Image Segmentation

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

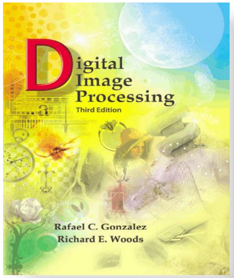


Image Segmentation

- Two Main Categories:
  - Edge Based
  - Region Based

Image Segmentation

- Example:
  - Constant Intensity
  - Textural Intensity

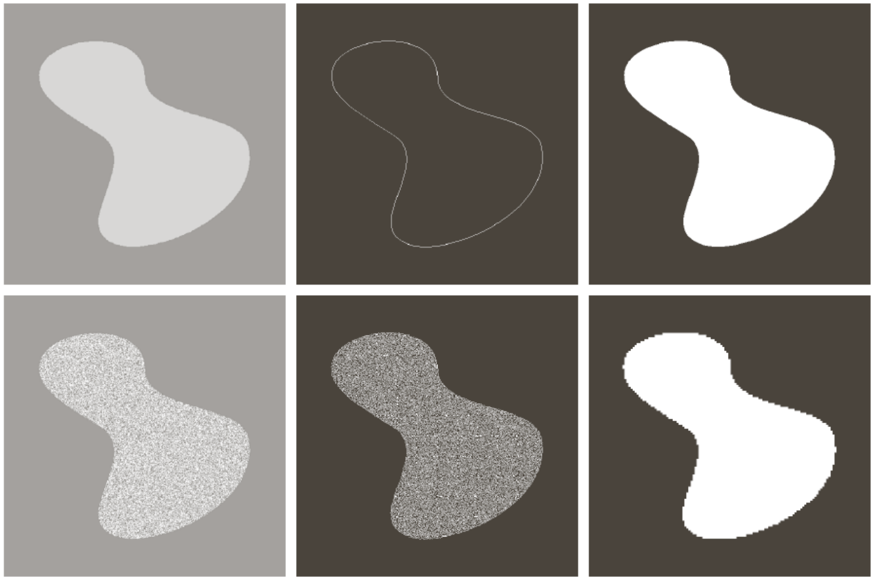


FIGURE 10.1 (a) Image containing a region of constant intensity. (b) Image showing the boundary of the inner region, obtained from intensity discontinuities. (c) Result of segmenting the image into two regions. (d) Image containing a textured region. (e) Result of edge computations. Note the large number 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.

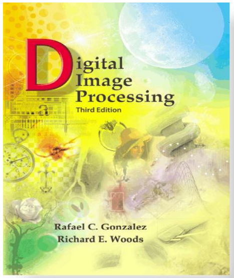


Image Segmentation

- Detection of Discontinuities:
  - Point
  - Line
  - Edge
- Approximation for first and second derivatives

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

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

Image Segmentation

- Example:

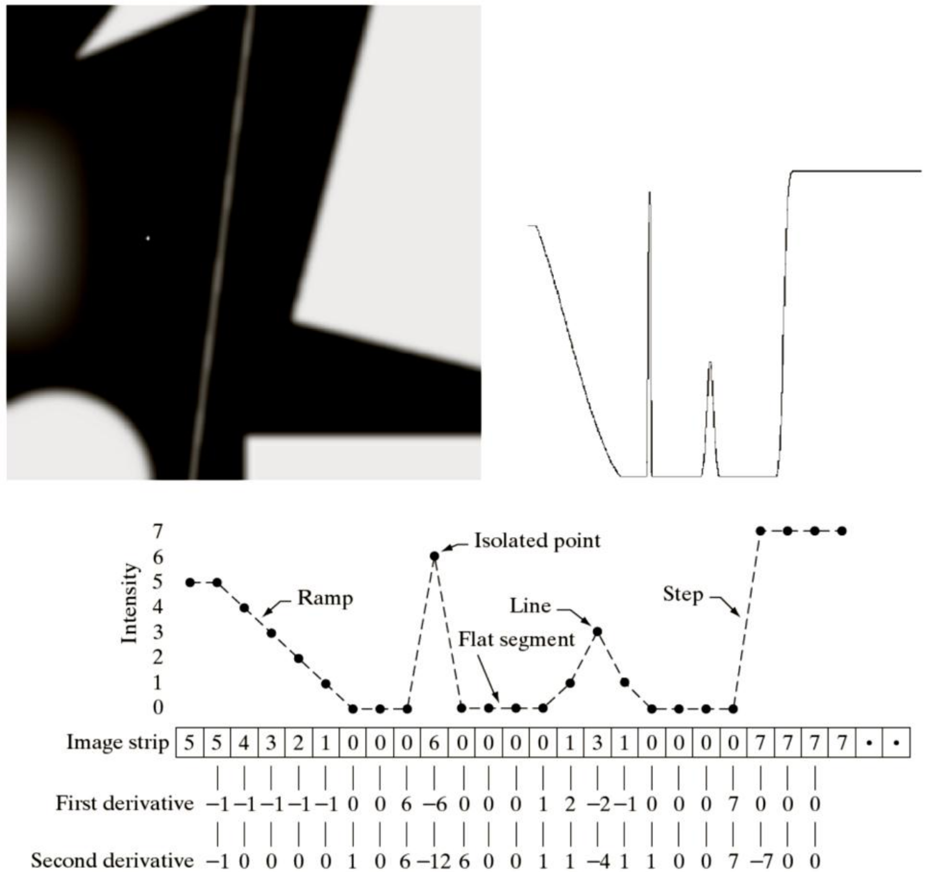


Image Segmentation

- 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 $|R| \geq T$

−1	−1	−1
−1	8	−1
−1	−1	−1

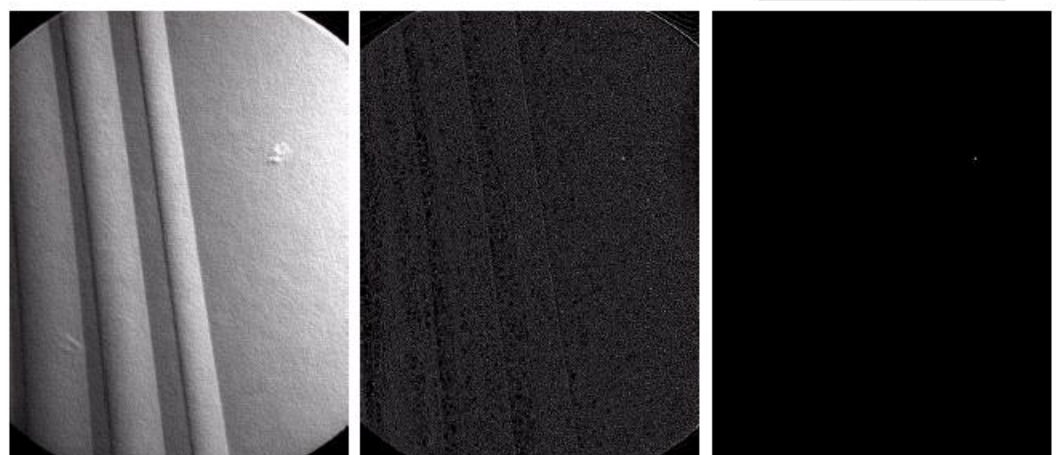


Image Segmentation

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

$$Line\ i : |R_i| \geq |R_j|, \forall j$$

−1	−1	−1
2	2	2
−1	−1	−1

Horizontal

2	−1	−1
−1	2	−1
−1	−1	2

+45°

−1	2	−1
−1	2	−1
−1	2	−1

Vertical

−1	−1	2
−1	2	−1
2	−1	−1

−45°

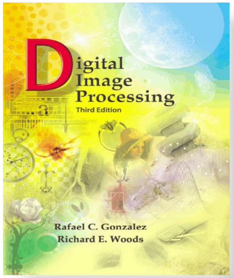
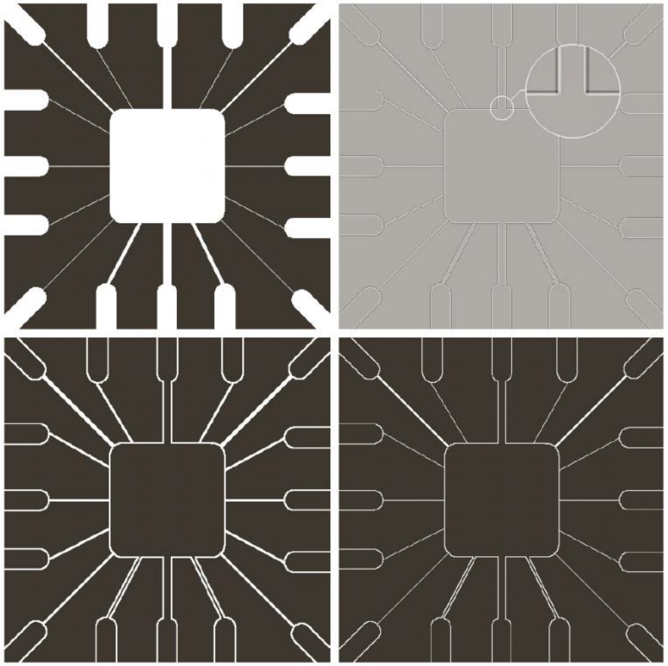


Image Segmentation

- Example (Laplacian):
  - Double line in absolute value
  - Single line in positive values

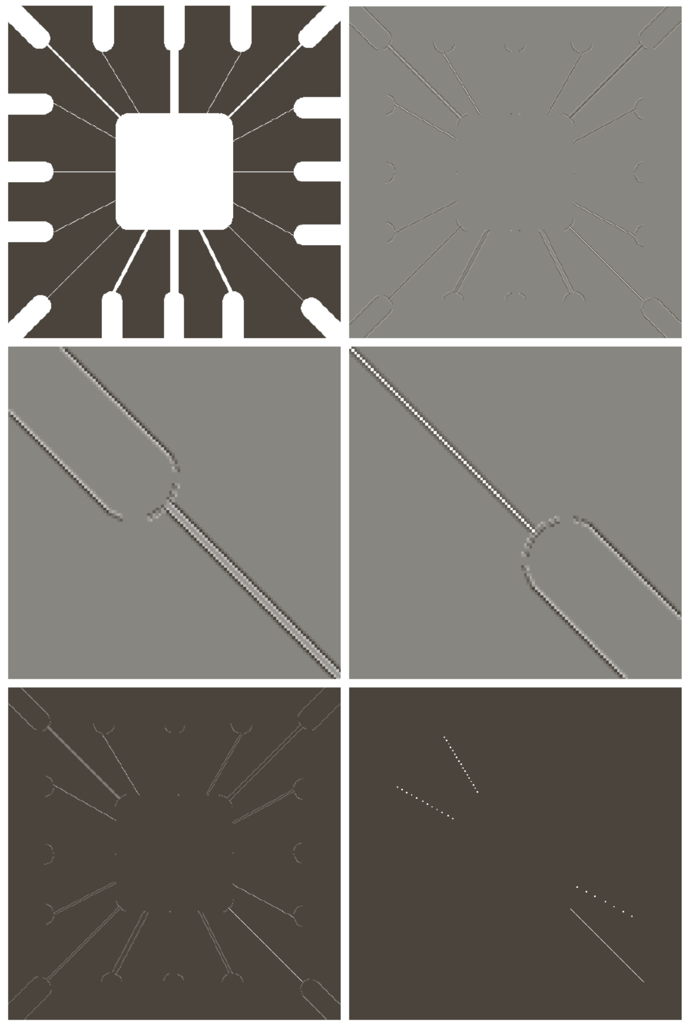


| a | b |
| c | d |

FIGURE 10.5
(a) Original image.
(b) Laplacian image; the magnified section shows the positive/negative double-line effect characteristic of the Laplacian.
(c) Absolute value of the Laplacian.
(d) Positive values of the Laplacian.

Image Segmentation

- Example:



Top-Left Zoom

Bottom-Right Zoom

Zeroed Neg. Values

Thr. Zeroed Neg. Values

Image Segmentation

- Edge Models:
  - Three Mathematical model:

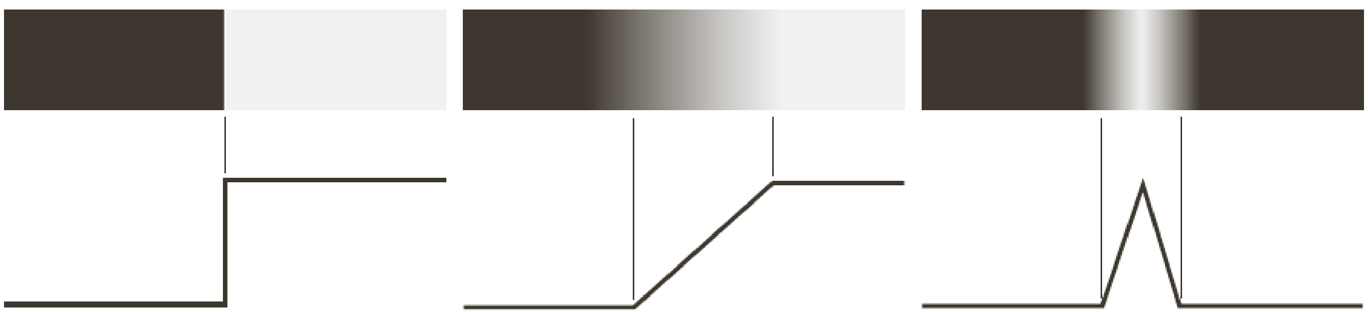


Image Segmentation

- Edge Models:
  - Example

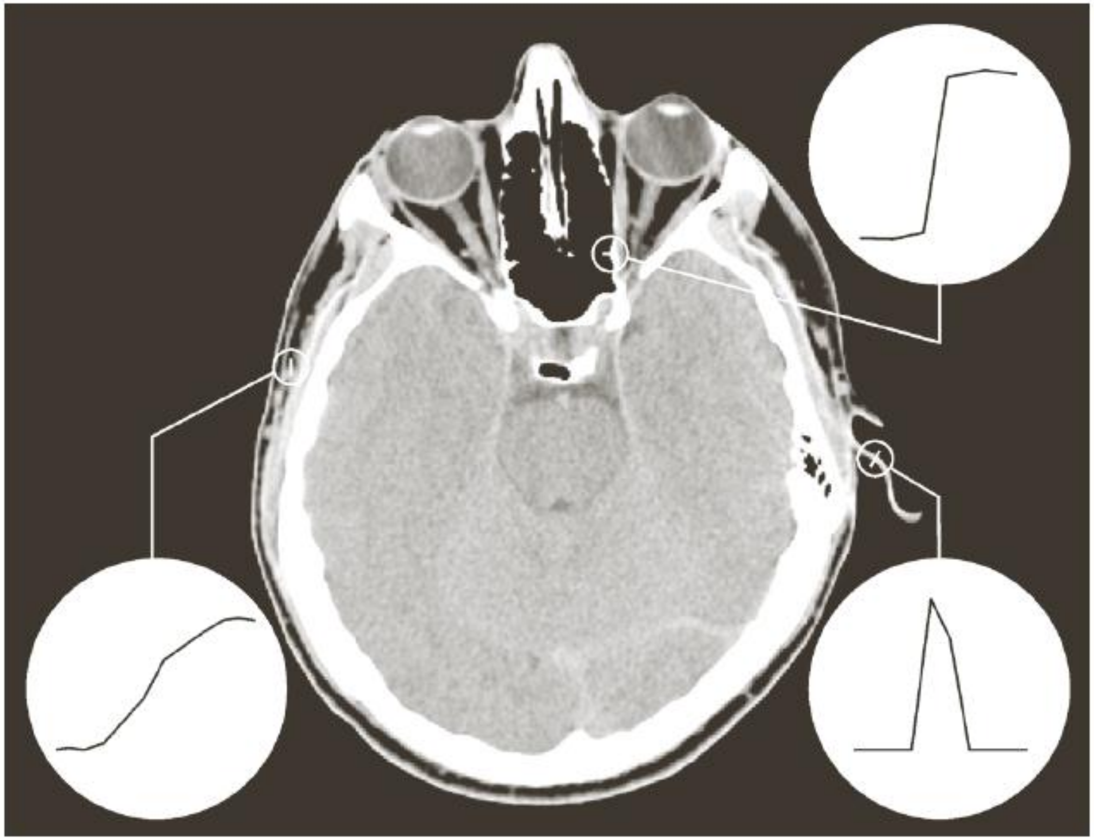
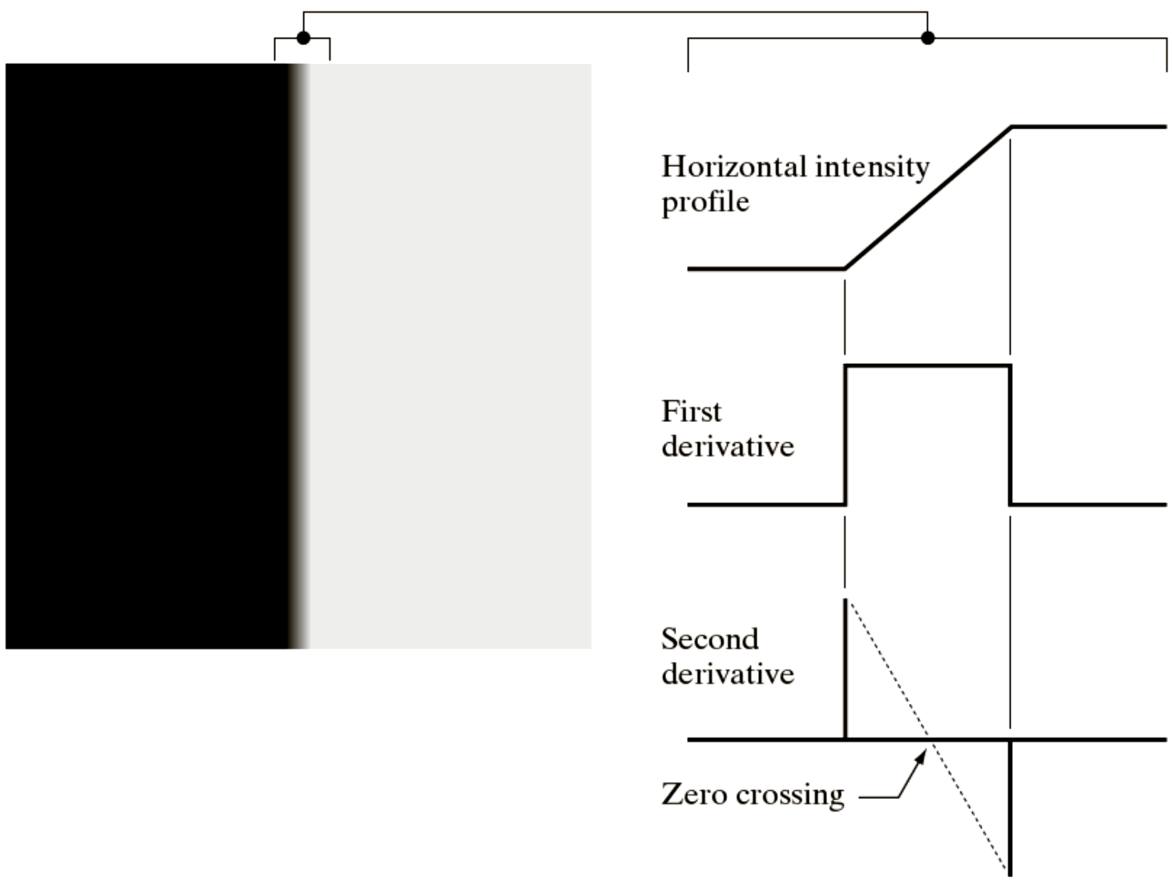


Image Segmentation

- Example:



$$\frac{\partial f}{\partial x}$$

$$\frac{\partial^2 f}{\partial x^2}$$

Image Segmentation

- Noise Problem:

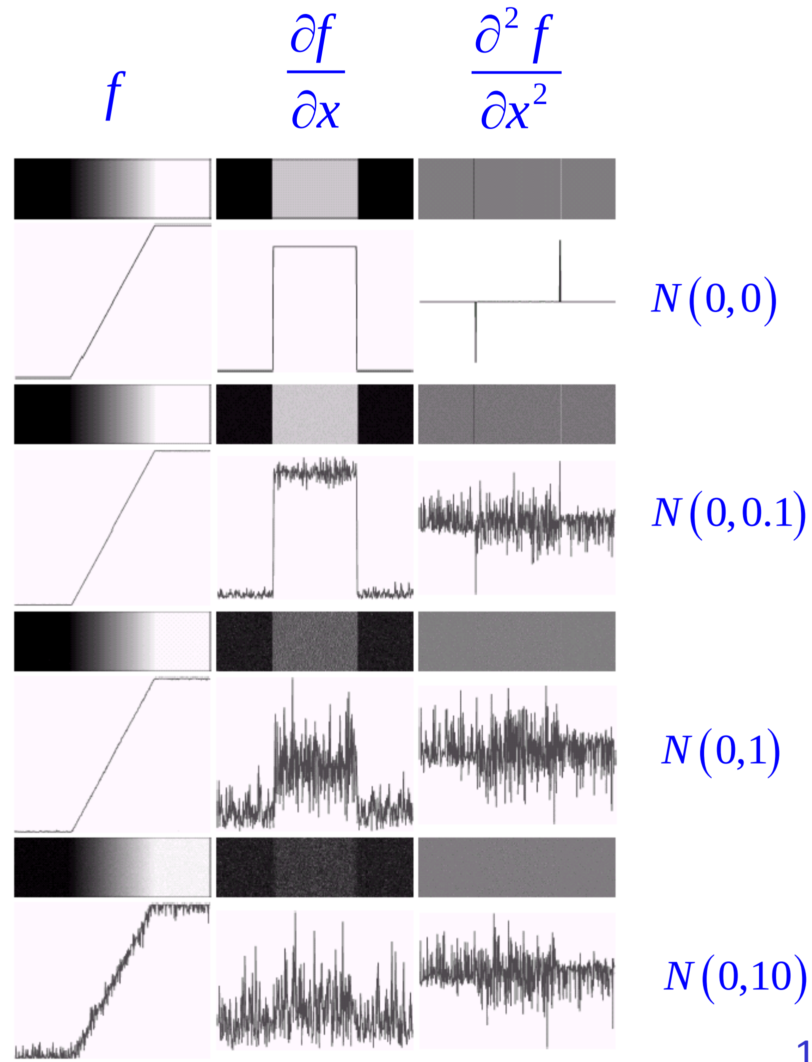


Image Segmentation

- Basic Edge Detection:
  - Gradient Operators:

$$\nabla f = \begin{bmatrix} g_x & g_y \end{bmatrix}^T = \begin{bmatrix} \dfrac{\partial f}{\partial x} & \dfrac{\partial f}{\partial y} \end{bmatrix}^T$$

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

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

Image Segmentation

- Graphical Illustration:

$$\nabla f = \begin{bmatrix} -2 \\ 2 \end{bmatrix}$$

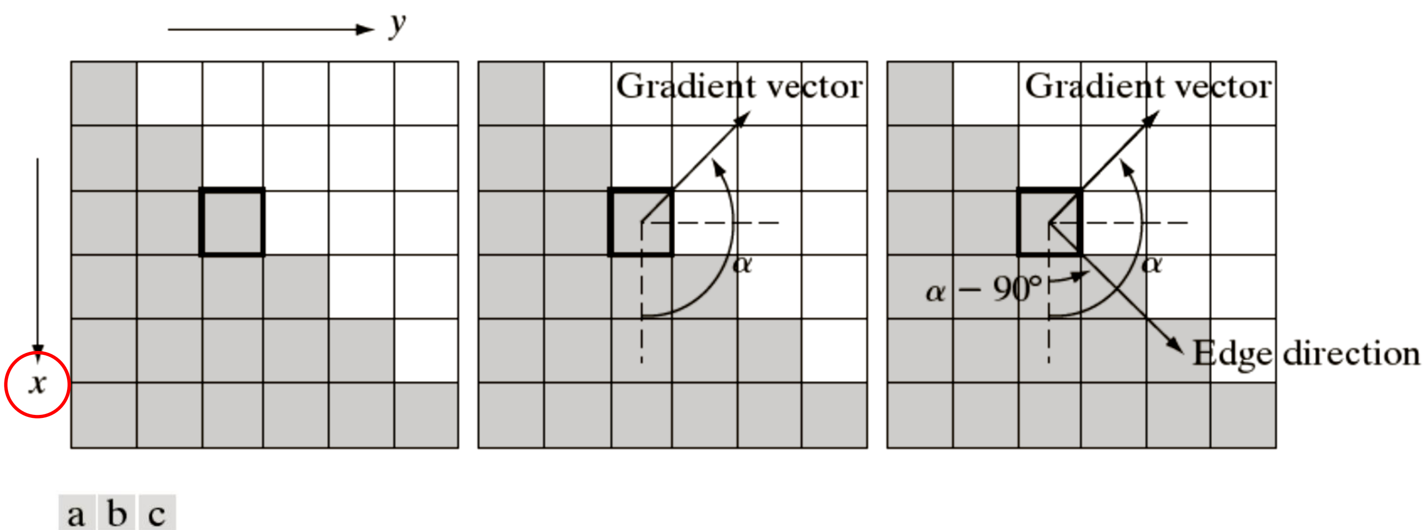


FIGURE 10.12 Using the gradient to determine edge strength and direction at a point. Note that the edge is perpendicular to the direction of the gradient vector at the point where the gradient is computed. Each square in the figure represents one pixel.

Image Segmentation

- Gradient Operators:
  - Roberts
  - Prewitt
  - Sobel

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

$$\begin{bmatrix} z_9 - z_5 \\ z_8 - z_6 \end{bmatrix}$$

−1	0
0	1

0	−1
1	0

Roberts

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

−1	−1	−1
0	0	0
1	1	1

−1	0	1
−1	0	1
−1	0	1

Prewitt

$$\begin{bmatrix} (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3) \\ (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7) \end{bmatrix}$$

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

Sobel

Image Segmentation

- Diagonal Edges:

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

Sobel

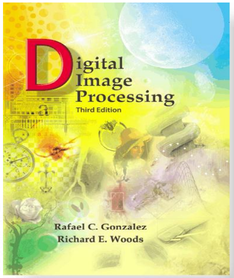


Image Segmentation

- Example:
  - Magnitude

Original

$\left|G_x\right|$

$\left|G_y\right|$

$\left|G_x\right| + \left|G_y\right|$

Image Segmentation

- Example:
  - Angle

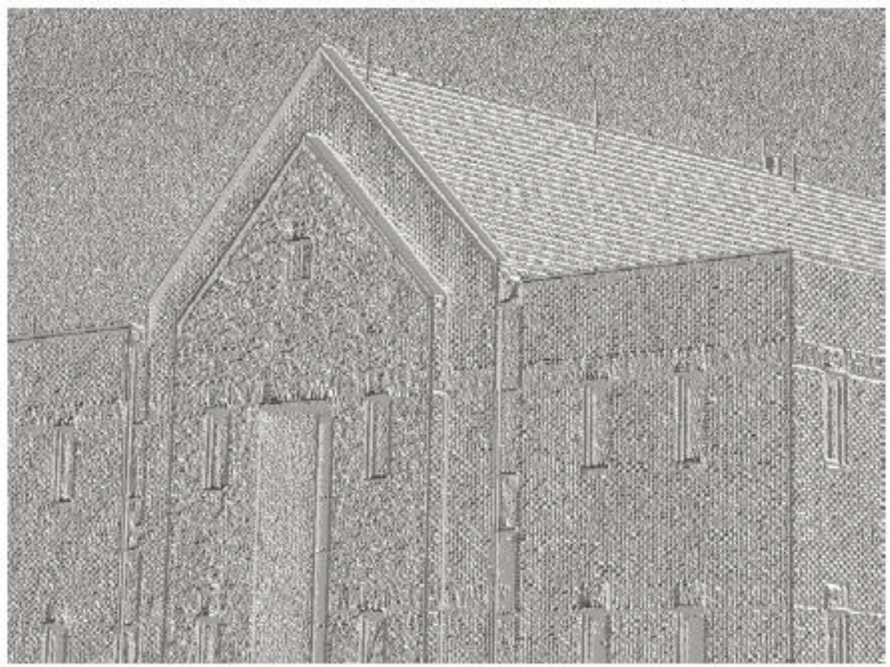
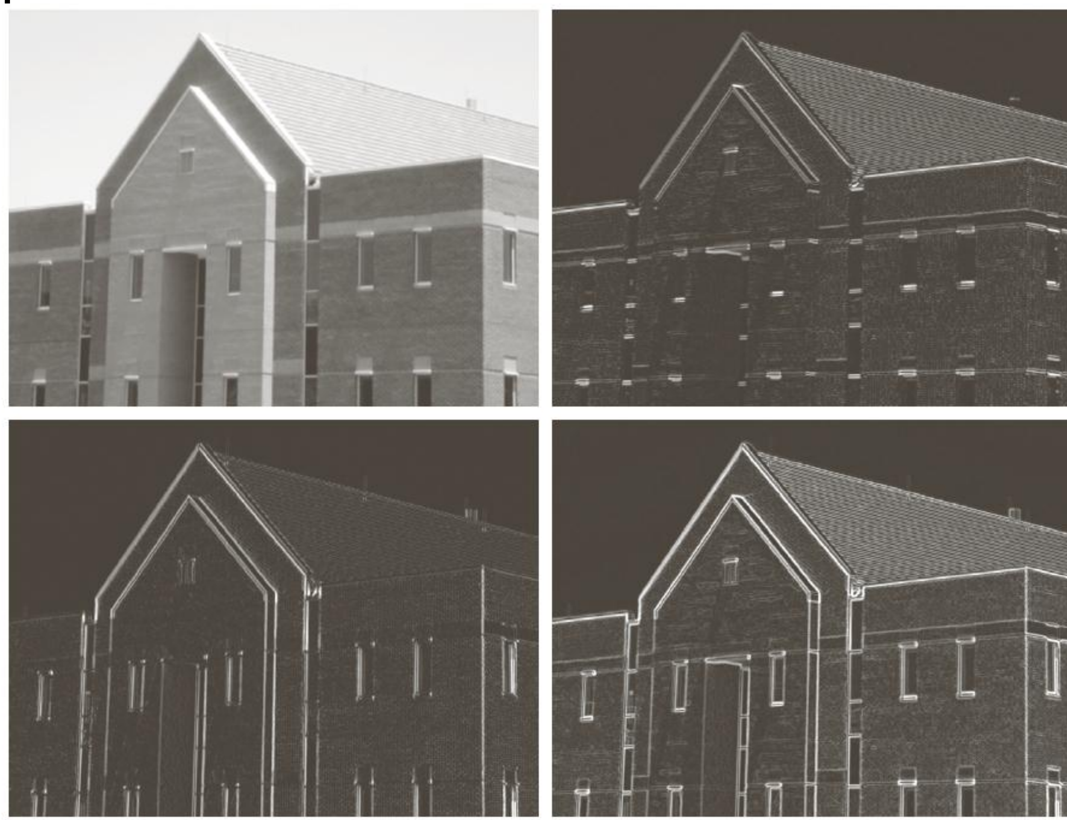


Image Segmentation

- Pre-Smoothing Effect:
  - 5×5 averaging filter

Original $\quad |G_x|$



$|G_y| \quad |G_x| + |G_y|$

Image Segmentation

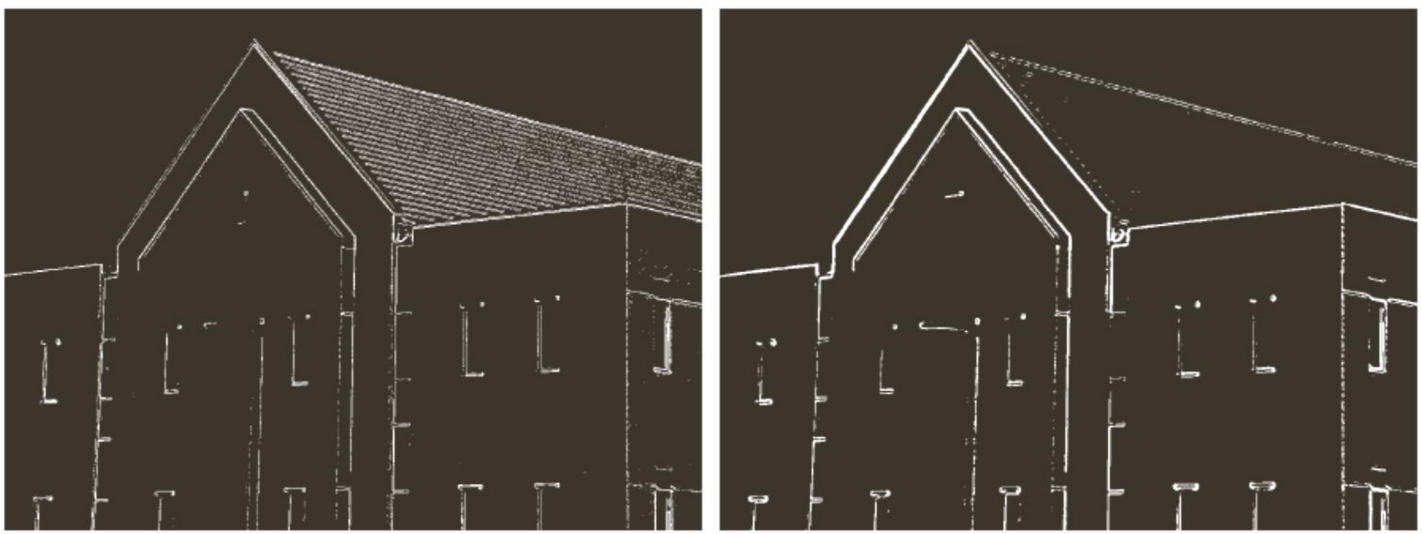
- Diagonal Edge:

$$\begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix} \qquad \begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

-45º and +45º lines

Image Segmentation

- Thresholding Effect:
  - Without smoothing (33%)
  - With smoothing (50%)



- The Marr-Hildreth Edge Detector:
  - LoG (Laplacian of Gaussian) Operator

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

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

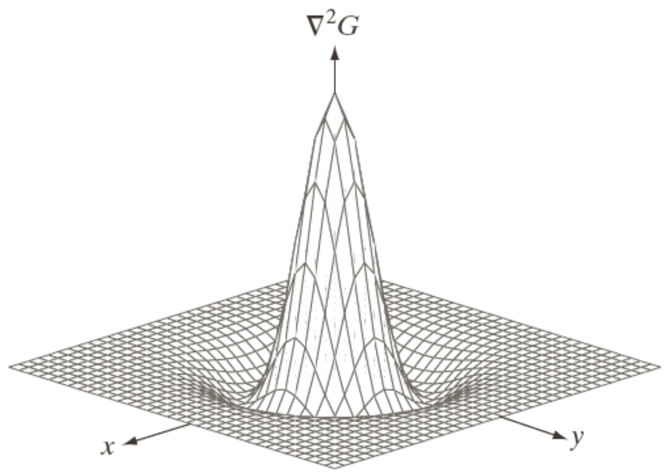
$$\nabla^2 G(x,y)=\left[\frac{x^2}{\sigma^4}-\frac{1}{\sigma^2}\right]G(x,y)+\left[\frac{y^2}{\sigma^4}-\frac{1}{\sigma^2}\right]G(x,y)$$

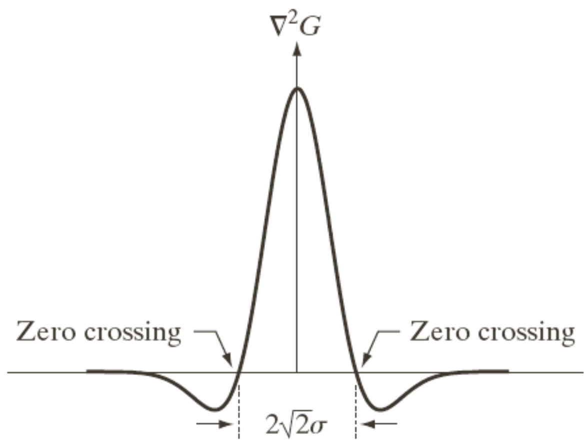
$$\nabla^2 G(x,y)=\left[\frac{x^2+y^2-2\sigma^2}{\sigma^4}\right]G(x,y)$$

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

Image Segmentation

- LoG Illustration:
  - Exact
  - Approximated





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

Image Segmentation

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

Smoothing with gaussian filter and then sharpening with laplacian of gaussian filter

Zero crossing is the line which is crossing main line of number line co-ordinate system. If a line passes through zero line, it is deviating from +ive to negative or vice versa, then its an edge.

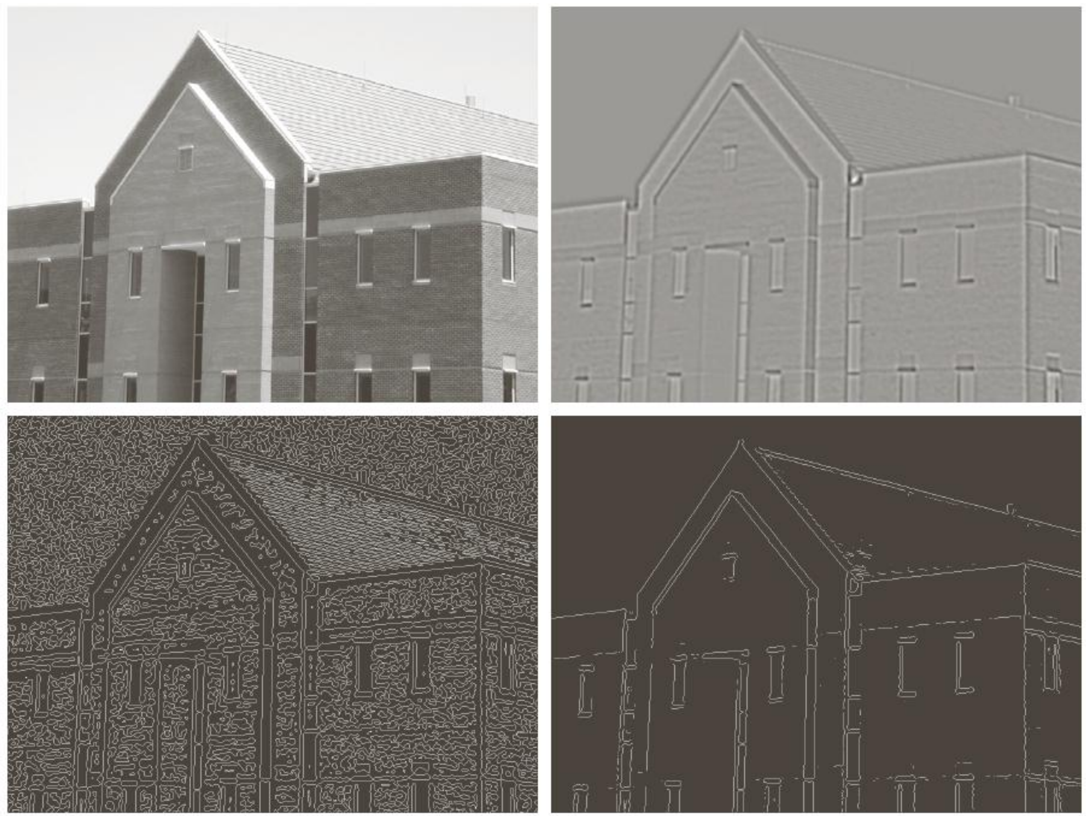
$$g(x,y)=\left[\nabla^2 G(x,y)\right]\star f(x,y)=\nabla^2\left[G(x,y)\star f(x,y)\right]$$

  - Algorithm:
    - Compute Laplacian of Gaussian filtered
    - Find zero-crossing
      - Use a 3×3 window around each pixel
      - ZC occurs if *at least* two opposing neighbors has different *signs*.

Image Segmentation

- Example:

Original **LoG**



a b
c d

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.

ZC using 0 Thr **ZC using 4%*Max as Thr**

- LoG Approximation with DoG:
  - DoG: Difference of Gaussian:

$$\mathrm{DoG}(x,y)=\frac{1}{2\pi\sigma_1^2}\exp\left(-\frac{x^2+y^2}{2\sigma_1^2}\right)-\frac{1}{2\pi\sigma_2^2}\exp\left(-\frac{x^2+y^2}{2\sigma_2^2}\right),\quad \sigma_1^2>\sigma_2^2$$

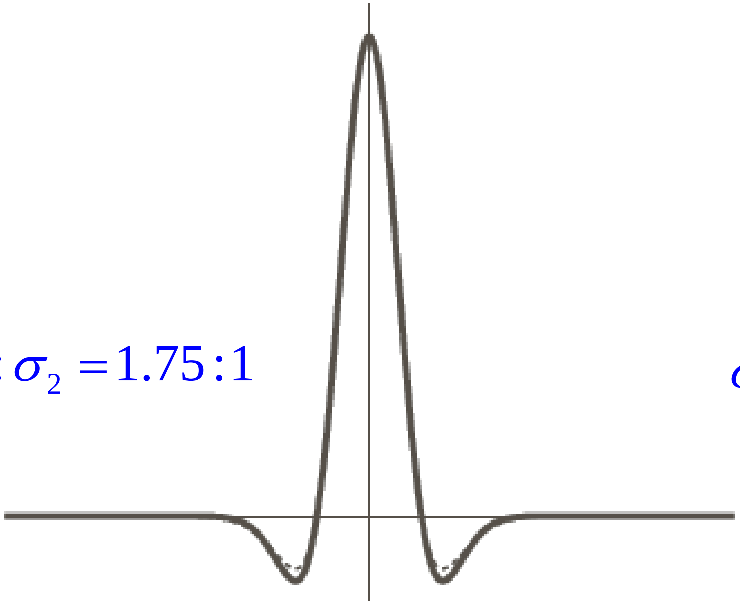
  - Ratio of 1.6:1, an engineering approximation of LoG.
  - For same zero-crossing:

$$\sigma^2=\frac{\sigma_1^2\sigma_2^2}{\sigma_1^2-\sigma_2^2}\ln\left[\frac{\sigma_1^2}{\sigma_2^2}\right]$$

Image Segmentation

- Approximation Accuracy:
  - Solid Line: Negative LoG
  - Dashed Line: Negative DoG

$\sigma_1 : \sigma_2 = 1.75 : 1$



$\sigma_1 : \sigma_2 = 1.6 : 1$

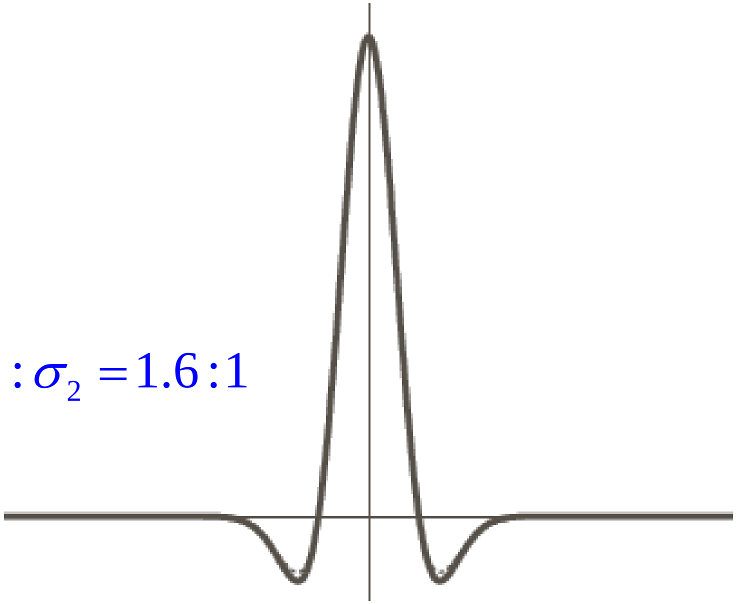


Image Segmentation

- Canny Edge Detector:
  - Low Error rate
  - Localized edge points
  - Single edge point response

Image Segmentation

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

- Canny – Smoothing:
  - Using Gaussian filter:

$$f_s(x,y) = G(x,y) \star f(x,y)$$

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

  - Practical Implementation:

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

- Canny – Gradient:
  - Using any gradient kernel:

$$g_x(x,y) = w_x(x,y) \star f_s(x,y)$$
$$g_y(x,y) = w_y(x,y) \star f_s(x,y)$$

  - Practical Implementation:
    - Sobel Kernel

$$w_x(x,y) = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}, \quad w_y(x,y) = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix}$$

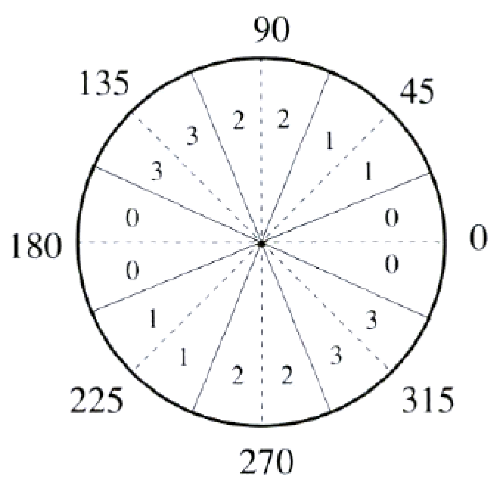
Image Segmentation

- Canny - Non-maximum Suppression (1):
  - Compute magnitude and angle of gradient:

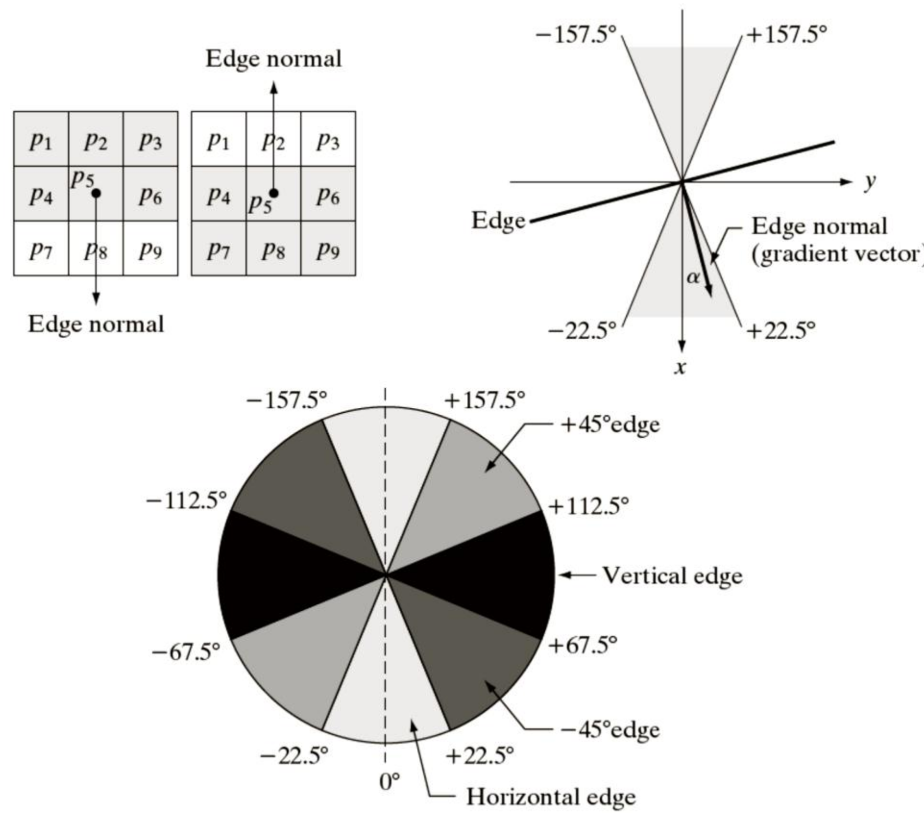
$$M(x,y)=\sqrt{g_x^2+g_y^2}\approx|g_x|+|g_y|$$

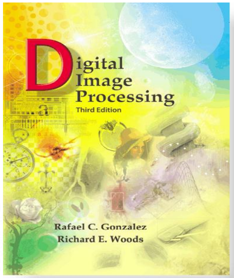
$$\theta(x,y)=\tan^{-1}\left(\frac{g_y}{g_x}\right)$$

  - Quantize the $\theta(x,y)$ to nearest 45°, $\theta_Q(x,y)$



- 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°, then the pixels (x+1, y), (x, y), and (x−1, y) are examined.
- If $\theta_Q(x,y)$ = 90°, then the pixels (x, y+1), (x, y), and (x, y−1) are examined.
- If $\theta_Q(x,y)$ = 45°, then the pixels (x+1, y+1), (x, y), and (x−1, y−1) are examined.
- If $\theta_Q(x,y)$ = 135°, then the pixels (x+1, y-1), (x, y), and (x−1, y+1) are examined.

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

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

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

  - Eliminate strong edge from g_{NH}

$$g_{NL}(x,y) = g_{NL}(x,y) - g_{NH}(x,y)$$

Image Segmentation

- 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 $g_{NL}(x,y)$

- Canny Example (1):
  - Original and Smoothed

(a) Original

(b) Smoothed

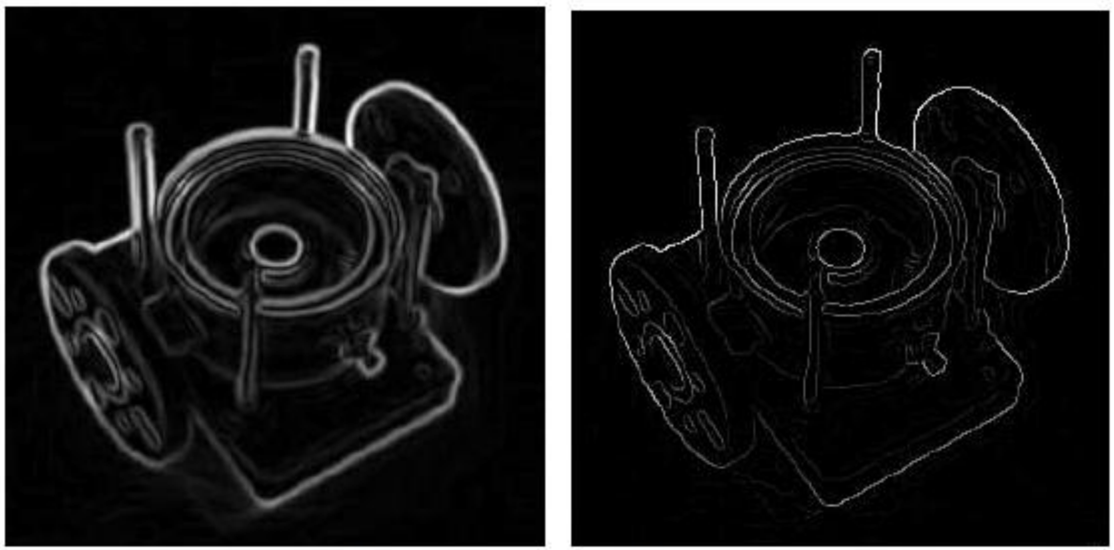
Image Segmentation

- Canny Example (1):
  - Gradient images:

(a) Smoothed

(b) Gradient magnitudes

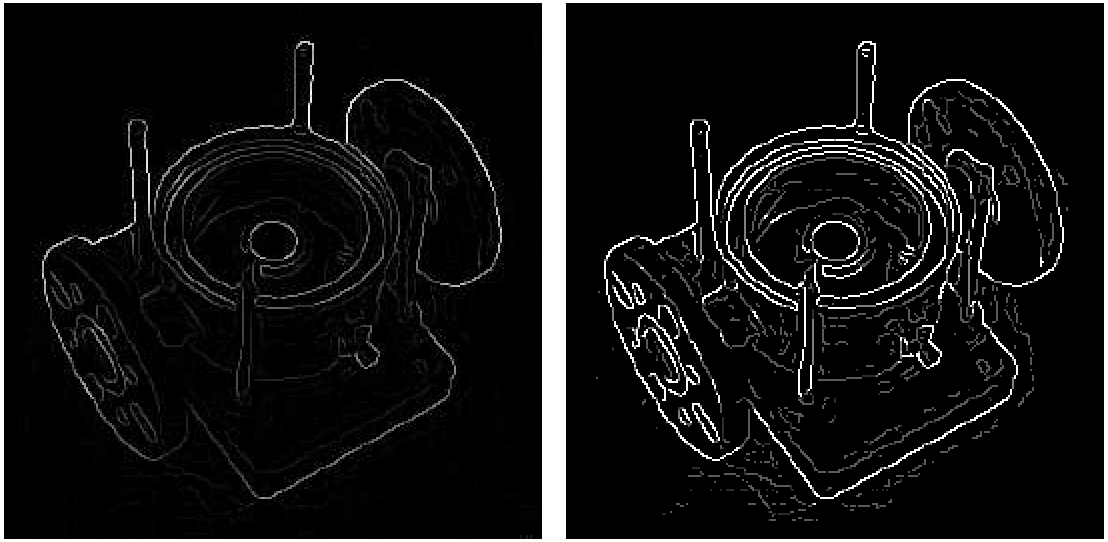
- Canny Example (1):
  - Non-maximum suppression:



(a) Gradient values (b) Edges after non-maximum suppression

Image Segmentation

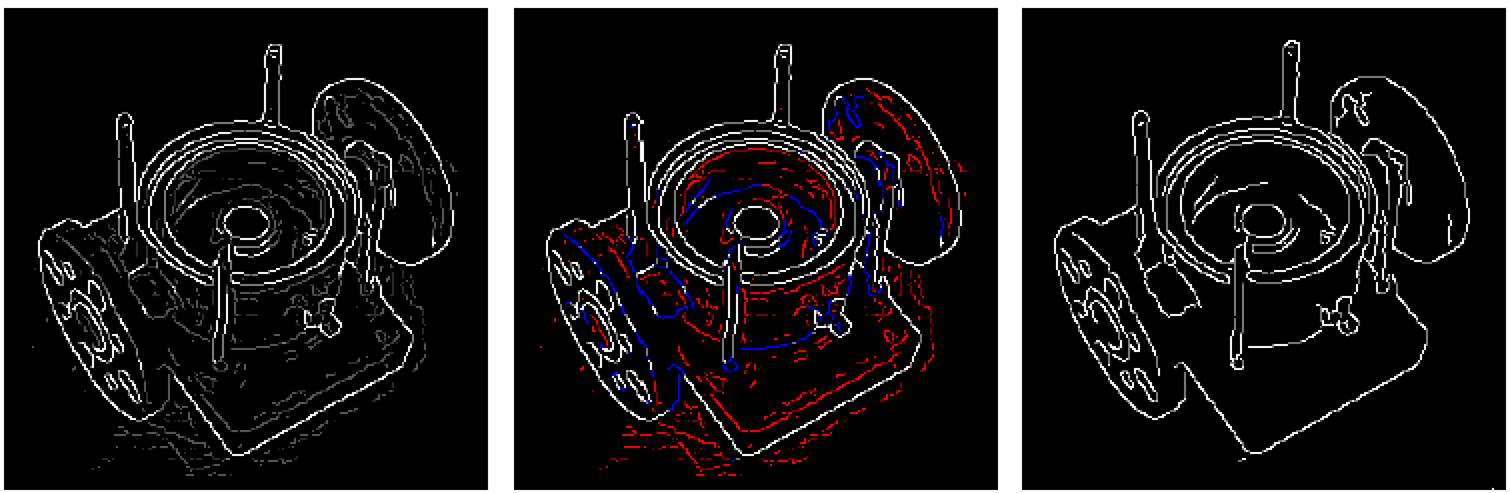
- Canny Example (1):
  - Double Thresholding:



(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

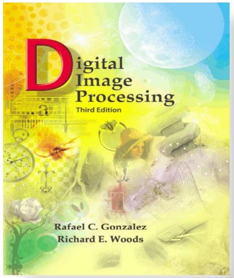


Image Segmentation

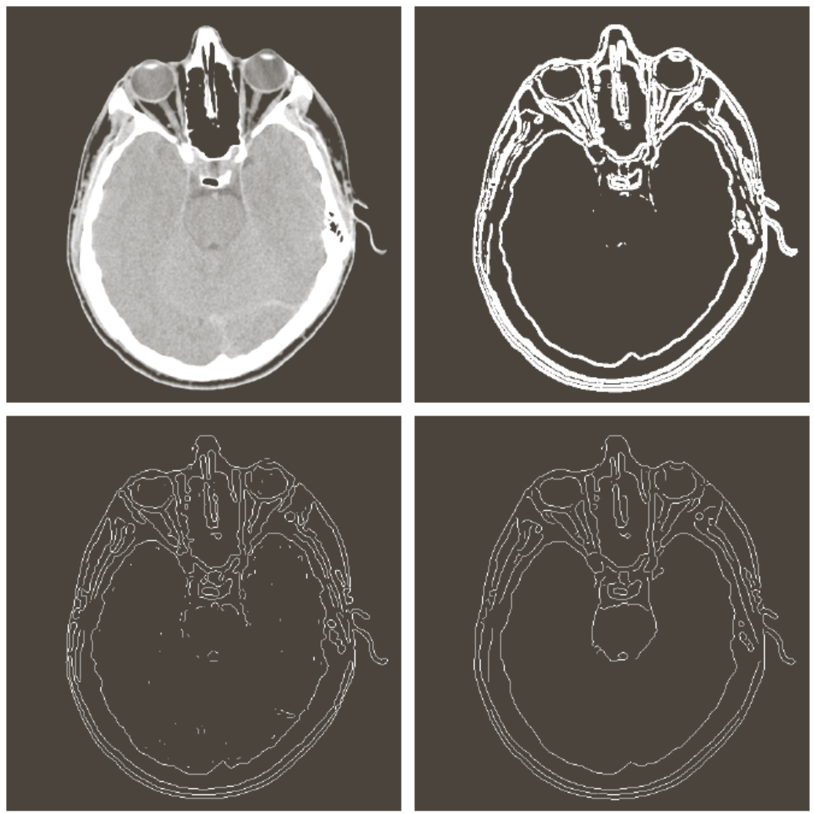
- Canny Example (2):

a b
c d

FIGURE 10.25 (a) Original image of size 834 $\times$ 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.

Image Segmentation

- Canny Example (3):



a b
c d

FIGURE 10.26
(a) Original head CT image of size 512 × 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.)

Image Segmentation

- Edge Linking (Local Processing):
  - Similarity of two edge pixels at (x,y) and (s,t):

$$|M(s,t)-M(x,y)|\le E \quad \text{and} \quad |\alpha(s,t)-\alpha(x,y)|\le A$$

  - Connect if both condition satisfied
- Computational Expensive.

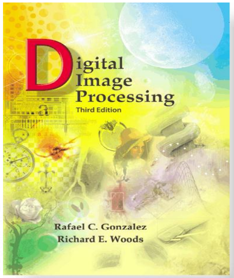


Image Segmentation

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

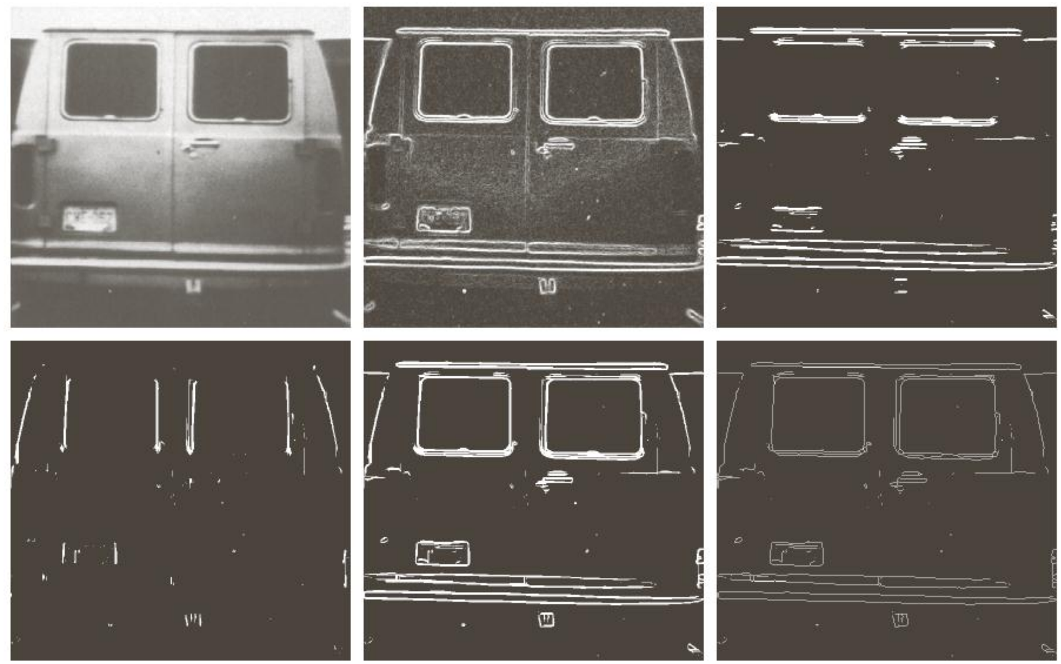
$$g(x,y)=\begin{cases}1 & M(x,y)>T_M \quad \textit{and} \quad \alpha(x,y)=A\pm T_A \\ 0 & \text{otherwise}\end{cases}$$

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, θ, by rotation the g by θ, and apply the horizontal scanning scheme. Rotate the result back to $-\theta$

Image Segmentation

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



a b c
d e f

FIGURE 10.27 (a) A 534 × 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.

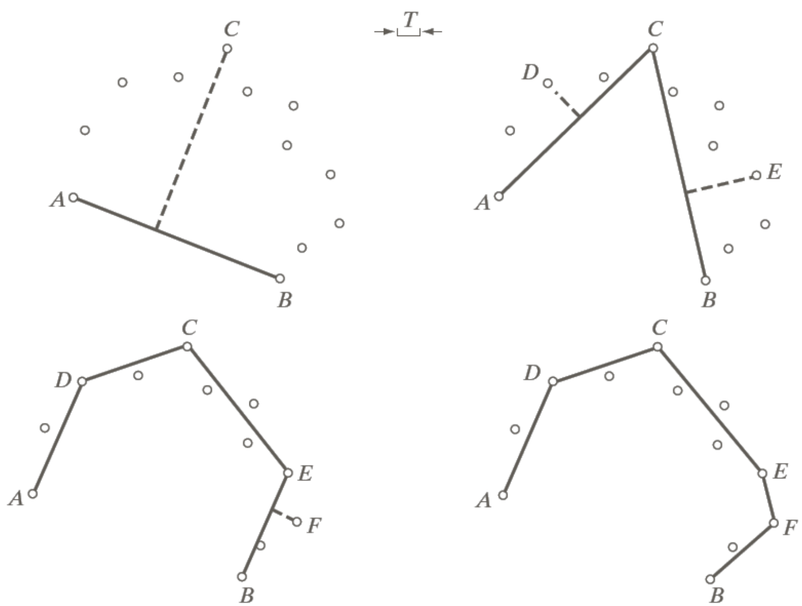
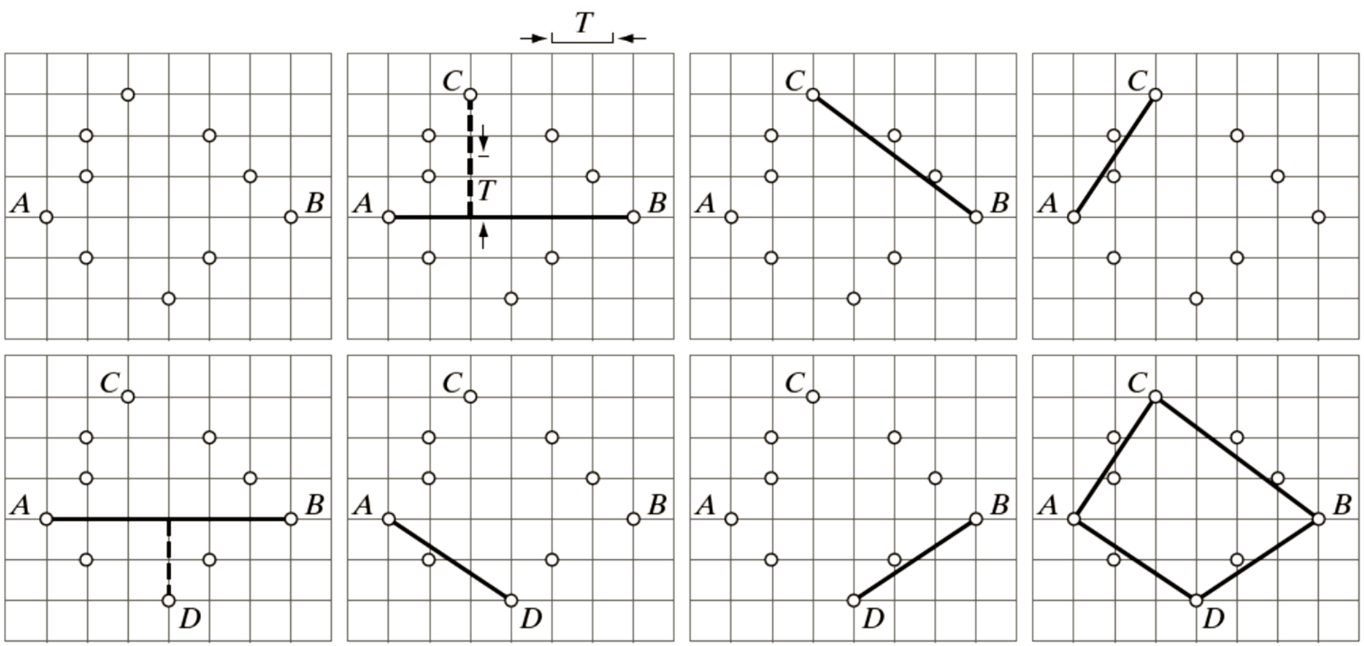


Image Segmentation

- Edge Linking Using Polygonal Fitting:
  - Read more: Pages: 728-732.



- Global Edge Linking by the Hough Transform:

$$\left(x_i, y_i\right) \ \& \ y = ax + b \Rightarrow y_i = ax_i + b$$

$$b = -x_i a + y_i$$

All $\left(x_i, y_i\right)$'s on a line intersect each other at $(\mathrm{a,b})$

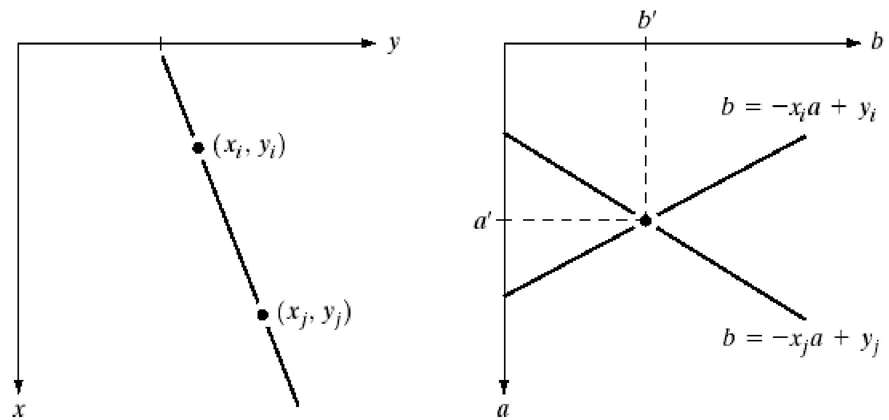
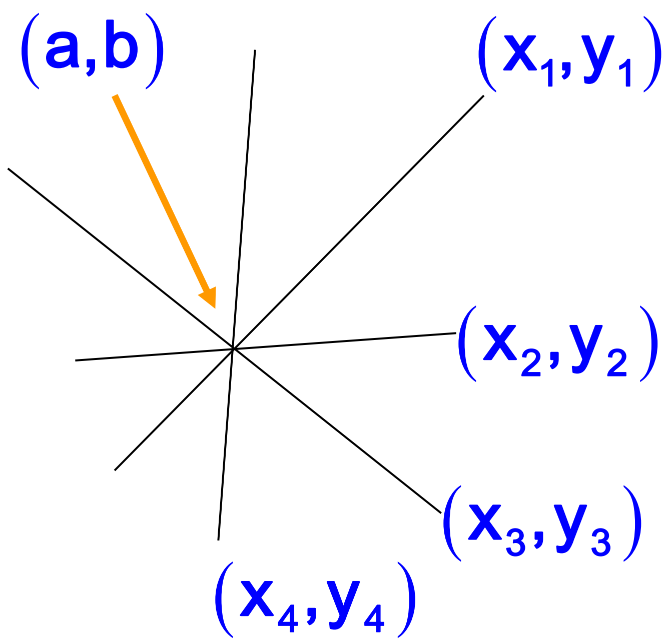
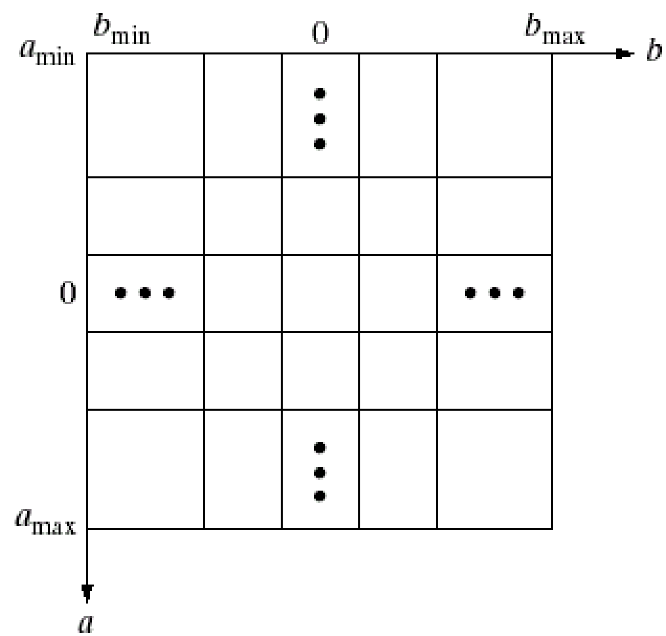


Image Segmentation

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





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

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

$$\text{All } (x_i, y_i)\text{'s on a sin intersect each other at } (\rho, \theta)$$

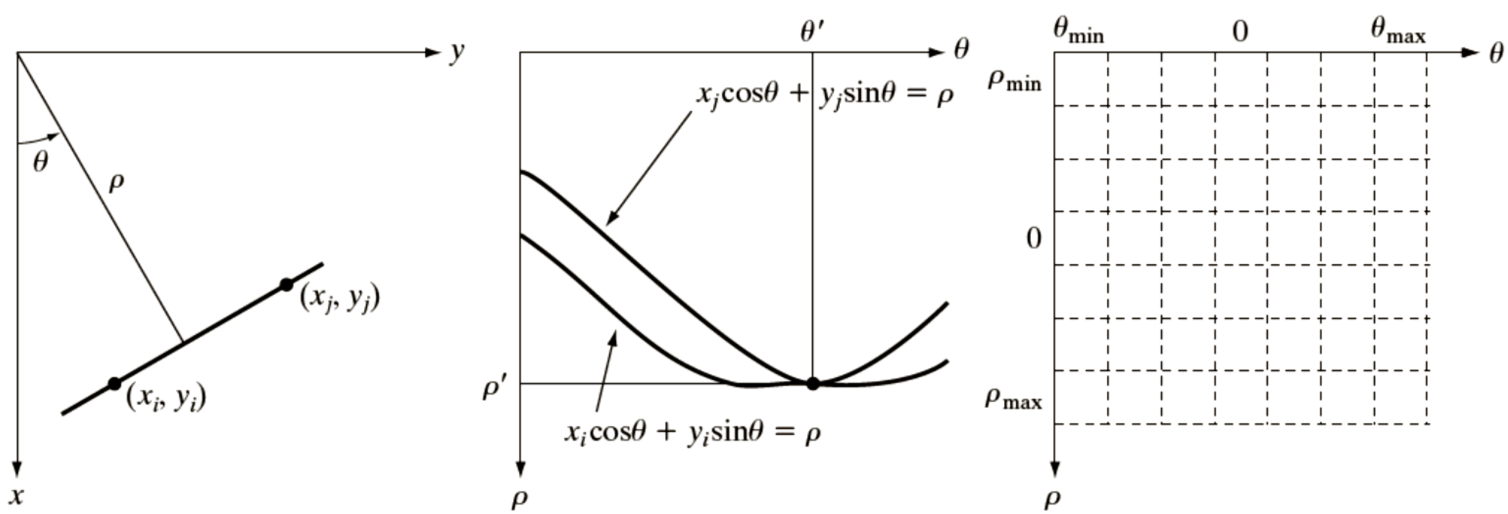
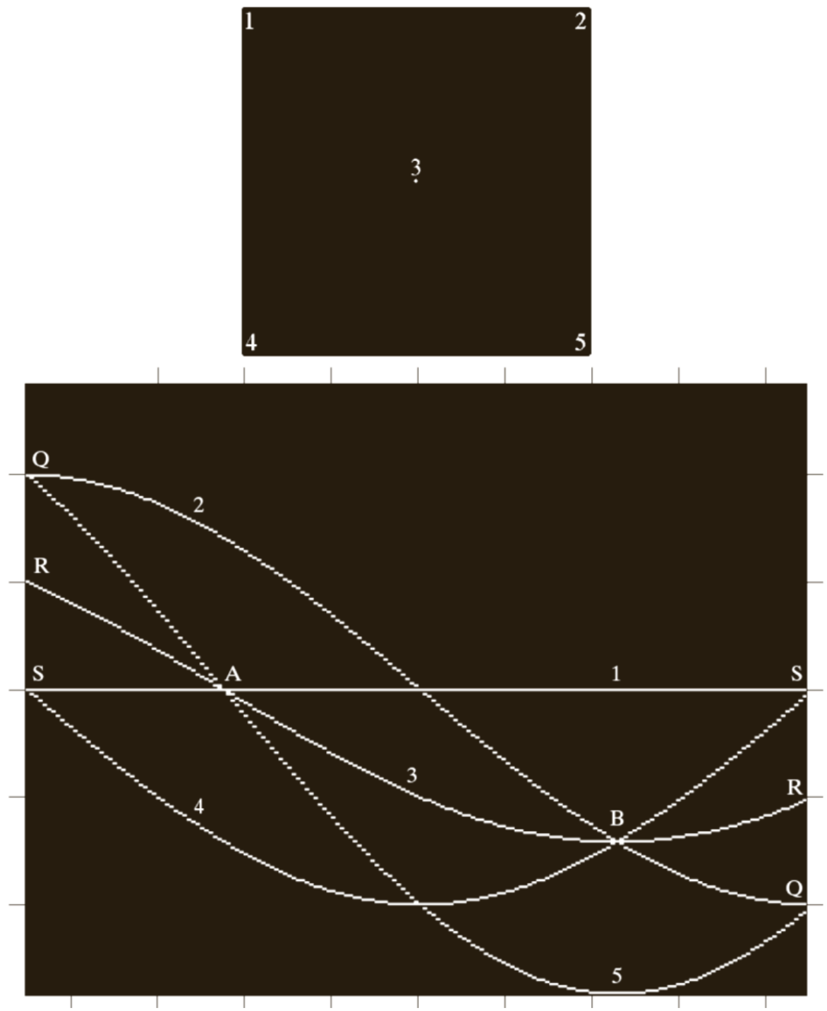


Image Segmentation

- Example:



a
b

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

Image Segmentation

- Example:

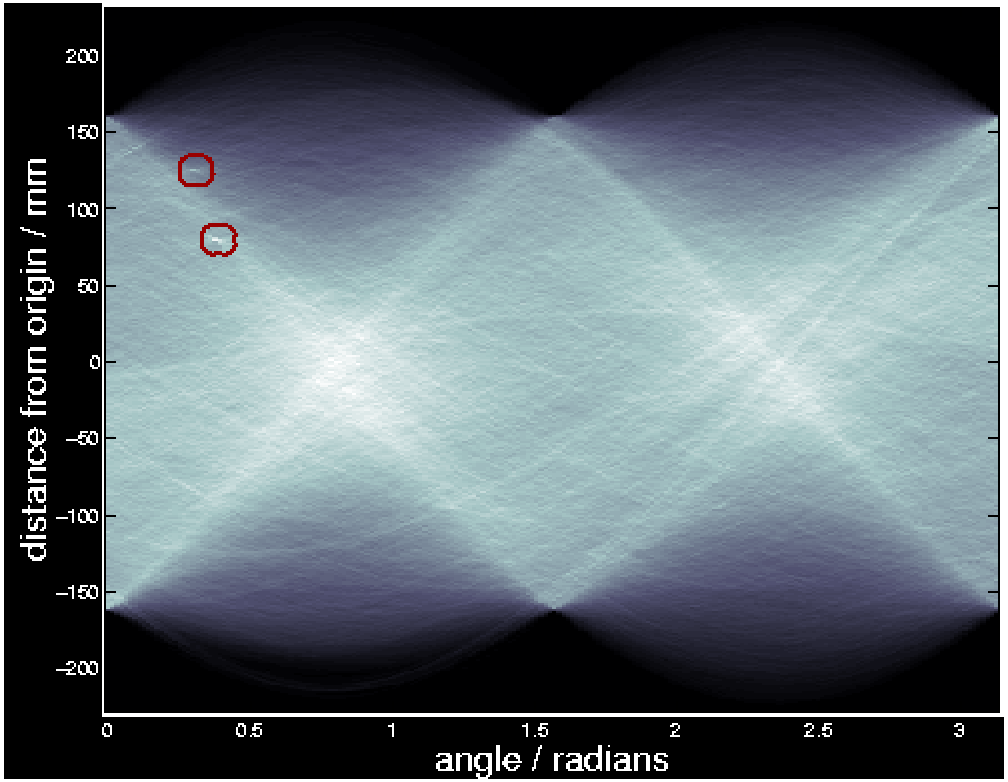


Image Segmentation

- Example:

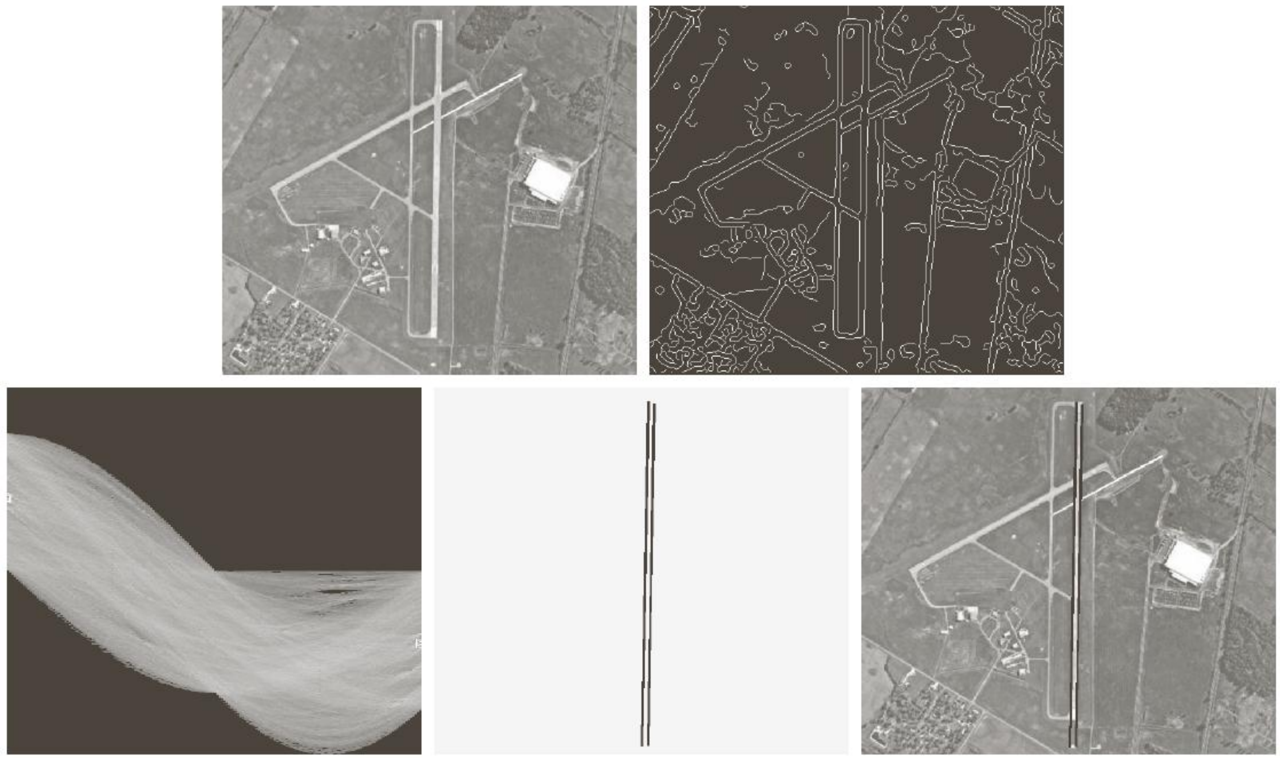


FIGURE 10.34 (a) A 502 × 564 aerial image of an airport. (b) Edge image obtained using Canny's algorithm. (c) Hough parameter space (the boxes highlight the points associated with long vertical lines). (d) Lines in the image plane corresponding to the points highlighted by the boxes. (e) Lines superimposed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

Image Segmentation

- Circle Hough Transform (CHT):

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

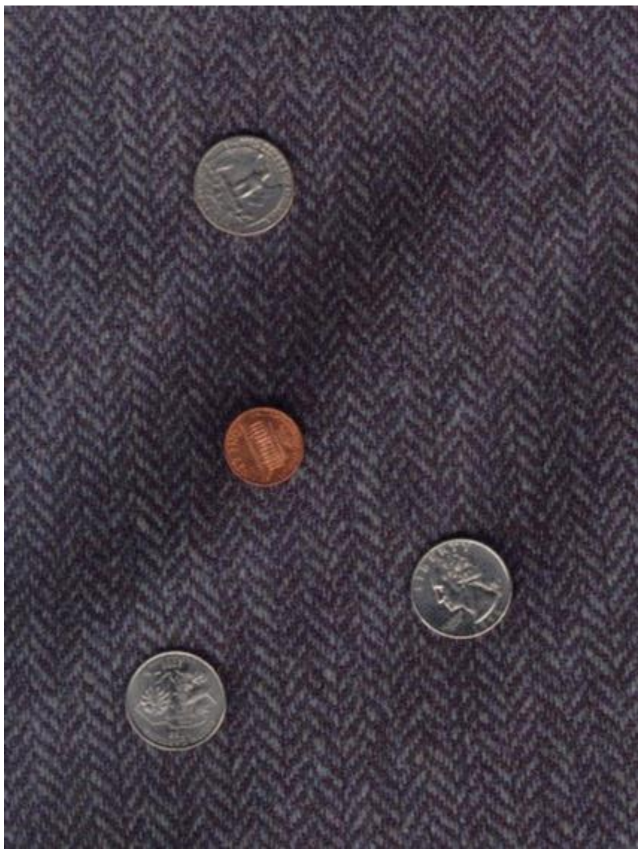
All (x_i, y_i)'s on a spherical surface intersect each other at (p_1, p_2, p_3)

  - 3D Parameter Space

- Extract each circle (independent of radius):

$$\left.\begin{array}{l} x_i=c_1+c_3\cos\theta \\ y_i=c_2+c_3\sin\theta \end{array}\right\} \Rightarrow \left.\begin{array}{l} c_1=x_i-c_3\cos\theta \\ c_2=y_i-c_3\sin\theta \end{array}\right\} \Rightarrow \text{Fill } (c_1, c_2) \text{ space at fixed } c_3$$

- Hough Transform for circle:



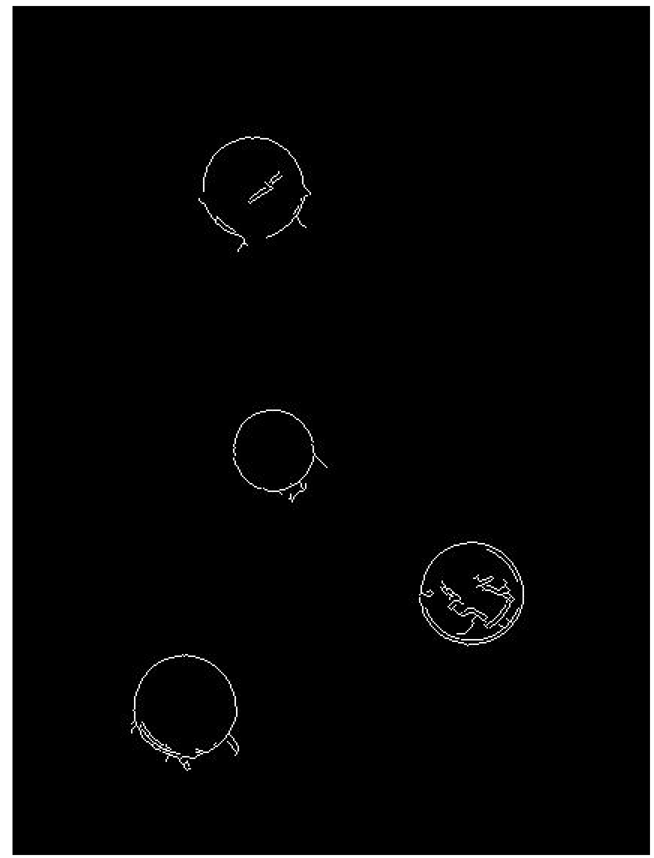


Image Segmentation

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

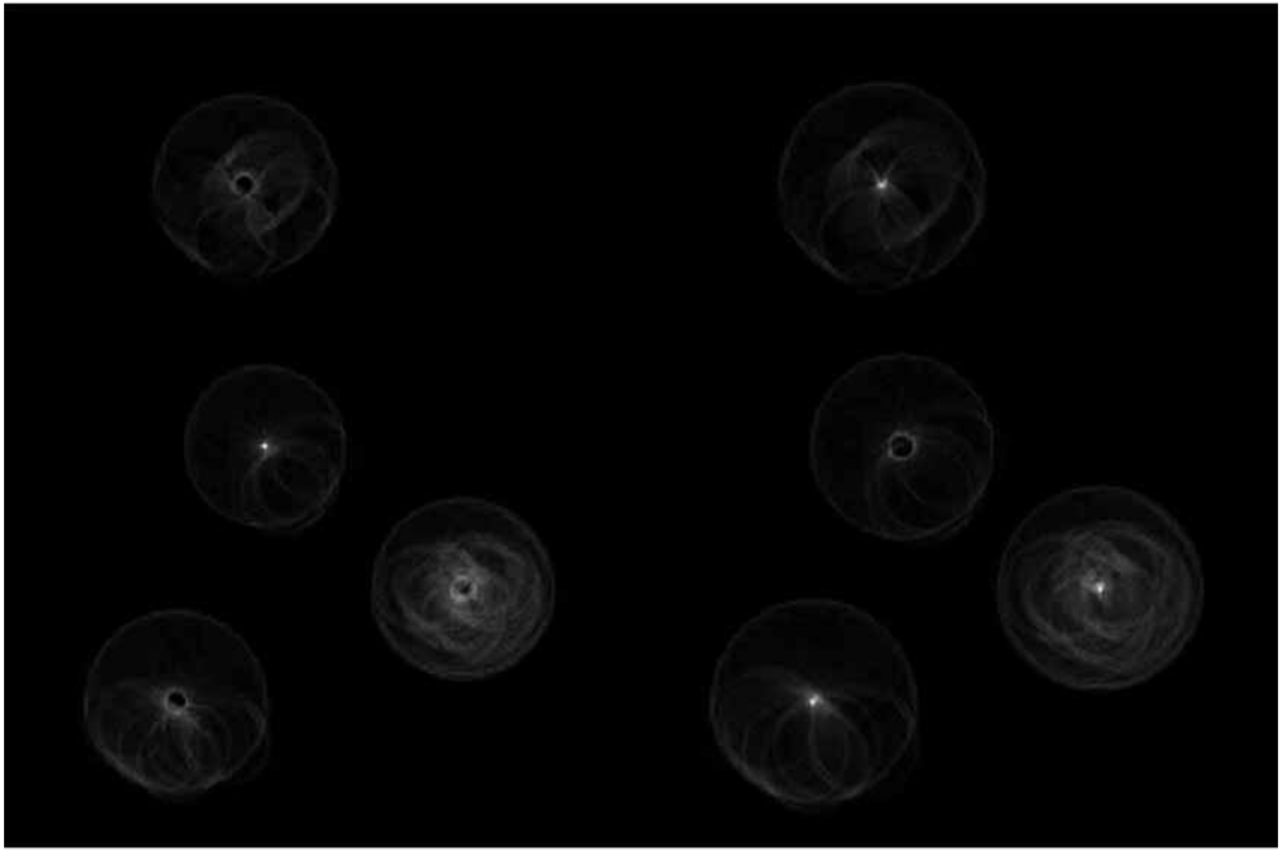


Image Segmentation

- Thresholding:
  - $f(x,y) > T$ then (x,y) is belong to the *object*, else (x,y) is belong to the *background*.
    - Bi-level (T):

$$g(x,y)=\begin{cases}1 & f(x,y)>T\\ 0 & f(x,y)\le T\end{cases}$$

    - Multi-level ($T_1, T_2, \ldots, T_n$)

$$g(x,y)=\begin{cases}a & f(x,y)>T_2\\ b & T_1<f(x,y)\le T_2\\ c & f(x,y)\le T_1\end{cases}$$

  - Challenge:
    - Threshold Selection
      - Histogram

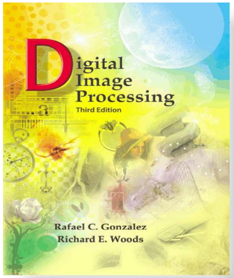


Image Segmentation

- Bi-Modal and Multi-Modal Histogram:

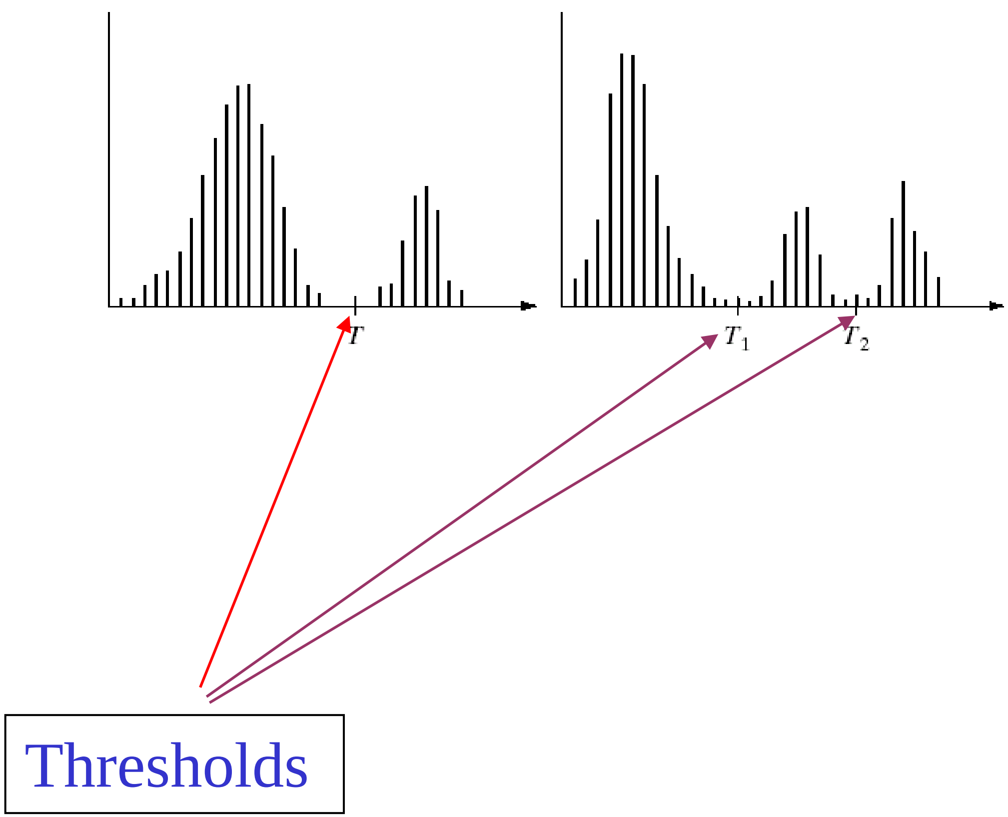


Image Segmentation

- Noise Effect on image Thresholding:

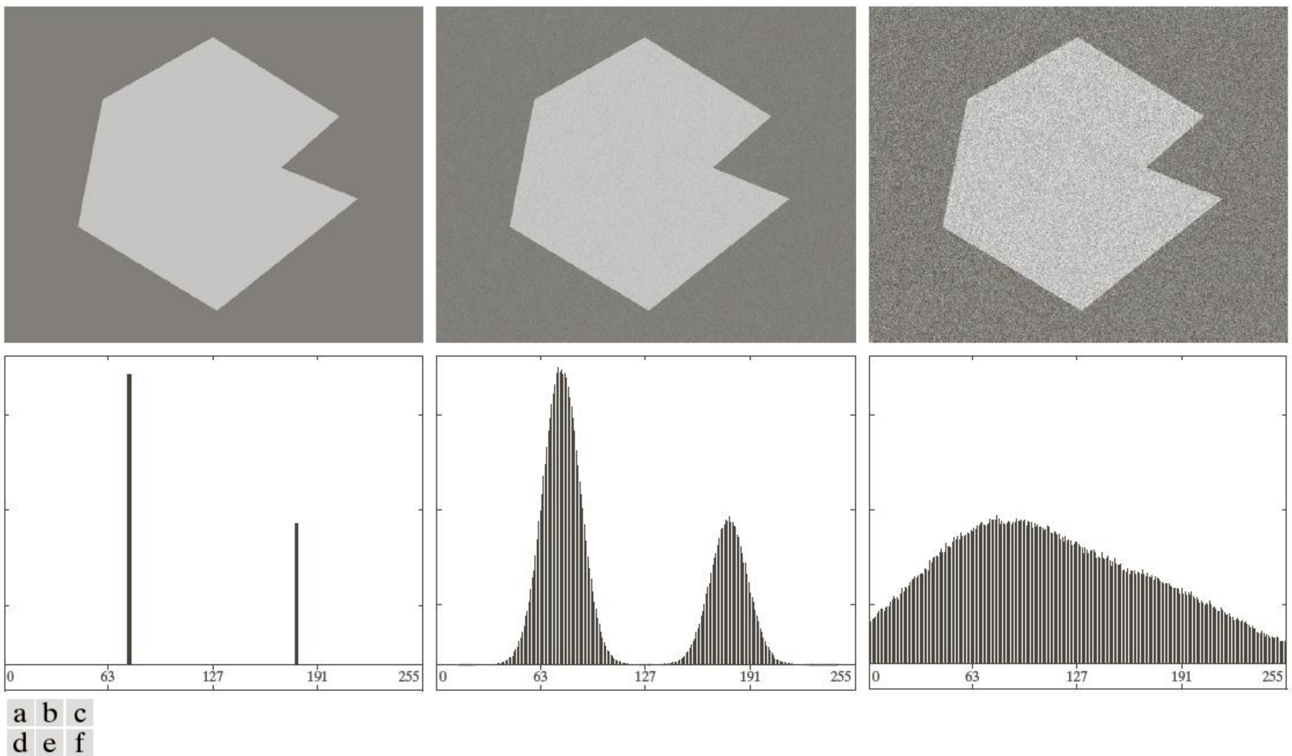


FIGURE 10.36 (a) Noiseless 8-bit image. (b) Image with additive Gaussian noise of mean 0 and standard deviation of 10 intensity levels. (c) Image with additive Gaussian noise of mean 0 and standard deviation of 50 intensity levels. (d)–(f) Corresponding histograms.

Image Segmentation

- Illumination and Reflectance Effect on image Thresholding:

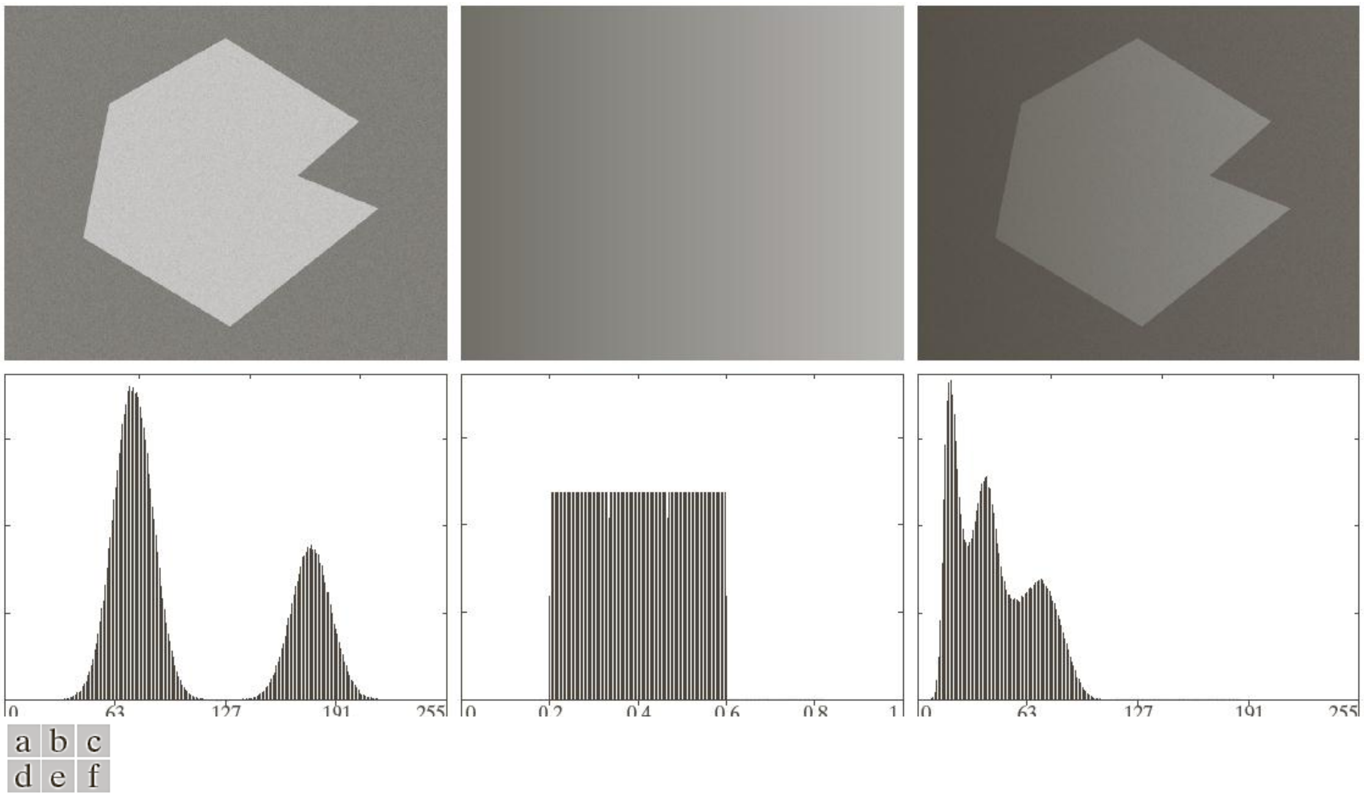


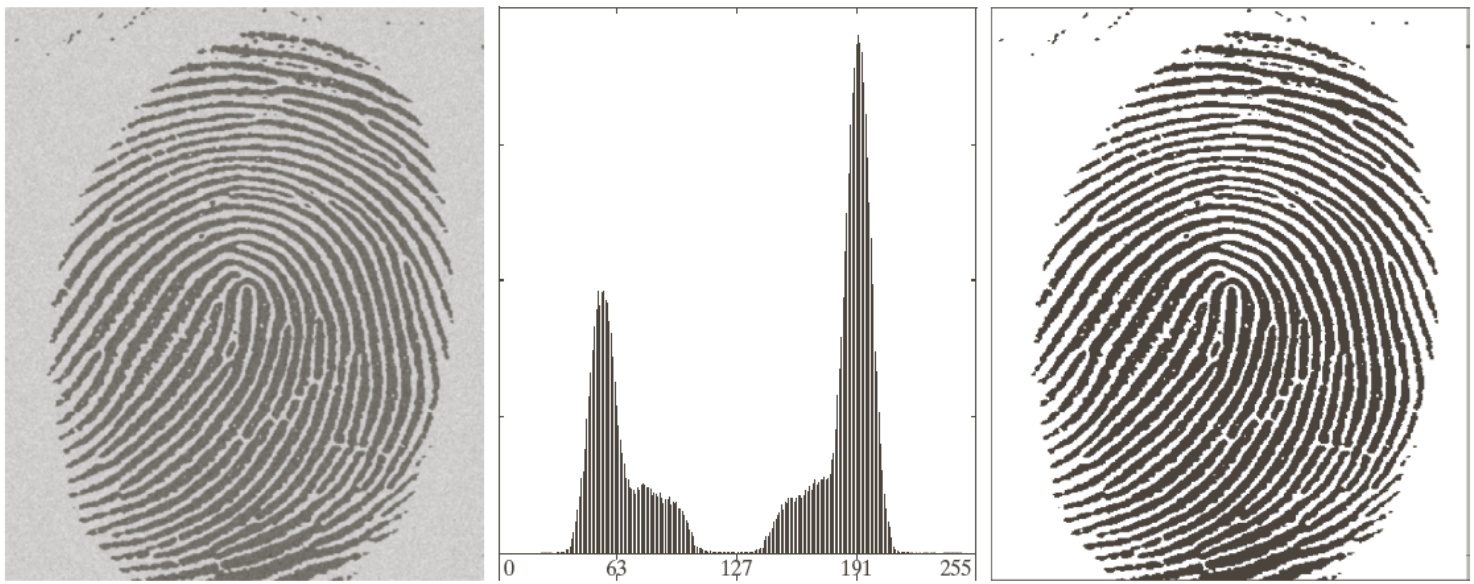
FIGURE 10.37 (a) Noisy image. (b) Intensity ramp in the range [0.2, 0.6]. (c) Product of (a) and (b). (d)–(f) Corresponding histograms.

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.(ΔT)

Image Segmentation

- Example:



$$T_0 = 0 \xrightarrow[\Delta T = 0]{\text{Convergence}} T_f = 125.4$$

Image Segmentation

- 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/

- Otsu Optimal Thresholding:
  - A $M \times N$ gray level image.
  - L distinct intensity level: $\{0, 1, 2, \ldots, L\text{-}1\}$
  - n_i = # of pixels with gray level i, $MN = n_0 + n_1 + \ldots + n_{L\text{-}1}$
  - 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:

$$C_1 = \{\text{pixels} \mid \text{intensity} \in [0, k]\}$$

$$C_2 = \{\text{pixels} \mid \text{intensity} \in k[\text{k}+1, L\text{-}1]\}$$

  - 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:

$$\text{Class mean: } m_1(k)=\sum_{i=0}^{k} iP(i/C_1)=\sum_{i=0}^{k} i\frac{P(C_1/i)P(i)}{P(C_1)}=\frac{1}{P_1(k)}\sum_{i=0}^{k} ip_i$$

$$\text{Class mean: } m_2(k)=\sum_{i=k+1}^{L-1} iP(i/C_2)=\sum_{i=k+1}^{L-1} i\frac{P(C_2/i)P(i)}{P(C_2)}=\frac{1}{P_2(k)}\sum_{i=k+1}^{L-1} ip_i$$

$$m(k)=\sum_{i=0}^{k} ip_i$$

$$m_G=P_1(k)m_1(k)+P_2(k)m_2(k)$$

- Otsu Optimal Thresholding:
  - Separability Index:

$$\eta(k) = \frac{\sigma_B^2(k)}{\sigma_G^2}$$

$$\sigma_B^2 = P_1(k)\left(m_1(k) - m_G\right)^2 + P_2(k)\left(m_2(k) - m_G\right)^2$$

$$= P_1(k)P_2(k)\left(m_1(k) - m_2(k)\right)^2$$

$$= \frac{\left(m_G P_1(k) - m(k)\right)^2}{P_1(k)\left(1 - P_1(k)\right)}$$

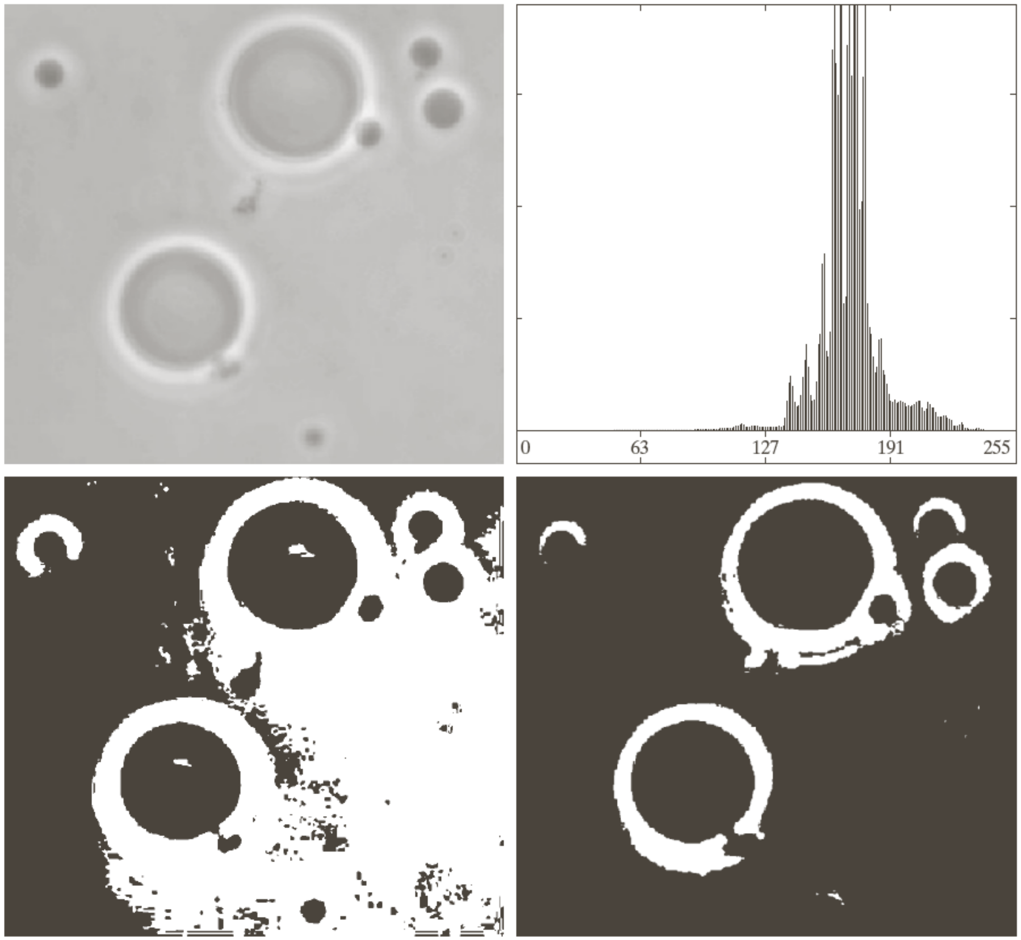
  - Optimal Solution:

$$k^* = \underset{0 \le k \le L-1}{\arg\max} \frac{\sigma_B^2(k)}{\sigma_G^2}$$

Measure of Seaparbility: $\eta\left(k^*\right)$

Image Segmentation

- Example:



a b
c d

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.)

- Example – Noise Effect:

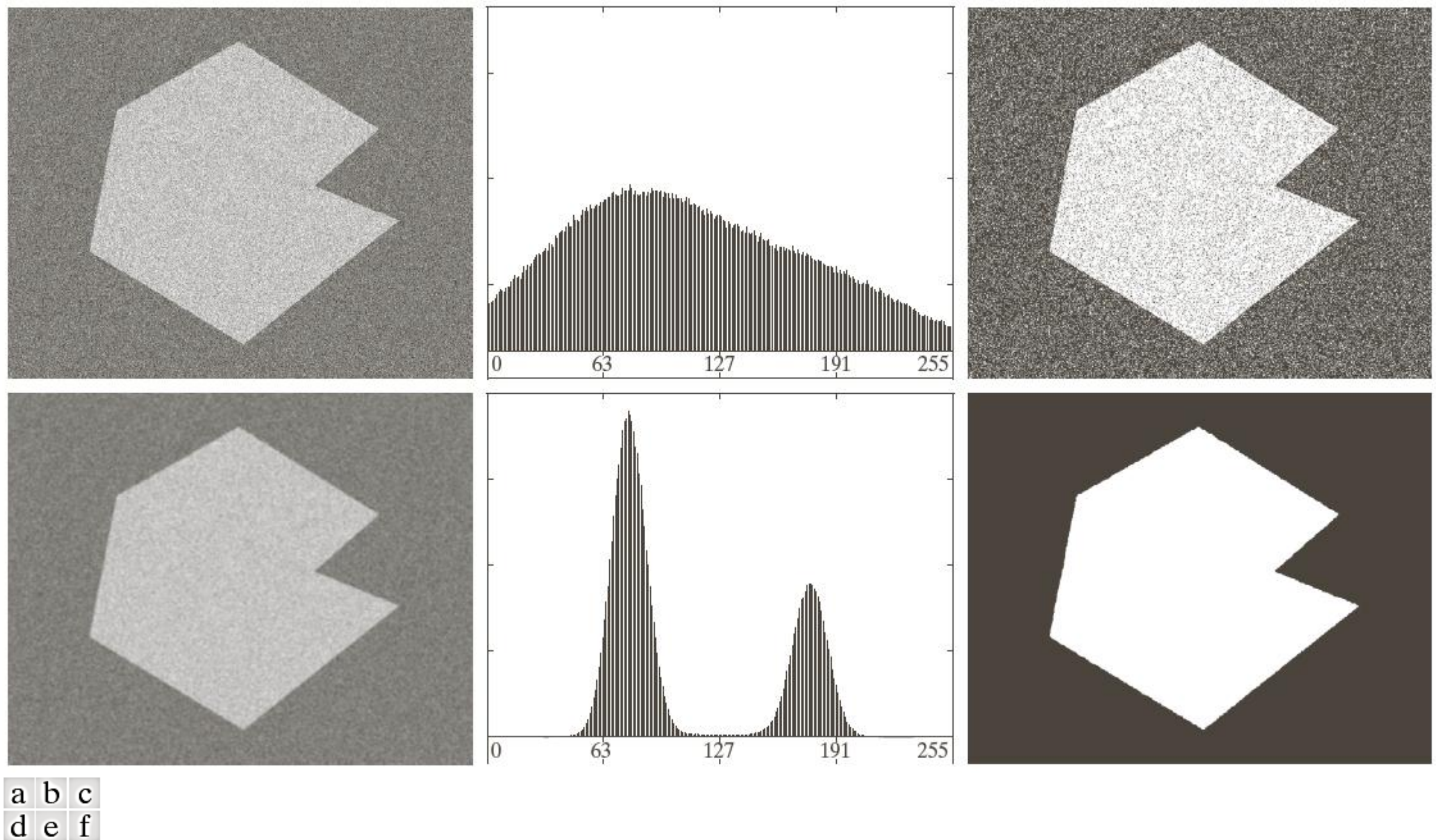


FIGURE 10.40 (a) Noisy image from Fig. 10.36 and (b) its histogram. (c) Result obtained using Otsu's method. (d) Noisy image smoothed using a 5 × 5 averaging mask and (e) its histogram. (f) Result of thresholding using Otsu's method.

Image Segmentation

- Example – Region Size Effect:

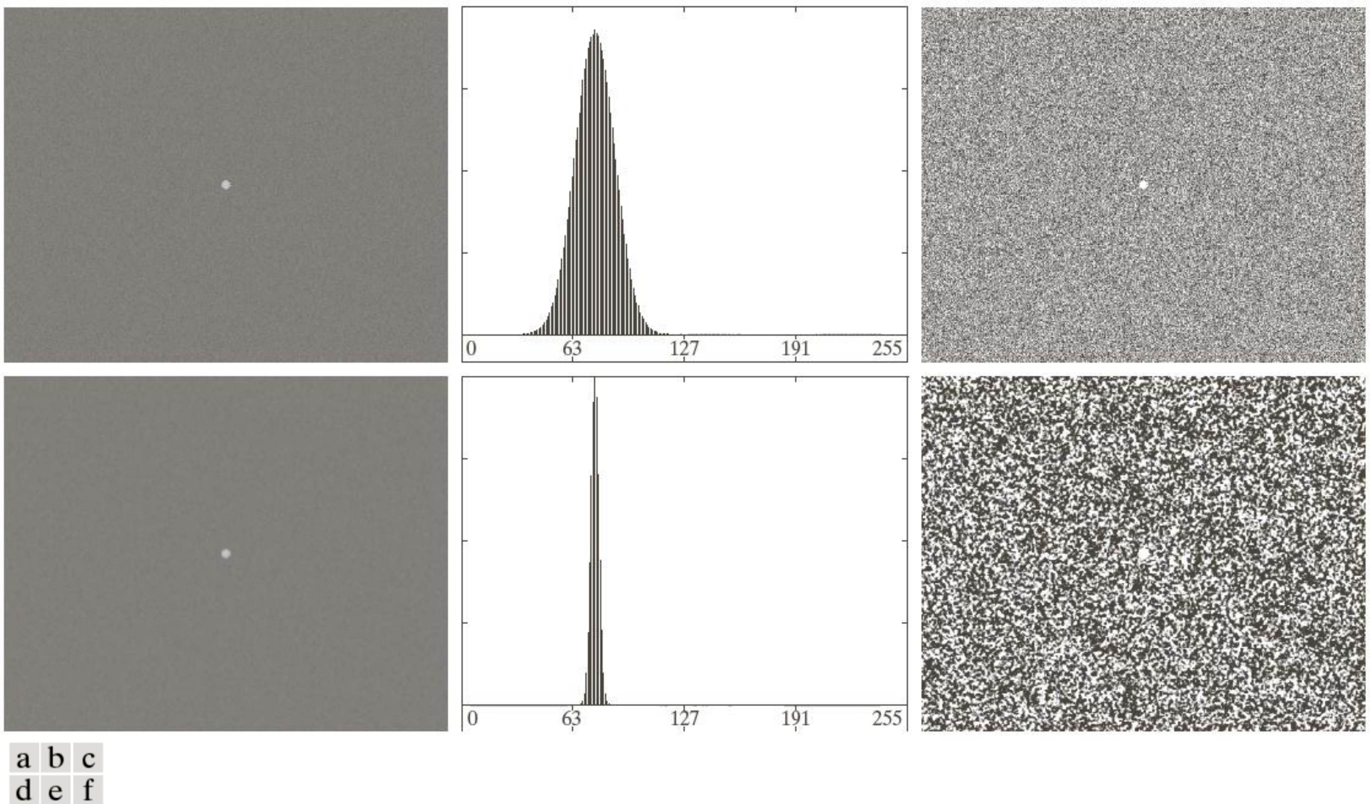


FIGURE 10.41 (a) Noisy image and (b) its histogram. (c) Result obtained using Otsu's method. (d) Noisy image smoothed using a 5×5 averaging mask and (e) its histogram. (f) Result of thresholding using Otsu's method. Thresholding failed in both cases.

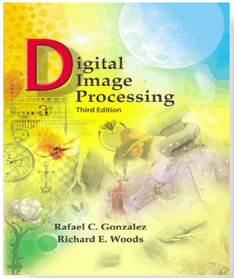


Image Segmentation

- 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:

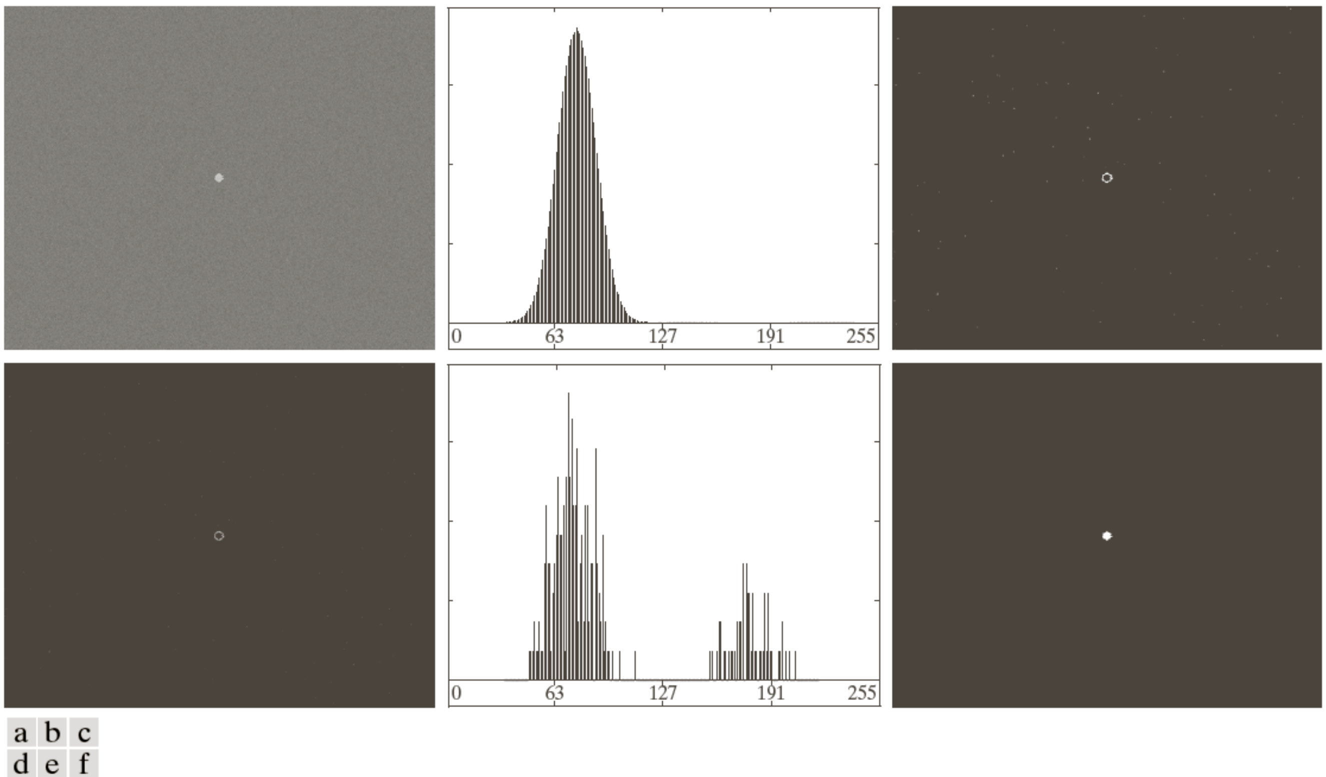


FIGURE 10.42 (a) Noisy image from Fig. 10.41(a) and (b) its histogram. (c) Gradient magnitude image thresholded at the 99.7 percentile. (d) Image formed as the product of (a) and (c). (e) Histogram of the nonzero pixels in the image in (d). (f) Result of segmenting image (a) with the Otsu threshold based 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:

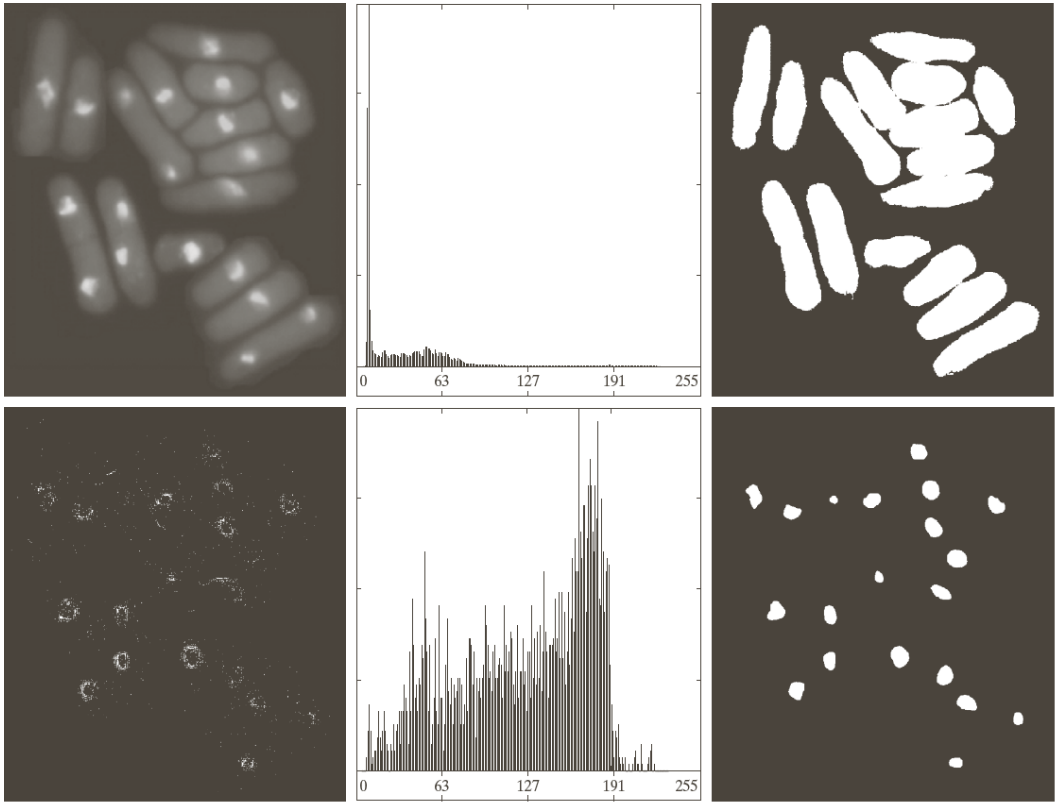


FIGURE 10.43 (a) Image of yeast cells. (b) Histogram of (a). (c) Segmentation of (a) with Otsu's method using the histogram in (b). (d) Thresholded absolute Laplacian. (e) Histogram of the nonzero pixels in the product of (a) and (d). (f) Original image thresholded using Otsu's method based on the histogram 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:

$$\sigma_B^2 = \sum_{k=1}^{K} P_k \left(m_k - m_G\right)^2$$

$$P_k = \sum_{i \in C_k} p_i$$

$$m_k = \frac{1}{P_k} \sum_{i \in C_k} i p_i$$

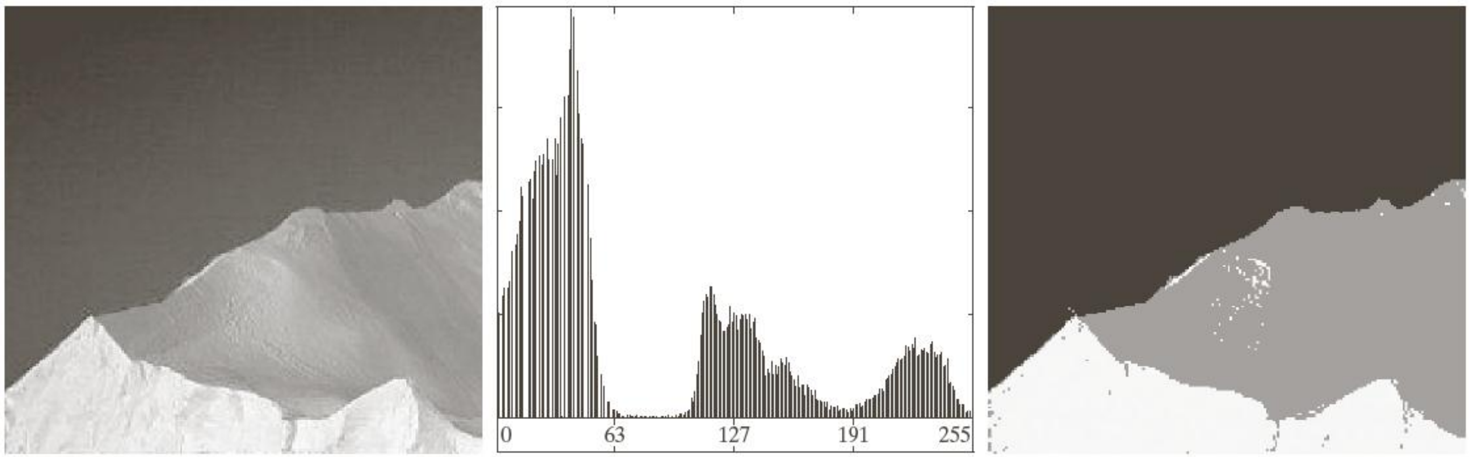
  - Optimal Solution:

$$\left\{k_1^*, k_2^*, \ldots, k_{K-1}^*\right\} = \underset{0<k_1<k_2<\ldots<k_{K-1}<L-1}{\arg\max} \left\{\sigma_B^2\left(k_1, k_2, \ldots, k_{K-1}\right)\right\}$$

  - A expensive computation task!
    - A search in $\mathbb{R}^{K-1}$

Image Segmentation

- Example



a b c

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 Segmentation

- Image Partitioning:

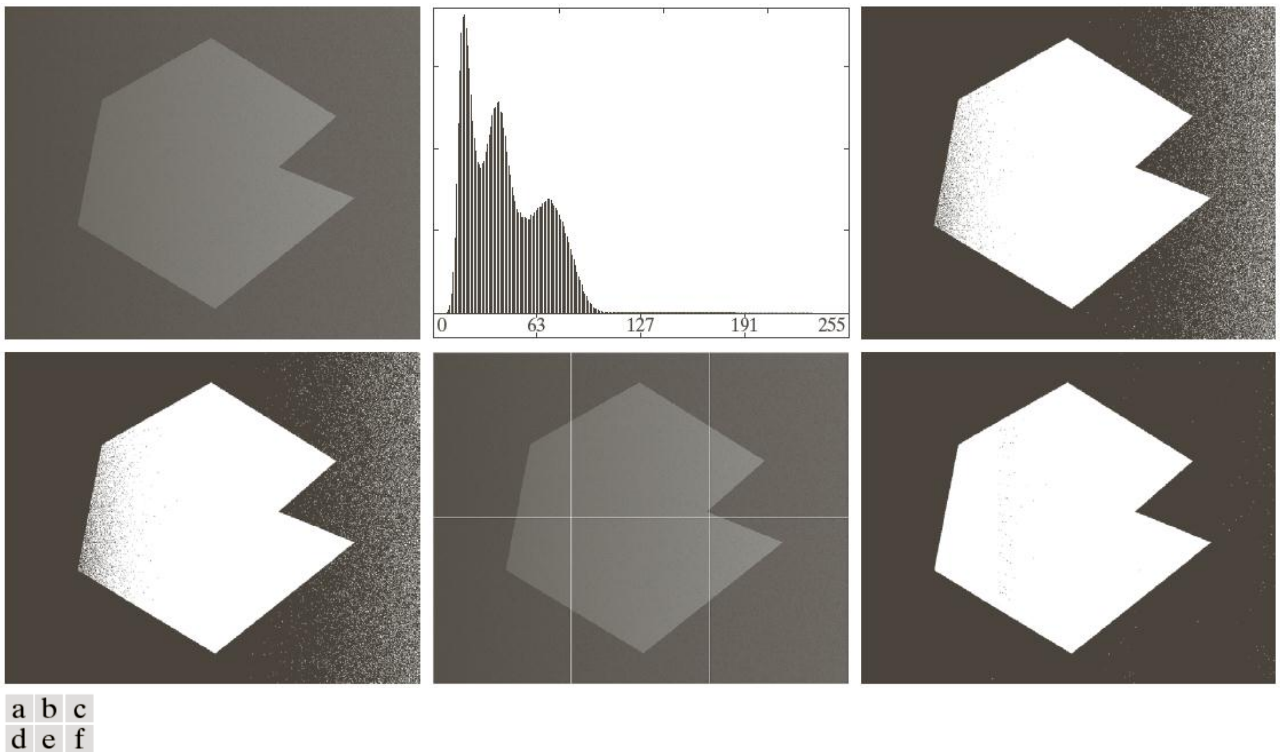
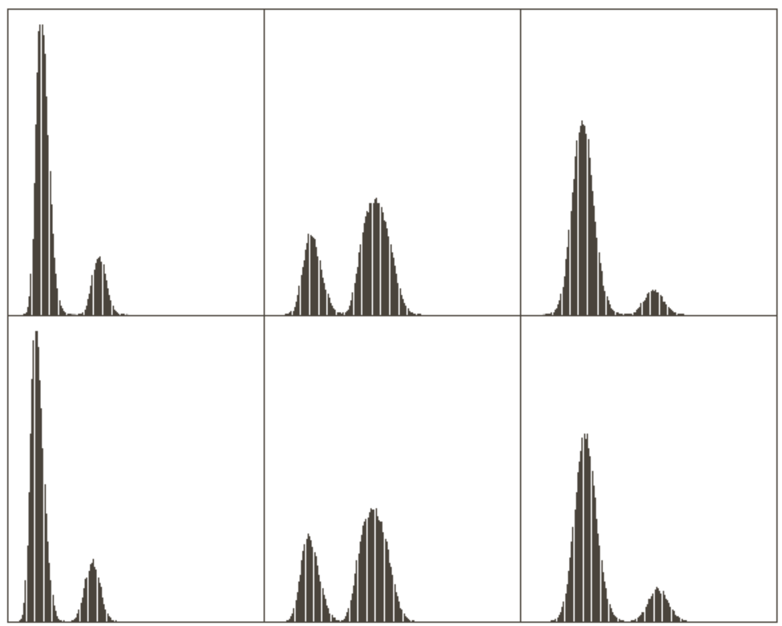


FIGURE 10.46 (a) Noisy, shaded image and (b) its histogram. (c) Segmentation of (a) using the iterative global algorithm from Section 10.3.2. (d) Result obtained using Otsu's method. (e) Image subdivided into six subimages. (f) Result of applying Otsu's method to each subimage individually.

Image Segmentation

- Image Partitioning:
  - Histogram of each partition
    - Bimodal → Otsu Optimal Thresholding

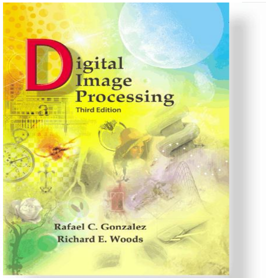


- Using Local Image Properties:
  - Compute Local mean and Local standard deviation
  - Design a rule, such as:

$$T(x,y) = a\sigma_L(x,y) + bm_L(x,y)$$
$$T(x,y) = a\sigma_L(x,y) + bm_G$$

Image Segmentation

- Example:



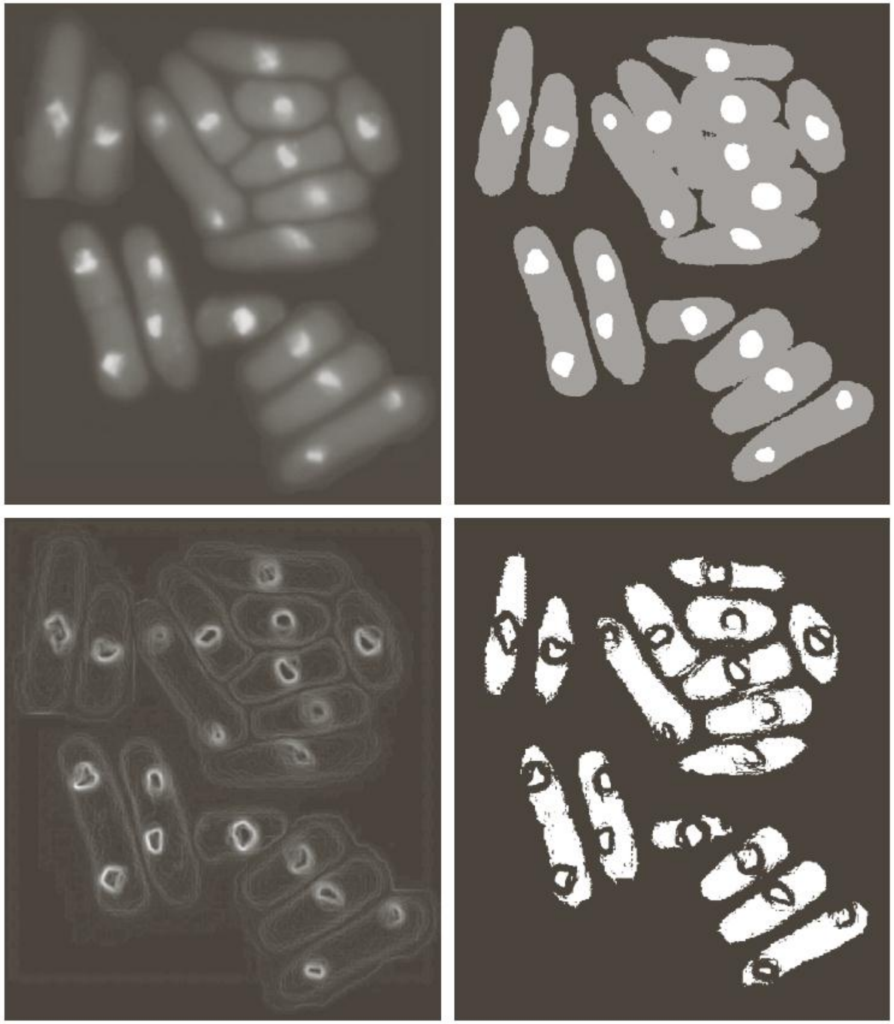


FIGURE 10.48 (a) Image from Fig. 10.43. (b) Image segmented using the dual thresholding approach discussed in Section 10.3.6. (c) Image of local standard deviations. (d) Result obtained using local thresholding.

Image Segmentation

- Using Moving Average:
  - Consider the following rule:

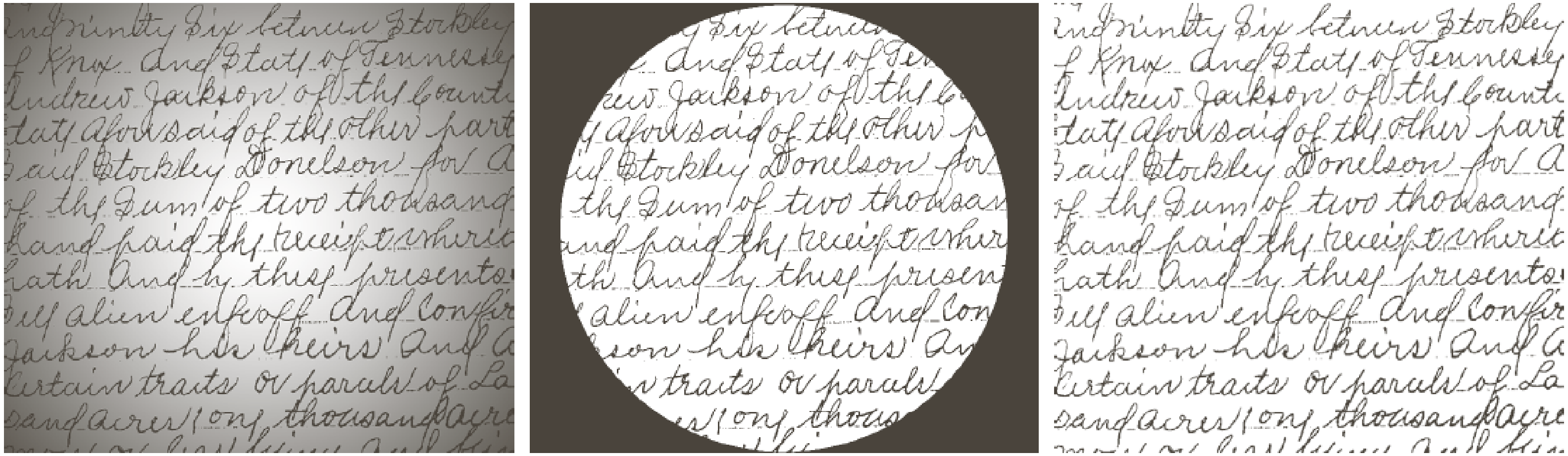
$$T(x,y) = a m_L(x,y)$$

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

$$\begin{aligned} m(k+1) &= \frac{1}{n}\sum_{i=k+2-n}^{k+1} z_i \\ &= \left[\frac{1}{n}\sum_{i=k+1-n}^{k} z_i\right] + \frac{1}{n}\left(z_{k+1} - z_{k-n}\right) \\ &= m(k) + \frac{1}{n}\left(z_{k+1} - z_{k-n}\right) \end{aligned}$$

Image Segmentation

- Example:
  - N=20, a=0.5

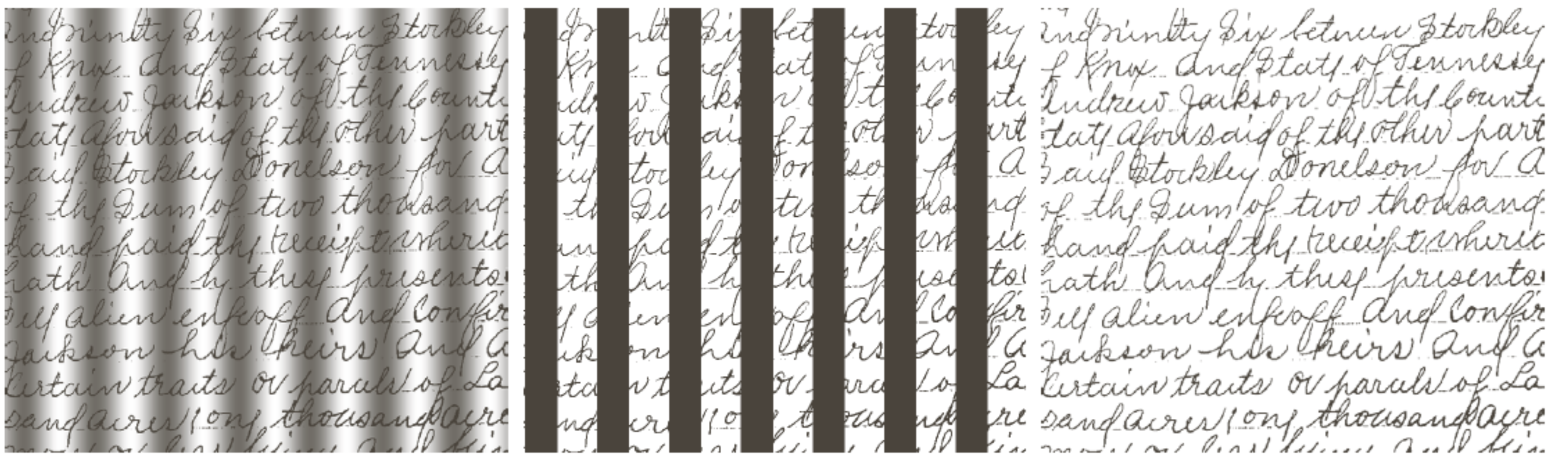


a b c

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.

Image Segmentation

- Example:
  - N=20, a=0.5



a b c

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: $\mathbf{z} \in \mathbb{R}^N$
    - Thresholding:

$$g = \begin{cases} 1 & \text{dist}(\mathbf{z},\mathbf{a}) < T \\ 0 & otherwise \end{cases}$$

    - $\mathbf{a}$: A specific color
  - Segmentation:
    - A Clustering task

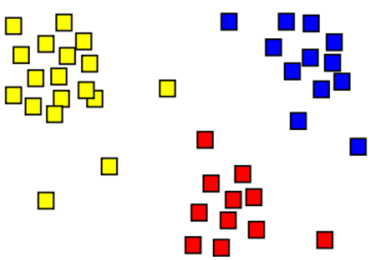
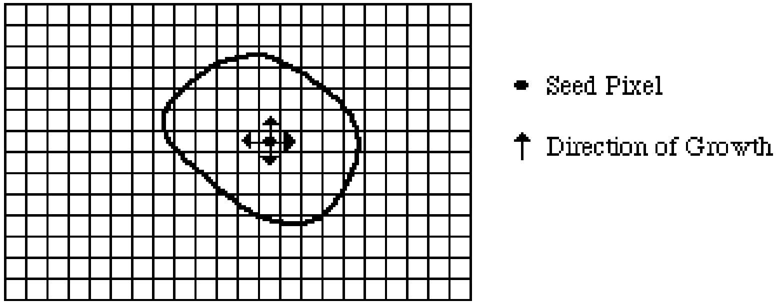


Image Segmentation

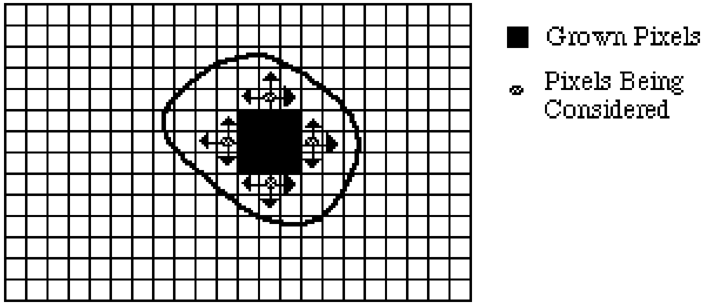
- Region Growing:
  - Select a start (seed) point
  - Grow the point based on a certain property
    - Connectivity should be considered.
  - Seed point selection:
    - Handy
    - Highlighted point (Due to specific property)

Image Segmentation

- Region Growing:



(a) Start of Growing a Region



(b) Growing Process After a Few Iterations

Image Segmentation

- 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_8 of region.

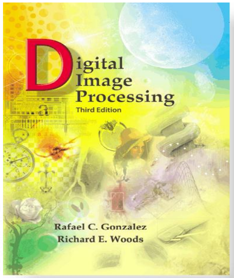


Image Segmentation

- Example – N8 Connectivity and T = 68, 126

a b c
d e f
g h i

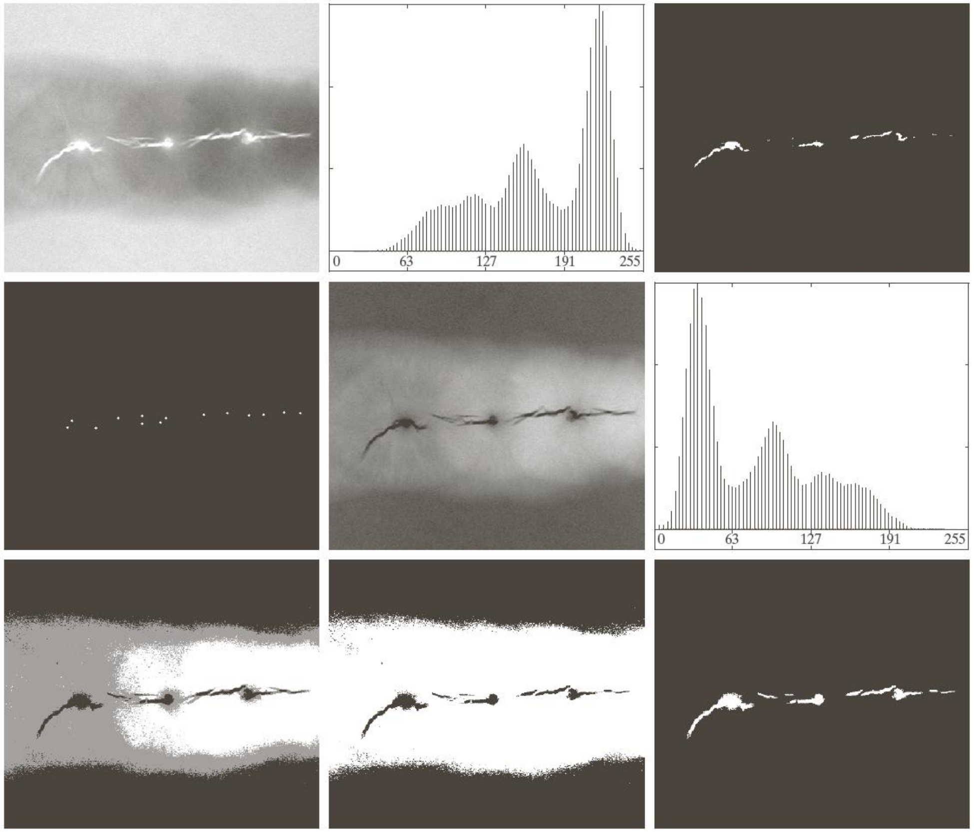


FIGURE 10.51 (a) X-ray image of a defective weld. (b) Histogram. (c) Initial seed image. (d) Final seed image (the points were enlarged for clarity). (e) Absolute value of the difference between (a) and (c). (f) Histogram of (e). (g) Difference image thresholded using dual thresholds. (h) Difference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

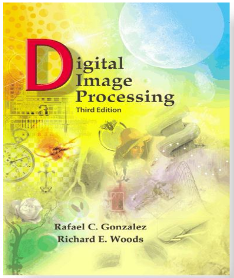
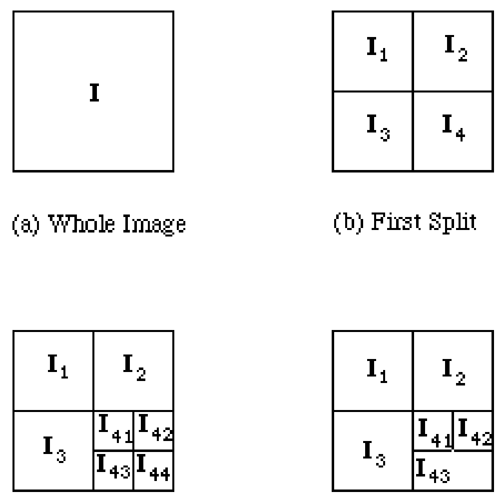


Image Segmentation

- 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

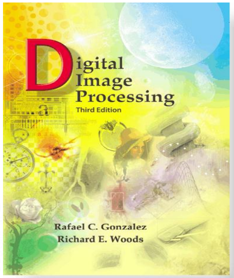
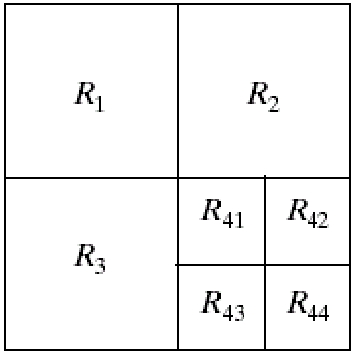


Image Segmentation

- Splitting and Merging



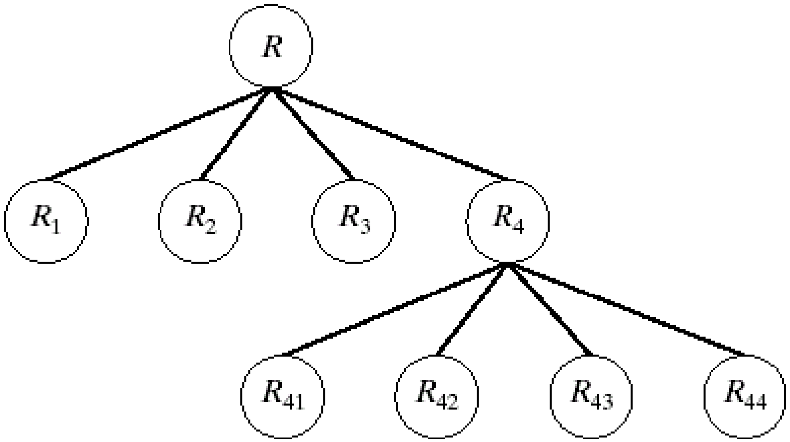
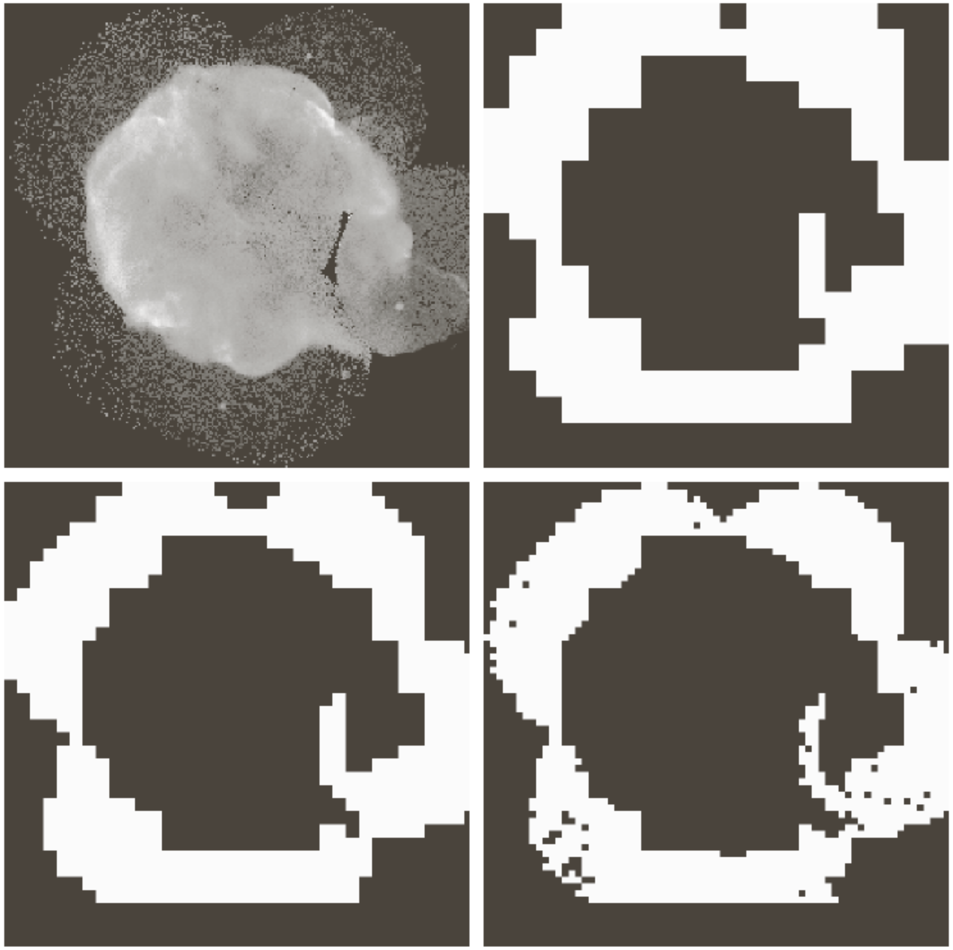


Image Segmentation

- Splitting-Merging Criteri
  - Example #1: n% of all pixels satisfy: $\left|z_j - m_i\right| \le 2\sigma_i$
  - Example #2: $\sigma > a \quad \text{AND} \quad 0 < m < b$

Image Segmentation

- Example:



a b
c d

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 32×32, 16×16, and 8×8 pixels, respectively. (Original image courtesy of NASA.)