Image Segmentation

- Definition
- Importance

- $\bigcup_{i=1}^{n} R_i$
- $R_i$'s are connected regions.
- $R_i \cap R_j = \emptyset, \quad i \neq j$
- $P(R_i) = \text{TRUE}$
- $P(R_i \bigcup R_j) = \text{FALSE}$
Image Segmentation

- Two Main Categories:
  - Edge Based
  - Region Based
Example:
- Constant Intensity
- Textural Intensity

**Figure 10.1** (a) Image containing a region of constant intensity. (b) Image showing the boundary of the inner region, obtained from intensity discontinuities. (c) Result of segmenting the image into two regions. (d) Image containing a textured region. (e) Result of edge computations. Note the large number of small edges that are connected to the original boundary, making it difficult to find a unique boundary using only edge information. (f) Result of segmentation based on region properties.
Digital Image Processing

Image Segmentation

- Detection of Discontinuities:
  - Point
  - Line
  - Edge

- Approximation for first and second derivatives

\[
\frac{\partial f}{\partial x} \approx \begin{cases} 
  f(x+1, y) - f(x, y) \\
  f(x, y) - f(x-1, y) \\
  0.5(f(x+1, y) - f(x-1, y)) 
\end{cases}
\]

\[
\frac{\partial^2 f}{\partial x^2} \approx f(x+1, y) - 2f(x, y) + f(x-1, y)
\]
Image Segmentation

Example:
Digital Image Processing

Image Segmentation

- **1\(^{st}\) and 2\(^{nd}\) Order Derivative Comparison:**
  - First Derivative:
    - Thicker Edge;
    - Strong Response for step changes;
  - Second Derivative:
    - Strong response for fine details and isolated points;
    - Double response at step changes.
Image Segmentation

- Isolated Point Detection:
  1. A Mask \( R = \sum_{i=1}^{9} w_i z_i \)
  2. Thresholding \( |R| \geq T \)

\[
\begin{array}{ccc}
-1 & -1 & -1 \\
-1 & 8 & -1 \\
-1 & -1 & -1 \\
\end{array}
\]
Digital Image Processing

Image Segmentation

- **Line Detection:**
  - Choose a Suitable Mask in desired direction
  - Thresholding

\[
\text{Line } i: |R_i| \geq |R_j|, \forall j
\]

<table>
<thead>
<tr>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>2</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>-1</th>
<th>2</th>
<th>-1</th>
<th>2</th>
<th>-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>-1</td>
<td>2</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>2</td>
<td>-1</td>
<td>2</td>
<td>-1</td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>2</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>2</td>
<td>-1</td>
<td>2</td>
<td>2</td>
<td>-1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Horizontal</th>
<th>+45°</th>
<th>Vertical</th>
<th>−45°</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

E. Fatemizadeh, Sharif University of Technology, 2011
Example (Laplacian):
- Double line in absolute value
- Single line in positive values
Image Segmentation

• Example:

Top-Left Zoom

Zeroed Neg. Values

Bottom-Right Zoom

Thr. Zeroed Neg. Values
• Edge Models:
  – Three Mathematical model:
Edge Models:

- Example
Image Segmentation

• Example:

\[ \frac{\partial f}{\partial x} \]

\[ \frac{\partial^2 f}{\partial x^2} \]
• Noise Problem:

![Diagram showing different noise levels with their corresponding partial derivatives]
Digital Image Processing

Image Segmentation

• Basic Edge Detection:
  – Gradient Operators:

\[
\nabla f = \begin{bmatrix} g_x & g_y \end{bmatrix}^T = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} \end{bmatrix}^T
\]

\[
M(x, y) = |\nabla f| = \left( (g_x)^2 + (g_y)^2 \right)^{1/2} \approx |g_x| + |g_y|
\]

\[
\alpha(x, y) = \tan^{-1}\left( \frac{g_y}{g_x} \right)
\]

E. Fatemizadeh, Sharif University of Technology, 2011
• Graphical Illustration:

\[ \nabla f = \begin{bmatrix} -2 \\ 2 \end{bmatrix} \]

**FIGURE 10.12** Using the gradient to determine edge strength and direction at a point. Note that the edge is perpendicular to the direction of the gradient vector at the point where the gradient is computed. Each square in the figure represents one pixel.
Digital Image Processing

Image Segmentation

• Gradient Operators:
  – Roberts
  – Prewitt
  – Sobel

\[
\begin{bmatrix}
  z_9 - z_5 \\
  z_8 - z_6
\end{bmatrix}
\]

\[
\begin{bmatrix}
  (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3) \\
  (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)
\end{bmatrix}
\]

\[
\begin{bmatrix}
  (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3) \\
  (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)
\end{bmatrix}
\]
Image Segmentation

- DiagonalEdges:

<table>
<thead>
<tr>
<th></th>
<th>Prewitt</th>
<th>Sobel</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>-1</td>
<td>0</td>
<td>-2</td>
</tr>
<tr>
<td>-1</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td>-2</td>
<td>-1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>-1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>-1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>
Digital Image Processing

Image Segmentation

Example:
- Magnitude

Original  $|G_x|$  

$|G_y|$  

$|G_x| + |G_y|$
Digital Image Processing

Image Segmentation

• Example:
  – Angle
• Pre-Smoothing Effect:
  – 5×5 averaging filter

\[ G_x \]

\[ G_y \]

\[ |G_x| + |G_y| \]
**Image Segmentation**

- **Diagonal Edge:**

  
  \[
  \begin{bmatrix}
  0 & 1 & 2 \\
  -1 & 0 & 1 \\
  -2 & -1 & 0 \\
  \end{bmatrix}
  \]

  
  \[
  \begin{bmatrix}
  -2 & -1 & 0 \\
  -1 & 0 & 1 \\
  0 & 1 & 2 \\
  \end{bmatrix}
  \]

-45° and +45° lines
Digital Image Processing

Image Segmentation

- Thresholding Effect:
  - Without smoothing (33%)
  - With smoothing (50%)
• The Marr-Hildreth Edge Detector:
  – LoG (Laplacian of Gaussian) Operator

\[ G(x, y) = \exp \left( -\frac{x^2 + y^2}{2\sigma^2} \right) \]

\[ \nabla^2 G(x, y) = \frac{\partial^2 G(x, y)}{\partial x^2} + \frac{\partial^2 G(x, y)}{\partial y^2} \]

\[ \nabla^2 G(x, y) = \left[ \frac{x^2}{\sigma^4} - \frac{1}{\sigma^2} \right] G(x, y) + \left[ \frac{y^2}{\sigma^4} - \frac{1}{\sigma^2} \right] G(x, y) \]

\[ \nabla^2 G(x, y) = \left[ \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} \right] G(x, y) \]

\[ \nabla^2 G(x, y) = \pm \left[ \frac{r^2 - 2\sigma^2}{\sigma^4} \right] \exp \left( -\frac{r^2}{2\sigma^2} \right) \]
Image Segmentation

• LoG Illustration:
  – Exact
  – Approximated
Image Segmentation

• The Marr-Hildreth Edge Detector:
  
  – Idea:
    • Smoothing
    • Sharpening

\[
g(x, y) = \left[ \nabla^2 G(x, y) \right] \ast f(x, y) = \nabla^2 \left[ G(x, y) \ast f(x, y) \right]
\]

  – Algorithm:
    • Compute Laplacian of Gaussian filtered
    • Find zero-crossing
      – Use a 3×3 window around each pixel
      – ZC occurs if \textit{at least} two opposing neighbors has different \textit{signs}.
Digital Image Processing

Image Segmentation

• Example:

Original

LoG

ZC using 0 Thr

ZC using 4%*Max as Thr

FIGURE 10.22
(a) Original image of size 834 × 1114 pixels, with intensity values scaled to the range [0, 1].
(b) Results of Steps 1 and 2 of the Marr-Hildreth algorithm using \( \sigma = 4 \) and \( n = 25 \).
(c) Zero crossings of (b) using a threshold of 0 (note the closed-loop edges).
(d) Zero crossings found using a threshold equal to 4% of the maximum value of the image in (b). Note the thin edges.
Digital Image Processing

Image Segmentation

• LoG Approximation with DoG:
  – DoG: Difference of Gaussian:
    \[ \text{DoG}(x, y) = \frac{1}{2\pi \sigma_1^2} \exp \left( -\frac{x^2 + y^2}{2\sigma_1^2} \right) - \frac{1}{2\pi \sigma_2^2} \exp \left( -\frac{x^2 + y^2}{2\sigma_2^2} \right), \quad \sigma_1^2 > \sigma_2^2 \]
  – Ratio of 1.6:1, an engineering approximation of LoG.
  – For same zero-crossing:
    \[ \sigma^2 = \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 - \sigma_2^2} \ln \left[ \frac{\sigma_1^2}{\sigma_2^2} \right] \]
Image Segmentation

• Approximation Accuracy:
  – Solid Line: Negative LoG
  – Dashed Line: Negative DoG

\[ \sigma_1 : \sigma_2 = 1.75 : 1 \]
\[ \sigma_1 : \sigma_2 = 1.6 : 1 \]
• Canny Edge Detector:
  – Low Error rate
  – Localized edge points
  – Single edge point response
Digital Image Processing

Image Segmentation

- Canny Edge Detector Steps:
  - Smoothing
  - Compute Gradients
  - Non-maximum Suppression
  - Edge Tracking by hysteresis (double) thresholding
Canny – Smoothing:

- Using Gaussian filter:

\[ f_s(x, y) = G(x, y) \ast f(x, y) \]

\[ G(x, y) = \exp \left( -\frac{x^2 + y^2}{2\sigma^2} \right), \quad \sigma = 1.4 \]

- Practical Implementation:

\[
G(x, y) = \frac{1}{159} \begin{bmatrix}
2 & 4 & 5 & 4 & 2 \\
4 & 9 & 12 & 9 & 4 \\
5 & 12 & 15 & 12 & 5 \\
4 & 9 & 12 & 9 & 4 \\
2 & 4 & 5 & 4 & 2
\end{bmatrix}
\]
Image Segmentation

- **Canny – Gradient:**
  - Using any gradient kernel:
    \[ g_x(x, y) = w_x(x, y) \ast f_s(x, y) \]
    \[ g_y(x, y) = w_y(x, y) \ast f_s(x, y) \]
  - Practical Implementation:
    - Sobel Kernel
    \[
    w_x(x, y) = \begin{bmatrix}
      -1 & 0 & +1 \\
      -2 & 0 & +2 \\
      -1 & 0 & +1
    \end{bmatrix}, \quad w_y(x, y) = \begin{bmatrix}
      0 & 0 & 0 \\
      +1 & +2 & +1 \\
      -1 & -2 & -1
    \end{bmatrix}
    \]
Digital Image Processing

Image Segmentation

• Canny - Non-maximum Suppression (1):
  – Compute magnitude and angle of gradient:
    \[
    M(x, y) = \sqrt{g_x^2 + g_y^2} \approx |g_x| + |g_y|
    \]
    \[
    \theta(x, y) = \tan^{-1}\left(\frac{g_y}{g_x}\right)
    \]
  – Quantize the \(\theta(x, y)\) to nearest 45°, \(\theta_Q(x, y)\)
Digital Image Processing

Image Segmentation

- Canny - Non-maximum Suppression (2):
  - Quantization:
Canny - Non-maximum Suppression (3):

- Compare $M(x, y)$ with $M(x', y')$ in positive and negative gradient direction.
  - If greater than both then keep it, $g_N(x, y) = M(x, y)$
    else suppress it, $g_N(x, y) = 0$
  - If $\theta_Q(x, y) = 0^\circ$, then the pixels $(x+1, y)$, $(x, y)$, and $(x-1, y)$ are examined.
  - If $\theta_Q(x, y) = 90^\circ$, then the pixels $(x, y+1)$, $(x, y)$, and $(x, y-1)$ are examined.
  - If $\theta_Q(x, y) = 45^\circ$, then the pixels $(x+1, y+1)$, $(x, y)$, and $(x-1, y-1)$ are examined.
  - If $\theta_Q(x, y) = 135^\circ$, then the pixels $(x+1, y-1)$, $(x, y)$, and $(x-1, y+1)$ are examined.
Image Segmentation

- **Canny – Edge Tracking (1):**
  - Select two threshold \((T_H, T_L), T_H = kT_L\)
  - Form two images using two threshold:

    \[
    g_{NH}(x, y) = \begin{cases} 
    1 & \text{if } g_N(x, y) \geq T_H \\
    0 & \text{if } g_N(x, y) < T_H
    \end{cases}
    \]
    Fewer and Strong Edge

    \[
    g_{NL}(x, y) = \begin{cases} 
    1 & \text{if } g_N(x, y) \geq T_L \\
    0 & \text{if } g_N(x, y) < T_L
    \end{cases}
    \]
    More and Weak/Strong Edge

  - Eliminate strong edge from \(g_{NH}\)

    \[
    g_{NL}(x, y) = g_{NL}(x, y) - g_{NH}(x, y)
    \]
• Canny – Edge Tracking (2):
  
  – All strong edge in $g_{NH}$ are store and marked immediately.

  – Gaps in $g_{NH}$ with fill using $g_{NL}$

  1. Locate the next unvisited pixel, $p$, in $g_{NH}(x,y)$
  2. Mark as valid edge all weak pixels in $g_{NL}(x,y)$ that are connected to $p$, using $N8$ criteria.
  3. If all nonzero pixel in $g_{NH}(x,y)$ has been visited go to step #4 else go to step #1
  4. Set to zero all pixels in $g_{NL}(x,y)$ that were not marked as valid edge.
  5. Append $g_{NH}(x,y)$ and nonzero elements of $g_{NL}(x,y)$
Image Segmentation

• Canny Example (1):
  – Original and Smoothed
Digital Image Processing

Image Segmentation

- Canny Example (1):
  - Gradient images:

(a) Smoothed
(b) Gradient magnitudes
Image Segmentation

- Canny Example (1):
  - Non-maximum suppression:

(a) Gradient values     (b) Edges after non-maximum suppression
• Canny Example (1):
  – Double Thresholding:

(a) Edges after non-maximum suppression

(b) Double thresholding
Image Segmentation

- Canny Example (1)
  - Edge Tracking:

(a) Double thresholding  (b) Edge tracking by hysteresis  (c) Final output
Digital Image Processing

Image Segmentation

• Canny Example (2):

FIGURE 10.25
(a) Original image of size 834 × 1114 pixels, with intensity values scaled to the range [0, 1].
(b) Thresholded gradient of smoothed image.
(c) Image obtained using the Marr-Hildreth algorithm.
(d) Image obtained using the Canny algorithm. Note the significant improvement of the Canny image compared to the other two.
• Canny Example (3):

**FIGURE 10.26**
(a) Original head CT image of size $512 \times 512$ pixels, with intensity values scaled to the range $[0, 1]$.
(b) Thresholded gradient of smoothed image.
(c) Image obtained using the Marr-Hildreth algorithm.
(d) Image obtained using the Canny algorithm.
(Original image courtesy of Dr. David R. Pickens, Vanderbilt University.)
• Edge Linking (Local Processing):
  – Similarity of two edge pixels at $(x,y)$ and $(s,t)$:
    $$\left| M(s,t) - M(x,y) \right| \leq E \quad \text{and} \quad \left| \alpha(s,t) - \alpha(x,y) \right| \leq A$$
  – Connect if both condition satisfied

• Computational Expensive.
Edge Linking (Local Processing):

- A simple algorithm:
  
  - Compute $M(x,y)$ and $\alpha(x,y)$ for input image.
  
  - Form a binary image $g(x,y)$:

    $$
g(x,y) = \begin{cases} 
    1 & M(x,y) > T_M \text{ and } \alpha(x,y) = A \pm T_A \\
    0 & \text{otherwise}
    \end{cases}
    $$

    $T_M$: Threshold, $A$: Specified angle direction, $\pm T_A$: acceptable direction margin

  - Scan rows of $g$ and fill (set to 1) all gap (0's) that do not exceed a specified length, $K$.

  - Detect gaps in any direction, $\theta$, by rotation the $g$ by $\theta$, and apply the horizontal scanning scheme. Rotate the result back to $-\theta$
Digital Image Processing

Image Segmentation

• Example:
  – Gradient Magnitude
  – Horizontally Connection
  – Vertical Connection
  – OR of V. and H.
  – Thinning

FIGURE 10.27 (a) A $534 \times 566$ image of the rear of a vehicle. (b) Gradient magnitude image. (c) Horizontally connected edge pixels. (d) Vertically connected edge pixels. (e) The logical OR of the two preceding images. (f) Final result obtained using morphological thinning. (Original image courtesy of Perceptics Corporation.)
Image Segmentation

- Edge Linking Using Polygonal Fitting:
  - Fit a polygon to a set of points.
**Image Segmentation**

- **Edge Linking Using Polygonal Fitting:**
  - Read more: Pages: 728-732.
Global Edge Linking by the Hough Transform:

\[(x_i, y_i) \& y = ax + b \Rightarrow y_i = ax_i + b\]

\[b = -x_i a + y_i\]

All \((x_i, y_i)\)'s on a line intersect each other at \((a,b)\)
Image Segmentation

- Hough Transform in Cartesian:
  - Scan and fill the parameter space

\[(a, b)\]
\[(x_1, y_1)\]
\[(x_2, y_2)\]
\[(x_3, y_3)\]
\[(x_4, y_4)\]
Digital Image Processing

Image Segmentation

- Hough Transform in Polar
  - Problem with slope of lines (for vertical line, \( a=\infty \))

\[
(x_i, y_i) \ & \ x \cos \theta + y \sin \theta = \rho \Rightarrow x_i \cos \theta + y_i \sin \theta = \rho
\]

All \((x_i, y_i)\)'s on a sin intersect each other at \((\rho, \theta)\)
Digital Image Processing

Image Segmentation

- Example:

**FIGURE 10.33**
(a) Image of size $101 \times 101$ pixels, containing five points.
(b) Corresponding parameter space. (The points in (a) were enlarged to make them easier to see.)
Image Segmentation

Example:
Digital Image Processing

Image Segmentation

• Example:

**FIGURE 10.34** (a) A 502 × 564 aerial image of an airport. (b) Edge image obtained using Canny’s algorithm. (c) Hough parameter space (the boxes highlight the points associated with long vertical lines). (d) Lines in the image plane corresponding to the points highlighted by the boxes. (e) Lines superimposed on the original image.
Digital Image Processing

Image Segmentation

• Circle Hough Transform (CHT):

\[(x_i, y_i) \& (x-c_1)^2 + (y-c_2)^2 = c_3^2 \Rightarrow (x_i-c_1)^2 + (y_i-c_2)^2 = c_3^2\]

All \((x_i, y_i)\)'s on a spherical surface intersect each other at \((p_1, p_2, p_3)\)

– 3D Parameter Space

• Extract each circle (independent of radius):

\[
\begin{align*}
    x_i &= c_1 + c_3 \cos \theta \\
    y_i &= c_2 + c_3 \sin \theta \\
    c_1 &= x_i - c_3 \cos \theta \\
    c_2 &= y_i - c_3 \sin \theta
\end{align*}
\]

\(\Rightarrow\) Fill \((c_1, c_2)\) space at fixed \(c_3\)
Image Segmentation

- Hough Transform for circle:
Image Segmentation

- Hough Transform for circle:
  - Small $r$ (Left) and Large $r$ (Right)
Image Segmentation

- Thresholding:
  
  \[ f(x, y) > T \] then \((x, y)\) is belong to the object, else \((x, y)\) is belong to the background.

  - Bi-level (T):
    \[ g(x, y) = \begin{cases} 
    1 & f(x, y) > T \\
    0 & f(x, y) \leq T 
    \end{cases} \]

  - Multi-level \((T_1, T_2, \ldots, T_n)\)
    \[ g(x, y) = \begin{cases} 
    a & f(x, y) > T_2 \\
    b & T_1 < f(x, y) \leq T_2 \\
    c & f(x, y) \leq T_1 
    \end{cases} \]

- Challenge:
  
  - Threshold Selection
    - Histogram
Image Segmentation

- Bi-Modal and Multi-Modal Histogram:
Image Segmentation

- Noise Effect on image Thresholding:

**Figure 10.36** (a) Noiseless 8-bit image. (b) Image with additive Gaussian noise of mean 0 and standard deviation of 10 intensity levels. (c) Image with additive Gaussian noise of mean 0 and standard deviation of 50 intensity levels. (d)–(f) Corresponding histograms.
Image Segmentation

Digital Image Processing

- Illumination and Reflectance Effect on image Thresholding:

**FIGURE 10.37** (a) Noisy image. (b) Intensity ramp in the range [0.2, 0.6]. (c) Product of (a) and (b). (d)–(f) Corresponding histograms.
Image Segmentation

- **Basic Global Thresholding:**
  - A Heuristic approach:
    1. Initial guess on $T$
    2. Segment image to $G_1(>T)$ and $G_2(\leq T)$
    3. Compute average value of $G_1, m_1$, and $G_2, m_2$.
    4. Set $T$ be average of $m_1$ and $m_2$
    5. Repeat 2-4 until small changed in successive $T$ values.$(\Delta T)$
Example:

\[ T_0 = 0 \xrightarrow{\text{Convergence}} T_f = 125.4 \]
• Optimal Global Thresholding, Otsu’s Methods:
  – Basic idea:
    • Separability of two class of data:
      – Large distance between two classes statistical means. (Between-Class Distance)
      – Small distance between each class samples (With-Class Distance)
      – Global variance to Between-Class variance ratio/
Digital Image Processing

Image Segmentation

- Otsu Optimal Thresholding:
  - A \( M \times N \) gray level image.
  - \( L \) distinct intensity level: \{0, 1, 2, ..., \( L-1 \}\)
  - \( n_i \) = # of pixels with gray level \( i \), \( MN = n_0 + n_1 + \ldots + n_{L-1} \)
  - Normalized histogram: \( p_i = n_i / MN \)
  \[
  \sum_{i=0}^{L-1} p_i = 1
  \]
  - We seek for optimal threshold \( T(k) = k \)
Image Segmentation

• Otsu Optimal Thresholding:
  – With threshold $k$, we have two classes of pixel:
    
    \[ C_1 = \{ \text{pixels} \mid \text{intensity} \in [0,k] \} \]
    \[ C_2 = \{ \text{pixels} \mid \text{intensity} \in k \left[ k+1, L-1 \right] \} \]

  – Statistics of entire image:
    • Global mean and Global variance:
      \[
      m_G = \sum_{i=0}^{L-1} i p_i \\
      \sigma_G^2 = \sum_{i=0}^{L-1} (i - m_G)^2 p_i
      \]
• Otsu Optimal Thresholding:
  – Statistics of each class:
    • Class Probability:
      \[ P_1(k) = \sum_{i=0}^{k} p_i \]
      \[ P_2(k) = \sum_{i=k+1}^{L-1} p_i = 1 - P_1(k) \]
    • Class mean:
      Class mean: \( m_1(k) = \sum_{i=0}^{k} i P(i/C_1) = \sum_{i=0}^{k} i \frac{P(C_1/i) P(i)}{P(C_1)} = \frac{1}{P_1(k)} \sum_{i=0}^{k} i p_i \)
      Class mean: \( m_2(k) = \sum_{i=k+1}^{L-1} i P(i/C_2) = \sum_{i=k+1}^{L-1} i \frac{P(C_2/i) P(i)}{P(C_2)} = \frac{1}{P_2(k)} \sum_{i=k+1}^{L-1} i p_i \)
      \[ m(k) = \sum_{i=0}^{k} i p_i \]
      \[ m_G = P_1(k) m_1(k) + P_2(k) m_2(k) \]
Digital Image Processing

Image Segmentation

• Otsu Optimal Thresholding:
  – Separability Index:
    \[ \eta(k) = \frac{\sigma_B^2(k)}{\sigma_G^2} \]
    \[ \sigma_B^2 = P_1(k)(m_1(k) - m_G)^2 + P_2(k)(m_2(k) - m_G)^2 \]
    \[ = P_1(k)P_2(k)(m_1(k) - m_2(k))^2 \]
    \[ = \frac{(m_GP_1(k) - m(k))^2}{P_1(k)(1 - P_1(k))} \]
  – Optimal Solution:
    \[ k^* = \arg \max_{0 \leq k \leq L-1} \frac{\sigma_B^2(k)}{\sigma_G^2} \]
    Measure of Seaparability: \( \eta(k^*) \)
Image Segmentation

Example:

**FIGURE 10.39**
(a) Original image.
(b) Histogram (high peaks were clipped to highlight details in the lower values).
(c) Segmentation result using the basic global algorithm from Section 10.3.2.
(d) Result obtained using Otsu’s method. (Original image courtesy of Professor Daniel A. Hammer, the University of Pennsylvania.)
Image Segmentation

• Example – Noise Effect:

**FIGURE 10.40** (a) Noisy image from Fig. 10.36 and (b) its histogram. (c) Result obtained using Otsu’s method. (d) Noisy image smoothed using a $5 \times 5$ averaging mask and (e) its histogram. (f) Result of thresholding using Otsu’s method.
**Image Segmentation**

- Example – Region Size Effect:

**FIGURE 10.41** (a) Noisy image and (b) its histogram. (c) Result obtained using Otsu’s method. (d) Noisy image smoothed using a $5 \times 5$ averaging mask and (e) its histogram. (f) Result of thresholding using Otsu’s method. Thresholding failed in both cases.
Image Segmentation

• Improve Using Edge Information:
  – Basic idea:
    • Estimate histogram using pixels around the edges
  – Algorithm:
    • Create an edge-map \((M(x,y)\) or Laplacian) from \(f(x,y)\)
    • Threshold the edge-map and produce a binary image \(g_T(x,y)\)
    • Compute histogram using only the ONE pixels in \(g_T(x,y)\)
    • Estimate the optimal threshold.
Image Segmentation

- Example:

**FIGURE 10.42** (a) Noisy image from Fig. 10.41(a) and (b) its histogram. (c) Gradient magnitude image thresholded at the 99.7 percentile. (d) Image formed as the product of (a) and (c). (e) Histogram of the nonzero pixels in the image in (d). (f) Result of segmenting image (a) with the Otsu threshold based on the histogram in (e). The threshold was 134, which is approximately midway between the peaks in this histogram.
Image Segmentation

- Example – Sequential Processing:
  - Otsu Thresholding
  - Laplacian Thresholding
  - Multiply original and Laplacian
  - Otsu Thresholding
Image Segmentation

• Example – Sequential Processing:

**FIGURE 10.43** (a) Image of yeast cells. (b) Histogram of (a). (c) Segmentation of (a) with Otsu’s method using the histogram in (b). (d) Thresholded absolute Laplacian. (e) Histogram of the nonzero pixels in the product of (a) and (d). (f) Original image thresholded using Otsu’s method based on the histogram in (e). (Original image courtesy of Professor Susan L. Forsburg, University of Southern California.)
Image Segmentation

• Example – Sequential Processing:
  – Using Lower threshold for Laplacian

**FIGURE 10.44**
Image in Fig. 10.43(a) segmented using the same procedure as explained in Figs. 10.43(d)–(f), but using a lower value to threshold the absolute Laplacian image.
Image Segmentation

• Optimal Multiple Thresholding:
  – We have $K$ classes, $C_1, C_2, \ldots, C_K$
  – Between Class variance:
    
    $$
    \sigma_B^2 = \sum_{k=1}^{K} P_k (m_k - m_G)^2
    $$
    
    $$
    P_k = \sum_{i \in C_k} p_i
    $$
    
    $$
    m_k = \frac{1}{P_k} \sum_{i \in C_k} ip_i
    $$
    
  – Optimal Solution:
    
    $$
    \{k_1^*, k_2^*, \ldots, k_{K-1}^*\} = \arg \max_{0<k_1<k_2<\ldots<k_{K-1}<L-1} \{\sigma_B^2(k_1, k_2, \ldots, k_{K-1})\}
    $$
    
  – A expensive computation task!
    
    • A search in $\mathbb{R}^{K-1}$
Image Segmentation

- Example

**FIGURE 10.45** (a) Image of iceberg. (b) Histogram. (c) Image segmented into three regions using dual Otsu thresholds. (Original image courtesy of NOAA.)
Image Segmentation

- Variable Thresholding:
  - Image Partitioning
  - Using Local Image Properties
  - Using Moving Average
Image Partitioning:

**FIGURE 10.46** (a) Noisy, shaded image and (b) its histogram. (c) Segmentation of (a) using the iterative global algorithm from Section 10.3.2. (d) Result obtained using Otsu’s method. (e) Image subdivided into six subimages. (f) Result of applying Otsu’s method to each subimage individually.
Image Segmentation

• Image Partitioning:
  – Histogram of each partition
    • Bimodal $\rightarrow$ Otsu Optimal Thresholding
Image Segmentation

• Using Local Image Properties:
  • Compute Local mean and Local standard deviation
  • Design a rule, such as:

\[
T(x, y) = a\sigma_L(x, y) + b m_L(x, y)
\]

\[
T(x, y) = a\sigma_L(x, y) + b m_G
\]
Image Segmentation

Example:

**FIGURE 10.48**
(a) Image from Fig. 10.43.
(b) Image segmented using the dual thresholding approach discussed in Section 10.3.6.
(c) Image of local standard deviations.
(d) Result obtained using local thresholding.
Using Moving Average:

Consider the following rule:

\[ T(x, y) = am_L(x, y) \]

Algorithm:

- Decide a scanning scheme
- Estimate Local mean via moving average techniques

\[
m(k + 1) = \frac{1}{n} \sum_{i=k+2-n}^{k+1} z_i
\]

\[
= \left[ \frac{1}{n} \sum_{i=k+1-n}^{k} z_i \right] + \frac{1}{n} (z_{k+1} - z_{k-n})
\]

\[
= m(k) + \frac{1}{n} (z_{k+1} - z_{k-n})
\]
Image Segmentation

- Example:
  - N=20, \( a=0.5 \)

FIGURE 10.49 (a) Text image corrupted by spot shading. (b) Result of global thresholding using Otsu’s method. (c) Result of local thresholding using moving averages.
Digital Image Processing

Image Segmentation

• Example:
  – $N=20, \ a=0.5$

![Image](image.png)

**FIGURE 10.50** (a) Text image corrupted by sinusoidal shading. (b) Result of global thresholding using Otsu’s method. (c) Result of local thresholding using moving averages.
Image Segmentation

• Multivariable Segmentation:
  – A Multiple sensor (R/G/B, Multi-Band, and etc.)
  – Image data: \( z \in \mathbb{R}^N \)
    • Thresholding:
      \[
g = \begin{cases} 
1 & \text{dist}(z,a) < T \\
0 & \text{otherwise}
\end{cases}
\]
    • \( a \): A specific color
  – Segmentation:
    • A Clustering task
Image Segmentation

• Region Growing:
  – Select a start (seed) point
  – Grow the point based on a certain property
    • Connectivity should be considered.
  – Seed point selection:
    • Handy
    • Highlighted point (Due to specific property)
Image Segmentation

- Region Growing:

- Seed Pixel
- Direction of Growth

(a) Start of Growing a Region

- Grown Pixels
- Pixels Being Considered

(b) Growing Process After a Few Iterations
Region Growing (Example):

– Determine seed points to maximum gray level.

– Growing criteria:
  
  • Gray level value difference (with respect to S.P.) less than a threshold.
  
  • Each candidate pixel should be $N_g$ of region.
Digital Image Processing

Image Segmentation

• Example – N8 Connectivity and $T = 68, 126$

FIGURE 10.51 (a) X-ray image of a defective weld. (b) Histogram. (c) Initial seed image. (d) Final seed image (the points were enlarged for clarity). (e) Absolute value of the difference between (a) and (c). (f) Histogram of (e). (g) Difference image thresholded using dual thresholds. (h) Difference image thresholded with the smallest of the dual thresholds. (i) Segmentation result obtained by region growing. (Original image courtesy of X-TEK Systems, Ltd.)
Image Segmentation

- Splitting and Merging:
  - Define a criteria for each region to be a valid segment.
  - Split each region which is not satisfy the criteria.
  - Merge two neighbor region based on criteria.
  - Split until a minimum size quadregions

(a) Whole Image
(b) First Split
(c) Second Split
(d) Merge

E. Fatemizadeh, Sharif University of Technology, 2011
Image Segmentation

• Splitting and Merging
Digital Image Processing

Image Segmentation

- Splitting-Merging Criteria
  - Example #1: n% of all pixels satisfy: $|z_j - m_i| \leq 2\sigma_i$
  - Example #2: $\sigma > a$ AND $0 < m < b$
Digital Image Processing

Image Segmentation

• Example:

![Example Image](image)

**FIGURE 10.53**
(a) Image of the Cygnus Loop supernova, taken in the X-ray band by NASA’s Hubble Telescope. (b)–(d) Results of limiting the smallest allowed quadregion to sizes of $32 \times 32$, $16 \times 16$, and $8 \times 8$ pixels, respectively. (Original image courtesy of NASA.)
Image Segmentation

- Morphological Watershed:
  - Topographic interpretation
• Morphological Watershed:
  – Three types of Points:
    • Points belonging to regional minimum;
    • A drop of water at these points, would fall (certainly) to a single minimum, **watershed or catchment basin**.
    • A drop of water at these points, would fall (equally) to more than one minimum. (**watershed line** or **divide line**)
Image Segmentation

• Segmentation:
  – Find analogy for each terms:
    • Punch a hole at each minimum regional;
    • Flood the topography by letting water rise through the holes.
    • Build a dam when rising water of two distinct catchment basin is about to merge.
Image Segmentation

- Example:
  - Image
  - Topographic Map
  - Two stage Flooding
Image Segmentation

- Example:
  - Flooding
  - Dam
Digital Image Processing

Image Segmentation

• Dam Construction:
  – Dilation
  – Connected Components
  – Intersection
  – ...
  – Constrained Dilation
  – Read Page 774-775
Image Segmentation

- **Watershed Segmentation**
  - Let $M_1, M_2, M_3, ..., M_n$ be the sets of coordinates of points in the regional minima of the image $g(x,y)$
  - $C(M_i)$ be the coordinates of points of the catchment basin associated with regional minima $M_i$
  - $T[n] = \{(s,t) \mid g(s,t) < n\}$
    - $T[n] = \text{Set of points in } g(x,y) \text{ which are lying below the plane } g(x,y) = n$
    - $n = \text{Stage of flooding, varies from } \text{min}+1 \text{ to } \text{max}+1$
    - $\text{min} = \text{minimum gray level value}$
    - $\text{max} = \text{maximum gray level value}$
Let $C_n(M_i)$ be the set of points in the catchment basin associated with $M_1$ that are flooded at stage $n$.

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

- if $(x,y) \in C(M_i)$ and $(x,y) \in T[n]$, then $C_n(M_i) = 1$ at $(x,y)$ otherwise it is 0.

$C[n]$: The union of flooded catchment basin portions at the stage $n$

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

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

- Algorithm keeps on increasing the level of flooding, and during the process $C_n(M_i)$ and $T[n]$ either increase or remain constant.
- Algorithm initializes $C[\text{min} + 1] = T[\text{min}+1]$, and then proceeds recursively assuming that at step $n$ $C[n-1]$ has been constructed.
- Let $Q$ be set of connected components in $T[n]$. 
For each connected component \( q \in Q[n] \), there are three possibilities of \( q \cap C[n-1] \):

- **Empty**
  - Occurs when a new minima is encountered, in this case \( q \) is added to set \( C[n-1] \) to form \( C[n] \).

- **contains one connected component of \( C[n-1] \):**
  - Occurs when \( q \) lies within a catchment basin of some regional minima, in that case

- **contains more than one connected components of \( C[n-1] \):**
  - occurs when ridge between two catchment basins is hit and further flooding will cause the waters from two basins will merge, so a dam must be built within \( q \).
Image Segmentation

Illustration (1):

\[ M_i \]

\[ C(M_i) \]

\[ T(n) \]

\[ C(n) \]
Illustration (2): 

\begin{align*}
C(n-1) & \quad T(n-1) \\
C_n-1(M_i) & \\
C(M_i) & \\
M_i & \\
q_3 & q_1 & q_2 \\
C(n) & \\
\end{align*}
Condition (a) occurs when a new minima is encountered, in this case q is added to set C[n-1] to form C[n].

Condition (b) occurs when q lies within a catchment basin of some regional minima, in that case

Condition (c) occurs when ridge between two catchment basins is hit and further flooding will cause the waters from two basins will merge, so a dam must be built within q.
Digital Image Processing

Image Segmentation

• Example:
Image Segmentation

• Oversegmentation Problem:
The Use of Markers:
- Selection of markers consists of two principal steps:
  - Preprocessing
  - Definition of a set of criteria
- There two types of markers:
  - External: associated with the background
  - Internal: associated with the objects of interest
- In the previous image due to large number of potential minima, image is over-segmented.
The Use of Markers:

- Smoothing filter will minimize the effect of small details
- A sample of Internal markers:
  - Region surrounded by the higher altitude points.
  - every region should be a connected component
  - Every point in the region should have same gray level value.
- External markers can be some regions of particular background color.
Image Segmentation

Example:
Image Segmentation

• The Use of Motion in Segmentation:
  – Spatial Domain
  – Frequency Domain

• Spatial Domain:
  – Main idea: Compare pixel by pixel (difference)

\[
d_{ij}(x, y) = \begin{cases} 
1 & \left| f(x, y, t_i) - f(x, y, t_j) \right| > T \\
0 & \text{O.W.} \rightarrow \text{Static Object}
\end{cases}
\]
Accumulative Difference Image (ADI):

- Consider $\{f(x, y, t_i)\}_{i=1}^{n}$ and $R(x, y)$ as reference frame.
- ADI compare reference frame with incoming frame.
- Increment counter of each pixel when a difference detected.
- ADI alternative:
  
  - Absolute
  - Positive
  - Negative
Accumulative Difference Image (ADI):

\[ A_k(x, y) = \begin{cases} 
A_{k-1}(x, y) + 1 & |R(x, y) - f(x, y, t_k)| > T \\
A_{k-1}(x, y) & \text{otherwise}
\end{cases} \]

\[ P_k(x, y) = \begin{cases} 
P_{k-1}(x, y) + 1 & \left[ R(x, y) - f(x, y, t_k) \right] > T \\
P_{k-1}(x, y) & \text{otherwise}
\end{cases} \]

\[ N_k(x, y) = \begin{cases} 
N_{k-1}(x, y) + 1 & \left[ R(x, y) - f(x, y, t_k) \right] < -T \\
N_{k-1}(x, y) & \text{otherwise}
\end{cases} \]
Digital Image Processing

Image Segmentation

• ADI Example

**FIGURE 10.49** ADIs of a rectangular object moving in a southeasterly direction. (a) Absolute ADI. (b) Positive ADI. (c) Negative ADI.
Image Segmentation

- Establishing a Reference Images:
  - Difference will erase static object:
    - When a dynamic object move out completely from its position, the back ground in replaced.

**FIGURE 10.50** Building a static reference image. (a) and (b) Two frames in a sequence. (c) Eastbound automobile subtracted from (a) and the background restored from the corresponding area in (b). (Jain and Jain.)
Image Segmentation

- Frequency Domain Methods:
  - A Static/black background
  - A Dynamic single pixel

\[
g_x(t, a_1) = \sum_{x=0}^{M-1} \left[ \sum_{y=0}^{N-1} f(x, y, t) \right] e^{j2\pi a_1 x \Delta t}, \quad t = 0, 1, 2, \ldots, K - 1
\]

\[
g_y(t, a_2) = \sum_{y=0}^{N-1} \left\{ \sum_{x=0}^{M-1} f(x, y, t) \right\} e^{j2\pi a_2 y \Delta t}, \quad t = 0, 1, 2, \ldots, K - 1
\]

\[
G_x(u_1, a_1) = \sum_{t=0}^{K-1} g_x(t, a_1) e^{-j2\pi u_1 t / K}, \quad u_1 = 0, 1, 2, \ldots, K - 1
\]

\[
G_y(u_2, a_1) = \sum_{t=0}^{K-1} g_y(t, a_1) e^{-j2\pi u_2 t / K}, \quad u_2 = 0, 1, 2, \ldots, K - 1
\]

\[
u_1 = a_1 V_1, \quad u_2 = a_2 V_2
\]
• An Example:
• Intensity Plot
Image Segmentation

- Spectrum Plot:

\[ G_x(u_1, a_1), \quad u_1 = 3 \]
Image Segmentation

• Spectrum Plot:

\[ G_x(u_2, a_2), \quad u_2 = 4 \]