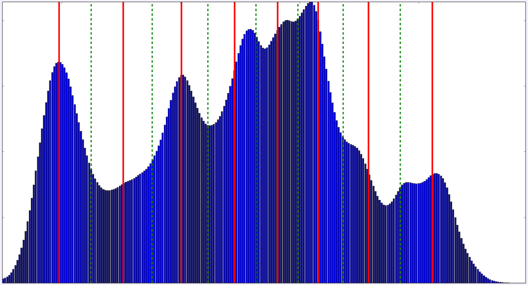
- Previous image clearly over segmented.
- (smoothed) histogram of Lena Image $\pm$ 8 “bumps”:



- For 1D features: *k*-means tries to separate the bumps in the histogram!
- What if *k* is far from the number of bumps,
- if bumps are unclear?
- if bumps are very close?
- What makes a bump? ML course?

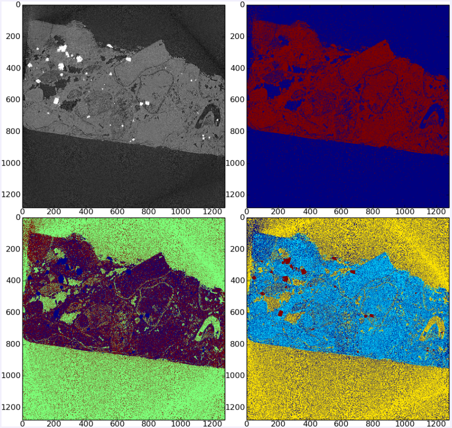
Choice of k

- Previous image clearly over segmented.
- (smoothed) histogram of Lena Image $\pm$ 8 “bumps”:



- For 1D features: k-means tries to separate the bumps in the histogram!
- What if k is far from the number of bumps,
- if bumps are unclear?
- if bumps are very close?
- What makes a bump? ML course?

Example

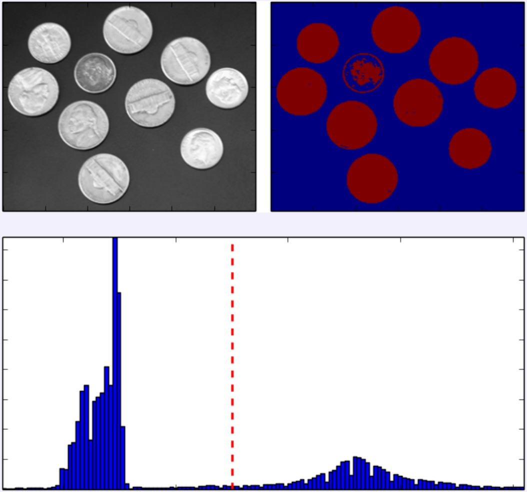


Original image and k-means clustering for k=2,3,4.

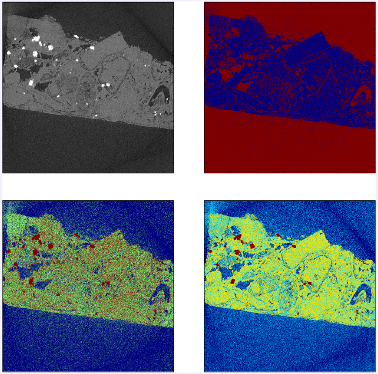
Otsu Clustering Method – 1979

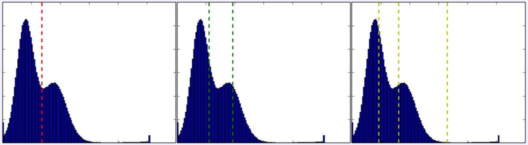
- For 2 regions:
  - Find threshold such that intraclass variance is minimized
  $$\sigma_W(t)^2 := \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t)$$
  - $\omega_1(t) = \frac{\#\text{pixels} < t}{\#\text{pixels in image}}$ = probability that pixel value is $< t$,
    $\omega_2(t) = \frac{\#\text{pixels} \geq t}{\#\text{pixels in image}}$ = probability that pixel value is $\geq t$.
  - $\sigma_1^2(t) =$ variance of the class of pixels $< t$,
    $\sigma_2^2(t) =$ variance of the class of pixels $\geq t$,
  - Search for t in range $[\min(\text{image value}), \max(\text{image value})]$.
- Performed on image histogram.
- Easily generalized to more classes/regions.
- Can search for local thresholds (on subimages).

Example



Multiple Classes Otsu

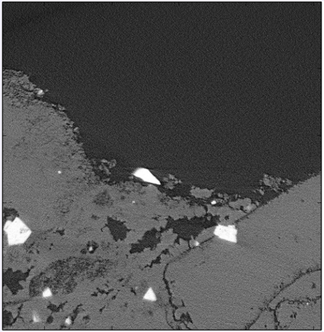


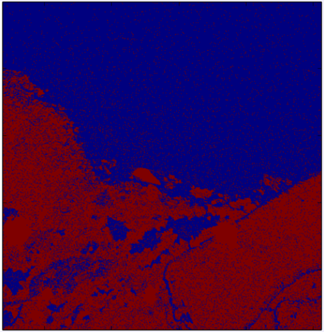


Outline

Clustering and Noise

- Intensity clustering discard does not use spatial proximity grouping
- Noise disturbs clustering.



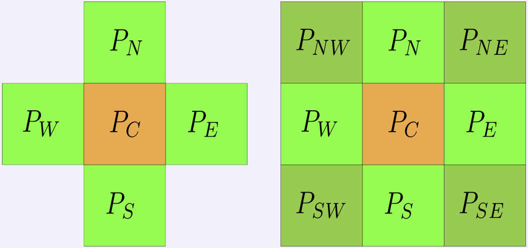


Computed X-Ray tomography from a sandstone sample. Relatively high noise level.

Local Spatial Coherence

- Prevent small holes: pixel of class A surrounded by pixels of class B: remember proximity grouping.

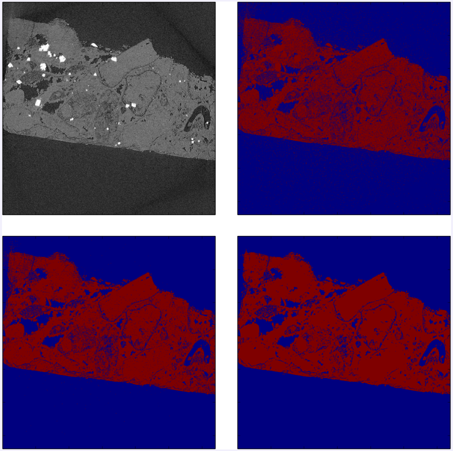
- Local proximity defined in term of pixel neighborhoods



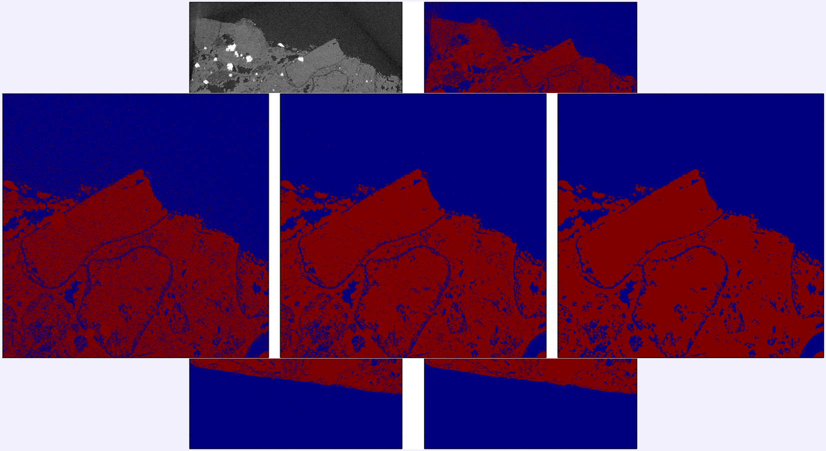
4 and 8 pixels neighborhood systems.

- If neighbor pixels have same (or almost same) labels, replace center pixel by dominant one.

Example



Example



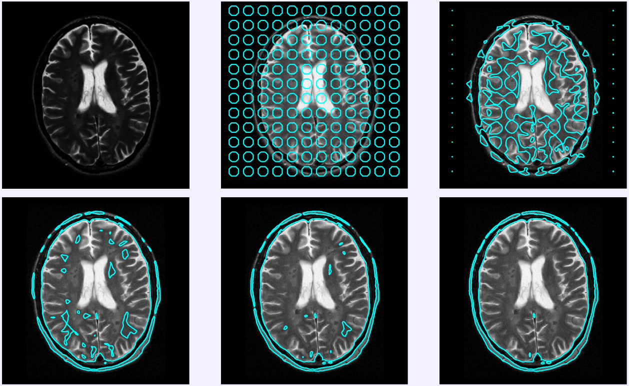
Clustering Snakes: Chan-Vese Algorithm

- Mixes ideas: Clustering (k-means), Snakes a.k.a Active contours, Spatial Regularization.
- Performs clustering while enforcing region coherence – remove small holes.
- Continuous formulation: find a curve C, numbers u_0, u_1 (class centroids) minimizing

$$\mathcal{E}(C, u_0, u_1) = \int_{\mathrm{int}(C)} (u - u_0)^2 \, dx + \int_{\mathrm{ext}(C)} (u - u_1)^2 \, dx + \lambda \mathrm{length}(C)$$

- Minimization by solving a Partial Differential equation for C.
- Complex, non linear.
- Very effective implementations – level sets, relaxations.
- Hundreds of derived methods.

Chan Vese Example



Outline

Summary

- Ideas from Gestalt theory: Figure-Ground, grouping: proximity, connectedness, similarity, continuity, closure, common regions.
- Segment as “small” semantic unit.
- Edges and segmentation: Marr-Hildreth edge detector, Canny Edge detector.
- Closing the gap. Perceptual aspects (amodal completion), operationalization: Snakes - a.k.a. Active contours.
- Feature proximity, clustering, K-Means, Otsu.
- Noise and Gap. Spatial regularization.
- Chan-Vese: Clustering Snakes.

Summary

- Ideas from Gestalt theory: Figure-Ground, grouping: proximity, connectedness, similarity, continuity, closure, common regions.
- Segment as “small” semantic unit.
- Edges and segmentation: Marr-Hildreth edge detector, Canny Edge detector.
- Closing the gap. Perceptual aspects (amodal completion), operationalization: Snakes - a.k.a. Active contours.
- Feature proximity, clustering, K-Means, Otsu.
- Noise and Gap. Spatial regularization.
- Chan-Vese: Clustering Snakes.