Image Segmentation

Image Processing with Biomedical Applications
ELEG-475/675
Prof. Barner

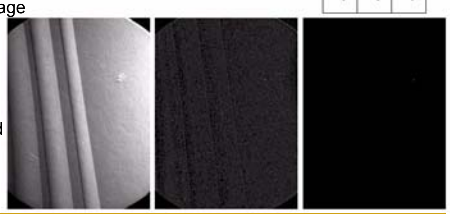
Image Segmentation

- Objective: extract attributes (objects) of interest from an image
  - Points, lines, regions, etc.
- Common properties considered in segmentation:
  - Discontinuities and similarities
- Approaches considered:
  - Point and line detection
    - Edge linking
  - Thresholding methods
    - Histogram, adaptive, etc.
  - Region growing and splitting

Detection of Discontinuities

- Mask filtering approach:
  $$R = \omega_1 z_1 + \omega_2 z_2 + \cdots + \omega_9 z_9 = \sum_{i=1}^{9} \omega_i z_i$$
- Isolated point detection: $|R| \geqslant T$
- Example:
  - X-ray image
  - T=90% of max value
  - Input, gradient, threshold output

–1	–1	–1
–1	8	–1
–1	–1	–1



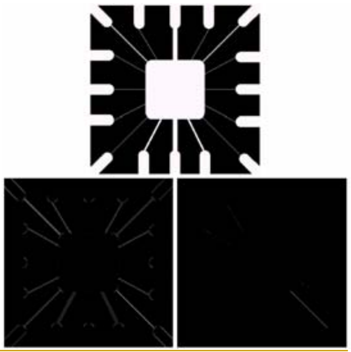
Line Detection

–1	–1	–1	–1	–1	2	–1	2	–1	2	–1	–1
2	2	2	–1	2	–1	–1	2	–1	–1	2	–1
–1	–1	–1	2	–1	–1	–1	2	–1	–1	–1	2
Horizontal			+45°			Vertical			–45°		

- Line detection masks
  - Detects lines one pixel wide
  - Line orientation specific
    - Set orientations specific thresholds
  - Second derivative based

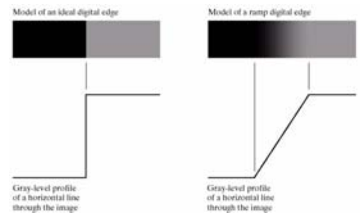
Line Detection Example

- Wire-bond mask for electronic circuit
- Application of -45° edge mask
- Result of thresholding

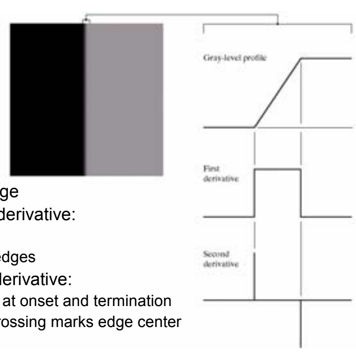


Edge Detection

- Concepts:
  - Edge – local
  - Boundary – global
- Ideal edge:
  - Step
- Practical edge:
  - Ramp
  - Ideal edges are smoothed by optics, sampling, illumination conditions
  - Inch thickness determined by transition region



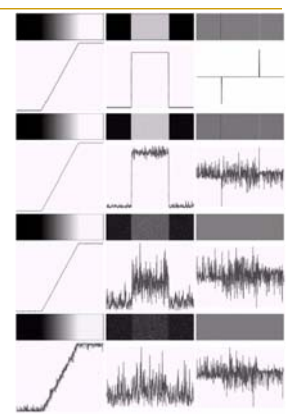
Edge Example – Noiseless Case



- Ramp edge
- The first derivative:
  - Pulse
  - Thick edges
- Second derivative:
  - Spikes at onset and termination
  - Zero crossing marks edge center

Edge Example – Noisy Case

- Gaussian noise corrupted edge
- Derivatives amplify noise
  - Even modest levels of noise severely degraded gradient-based edge detection
  - Possible solution: noise smoothing prior to edge detection



Gradient Operators

- Two-dimensional gradient:
$$\nabla \mathbf{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$
  - Magnitude:
$$\nabla f = mag(\nabla \mathbf{f}) = \left[G_x^2 + G_y^2 \right]^{1/2}$$
  - Direction (angle)
$$\alpha(x, y) = \tan^{-1}\left(\frac{G_y}{G_x} \right)$$
    - Perpendicular to edge
  - Approximation:
$$\nabla f \approx |G_x| + |G_y|$$
- Shown: mask realizations

−1	0		0	−1
0	1		1	0

Roberts

−1	−1	−1		−1	0	1
0	0	0		−1	0	1
1	1	1		−1	0	1

Prewitt

−1	−2	−1		−1	0	1
0	0	0		−2	0	2
1	2	1		−1	0	1

Sobel

Gradient Operators and Example

- Application: Horizontal, vertical and (additive) gradient

Gradient Example (I)

- Preprocess image
  - Smooth detail textures
  - Thicken edges
  - Filter: 5x5 averaging filter

Gradient Example (II)

- Extension to 45° gradients and their application

0	1	1		−1	−1	0
−1	0	1		−1	0	1
−1	−1	0		0	1	1

Prewitt

0	1	2		−2	−1	0
−1	0	1		−1	0	1
−2	−1	0		0	1	2

Laplacian of a Gaussian (LoG)

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

- Recall Laplacian:

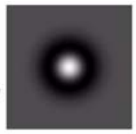
$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

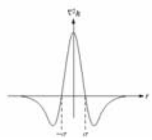
- Edge detection limitations:
  - Produces double edges, insensitive to edge direction, sensitive to noise
- Pre-smooth with Gaussian filter

$$h(r) = -e^{-\frac{r^2}{2\sigma^2}}$$

  - Combined (linear) smoothing and derivative operations

$$\nabla^2 h(r) = -\left[\frac{r^2 - \sigma^2}{\sigma^4}\right] e^{-\frac{r^2}{2\sigma^2}}$$

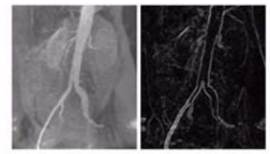


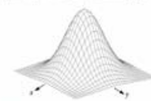


0	0	−1	0	0
0	−1	−2	−1	0
−1	−2	16	−2	−1
0	−1	−2	−1	0
0	0	−1	0	0

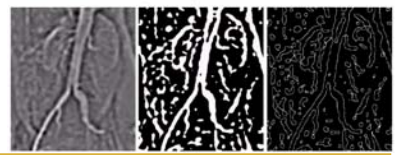
LoG Example

- Angiogram example
  - Sobel output shown for reference
- To obtain edges:
  - Threshold LoG
  - Mark zero crossings
    - Numerous false (spaghetti) edges
    - First derivative more widely used
- LoG models certain aspects of the human visual system





−1	−1	−1
−1	8	−1
−1	−1	−1



Edge Linking

- Procedures often yield broken edges
  - Noise, illumination irregularities, etc.
- Link neighboring segments based on predefined criteria
  - Example criteria:
    - Strength of gradients

$$|\nabla f(x,y) - \nabla f(x_0,y_0)| \le E$$

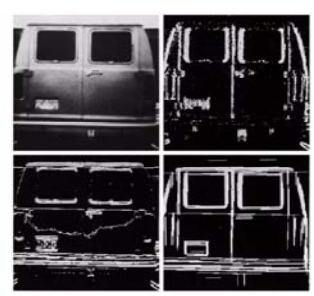
    - Direction of gradients

$$|\alpha(x,y) - \alpha(x_0,y_0)| < A$$

    - Applied over predefined search neighborhood

Edge Linking Example

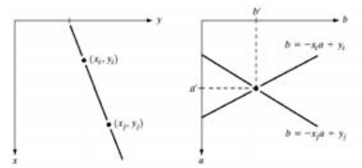
- Goal: license plate localization
- Shown: horizontal and vertical gradient images
- Linking criteria:
  - Gradient $\geqslant$ 25
  - Angle differences $\leqslant$ 15°
- Final result:
  - Linked edges
  - Search for license plate based on rectangle side ratios



Hough Transform (I)

- General approach:
  - Project feature into a parameter space
  - Examples: lines, circles, etc.
- Line case:
  - Defining parameters: slope and intercept
  - Map lines into the single (slope, intercept) 2-tuple
    - Advantage: an infinite number of points get mapped to a single 2-tuple
- Reverse operation for isolated (binary) points
  - Line case: a point is located on an infinite number of lines
    - Map to all (slope, intercept) 2-tuples corresponding to the infinite number of lines passing through the point
    - Result: a curve in the (slope, intercept)

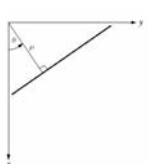
Hough Transform (II)

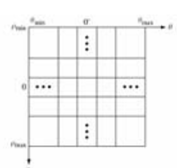


- Line equation: $y_i = ax_i + b$
- Parameter space:
  - Fix x_i and y_i
  - Line in parameter space:

    $b = -x_i a + y_i$
  - All lines (in parameter space) for points on a line in image space cross at a single point
    - Crossing point: common (slope, intercept)

Hough Transform (II)

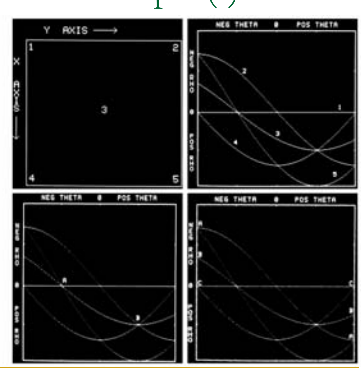




- Problem: infinite slopes
- Solution: normal representation of a line

  $x\cos\theta + y\sin\theta = \rho$
- Procedure for binary images:
  - Subdivide ρ,θ plane
  - Map each point in the image plane to a curve in the ρ,θ plane
    - Increment cells in subdivided ρ,θ plane crossed by curve
  - Peak values in ρ,θ plane correspond to lines in the image planes

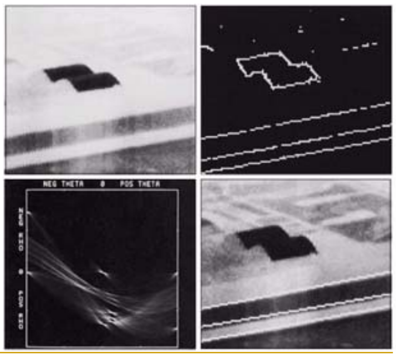
Hough Transform Example (I)



- Image space:
  - Five points
- Parameter space
  - Five curves
- Parameters space curve intersections:
  - Lines connecting points in image space

Hough Transform Example (II)

- Edge localization example
- Input: infrared image
- Process:
  - Edge detection
  - Hough transform
  - Peak detection
  - Map (lines) back to image space
- Generalizations to other shapes



Collimation of X-Ray Images (I)

- Problem: identify region of exposure
- Problem: x-ray scattering smoothes edges
- Solution:
  - Enhance edges
  - Detect edges
  - Radon transform detected edges
    - Hough transform generalization
  - Identify lines
  - Map border (lines) back to image space

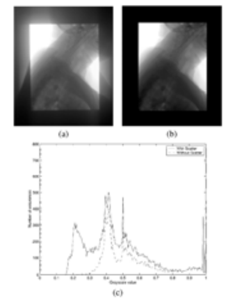


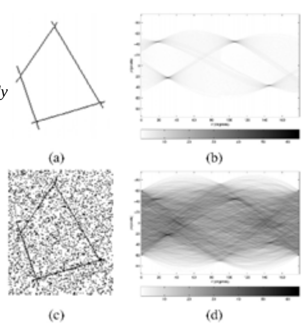
Figure 1: Illustration showing effects of x-ray scattering: (a) Typical scattering pattern in collimated x-rays. (b) The ROI in the ideal case of zero exterior scattering. (c) Corruption of greyscale histogram in the presence of scattering.

Collimation of X-Ray Images (II)

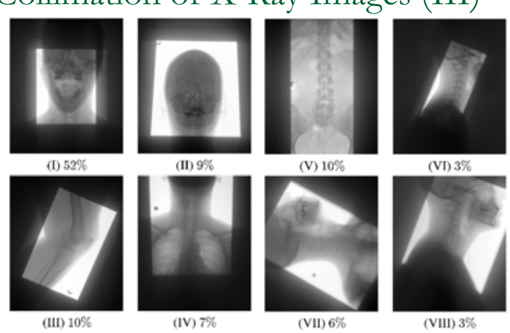
- Radon transform:

$$R(\rho,\theta)=\iint_{x\,y} f(x,y)\delta(x\cos\theta+y\sin\theta-\rho)dxdy$$

- Examples:
  - Sample border
    - Noise free and noisy cases
  - Image and transform domain representations
    - Peaks in transform domain correspond to lines in image domain

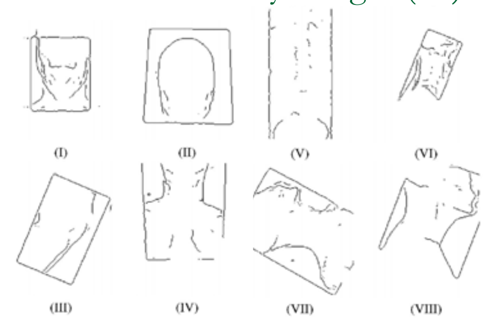


Collimation of X-Ray Images (III)



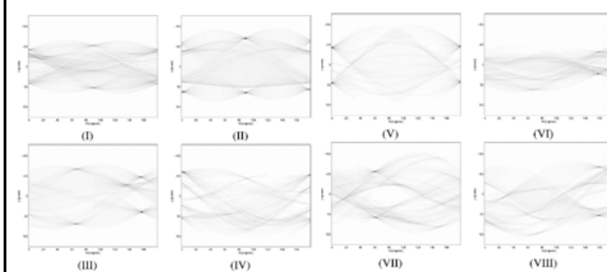
Test case categories and the percentage of images in each category

Collimation of X-Ray Images (IV)



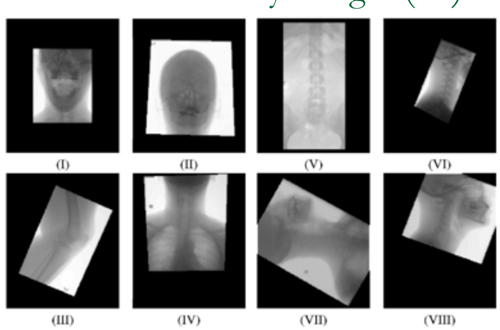
Sobel edge detection results

Collimation of X-Ray Images (V)



Radon transform of edge detection results

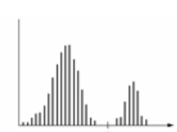
Collimation of X-Ray Images (VI)

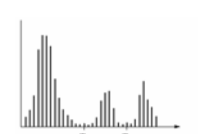


Collimation results based on lines detected in the radon transform domain

Thresholding Approaches

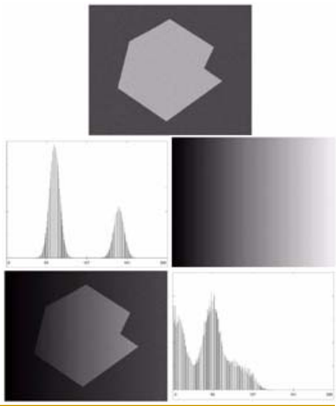
- Thresholding is appropriate when:
  - Objects and background have different intensities
  - Multimodal distribution
- Threshold can be set:
  - Globally
  - Locally
  - Adaptively
- Utilized multiple thresholds





Illumination Effects

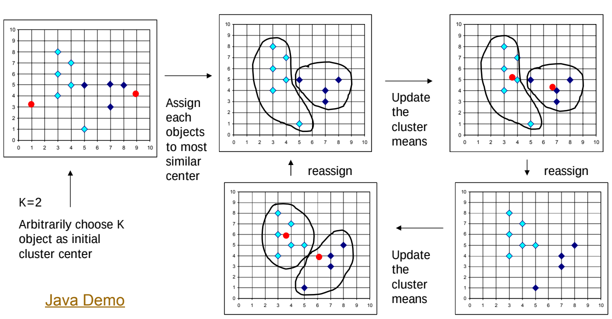
- Recall image model:
  $f(x,y)-i(x,y)r(x,y)$
- Utilizing the log:
  $$z(x,y)=\ln f(x,y)$$
  $$=\ln i(x,y)+\ln r(x,y)$$
  $$=i'(x,y)+r'(x,y)$$
- If components independent:
  - Convolve distributions
  - If $i(x,y)$ is constant
    - Density is a delta
  - Uneven illumination yields convolved, distorted distributions
    - No longer separable



K-Means Algorithm

- Clustering algorithm
  - Apply to spatial or multidimensional samples
  - Apply to intensities (one-dimensional) to cluster histogram
- Procedure:
  1. Place *K* points in the space
     - Initial cluster centroids
     - Set randomly or with *a priori* knowledge
  2. Assign all points (pixel values) to the cluster defined by the closest centroid
     - Utilized appropriate distance metric (Euclidian, city block, etc.)
  3. Recalculate the positions (values) of the *K* centroids
     - For Euclidian distance, calculate the mean of all points in a specific cluster
  4. Repeat Steps 2 and 3 until centroid movements areto below a fixed threshold

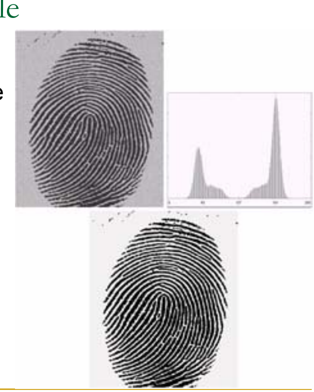
K-Means Clustering Example



Java Demo

Fingerprint Example

- Input image: grayscale
- Histogram shows two modes
- Set threshold with K-means algorithm
  - K=2



Mixture Model and the EM Algorithm (I)

- Statistical modeling
  - Relaxation of K-means
- Assume samples are from a mixture of two Gaussians:

$$Y_1 \sim N(\mu_1,\sigma^2_1)$$
$$Y_2 \sim N(\mu_2,\sigma^2_2)$$
$$Y = (1-\Delta)Y_1 + \Delta Y_2$$

where $\Delta\in\{0,1\}$ with $\Pr(\Delta=1)=\pi$

- The PDF of the samples is thus

$$p_Y(y) = (1-\pi)p_{\theta_1}(y) + \pi p_{\theta_2}(y)$$

where $p_{\theta_1}(y)$ is a Gaussian PDF with parameters $\theta=(\mu,\sigma^2)$

Mixture Model and the EM Algorithm (II)

- Objective: given N observed samples, determined all parameters

$$\theta = (\pi,\theta_1,\theta_2) = (\pi,\mu_1,\sigma^2_1,\mu_2,\sigma^2_2)$$

- Estimation technique: Maximum Likelihood
  - Log-likelihood function:

$$\mathcal{L}(\theta;\mathbf{Z}) = \sum_{i=1}^{N}\log[(1-\pi_i)p_{\theta_1}(y_i)+\pi_i p_{\theta 2}(y_i)]$$

    - Direct maximization is difficult
    - Solution: suppose we know the values of the Δ_i's
  - Log-likelihood function reduces to:

$$\mathcal{L}(\theta;\mathbf{Z},\mathbf{\Delta}) = \sum_{i=1}^{N}\log[(1-\Delta_i)p_{\theta_1}(y_i)+\Delta_i p_{\theta 2}(y_i)]$$

Mixture Model and the EM Algorithm (III)

- Given $\mathcal{L}(\theta;\mathbf{Z},\mathbf{\Delta}) = \sum_{i=1}^{N}\log[(1-\Delta_i)p_{\theta_1}(y_i)+\Delta_i p_{\theta 2}(y_i)]$
  - Case 1:
    - ML estimates of μ_1 and σ^2_1 are the sample mean and variance for all samples with $\Delta_i = 0$
  - Case 2:
    - ML estimates of μ_2 and σ^2_2 are the sample mean and variance for all samples with $\Delta_i = 1$
- But Δ_i's are unknown
  - Solution: proceed in an iterative fashion utilizing a the expected values of Δ_i's

$$\lambda(\theta) = E(\Delta_i\,|\,\theta,\mathbf{Z}) = \Pr(\Delta_i = 1\,|\,\theta,\mathbf{Z})$$

  - These are hidden terms, referred to as responsibilities
    - Determined using soft assignments of all samples and the probability of a sample being from distribution #2

EM Algorithm for Two-Component Gaussian Mixture

1. Make an initial guesses for the parameters $\hat{\mu}_1,\hat{\sigma}_1^2,\hat{\mu}_2,\hat{\sigma}_2^2,\hat{\pi}$
2. *Expectation Step*: compute the responsibilities

$$\hat{\gamma}_i = \frac{\hat{\pi}p_{\hat{\theta}_2}(y_i)}{(1-\hat{\pi})p_{\hat{\theta}_1}(y_i)+\hat{\pi}p_{\hat{\theta}_2}(y_i)},\quad i=1,2,\ldots,N$$

3. *Maximization Step*: compute the weighted means and variances

$$\hat{\mu}_1 = \frac{\sum_{i=1}^{N}(1-\hat{\gamma}_i)y_i}{\sum_{i=1}^{N}(1-\hat{\gamma}_i)} \qquad \hat{\sigma}_1^2 = \frac{\sum_{i=1}^{N}(1-\hat{\gamma}_i)(y_i-\hat{\mu}_1)^2}{\sum_{i=1}^{N}(1-\hat{\gamma}_i)}$$

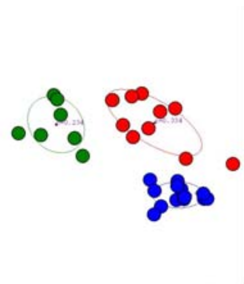
$$\hat{\mu}_2 = \frac{\sum_{i=1}^{N}\hat{\gamma}_i y_i}{\sum_{i=1}^{N}\hat{\gamma}_i} \qquad \hat{\sigma}_2^2 = \frac{\sum_{i=1}^{N}\hat{\gamma}_i(y_i-\hat{\mu}_2)^2}{\sum_{i=1}^{N}\hat{\gamma}_i}$$

and the mixing probability $\hat{\pi} = {}^{1}\!/\!_{N}\sum_{i=1}^{N}\hat{\gamma}_i$

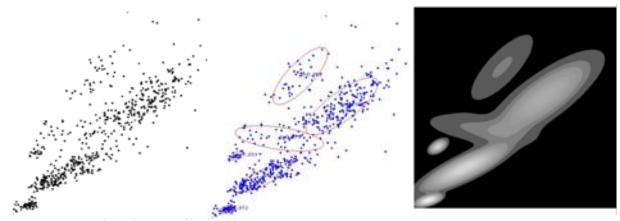
4. Integrate steps 2 and 3 until convergence

Three Mixture Example

- Model samples as three Gaussian mixtures
- Initialize with guess
- Colors indicate probability of belonging to each parent distribution
- Shown:
  - Samples with probabilities
  - Initial guess distributions
  - Distribution variance contour
  - Iterations 1-6 and final result (iteration 20)



Biomedical Example – Clustering of Assay Data



- Shown:
  - Observed data
  - Sixth mixture model with contours
  - Mixture distribution

Java Demo

Relation Between EM and K-Means

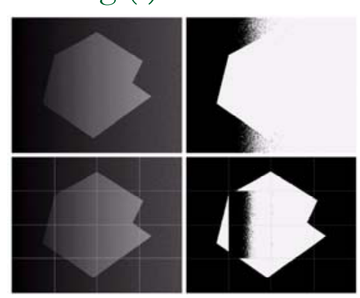
- In the EM (Baum-Welch) algorithm
  - Utilized binary decisions: $\hat{\gamma}_i = 1 \text{ if } |y_i - \hat{\mu}_2| < |y_i - \hat{\mu}_1|$
  - Then $\hat{\mu}_1$ and $\hat{\mu}_2$ are unweighted means

$$\hat{\mu}_1 = \frac{\sum_{i=1}^{N}(1-\hat{\gamma}_i)y_i}{\sum_{i=1}^{N}(1-\hat{\gamma}_i)} \qquad \hat{\mu}_2 = \frac{\sum_{i=1}^{N}\hat{\gamma}_i y_i}{\sum_{i=1}^{N}\hat{\gamma}_i}$$

    - Equivalent to K-means (K=2)
      - Trivially generalized to a larger number of partitions/mixtures
    - Mixture model gives soft (probability) cluster assignments
- Generalizations
  - Fuzzy C-Means
    - Samples assigned to more than one cluster – membership function
    - Assignments are functions of distance
  - K-medoids – use cluster median as central representative point
    - More robust

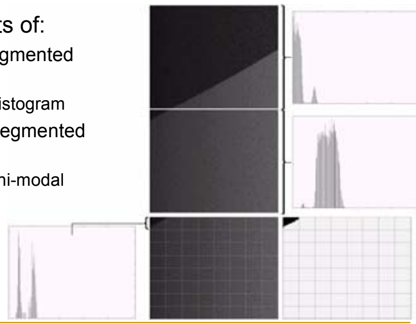
Adapted Thresholding (I)

- Simple approach:
  - Partition image
  - Check homogeneity in each partition
    - Example: test variance
  - Segment nonhomogeneous partitions
    - Example: K-means
- Shown:
  - Global thresholding
  - Adaptive thresholding



Adapted Thresholding (II)

- Enlargements of:
  - Correctly segmented partition
    - Bimodal histogram
  - Incorrectly segmented partition
    - (Nearly) uni-modal histogram
- Solution:
  - Finer partitioning



Optimal Thresholding (I)

- Considered two objects
  - Foreground/background
  - Overlapping PDFs
- Overall (mixture) PDF:

$$p(z) = P_1 p_1(z) + P_2 p_2(z)$$

where $P_1 + P_2 = 1$

- Probability of classifying Object 2 as Object 1:

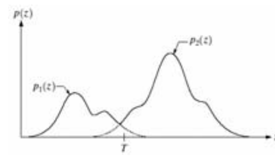
$$E_1(T) = \int_{-\infty}^{T} p_2(z)dz$$

- Probability of classifying Object 1 as Object 2:

$$E_2(T) = \int_{T}^{\infty} p_1(z)dz$$

- Overall probability of error:

$$E(T) = P_2 E_1(T) + P_1 E_2(T)$$



Optimal Thresholding (II)

- Solve for optimal threshold T:

$$\frac{dE(T)}{dT} = \frac{d}{dT}\left(P_2 \int_{-\infty}^{T} p_2(z)dz + P_1 \int_{-T}^{\infty} p_1(z)dz\right)$$

$$= P_2 p_2(T) - P_1 p_1(T) \quad \Rightarrow \quad P_2 p_2(T) = P_1 p_1(T)$$

- In the Gaussian case:

$$p(z) = \frac{P_1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(z-\mu_1)^2}{2\sigma_1^2}} + \frac{P_2}{\sqrt{2\pi}\sigma_2} e^{-\frac{(z-\mu_2)^2}{2\sigma_2^2}}$$

  - Solution is in the form of the quadratic: $AT^2 + BT + C = 0$

$$A = \sigma_1^2 - \sigma_2^2$$

$$B = 2\left(\mu_1\sigma_2^2 - \mu_2\sigma_1^2\right)$$

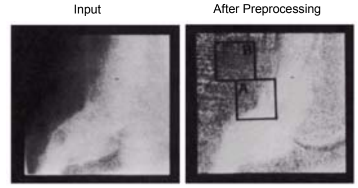
$$C = \mu_1\sigma_2^2 - \mu_2\sigma_1^2 + 2\sigma_1^2 2\sigma_2^2 \ln\left(\sigma_2 P_1 / \sigma_1 P_2\right)$$

    - Results may yield two thresholds
    - If both distributions have a common variance, σ^2:

$$T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^1}{\mu_1 - \mu_2} \ln\left(\frac{P_2}{P_1}\right)$$

Cardiogram Example (I)

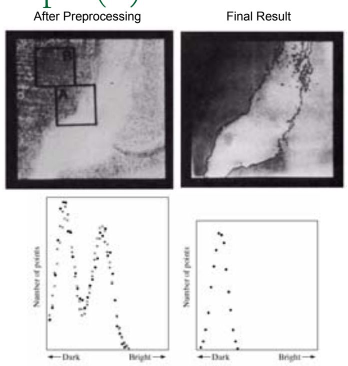
Input　　After Preprocessing



- Objective:
  - Automatically outline heart ventricle boundaries
  - Utilizes contrast medium
- Preprocessing:
  - Intensity log mapping to counter exponential radioactive absorption effects
  - Subtraction of base (noncontrast) image
  - Image (frame) averaging to reduce noise
- Procedure:
  - Subdivide image
  - Generate local histograms

Cardiogram Example (II)

- Procedure (cont'd):
  - Fit (uni/bi-modal) Gaussian distributions to histograms
  - For blocks with bimodal distributions:
    - Set adaptively determined threshold
  - Set boundaries by taking derivative of thresholded image



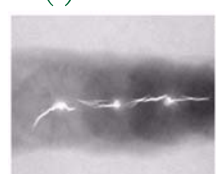
Region-Based Segmentation

- Previous approaches utilized continuities and/or pixel value attributes (gray value)
  - They do not operate on or directly consider regions
- Region-based formulation:
  - Let R be the entire image region
  - Segment R into n subregions, R_1, R_2,…, R_n, such that:
    - (a) $\bigcup_{i=1}^{n} R_i = R$.
    - (b) R_i is a connected region, $i = 1, 2, \ldots, n$.
    - (c) $R_i \bigcap R_j = \varnothing$ for all i and $j, i \neq j$.
    - (d) $P(R_i)$=TRUE for i=1,2,...,n.
    - (e) $P(R_i \bigcup R_j)$=FALSE for $i \neq j$.
  - $P(R_i)$ is a logical operator that defines the properties of the region
    - Example: $P(R_i)$=TRUE if the pixel values in R_i are from a predefined set

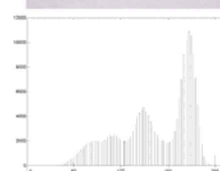
Region Growing

- Approach:
  - Group pixels/subregions into larger subregions based on a set criteria
    - Criteria examples: gray level, texture, color, size, shape
    - Multiple criteria: gray value and size, etc.
  - Iterative procedure
    - How to set the seed regions, number of regions?
    - How to set criteria?
    - When to stop?

Region Growing Example (I)

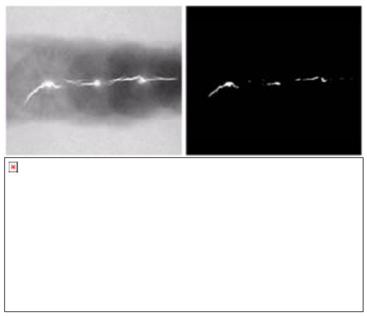


- Objective:
  - Segment x-ray image to identify weld failures
- Set seed points:
  - All pixels having maximum (255) value
- Region growing criteria:
  - Absolute gray value difference $\leqslant 65$
    - Set as difference between maximum value and first mode in histogram
  - Pixel is 8-connected to at least one pixel in the region



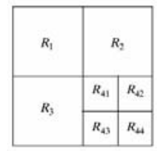
Region Growing Example (II)

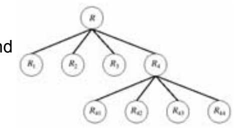
- Shown results:
  - Input
  - Seed regions
  - Results of region growing
  - Region boundaries
- Observations:
  - Histogram is not suited to strict thresholding
  - Connectivity criteria critical to satisfactory result



Region Splitting and Merging

- Alternative approach to seed regions:
  - Subdivide image into a set of arbitrary, disjoint regions
    - Merge and/or split the set of regions to satisfy region segmentation conditions
      - Value similarity, connectivity, etc.
- Quadtree method:
  - Split into four disjoint quadrants any region R_i for which $P(R_i)$=FALSE
  - Merged in the adjacent regions R_j and R_k for which $P(R_j \cup R_k)$=TRUE
  - Stop when no further merging or splitting is possible





Quadtree Split-Merge Example

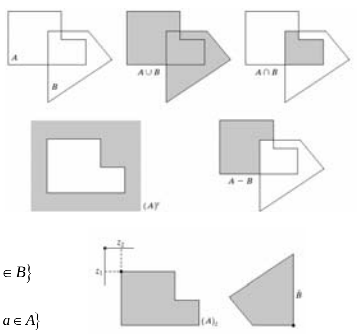
- Homogeneity criteria:
  - $P(R_i)$=TRUE if $|z_j - m_i| \leq 2\sigma_i$ for at least 80% of the pixels in R_i
    - Region mean: m_i
    - Region standard deviation: σ_i
- Shown: input, quadtree and threshold segmentations
  - Threshold set as midpoint between main histogram modes
    - Thresholding loses details

Morphology

- *Mathematical Morphology* focuses on extracting image components
  - Useful in the representation and description of region shapes
  - Examples: boundaries, skeletons and convex hulls
- Set operations
  - Typically applied to binary images
  - Multilevel extensions exist

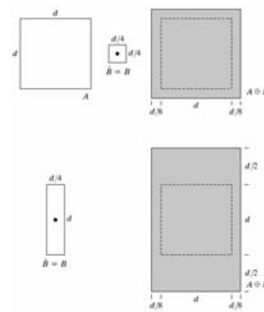
Basic Set Operations

- Let A and B be sets in Z^2
- Standard operations:
  - Union
  - Intersection
  - Complement
  - Difference
  - Reflection
    $\hat{B} = \{\omega \mid \omega = -b, \text{ for } b \in B\}$
  - Translation
    $(A)_z = \{c \mid c = a + z, \text{ for } a \in A\}$



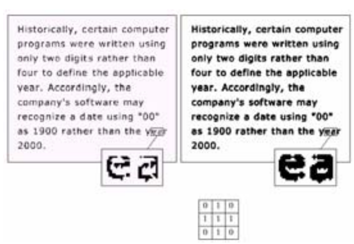
Dilation

- The dilation of *A* by *B* :
  $A \oplus B = \{z \mid (\hat{B})_z \bigcap A \neq \varnothing\}$
- Result:
  - The set of all displacements, *z*, such that the reflection of *B* and *A* overlap by at least one element
- Rewrite dilation:
  $A \oplus B = \{z \mid [(\hat{B})_z \bigcap A] \subseteq A\}$
  - Dilation *structuring element*: B



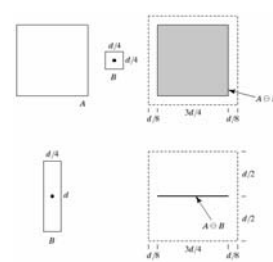
Dilation Example

- Dilation application:
  - Filling in gaps
- Example:
  - Scanned text



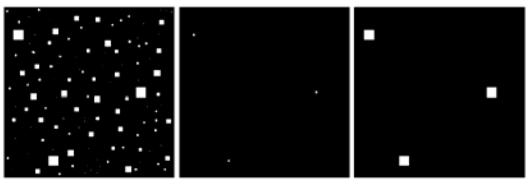
Erosion

- The erosion of *A* by *B* :
  $A \ominus B = \{z \mid (B)_z \subseteq A\}$
- Result:
  - The set of all displacements, *z*, such that *B*, translated by *z*, is contained in *A*
- Note dilation and erosion are duels of each other with the respective complementation and reflection:
  $(A \ominus B)^c = A^c \oplus \hat{B}$



Erosion Example

- Erosion application: Elimination of irrelevant detail
  - A function of detail (structuring element) size
- Example: 13x13 structuring element removes small details
  - Erosion removes (and shrinks) details
  - Postprocess through dilation: expands remaining details

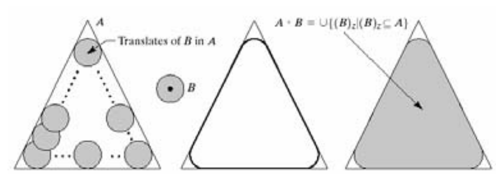


Opening

- *Opening* smooths the contour of an object, breaks narrow isthmuses, and eliminates in protrusions

  $$A \circ B = (A \ominus B) \oplus B$$

  - The erosion of *A* by *B*, followed by dilation of the result by *B*



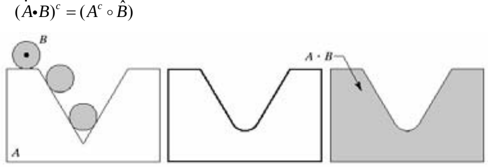
Closing

- *Closing* smooths the contour of an object, fuses narrow breaks and long thin gulfs, eliminate small holes, and fills gaps in the contour

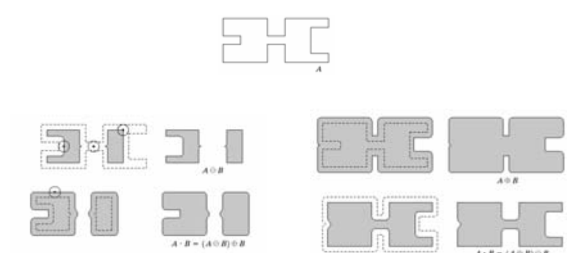
  $$A \bullet B = (A \oplus B) \ominus B$$

  - The dilation of *A* by *B*, followed by erosion of the result by *B*
- Opening and closing our duels of each other with respect to complementation and reflection

  $$(A \bullet B)^c = (A^c \circ \hat{B})$$



Opening and Closing Example (I)



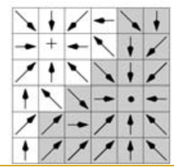
Opening and Closing Example (II)

- Fingerprint example
- Objective:
  - Remove noise and connect broken lines
- Operations:
  - Opening
    - Eliminates noise
  - Closing
    - Connects broken lines

Watershed Segmentation (I)

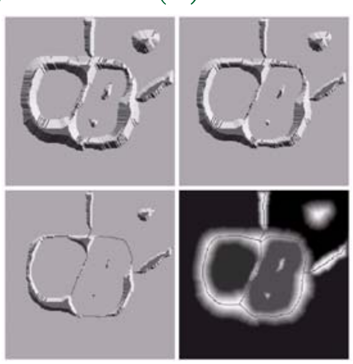
- Methodology: topographical interpretation of image
  - Three types of points:
    - Points belonging to a regional minimum
    - Points that drain to a common minimum point
      - A drop of water released at all such points flows downhill, reaching a common minimum
      - Points referred to as *catchment basin* or *watershed* points
    - Points that can drain to more than one minimum point
      - Points referred to as *divide* or *watershed lines*

16.8	19.2	13.5	20.5	31.2	30.1
18.7	11.9	15.4	18.5	22.1	18.4
20.1	21.9	26.6	20.8	17.3	18.1
25.3	22.8	20.9	19.8	15.1	15.9
30.7	35.5	29.9	18.7	17.6	39.9
34.8	38.6	33.4	32.7	33.5	36.7



Watershed Segmentation (II)

- Interpretation:
  - Punch a hole in each regional minimum
  - Flood entire topography from below with rising water
  - Build dams to prevent catchment basins from merging
  - Once fully flooded, only dams remain
    - Dams define (closed) boundaries

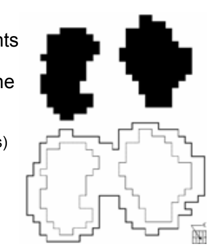


Watershed Segmentation (III)

- Let M_1 and M_2 denote the set of points and two regional minima
- Catchment basins associated with the two regional minima:
  - $C_{n-1}(M_1)$ and $C_{n-1}(M_2)$
    - Defines a set of points (pixel locations) that drain to each of the minima
    - Flooding stage denoted by: n-1
  - Union of points:

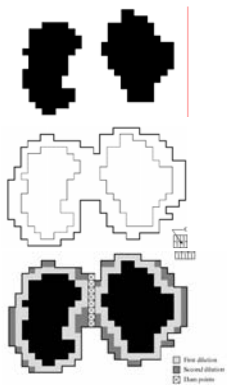
$$C[n-1]=C_{n-1}(M_1) \cup C_{n-1}(M_2)$$

- Catchment basin merging occurs at step n if:
  - $C[n-1]$ has two connected components
  - At step n there is a single, one connected component, q



Watershed Segmentation (IV)

- Dam construction:
  - Dilate $C_{n-1}(M_1)$ and $C_{n-1}(M_2)$ subject to:
    1. The dilation is constrained to q
    2. Dilation is not performed on points that cause the sets to merge
- Example:
  - Top: $C_{n-1}(M_1)$ and $C_{n-1}(M_2)$
  - Middle: q
  - Bottom: dilation
    - First dilation (light gray)
    - Second dilation (medium gray)
      - Condition 1 violations: some dilations outside q (not shown)
      - Point satisfying both conditions constitute dams
        - Marked with x's



Watershed Algorithm (I)

- The watershed algorithm is typically applied to a gradient image
- Regional minima (coordinates) in image $g(x,y)$: $M_1, M_2, \ldots, M_R$
- Set containing the coordinates of the samples in the catchment basin associated with M_i: $C(M_i)$
- Set of image points less than threshold n:

$$T[n] = \{(s,t) \mid g(s,t) < n\}$$

- Flood the image, and mark all pixels < the flood plane $g(x,y)=n$
- Set containing (coordinates) of points in catchment basin $C(M_i)$ that are < n

$$C_n(M_i) = C(M_i) \bigcap T[n]$$

  - Binary image (set) indicating if catchment basin points are $> n$ (1) $< n$ (0)

Watershed Algorithm (II)

- Union of flooded catchment basins:

$$C[n] = \bigcup_{i=1}^{R} C_n(M_i)$$

- Union of all catchment basins:

$$C[\max+1] = \bigcup_{i=1}^{R} C(M_i)$$

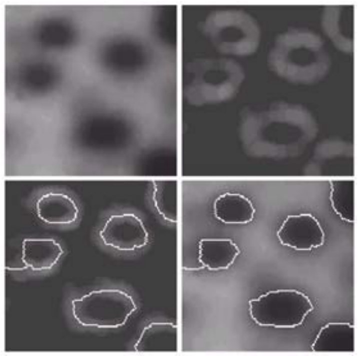
  - Maximum image value: max
  - Minimum image value: min
- Note: $C[n\text{-}1] \subseteq C[n] \subseteq T[n]$
  - Each connected component in $C[n\text{-}1]$ is contained in exactly one connected component of $T[n]$

Watershed Algorithm (III)

- Initialization: $C[\min+1] = T[\min+1]$
- Recursively determine $C[n]$ from $C[n\text{-}1]$:
  - Set of connected components in $T[n]$: $Q[n]$
  - Three possibilities for each $q \in Q[n]$:
    1. $q \cap C[n\text{-}1]$ is empty
       - A new minimum is encountered
       - Incorporate q into $C[n\text{-}1]$ to form $C[n]$: $C[n] = q \cup C[n\text{-}1]$
    2. $q \cap C[n\text{-}1]$ contains one connected component of $C[n\text{-}1]$
       - Connected component q lies within the catchment basin of the some regional minimum ($q \subseteq C_{n-1}(M_i)$)
       - Incorporate q into $C[n\text{-}1]$ to form $C[n]$: $C[n] = q \cup C[n\text{-}1]$
    3. $q \cap C[n\text{-}1]$ contains more than one connected component of $C[n\text{-}1]$
       - A ridge separating two (or more) catchment basins is encountered
       - A damn is built to prevent flooding across basins
       - Dilate $q \cap C[n\text{-}1]$ with a 3x3 structuring element, restricting the dilation to q
         - See previous description

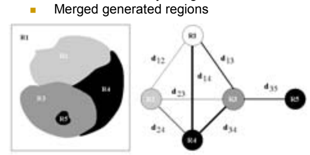
Watershed Segmentation Example

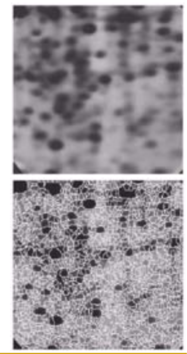
- Diffuse object example
- Images shown:
  - Observation
  - Gradient
  - Watershed result
  - Watershed result superimposed on observation



Over Segmentation

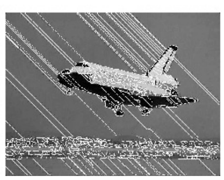
- Watershed advantages:
  - Closed boundaries
  - Good edge localization
- Watershed disadvantages:
  - Over segmentation
    - Excessive number of minima
  - Over segmentation solutions:
    - Preprocessing filtering
    - Restrict minima by using markers
    - Merged generated regions

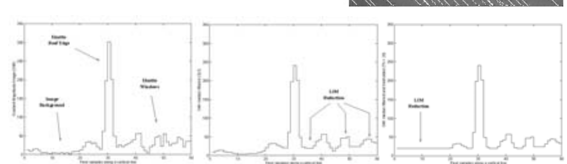




Over Segmentation – Prefiltering

- Original image gradient contains excessive minima
- Prefiltering:
  - Median removes isolated minima
  - Thresholding removes inconsequential background minima
- Postprocess to remove background lines
  - Results of gradient definition





Over Segmentation – Markers

- Use markers to define “super minima”
  - Region that is surrounded by greater magnitude points
  - Points in region form a connected component
  - Points in the connected component have the same gray level value
- Marked points shown on a smoothed image
  - Light gray denotes markers
- Apply watershed
  - Markers are the only allowable minima
  - Each region contains a single marker and background
  - Partition each region into foreground and background
    - Each region is considered in independent “image”
    - Final results consist of boundaries around the foreground in each marker defined region