Lecture 9 (4.2.07)

Image Segmentation

Shahram Ebadollahi
Lecture Outline

- Skeletonization
- GSAT
- Watershed Algorithm
- Image Segmentation – Introduction
- Edge detection and linking
- Thresholding
- Region-based Approach
- Motion Segmentation
Skeletonization (Medial Axis Transform)

B is a “Maximal Disc” in set X if there are no other discs included in X and containing B

Skeleton is the loci of the centers of all “maximal discs”

\[ S(X) = \bigcup_{k \geq 0} \{ \mathcal{E}_{kB}(X) \setminus \gamma_B[\mathcal{E}_{kB}(X)] \} \]
Skeletonization

\[ S(X) = \bigcup_{k=0}^{K} S_k(X) \]

\[ S_k(X) = e_{kB}(X) - \gamma_B(e_{kB}(X)) \]

\[ e_{kB}(X) = e_B(e_B(\cdots(e_B(X)))) \]

\[ K = \max\{k \mid e_{kB}(X) \neq \phi\} \]

Reconstruction

\[ X = \bigcup_{k=0}^{K} \delta_{kB}(S_k(X)) \]

\[ \delta_{kB}(X) = \delta_B(\delta_B(\cdots(\delta_B(S_k(X)))))) \]

Notion of “Maximal Disc”

Skeleton is the loci of the centers of all “maximal discs”
Gray-level SAT
(Skeletonization for Gray-level Images)
GSAT

• GSAT is the extension of skeletonization to gray-level images
• GSAT is the locus of the centers of maximal discs whose plane is perpendicular to the gray-level axis and which fit in the region above or below the topographic map of image
• Symmetry surface for gray-level == skeleton for bi-level
• Represent the collection of symmetry surfaces as a graph → GSAT graph

• nodes:
  • g_min: gray-level at bottom of surface path
  • delta_g: difference in gray-level between top and bottom
  • p_avg: average location of maximal discs on the path
  • n_avg: average size of maximal discs on the path

\[ D_n(f) = \delta_{nB}(f) \]
\[ S_n(f) = \phi_B(D_n(f)) - D_n(f) = \epsilon_B(D_{n+1}(f)) - D_n(f) \]
$B \sim \tilde{B} = \delta_{B_1}(B_0)$

GSAT - Implementation

Dual set of operations on odd image
Use structuring element $B_1$
Image Segmentation by Region Growing on GSAT graph
**Image Segmentation - Intro**

*Goal* decompose image into regions $R_k$ such that:

$$f = \bigcup_{k=1}^{K} R_k$$

$$R_i \cap R_j = \emptyset \quad i \neq j$$

$$H(R_k) = true \quad \forall k \in \{1, \cdots, K\}$$

$$H(R_i \cup R_j) = false \quad i \neq j$$

There are various approaches:

- edge-based vs. region-based
- global vs. local
- feature – texture, motion, color
Watershed

Goal: Find watershed lines
Watershed: Intuition
Recursion relation

\[ h_{X_{h+1}} + h_{T_{h+1}} + h_{X_h} \]
Boundaries between flooded catchment basins
**Edge Detection**

**Figure 10.6**
(a) Two regions separated by a vertical edge. 
(b) Detail near the edge, showing a gray-level profile, and the first and second derivatives of the profile.

\[
\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}
\]

**Laplacian**

\[
\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}
\]
Gradient Operation on Images

Vertical & Horizontal Edges

\[
\begin{array}{ccc}
  z_1 & z_2 & z_3 \\
  z_4 & z_5 & z_6 \\
  z_7 & z_8 & z_9 \\
\end{array}
\]

\[
\begin{array}{ccc}
  -1 & 0 & 0 \\
  0 & -1 & 1 \\
  1 & 1 & 1 \\
\end{array}
\]

Roberts

\[
\begin{array}{ccc}
  -1 & -1 & -1 \\
  0 & 0 & 0 \\
  1 & 1 & 1 \\
\end{array}
\]

Prewitt

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

Sobel

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

Isotropic condition is lost

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]

\[
\text{Isotropic condition is lost}
\]
Edge Detection - Example

**FIGURE 10.11**
Same sequence as in Fig. 10.10, but with the original image smoothed with a $5 \times 5$ averaging filter.

**FIGURE 10.12**
Diagonal edge detection.  
(a) Result of using the mask in Fig. 10.9(c).  
(b) Result of using the mask in Fig. 10.9(d). The input in both cases was Fig. 10.11(a).
Laplacian Operation on Images

\[ \nabla^2 f \cong 4z_5 - (z_2 + z_4 + z_6 + z_8) \]

Problems:
- Too sensitive to noise
- Double edges
- Edge direction not detectable

User for:
- Detecting edge *location*
- Dark or light side of edge

LoG

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

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

Sobel Gradient

Gaussian Function

Laplacian Operator

<table>
<thead>
<tr>
<th>-1</th>
<th>-1</th>
<th>-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>8</td>
<td>-1</td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
</tbody>
</table>

LoG

LoG Thresholded

Zero Crossings
Hough Method for Curve Detection

Hough Transform: Intuition

Algorithm:

1. For each point \((x, y)\) in image determine \(a_p\) and \(b_q\) that satisfy the line equation
2. Increment \(A(p,q)\) by 1
Hough Method: Normal line representation

\[ x \cos \theta + y \sin \theta = \rho \]
FIGURE 10.21
(a) Infrared image.
(b) Thresholded gradient image.
(c) Hough transform.
(d) Linked pixels.
(Courtesy of Mr. D. R. Cate, Texas Instruments, Inc.)
Edge-based segmentation: Problems

- Spurious edges due to noise and low quality image. Difficult to identify spurious edges.
- Dependent on local neighborhood information
- No notion of higher order organization of the image
- Gaps and discontinuities in the linked edges
Thresholding

FIGURE 10.26 (a) Gray-level histograms that can be partitioned by (a) a single threshold, and (b) multiple thresholds.

\[ T = T[x, y, p(x, y), f(x, y)] \]

\[ g(x, y) = \begin{cases} 
1 & \text{if } f(x, y) > T \\
0 & \text{if } f(x, y) \leq T 
\end{cases} \]

- local vs. global
- static vs. dynamic
Global Thresholding

1. \(T = T_0\)
2. Segment using \(T\)
3. Get average gray-level for region \(G_1\) \((f > T)\) and region \(G_2\) \((f \leq T)\)
4. \(T_{\text{new}} = \text{average of average of gray-levels}\)
5. Repeat until convergence

---

Input image

segmented

histogram

\(T = 30\)

\(T = 42.5\)

\(T = 46.7\)

\(T = 52.1\)
Adaptive Thresholding
Optimal Threshold

\[
\Pr(f(x, y) = z \mid (x, y) \in FG) \quad \Pr(f(x, y) = z \mid (x, y) \in BG)
\]

\[
\Pr(f(x, y) = z) = \Pr([f(x, y) = z, (x, y) \in BG] \lor [f(x, y) = z, (x, y) \in FG])
\]

\[
\Pr(f(x, y) = z) = \Pr((x, y) \in FG) \Pr(f(x, y) = z \mid (x, y) \in FG) + \\
\Pr((x, y) \in BG) \Pr(f(x, y) = z \mid (x, y) \in BG)
\]

\[
\Pr(f(x, y) = z) = P_{FG}P_1(z) + P_{BG}P_2(z) \quad \text{where,} \quad P_{FG} + P_{BG} = 1
\]
Optimal Threshold (cont.)

\[ E_1(T) = \int_{-\infty}^{T} p_2(z) \, dz \quad \text{(Probability of classifying a BG pixel as FG by mistake)} \]

\[ E_2(T) = \int_{T}^{\infty} p_1(z) \, dz \quad \text{(Probability of classifying a FG pixel as BG by mistake)} \]

Overall probability of error:

\[ E(T) = P_{BG} E_1(T) + P_{FG} E_2(T) \]

\[ T_{opt} = \arg\min_{T} E(T) \quad \text{(Optimal threshold minimizes the probability of misclassification)} \]

Gaussian densities case:

\[ p(z) = \frac{P_{FG}}{\sqrt{2\pi}\sigma_{FG}} e^{-\frac{(z-\mu_{FG})^2}{2\sigma_{FG}^2}} + \frac{P_{BG}}{\sqrt{2\pi}\sigma_{BG}} e^{-\frac{(z-\mu_{BG})^2}{2\sigma_{BG}^2}} \]

\[ T_{opt} = \frac{\mu_{FG} + \mu_{BG}}{2} + \frac{\sigma^2}{\mu_{FG} - \mu_{BG}} \ln\left( \frac{P_{FG}}{P_{BG}} \right) \]

\[ \sigma_{FG}^2 = \sigma_{BG}^2 = \sigma^2 \]
Split & Merge Method

1. Pick a grid structure and homogeneity property $H$

2. If for any region $R$, $H(R)=false$, split region into 4

3. If for any neighboring regions $H(R_1∪R_2)=true$, merge regions into single region

4. Stop when no more split or merge

\[ H(R_k) = true \quad cnt(|z_j - m_k| \leq 2\sigma_k) \geq 0.8 \]
Segmentation as Clustering in Feature Space

Feature Space (Color, Texture)

Clustering algorithms:
• K-means
• Gaussian Mixture Model
• Neural Network

Issues/Choices:
• Feature
• Distance
• Method
K-means Clustering Algorithm

- Select K initial cluster centers: $C_1, C_2, \ldots, C_K$
- Assign each pixel representation $x_i$ to nearest cluster $C_k$
- Recompute cluster centroids for all clusters
- Iterate until convergence
Accumulative Difference

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