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:
  \[ R(x, y) = x_1 x_2 x_3 + x_4 + x_5 + x_6 + x_7 + x_8 \]
- Isolated point detection: \(|R| \geq T\)
- Example:
  - X-ray image
  - T=90% of max value
  - Input, gradient, threshold output

Line Detection
- 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:
  \[ \nabla \left[ \frac{f_x}{f_y} \right] = \frac{f_y}{f_x} \]
  - Magnitude:
    \[ \sqrt{f_x^2 + f_y^2} \]
  - Direction (angle)
    \[ \tan^{-1} \left( \frac{f_y}{f_x} \right) \]
  - Perpendicular to edge
  - Approximation:
    \[ \nabla \approx \frac{1}{2} \left( \begin{array}{c} f_x \\ f_y \end{array} \right) \]
  - Shown: mask realizations

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
Laplacian of a Gaussian (LoG)

- Recall Laplacian:
  \[ \nabla^2 f \quad \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \]

- Edge detection limitations:
  - Produces double edges
  - Insensitive to edge direction
  - Sensitive to noise

- Pre-smooth with Gaussian filter

- Combined (linear) smoothing and derivative operations

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

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

Edge Linking

- Procedures often yield broken edges
  - Noise, illumination irregularities, etc.
- Link neighboring segments based on predefined criteria
  - Example criteria:
    - Strength of gradients
      \[ \| \nabla f(x_0, y_0) \| \leq E \]
    - Direction of gradients
      \[ \| \nabla f(x_0, y_0) \cdot \alpha(x_0, y_0) \| < \Delta \]
    - Applied over predefined search neighborhood

Edge Linking Example

- Goal: license plate localization
- Shown: horizontal and vertical gradient images
- Linking criteria:
  - Gradient \( \geq 25 \)
  - Angle differences \( \leq 15^\circ \)
- 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) plane

Hough Transform (II)

- Line equation:
  \[ y = ax + b \]
- Parameter space:
  - Fix \( x_i \) and \( y_j \)
  - Line in parameter space:
    \[ b = -x_i a + y_j \]
  - 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 Example (I)

- Image space:
  - Five points
- Parameter 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

---

**Collimation of X-Ray Images (II)**

- Radon transform:
  \[ R(\rho, \theta) = \int f(x, y) \delta(x \cos \theta + y \sin \theta - \rho) \, dx \, dy \]
- 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:
  - (I) 52%
  - (II) 9%
  - (III) 14%
  - (IV) 23%
  - (V) 10%
  - (VI) 7%
  - (VII) 6%
  - (VIII) 5%
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:
  \[ f(x,y) = f_{(x,y)_{(c,y)}} \]
- Utilizing the log:
  \[ \tilde{f}(x,y) = \ln(f(x,y)) = \ln(f_{(x,y)_{(c,y)}}) = f_{(x,y)_{(c,y)}} \]

- If components independent:
  - Convolve distributions
  - If \( f_{(x,y)} \) 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
  3. Recalculate the positions (values) of the \( K \) centroids
     - Utilized appropriate distance metric (Euclidean, city block, etc.)
  4. Repeat Steps 2 and 3 until centroid movements are below a fixed threshold

K-Means Clustering Example

- \( K=2 \)
  - Arbitrarily choose \( K \) objects as initial cluster centers
  - Assign each object to most similar center
  - Update the cluster means
  - Reassign

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:
  \[ L(\theta; Z) = \sum_{i=1}^{N} \log(\pi p_{\theta_1}(y_i) + (1-\pi) p_{\theta_2}(y_i)) \]
  - Direct maximization is difficult
  - Solution: suppose we know the values of the \( \Delta_i \)’s
    - Log-likelihood function reduces to:
      \[ L(\theta; Z, \Lambda) = \sum_{i=1}^{N} \log(\pi p_{\theta_1}(y_i) + (1-\pi) p_{\theta_2}(y_i)) \]

EM Algorithm for Two-Component Gaussian Mixture

1. Make an initial guesses for the parameters \( \hat{\mu}_i, \hat{\sigma}_i^2, \hat{\pi}_i, \hat{\Delta}_i \)
2. Expectation Step: compute the responsibilities
   \[ \hat{\gamma}_i = \frac{\pi p_{\theta_1}(y_i)}{(1-\pi)p_{\theta_1}(y_i) + \pi p_{\theta_2}(y_i)}, \quad i=1, 2, \ldots, N \]
3. Maximization Step: compute the weighted means and variances
   \[ \hat{\mu}_i = \frac{\sum_{i=1}^{N} (1-\hat{\gamma}_i) \hat{Y}_i}{\sum_{i=1}^{N} (1-\hat{\gamma}_i)} \]
   \[ \hat{\sigma}_i^2 = \frac{\sum_{i=1}^{N} (1-\hat{\gamma}_i)(\hat{Y}_i - \hat{\mu}_i)^2}{\sum_{i=1}^{N} (1-\hat{\gamma}_i)} \]
   \[ \hat{\mu}_2 = \frac{\sum_{i=1}^{N} \hat{\gamma}_i \hat{Y}_i}{\sum_{i=1}^{N} \hat{\gamma}_i} \]
   \[ \hat{\sigma}_2^2 = \frac{\sum_{i=1}^{N} \hat{\gamma}_i (\hat{Y}_i - \hat{\mu}_2)^2}{\sum_{i=1}^{N} \hat{\gamma}_i} \]
   and the mixing probability \( \hat{\pi} = \frac{\sum_{i=1}^{N} \hat{\gamma}_i}{N} \)
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)

Relation Between EM and K-Means

- In the EM (Baum-Welch) algorithm
  - Utilized binary decisions: \( f_i = 1 \) if \( y_i - \hat{\mu}_2 < |y_i - \hat{\mu}_1| \)
  - Then \( \hat{\mu}_1 \) and \( \hat{\mu}_2 \) are unweighted means
    \[ \hat{\mu}_1 = \frac{\sum f_i y_i}{\sum f_i} \quad \hat{\mu}_2 = \frac{\sum (1-f_i) y_i}{\sum (1-f_i)} \]
  - 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:
  \[ p(x) = \frac{1}{P_1 + P_2} \]
- Probability of classifying Object 2 as Object 1:
  \[ E_2(T) = \int p_2(z) dz \]
- Probability of classifying Object 1 as Object 2:
  \[ E_1(T) = \int p_1(z) dz \]
- Overall probability of error:
  \[ E(T) = P_1 E_2(T) + P_2 E_1(T) \]

Optimal Thresholding (II)

- Solve for optimal threshold \( T \):
  \[ \frac{d E(T)}{dT} = \frac{1}{P_1} \int p_1(z) dz + \frac{1}{P_2} \int p_2(z) dz \]
  \[ = P_1 p_1(T) - P_2 p_2(T) \]
  \[ \Rightarrow P_1 p_1(T) = P_2 p_2(T) \]
- In the Gaussian case:
  \[ p(x) = \frac{1}{\sqrt{2\pi} \sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \]
  - Solution is in the form of the quadratic:
    \[ AT^2 + BT + C = 0 \]
  - Results may yield two thresholds:
    - If both distributions have a common variance, \( \sigma^2 \):
      \[ T = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A} = \frac{\mu_1 - \mu_2}{2\sigma^2} \left( \frac{\sigma_1^2 + \sigma_2^2}{\sigma_1^2} \right) \]

Cardiogram Example (I)

- 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, $R_1, R_2, \ldots, R_n$ such that:
    1. $R = R_i$ for all $i$
    2. $R_i$ is a connected region, $i = 1, 2, \ldots, n$
    3. $P(R) = \text{TRUE}$ for $i = 1, 2, \ldots, n$
    4. $P(R_i) = \text{FALSE}$ for $i \neq j$
  - $P(R_i)$ is a logical operator that defines the properties of the region
  - Example: $P(R_i) = \text{TRUE}$ 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 $\leq 65$
    - Set as difference between maximum value and first mode in histogram
  - Pixel is B-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 \( P(R_i) = \text{FALSE} \)
  - Merged in the adjacent regions \( R_i \) and \( R_k \) for which \( P(R_i \cup R_k) = \text{TRUE} \)
  - Stop when no further merging or splitting is possible

Quadtree Split-Merge Example

- Homogeneity criteria:
  - \( P(R_i) = \text{TRUE} \) if \( \sum |z_i - m_i| \leq 2 \sigma_i \) for at least 80% of the pixels in \( R_i \)
    - Region mean: \( m_i \)
    - Region standard deviation: \( \sigma_i \)

- 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 $\mathbb{Z}^2$
- Standard operations:
  - Union
  - Intersection
  - Complement
  - Difference
  - Reflection
  - Translation

Dilation

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

Dilation Example

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

Erosion

- The erosion of $A$ by $B$:
  \[ A \ominus B = \{ x \mid (y, y) \in B \cap (x, x) \notin A \} \]
- Result:
  - The set of all displacements, $z$, such that $B$, translated by $z$, is contained in $A$
- Note dilation and erosion are duals of each other with the respective complementation and reflection:
  \[ (A \ominus B)^c = A \ominus B \]
**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
  \[ A \ominus B = (A \ominus B) \ominus B \]
  - The erosion of \( A \) by \( B \), followed by dilation of the result by \( B \)

- Opening and closing our duels of each other with respect to complementation and reflection
  \[ (A \oplus B)' = (A' \ominus B) \]
  - 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

**Closing**

- Closing smooths the contour of an object, fuses narrow breaks and long thin gulfs, eliminate small holes, and fills gaps in the contour
  \[ A \odot B = (A \odot B) \odot B \]
  - 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
  \[ (A \odot B)' = (A' \oplus B) \]
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

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:
    - $C_{n-1}(M_1) \cup C_{n-1}(M_2)$
  - 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

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 \( n \):
  \[ T[n] = \{(x,y) | g(x,y) < n \} \]
- Flood the image, and mark all pixels \( < \) the flood plane.

Watershed Algorithm (II)

- Union of flooded catchment basins:
  \[ C[n] = \bigcup_{M_i} C(M_i) \]

- Union of all catchment basins:
  \[ C[\max] = \bigcup_{M_i} C(M_i) \]

- Maximum image value: max
- Minimum image value: min

- Note: \( C[n-1] \subseteq C[n] \subseteq T[n] \)

- Each connected component in \( C[n-1] \) is contained in exactly one connected component of \( T[n] \)

Watershed Algorithm (III)

- Initialization: \( C[\min] = T[\min] \)
- Recursively determine \( C[n] \) from \( C[n-1] \):
  - Set of connected components in \( T[n] \): \( \mathcal{X}[n] \)
  - Three possibilities for each \( q \in \mathcal{X}[n] \):
    1. \( q \cap C[n-1] \) is empty
    2. A new minimum is encountered
    3. \( q \cap C[n-1] \) contains one connected component of \( C[n-1] \)
    4. \( q \cap C[n-1] \) contains more than one connected component of \( C[n-1] \)
  - A ridge separating two (or more) catchment basins is encountered
  - A dam is built to prevent flooding across basins
  - Dilate \( q \cap C[n-1] \) 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