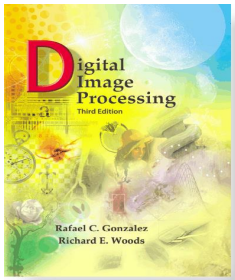


Chapter 10
Segmentation

10.1 Fundamentals

10.2 Point, Line, and Edge Detection

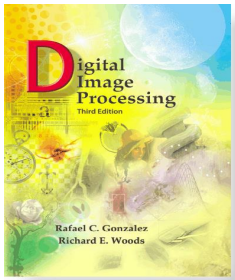
10.3 Thresholding

10.4 Region-Based Segmentation

10.5 Segmentation Using Morphological Watersheds

10.6 The Use of Motion in Segmentation

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10

Segmentation

10.1 Fundamentals

- **Basic Formulation**
  - Partition the entire image R into n subimages $R_1, R_2, \ldots, R_n$, such taht

(a) $\bigcup_{i=1}^{n} R_i = R.$

(b) R_i is a connected region, $i = 1, 2, \ldots, n.$

(c) $R_i \cap R_j = \varnothing$ for all i and j, $i \neq j.$

(d) $Q(R_i) = TRUE$ for $i = 1, 2, \ldots, n.$

(e) $Q(R_i \cup R_j) = FALSE$ for any adjacent regions R_i *and* R_j.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Condition (d): If a pixel is on, or inside the boundary, label it white; otherwise label it black.**
- **Condition (e).**

- **It is fail by Edge information !**

- **Region information : standard deviation of pixel values.**

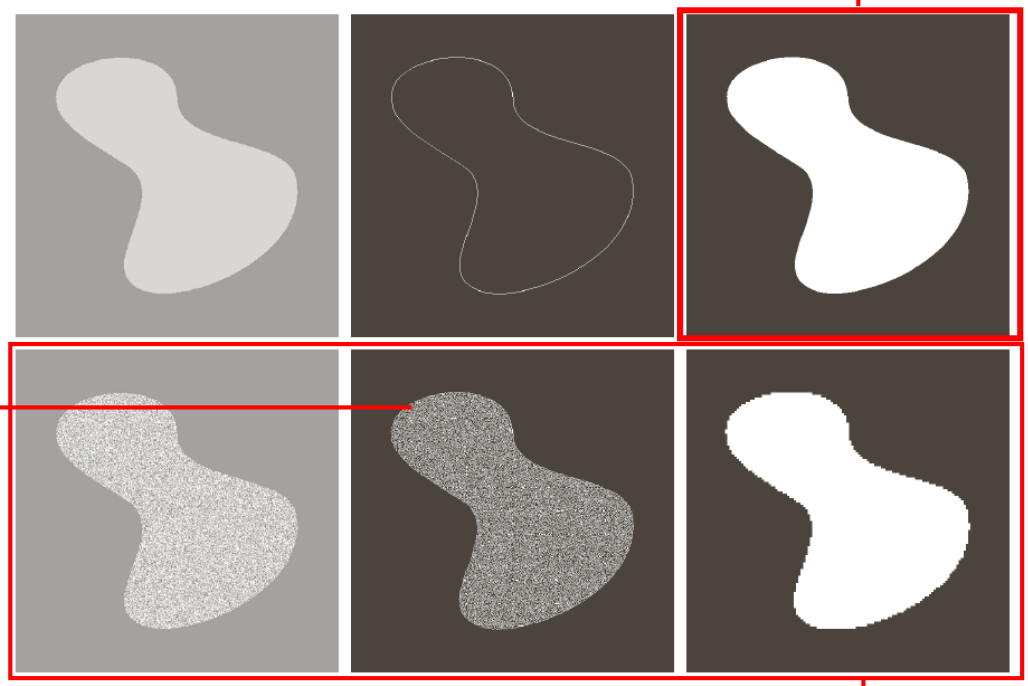


FIGURE 10.1 (a) Image containing a region of constant intensity. (b) Image showing the boundary of the inner region, obtained from intensity discontinuities. (c) Result of segmenting the image into two regions. (d) Image containing a textured region. (e) Result of edge computations. Note the large number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

© 1992–2008 R. C. Gonzalez & R. E. Woods

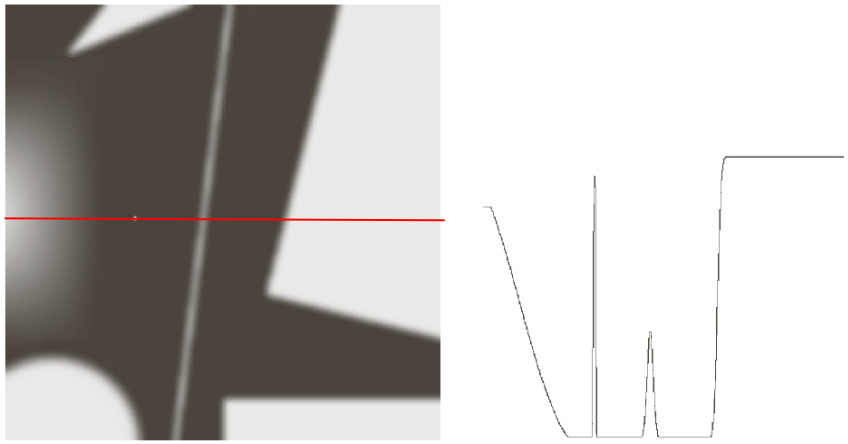
10.2 Point, Line, and Edge detection

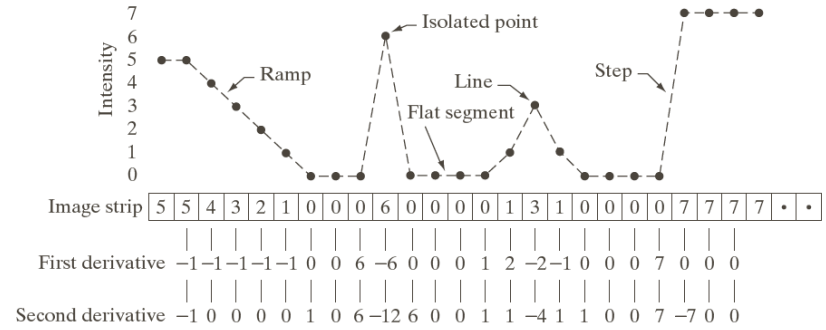
- *First-order derivatives*

$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

- *Second-order derivatives*

$$\frac{\partial^2 f}{\partial^2 x} = f(x+1) + f(x-1) - 2f(x)$$





a b
c

FIGURE 10.2 (a) Image. (b) Horizontal intensity profile through the center of the image, including the isolated noise point. (c) Simplified profile (the points are joined by dashes for clarity). The image strip corresponds to the intensity profile, and the numbers in the boxes are the intensity values of the dots shown in the profile. The derivatives were obtained using Eqs. (10.2-1) and (10.2-2).

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- *Compute first-order and second-order derivatives using spatial filters.*

$$R = w_1 z_1 + w_2 z_2 + \cdots + w_9 z_9$$
$$= \sum_{i=1}^{9} w_i z_i$$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

FIGURE 10.3 A general 3×3 spatial filter mask.

© 1992–2008 R. C. Gonzalez & R. E. Woods

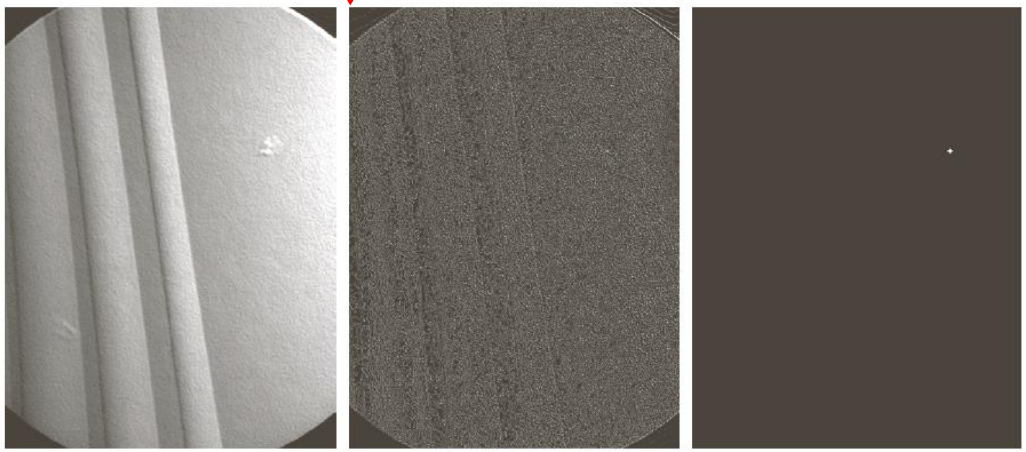
Chapter 10
Segmentation

- **Detection of isolated point**

$$g(x,y)=\begin{cases}1 & if\ |R(x,y)| \geq T \\ 0 & \text{otherwise}\end{cases}$$

T: 90% of the highest absolute pixel value of Fig. 10.4(c)

1	1	1
1	−8	1
1	1	1



a
b c d

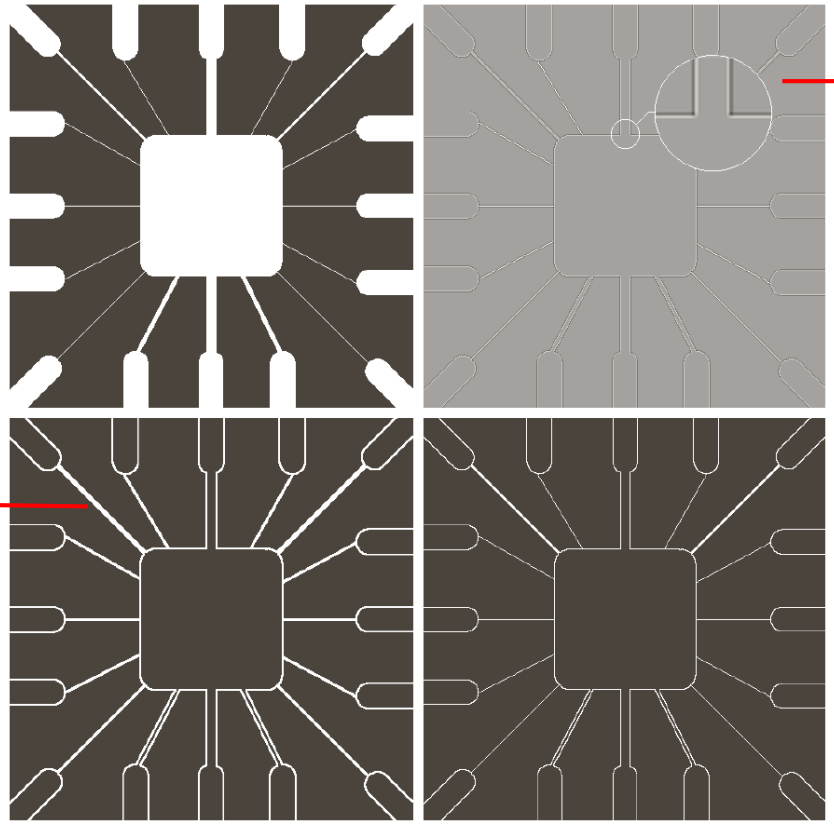
FIGURE 10.4 (a) Point detection (Laplacian) mask. (b) X-ray image of turbine blade with a porosity. The porosity contains a single black pixel. (c) Result of convolving the mask with the image. (d) Result of using Eq. (10.2-8) showing a single point (the point was enlarged to make it easier to see). (Original image courtesy of X-TEK Systems, Ltd.)

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Line Detection**

The problem of Laplacian operator in line detection !



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.

Double line effect

Double the thickness of line

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Line Detection**

$$|R_k| > |R_j| \quad for\ all\ \ j \neq k$$

$$Example:$$

$$|R_1| > |R_j| \quad for\ \ j = 2,3,4$$

The particular point is said to be more likely associated with a horizontal line.

R_1			R_2			R_3			R_4		
−1	−1	−1	2	−1	−1	−1	2	−1	−1	−1	2
2	2	2	−1	2	−1	−1	2	−1	−1	2	−1
−1	−1	−1	−1	−1	2	−1	2	−1	2	−1	−1
Horizontal			+45°			Vertical			−45°		

FIGURE 10.6 Line detection masks. Angles are with respect to the axis system in Fig. 2.18(b).

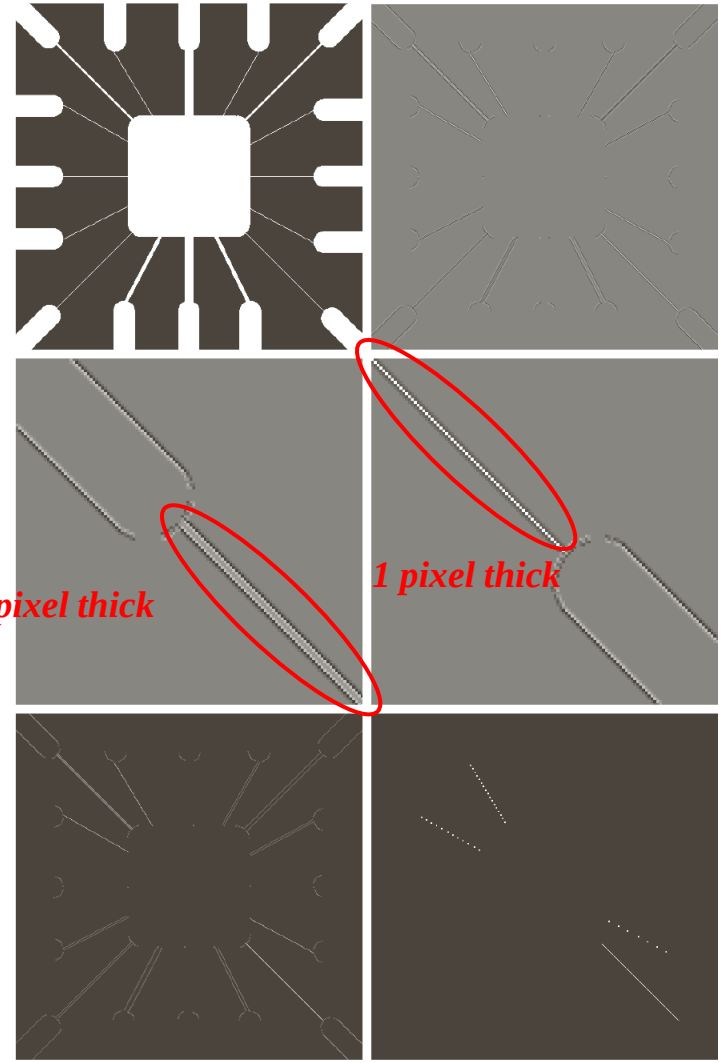
© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Example of line detection**

2	−1	−1
−1	2	−1
−1	−1	2

+45°



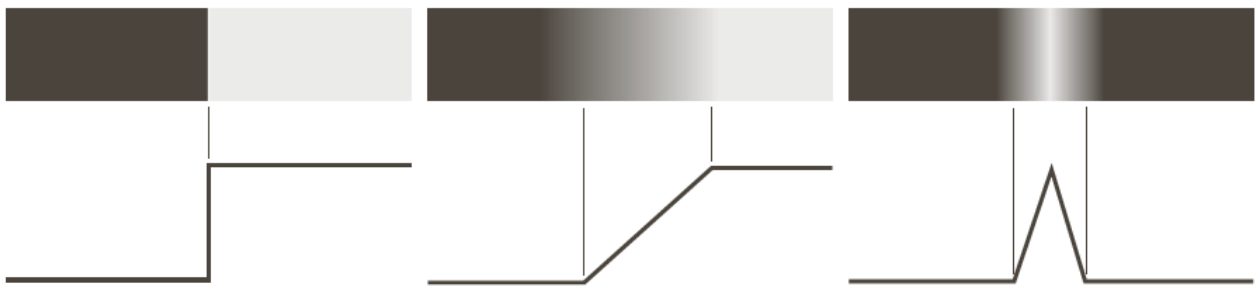
a b
c d
e f

FIGURE 10.7 (a) Image of a wire-bond template. (b) Result of processing with the +45° line detector mask in Fig. 10.6. (c) Zoomed view of the top left region of (b). (d) Zoomed view of the bottom right region of (b). (e) The image in (b) with all negative values set to zero. (f) All points (in white) whose values satisfied the condition $g \geq T$, where g is the image in (e). (The points in (f) were enlarged to make them easier to see.)

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Edge Models**



a b c

FIGURE 10.8 From left to right, models (ideal representations) of a step, a ramp, and a roof edge, and their corresponding intensity profiles.

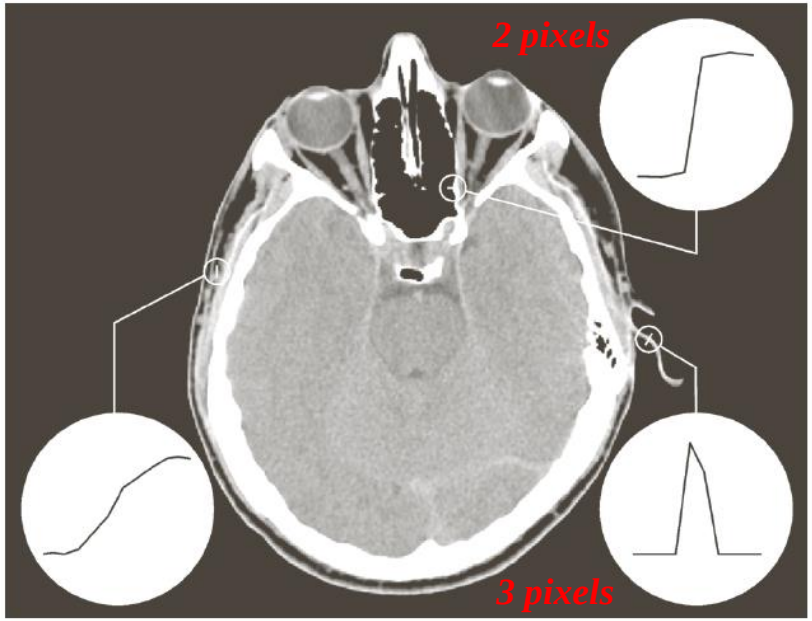
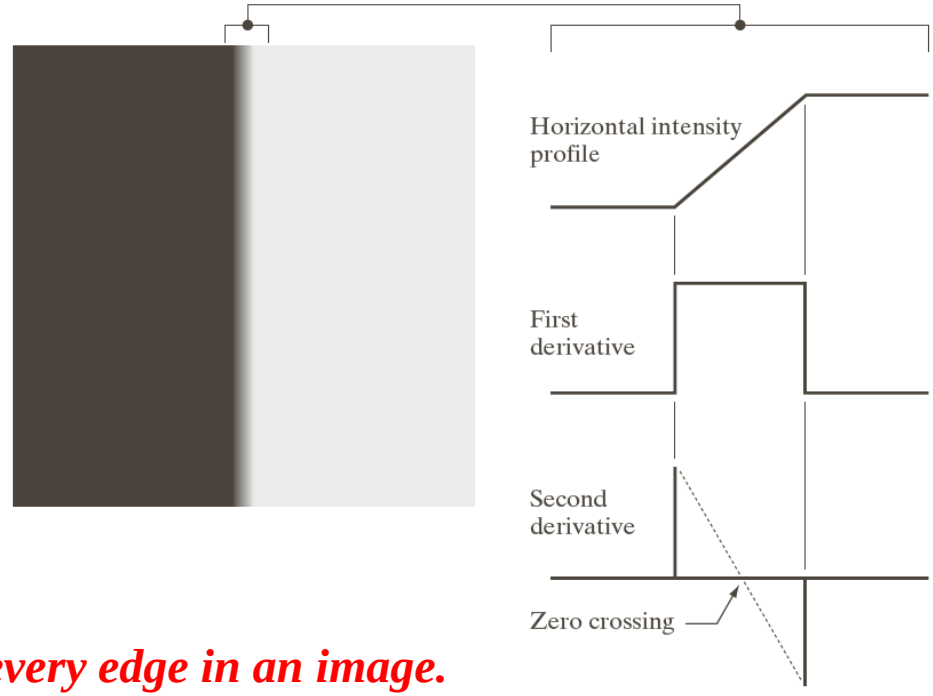


FIGURE 10.9 A 1508 × 1970 image showing (zoomed) actual ramp (bottom, left), step (top, right), and roof edge profiles. The profiles are from dark to light, in the areas indicated by the short line segments shown in the small circles. The ramp and "step" profiles span 9 pixels and 2 pixels, respectively. The base of the roof edge is 3 pixels. (Original image courtesy of Dr. David R. Pickens, Vanderbilt University.)

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

– 1st and 2nd derivative



a b

FIGURE 10.10 (a) Two regions of constant intensity separated by an ideal vertical ramp edge. (b) Detail near the edge, showing a horizontal intensity profile, together with its first and second derivatives.

- **2nd derivative**

(1)It produces 2 values for every edge in an image.

(2)It's zero crossings can be used for locating the centers of thick edges.

© 1992–2008 R. C. Gonzalez & R. E. Woods

– Behavior of the 1st and 2nd derivatives around a noisy edge

- ***There are 3 fundamental steps performed in edge detection:***
  - ***Image smoothing for noise reduction.***
  - ***Detection of edge points: extract the potential candidates from an image.***
  - ***Edge localization: select the true members.***

How to do ?

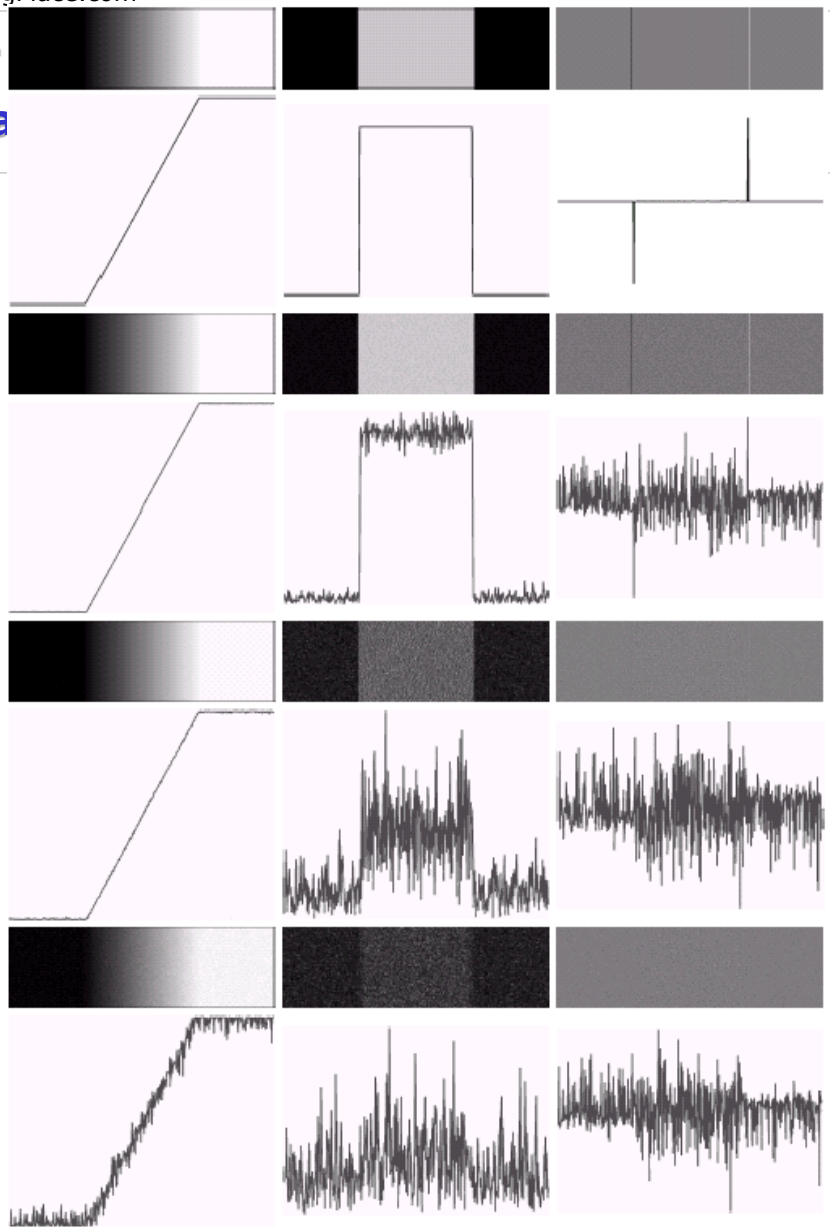


FIGURE 10.7 First column: images and gray-level profiles of a ramp edge corrupted by random Gaussian noise of mean 0 and $\sigma = 0.0, 0.1, 1.0,$ and 10.0, respectively. Second column: first-derivative images and gray-level profiles. Third column: second-derivative images and gray-level profiles.

a
b
c
d

© 1992–2008 R. C. Gonzalez & R. E. Woods

– Gradient operators

$$\nabla f \equiv grad(f) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix}$$

Magnitude:

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

Direction:

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

© 1992–2008 R. C. Gonzalez & R. E. Woods

- **Examples**

$$\nabla f = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} -2 \\ 2 \end{bmatrix}$$

$$M(x, y) = 2\sqrt{2}$$

$$\alpha(x, y) = \tan^{-1}\left(\frac{g_y}{g_x}\right) = -45^\circ \left(135^\circ\right)$$

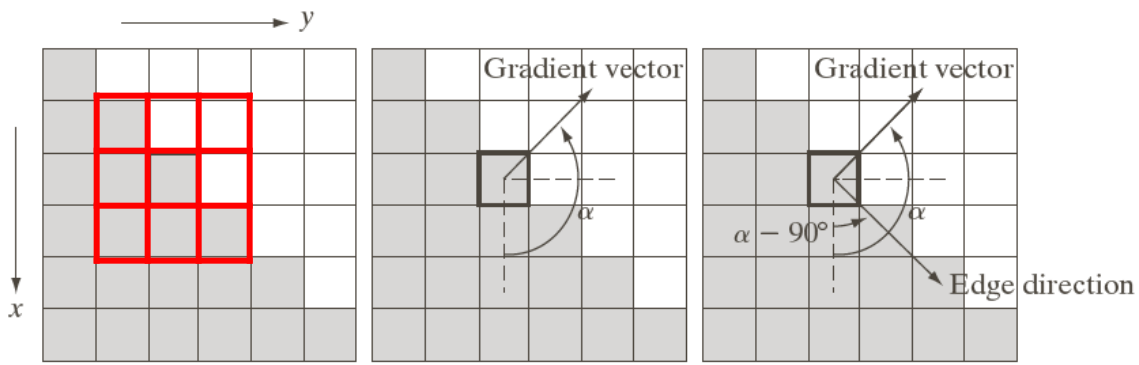


FIGURE 10.12 Using the gradient to determine edge strength and direction at a point. Note that the edge is perpendicular to the direction of the gradient vector at the point where the gradient is computed. Each square in the figure represents one pixel.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

– Gradient operators

a
b c
d e
f g

FIGURE 10.14 A 3 × 3 region of an image (the z's are intensity values) and various masks used to compute the gradient at the point labeled z_5.

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	0	0	−1
0	1	1	0

Roberts

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

Prewitt

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

Sobel

Roberts Cross-Gradient

$g_x = (z_9 - z_5)$

$g_y = (z_8 - z_6)$

Prewitt Operators

$g_x = (z_7 - z_1) + (z_8 - z_2) + (z_9 - z_3)$

$g_y = (z_3 - z_1) + (z_6 - z_4) + (z_9 - z_7)$

Sobel Operators

$g_x = (z_7 - z_1) + 2(z_8 - z_2) + (z_9 - z_3)$

$g_y = (z_3 - z_1) + 2(z_6 - z_4) + (z_9 - z_7)$

- Approximation of Gradient Magnitude

$$M(x,y) \approx |g_x| + |g_y|$$

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- Modification to obtain strongest response along the diagonal directions.

0	1	1
−1	0	1
−1	−1	0

−1	−1	0
−1	0	1
0	1	1

Prewitt

0	1	2
−1	0	1
−2	−1	0

−2	−1	0
−1	0	1
0	1	2

Sobel

a b
c d

FIGURE 10.15 Prewitt and Sobel masks for detecting diagonal edges.

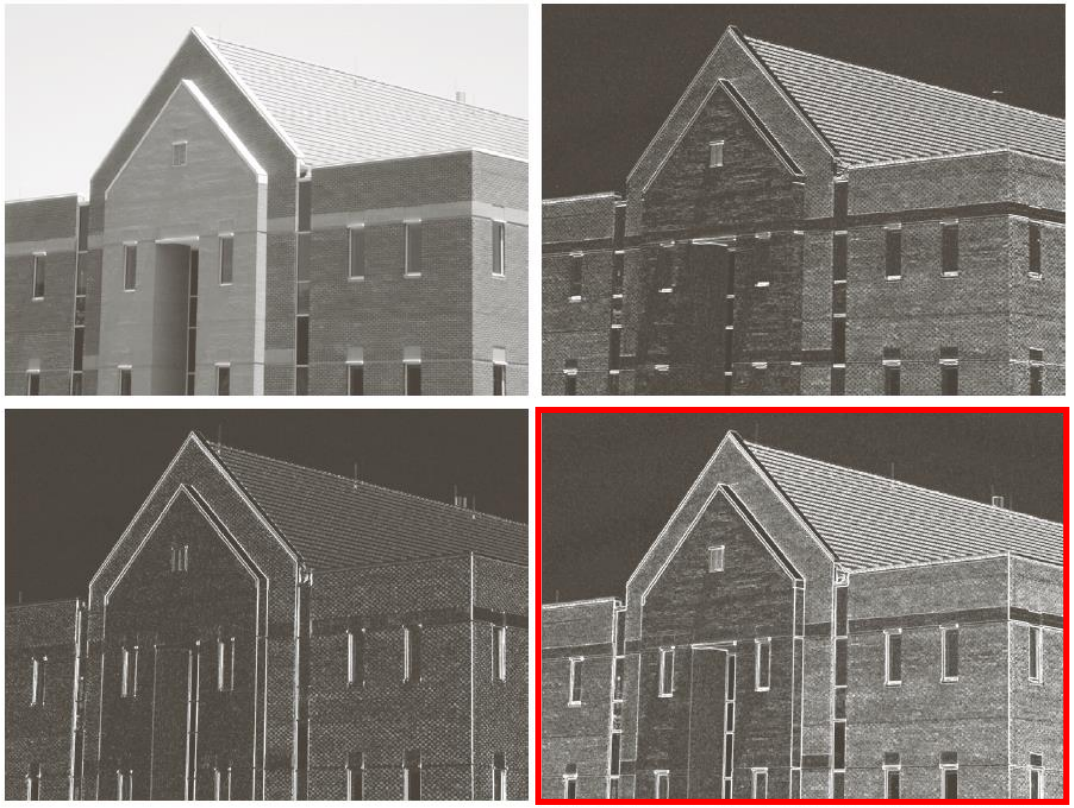
© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- Example

$|g_x|$

$|g_y|$



$$M(x,y) \approx |g_x| + |g_y|$$

a b
c d

FIGURE 10.16 (a) Original image of size 834 × 1114 pixels, with intensity values scaled to the range [0, 1]. (b) $|g_x|$, the component of the gradient in the x-direction, obtained using the Sobel mask in Fig. 10.14(f) to filter the image. (c) $|g_y|$, obtained using the mask in Fig. 10.14(g). (d) The gradient image, $|g_x| + |g_y|$.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

$$\text{Direction}:$$

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

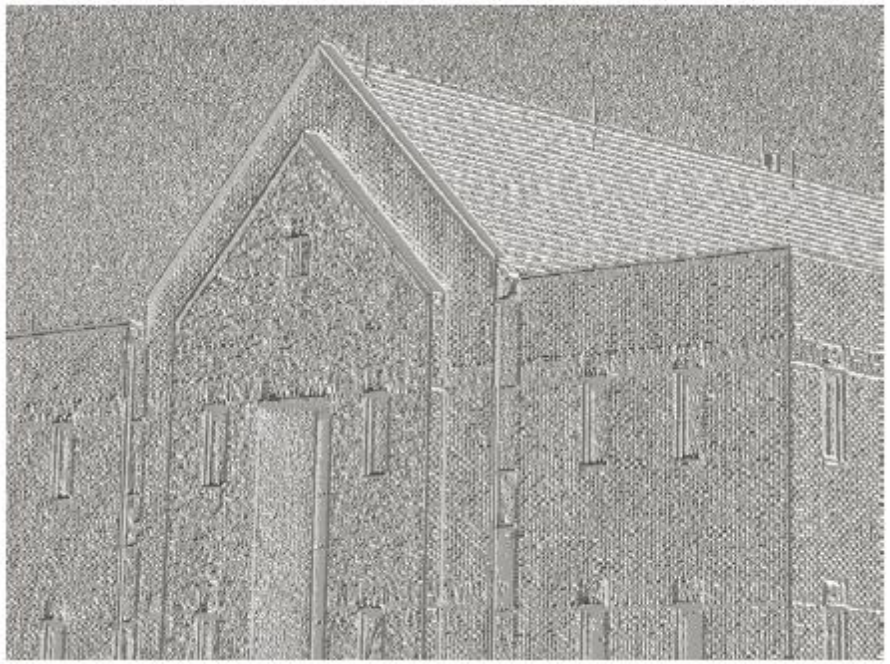


FIGURE 10.17 Gradient angle image computed using Eq. (10.2-11). Areas of constant intensity in this image indicate that the direction of the gradient vector is the same at all the pixel locations in those regions.

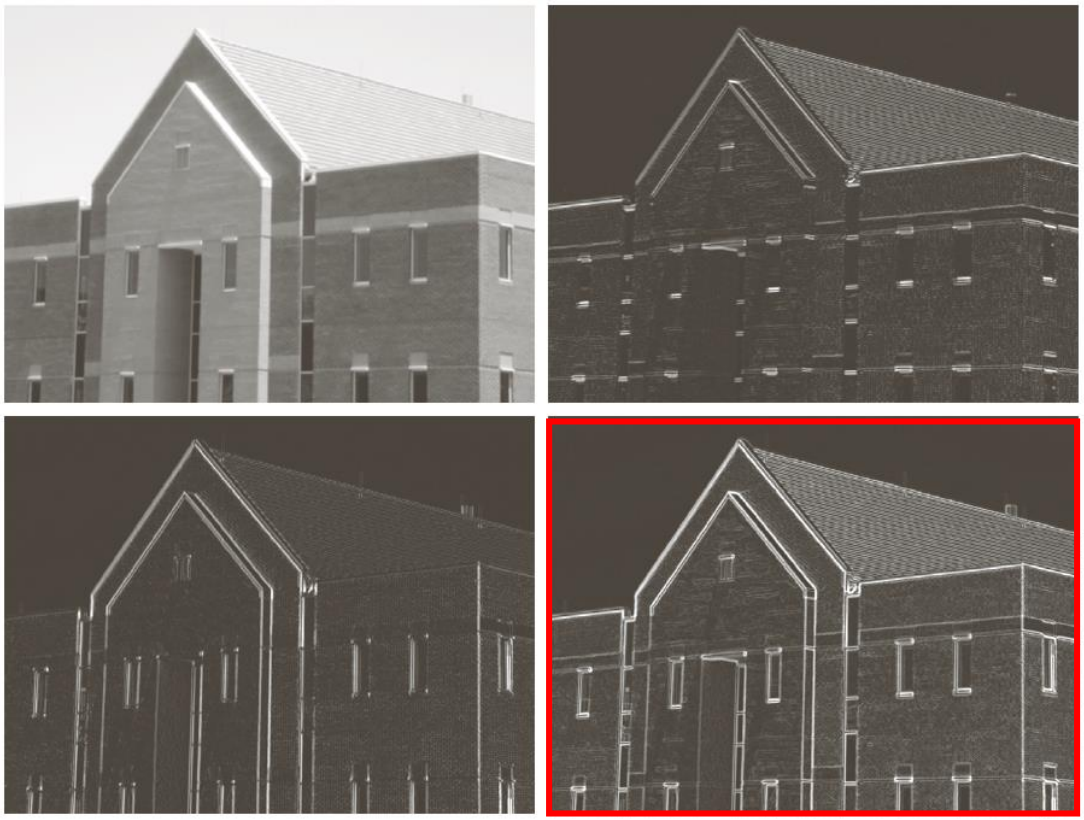
© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- Reducing the wall bricks effect by adding a 5x5 averaging filter.

5x5 averaging filter

5x5 averaging filter



$|g_x|$

$|g_y|$

a b
c d

FIGURE 10.18 Same sequence as in Fig. 10.16, but with the original image smoothed using a 5×5 averaging filter prior to edge detection.

$$M(x,y) \approx |g_x| + |g_y|$$

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- Modification to obtain strongest response along the diagonal directions.

0	1	2
−1	0	1
−2	−1	0

−2	−1	0
−1	0	1
0	1	2

a b

FIGURE 10.19 Diagonal edge detection. (a) Result of using the mask in Fig. 10.15(c). (b) Result of using the mask in Fig. 10.15(d). The input image in both cases was Fig. 10.18(a).

© 1992–2008 R. C. Gonzalez & R. E. Woods

- Combining the gradient with thresholding

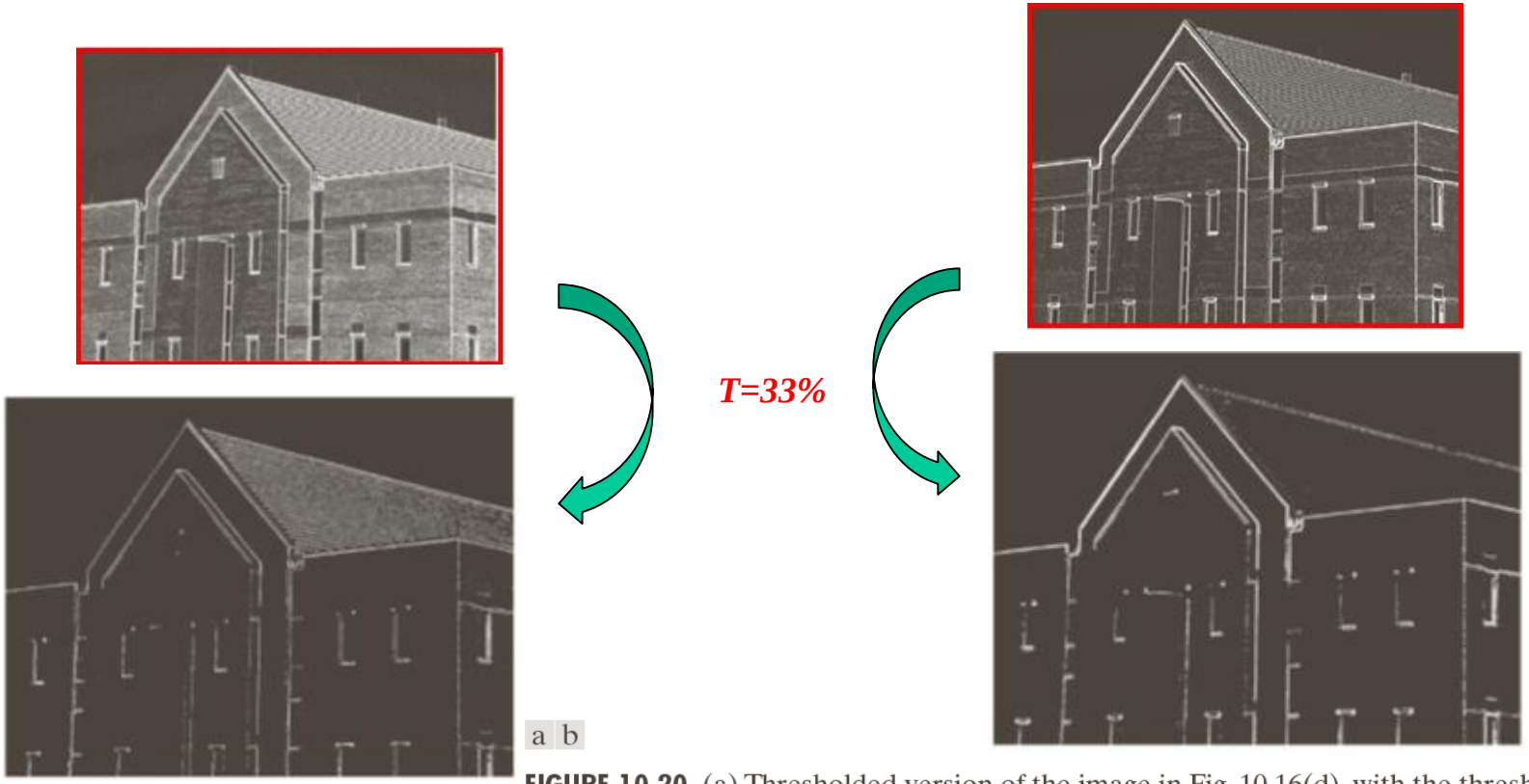
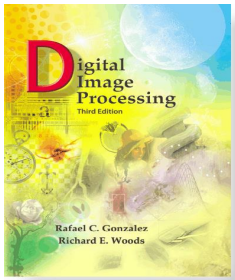


FIGURE 10.20 (a) Thresholded version of the image in Fig. 10.16(d), with the threshold selected as 33% of the highest value in the image; this threshold was just high enough to eliminate most of the brick edges in the gradient image. (b) Thresholded version of the image in Fig. 10.18(d), obtained using a threshold equal to 33% of the highest value in that image.

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10
Segmentation

- **More Advanced Techniques for Edge Detection**
  - The Marr-Hildreth edge detector

    (1) It should be a differential operator capable of computing a digital approximation of the 1st or 2nd derivative at every point in the image.

    (2) It should be capable of being "tuned" to act at any desired scale, so that large operators can be used to detect blurry edges and small operators to detect sharply focused fine detail

Laplacian of a Gaussian (LoG)

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

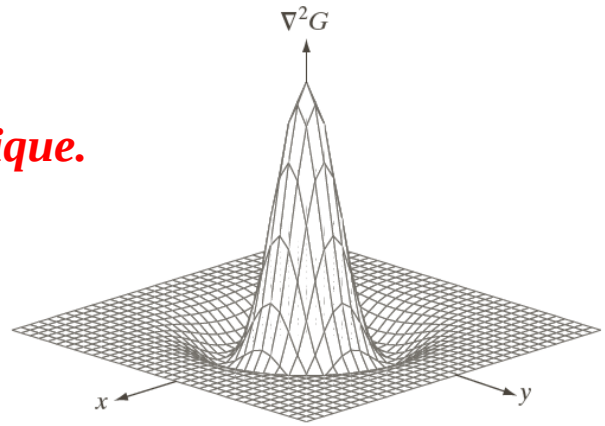
$$\nabla^2 G(x,y) = \frac{\partial^2 G(x,y)}{\partial x^2} + \frac{\partial^2 G(x,y)}{\partial y^2} = \frac{\partial}{\partial x}\left[\frac{-x}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}\right] + \frac{\partial}{\partial y}\left[\frac{-y}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}\right]$$

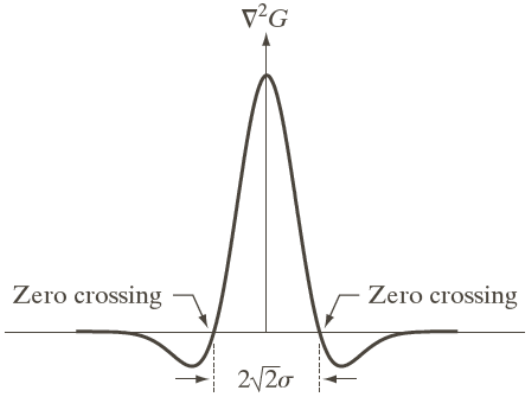
$$= \left[\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2}\right] e^{-\frac{x^2+y^2}{2\sigma^2}} + \left[\frac{y^2}{\sigma^4} - \frac{1}{\sigma^2}\right] e^{-\frac{x^2+y^2}{2\sigma^2}} = \left[\frac{x^2+y^2-2\sigma^2}{\sigma^4}\right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

© 1992–2008 R. C. Gonzalez & R. E. Woods

– The *Laplacian of a Gaussian* (***LoG***)

- ***Mexican operator.***
- ***The approximation is not unique.***





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

a b
c d

FIGURE 10.21 (a) Three-dimensional plot of the *negative* of the LoG. (b) Negative of the LoG displayed as an image. (c) Cross section of (a) showing zero crossings. (d) 5 × 5 mask approximation to the shape in (a). The negative of this mask would be used in practice.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

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

The Marr-Hildreth edge detector algorithm :

1. ***Filter the input image with an n x n Gaussian filter.***
2. ***Compute the Laplacian of the image resulting from Step 1 using, for example, the 3 x 3 mask.***
3. ***Find zero crossing of the image from Step 2.***

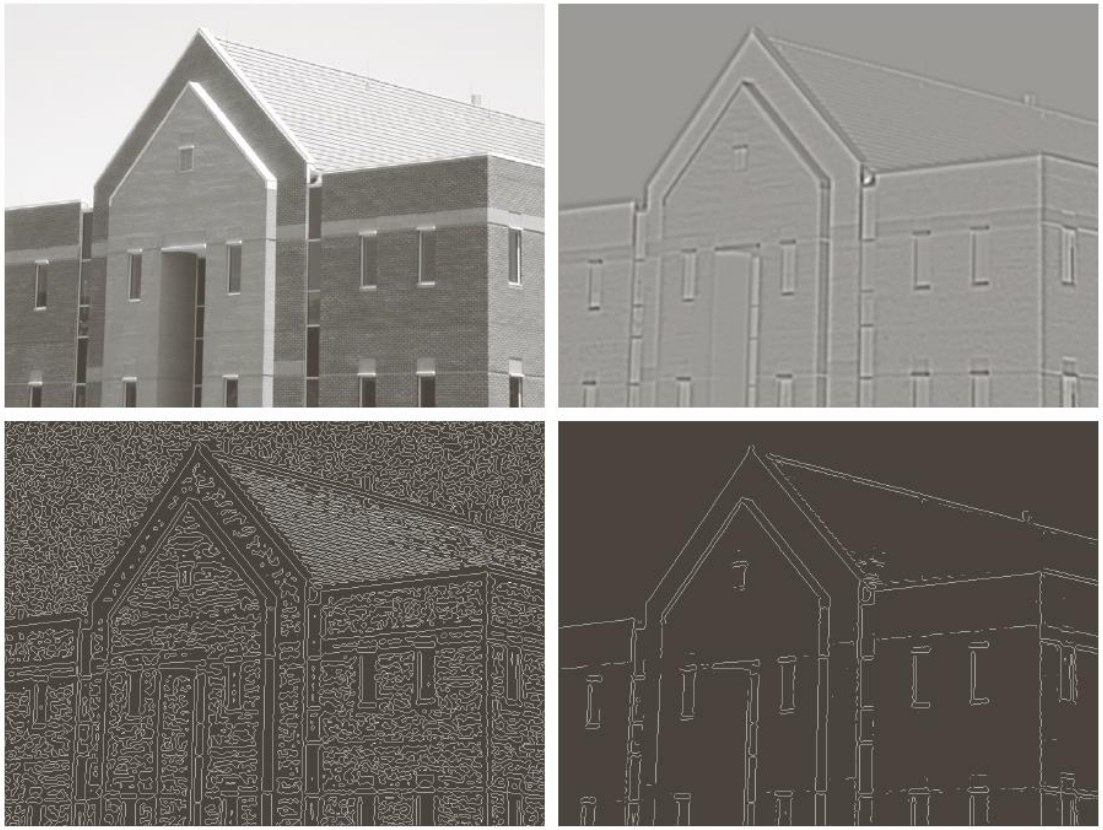
- ***The size of an n x n LoG discrete filter should be such that n is the smallest odd integer greater than or equal to 6σ.***
- ***How to find the zero crossing point ?***

Resulting edges are 1 pixel thick.

© 1992–2008 R. C. Gonzalez & R. E. Woods

- **Example of Marr-Hildreth edge detector**

$\sigma = 4, n = 25$



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.

"Spaghetti" effect !

Threshold: 4% of the maximum value in (b)

© 1992–2008 R. C. Gonzalez & R. E. Woods

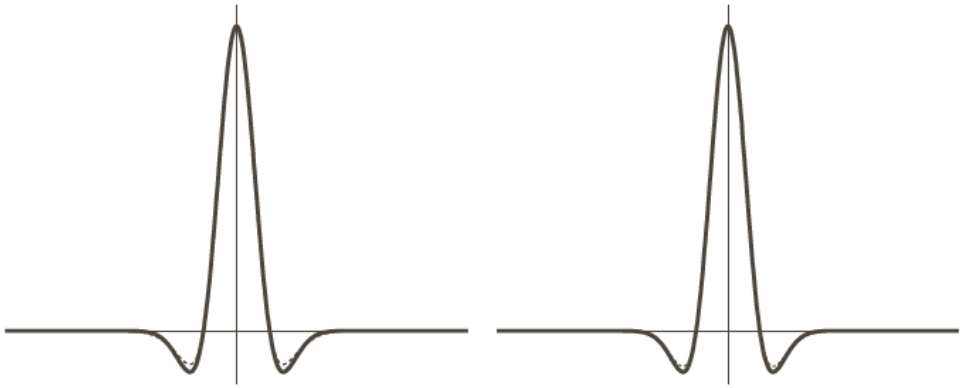
- **Using Difference of Gaussians, *DoG* to approximate *LoG***

[Ref] Martin, D. and Hildreth, E. [1980]. "Theory of Edge Detection," Proc. R. Soc. Lond., Vol. B207, pp. 187~217.

$$DoG(x\,,\,y) = \frac{1}{2\pi\sigma_1^2} e^{-\frac{x^2+y^2}{2\sigma_1^2}} - \frac{1}{2\pi\sigma_2^2} e^{-\frac{x^2+y^2}{2\sigma_2^2}} \quad \text{with } \sigma_1 > \sigma_2$$

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

Same zero-crossing between *DoG* and *LoG*



a b

FIGURE 10.23 (a) Negatives of the LoG (solid) and DoG (dotted) profiles using a standard deviation ratio of 1.75:1. (b) Profiles obtained using a ratio of 1.6:1.

© 1992–2008 R. C. Gonzalez & R. E. Woods

- **Both the LoG and the DoG filtering operations can be implemented with 1-D convolutions instead of using 2-D convolutions**

$$G(x,y) \star f(x,y) = G(x) \star [G(y) \star f(x,y)]$$

Assume that G and f are discrete arrays of size n x n and M x N respectively. Then the computational advantage is

2-D convolution	1-D convolutions	Computational advantage
n^2MN	$2nMN$	$n/2$

© 1992–2008 R. C. Gonzalez & R. E. Woods

(a) As Eq. (10.2-25) shows, the first two steps of the algorithm can be summarized into one equation:

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

Using the definition of the Laplacian operator we can express this equation as

$$\begin{aligned} g(x,y) &= \frac{\partial^2}{\partial x^2}[G(x,y) \star f(x,y)] + \frac{\partial^2}{\partial y^2}[G(x,y) \star f(x,y)] \\ &= \frac{\partial^2}{\partial x^2}[G(x) \star G(y) \star f(x,y)] + \frac{\partial^2}{\partial y^2}[G(x) \star G(y) \star f(x,y)] \end{aligned}$$

where the second step follows from Problem 10.18, with $G(x) = e^{-\frac{}{2\sigma^2}}$ and $G(y) = e^{-\frac{y^2}{2\sigma^2}}$. The terms inside the two brackets are the same, so only two convolutions are required to implement them. Using the definitions in Section 10.2.1, the partials may be written as

$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$$

and

$$\frac{\partial^2 f}{\partial y^2} = f(y+1) + f(y-1) - 2f(y).$$

© 1992–2008 R. C. Gonzalez & R. E. Woods

The first term can be implemented via convolution with a 1×3 mask having coefficients coefficients, $[1 \; -2 \;\; 1]$, and the second with a 3×1 mask having the same coefficients. Letting ∇_x^2 and ∇_y^2 represent these two operator masks, we have the final result:

$$g(x,y) = \nabla_x^2 \star [G(x) \star G(y) \star f(x,y)] + \nabla_y^2 \star [G(x) \star G(y) \star f(x,y)]$$

which requires a total of four different 1-D convolution operations.

(b) If we use the algorithm as stated in the book, convolving an $M \times N$ image with an $n \times n$ mask will require $n^2 \times M \times N$ multiplications (see the solution to Problem 10.18). Then convolution with a 3×3 Laplacian mask will add another $9 \times M \times N$ multiplications for a total of $(n^2 + 9) \times M \times N$ multiplications. Decomposing a 2-D convolution into 1-D passes requires $2nMN$ multiplications, as indicated in the solution to Problem 10.18. Two more convolutions of the resulting image with the 3×1 and 1×3 derivative masks adds $3MN + 3MN = 6MN$ multiplications. The computational advantage is then

$$A = \frac{(n^2+9)MN}{2nMN + 6MN} = \frac{n^2+9}{2n+6}$$

which is independent of image size. For example, for $n = 25$, $A = 11.32$, so it takes on the order of 11 times more multiplications if direct 2-D convolution is used.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Canny Edge Detection:**

The Canny Edge Detector is one of the most commonly used image processing tools, detecting edges in a very robust manner.

- **Canny's approach is based on 3 basic objectives:**
  1. **Low error rate. The edges detected must be as close as possible to the true edge.**
  2. **Edge points should be well localized. The distance between a point marked as an edge by the detector and the center of the true edge should be minimum.**
  3. **Single edge point response. The detector should not identify multiple edge pixels where only a signal edge point exists.**

How to express the preceding 3 criteria mathematically ?

© 1992–2008 R. C. Gonzalez & R. E. Woods

Step 1: Smooth the input image with a Gaussian filter

$$G(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \quad \Rightarrow \quad f_s(x,y) = G(x,y) \star f(x,y)$$

Usually noise reduction implies some sort of blurring operation. Most people use a Gaussian filter to do this. One suggestion is to use the following 5×5 filter:

$$\frac{1}{159} * \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

Note that this 5×5 filter is roughly equivalent to the Gaussian filter, using a

.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Step 2: Compute the gradient magnitude and angle images

Compute the derivatives ($g_x(x, y)$ and $g_y(x, y)$) of the image in the x and y directions. Any of the filter mask pairs in preceding section can be used to obtain do g_x and g_y.

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

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

Step 3: Nonmaxima suppression to the gradient magnitude image

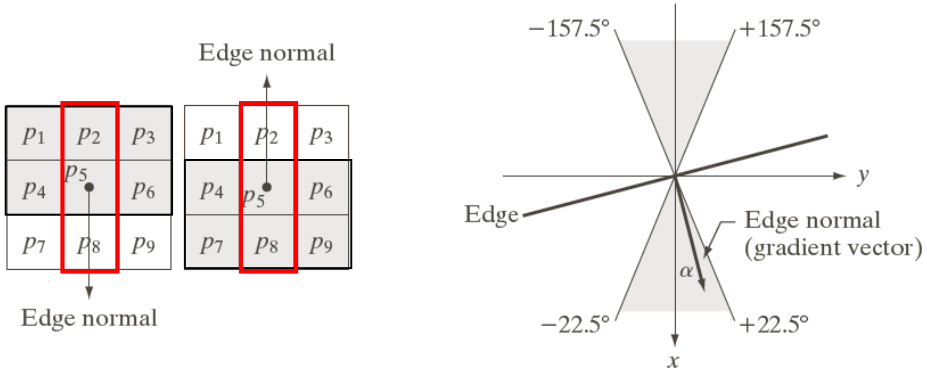
$M(x, y)$ typically contains wide ridges around local maxima. One would like to have edges that are only one pixel wide. The "nonmaxima suppression" step keeps only those pixels on an edge with the highest gradient magnitude.

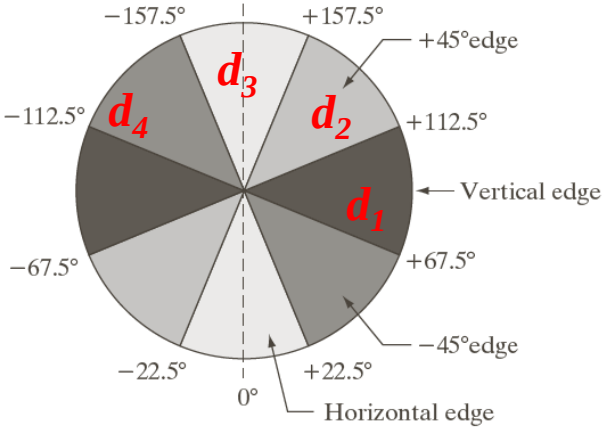
© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **For a 3 × 3 region centered at every point (x, y) in α(x, y):**

1. Find the direction d_k that is closet to $\alpha(x, y)$.
2. If the value of $M(x, y)$ is less than at least one of its two neighbors along d_k , let $g_N(x, y)=0$ (supression); otherwise, let $g_N(x, y)= M(x,y)$.

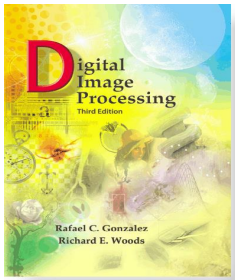




a b
c

FIGURE 10.24 (a) Two possible orientations of a horizontal edge (in gray) in a 3 × 3 neighborhood. (b) Range of values (in gray) of α, the direction angle of the *edge normal*, for a horizontal edge. (c) The angle ranges of the edge normals for the four types of edge directions in a 3 × 3 neighborhood. Each edge direction has two ranges, shown in corresponding shades of gray.

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10
Segmentation

Step 4: Double thresholding and connectivity analysis to detect and link edges

Some of the edges detected by Steps 1–3 will not actually be valid, but will just be noise. Canny's algorithm would like to filter this noise out by using ***hysteresis thresholding***.

$$g_{NH}(x,y) = g_N(x,y) \geq T_H$$
$$g_{NL}(x,y) = g_N(x,y) \geq T_L$$

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

"strong" edge pixels are assumed to be valid edge pixels

"week" edge pixels

Are they valid pixels ?

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Longer edges are formed using the following procedure:**

  a. Locate the next unvisited edge pixel, p, in $g_{NH}(x, y)$.
  b. Mark as valid edge pixels all the weak pixels in $g_{NL}(x, y)$ that connected to p.
  c. If all nonzero pixels in $g_{NH}(x, y)$ have been visited go to Step d. Else, return to Step a.
  d. Set to zero all pixels in $g_{NL}(x, y)$ that were not marked as valid edge pixels.

Canny algorithm is formed by appending to $g_{NH}(x, y)$ all the nonzero pixels from $g_{NL}(x, y)$.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Example of Canny edge detector**

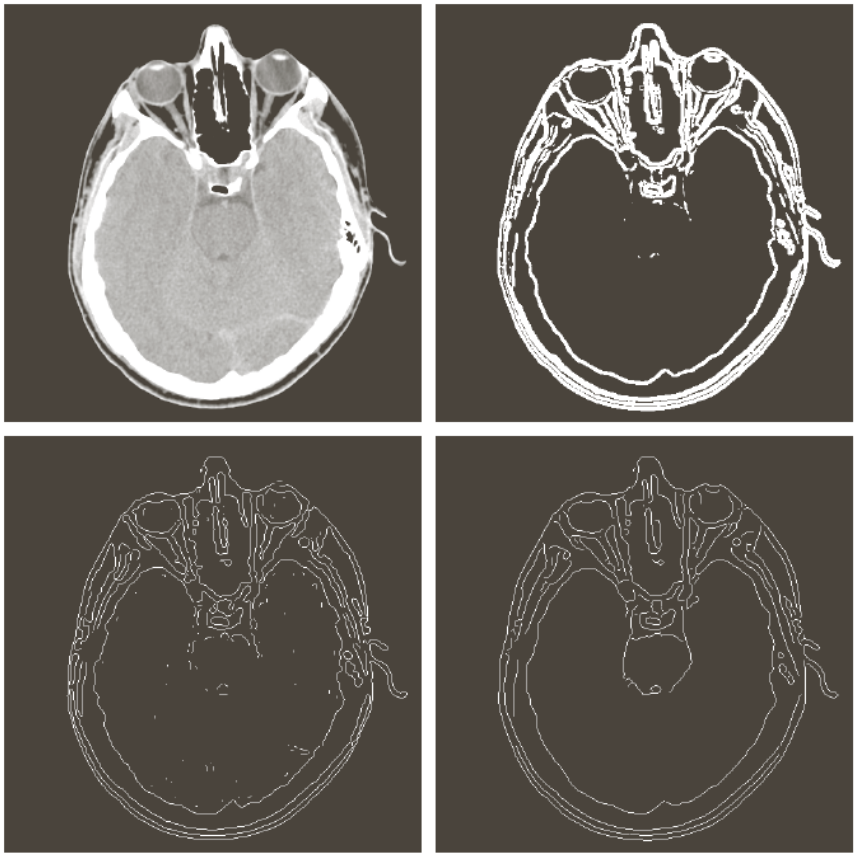
a b
c d

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.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

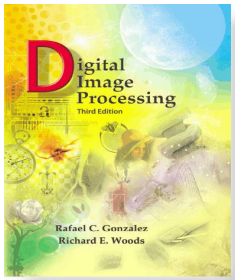
- **Example of Canny edge detector**



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.)

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10
Segmentation

- **Edge Linking and Boundary Detection**
- **Local Processing**
  - The strength of the response of the gradient operator used to produce the edge pixel.
  - The direction of the gradient vector.
  - Let S_{xy} denote the set of coordinates of a neighborhood centered at point (x, y) in an image.

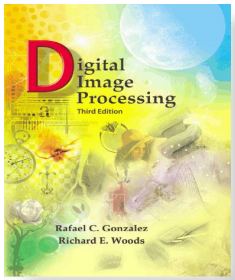
$$\left|M(s,t)-M(x,y)\right| \le E$$

$$\left|\alpha(s,t)-\alpha(x,y)\right| \le A$$

A pixel with coordinates (s, t) in S_{xy} is linked to the pixel at (x, y) if both magnitude and direction criterion are satisfied.

Computationally expensive !

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10
Segmentation

- **A simplification for real-time application consists the following steps:**

1. Compute the gradient magnitude and angle arrays, $M(x, y)$ and $\alpha(x, y)$, of the input image $f(x, y)$.
2. Form a binary image, g, whose value at any pair of coordinates (x, y) is given by:

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

3. Scan the rows of g and fill all gaps in each row that do not exceed a specified length, K.
4. Detect gaps in any other direction, θ, rotate g by this angle and apply the horizontal scanning procedure in Step 3. Rotate the result back by -θ.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

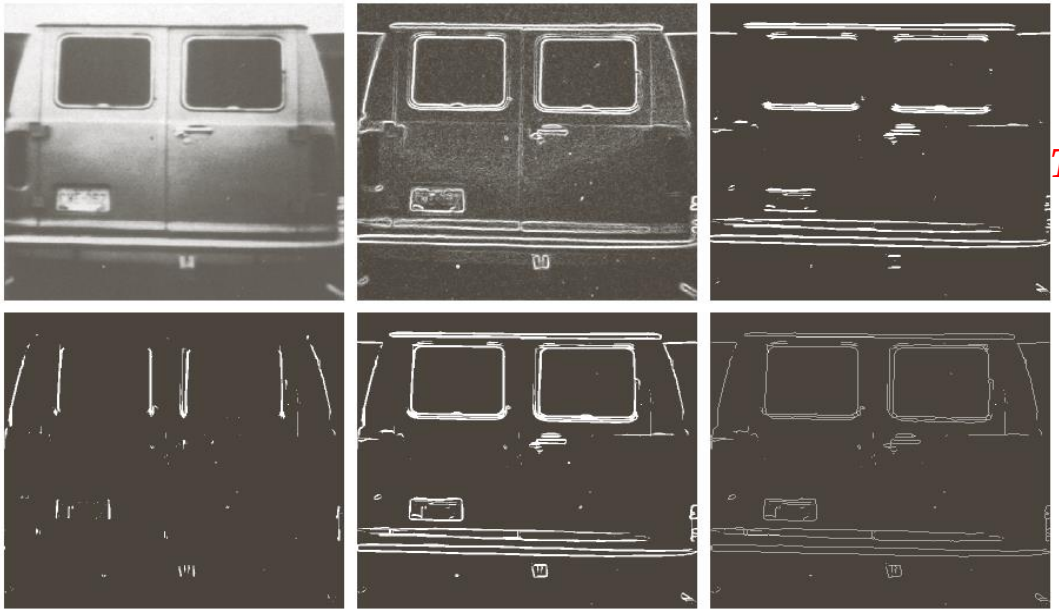
- Example of edge-point linking based on local processing.

→ *Detecting strong horizontal and vertical edges.*

Step 3

T_M =30% of maximum gradient value , $A=90^0$, $T_A= 45^0$

Step 4



a b c
d e f

Morphological thinning

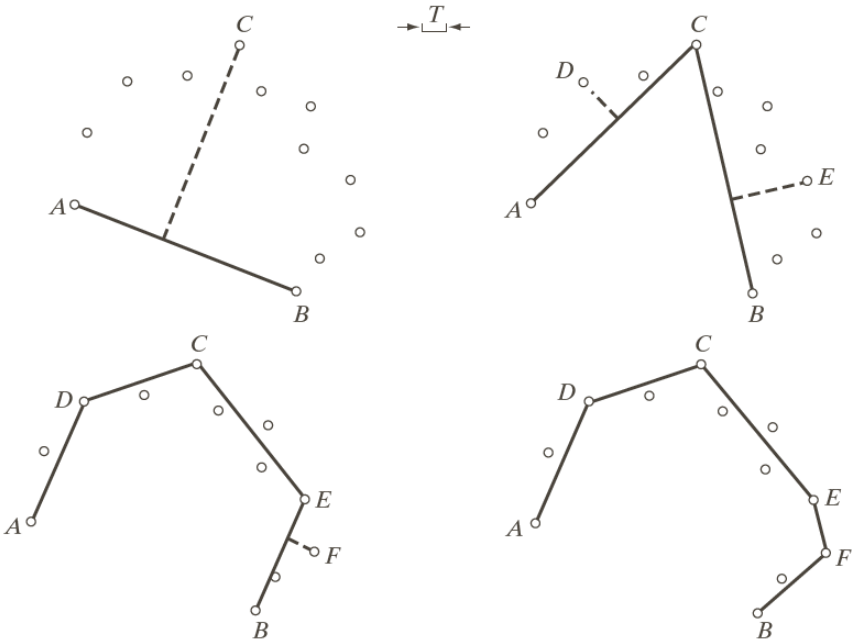
FIGURE 10.27 (a) A 534 × 566 image of the rear of a vehicle. (b) Gradient magnitude image. (c) Horizontally connected edge pixels. (d) Vertically connected edge pixels. (e) The logical OR of the two preceding images. (f) Final result obtained using morphological thinning. (Original image courtesy of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

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Regional Processing**
  - Approximate a 2-D curve to the known points.
  - Polygonal approximations are attractive because they can capture the essential shape feature of a region.

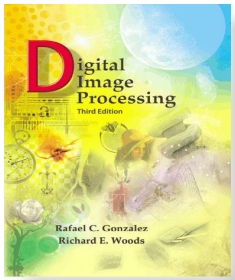
Select the point yielded the maximum distance



a b
c d

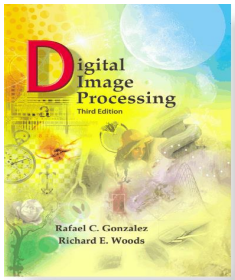
FIGURE 10.28 Illustration of the iterative polygonal fit algorithm.

© 1992–2008 R. C. Gonzalez & R. E. Woods



- **An algorithm for finding a polygon fit to open and closed curves may be stated as follows:**
  1. Let P be a sequence of ordered, distinct, 1-valued points of a boundary image. Specify 2 starting points, A and B.
  2. Specify a threshold, ***T***, and 2 empty stacks, OPEN and CLOSED.
  3. If the points in P correspond to a closed curve, put A into OPEN and put B into OPEN and into CLOSED. If the points correspond to an open curve, put A into OPEN and B into CLOSED
  4. Compute the parameters of the line passing from the last vertex in CLOSED to the last vertex in OPEN.
  5. Compute the distance from the line in Step 4 to all the points in P whose sequence places them between the vertices from Step 4. Select the point, V_{max}, with the maximum diatance, D_{max}.
  6. If $D_{max} > T$, place V_{max} at the end of the OPEN stack as a new vertex. Go to Step 4.
  7. Else, remove the last vertex from OPEN and insert it as the last vertex of CLOSED.
  8. If OPEN is not empty, go to Step 4.
  9. Else, exit. The vertices in CLOSED are the vertices of the polygon fit to the points in P.

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10
Segmentation

CLOSED	OPEN	Curve segment processed	Vertex generated
B	B, A	—	A, B
B	B, A	(BA)	C
B	B, A, C	(BC)	—
B, C	B, A	(CA)	—
B, C, A	B	(AB)	D
B, C, A	B, D	(AD)	—
B, C, A, D	B	(DB)	—
B, C, A, D, B	Empty	—	—

TABLE 10.1 Step-by-step details of the mechanics in Example 10.11.

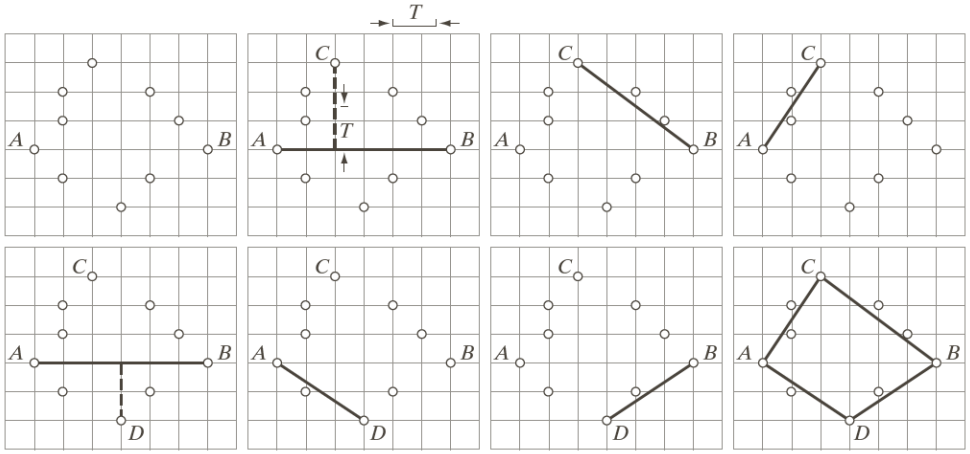
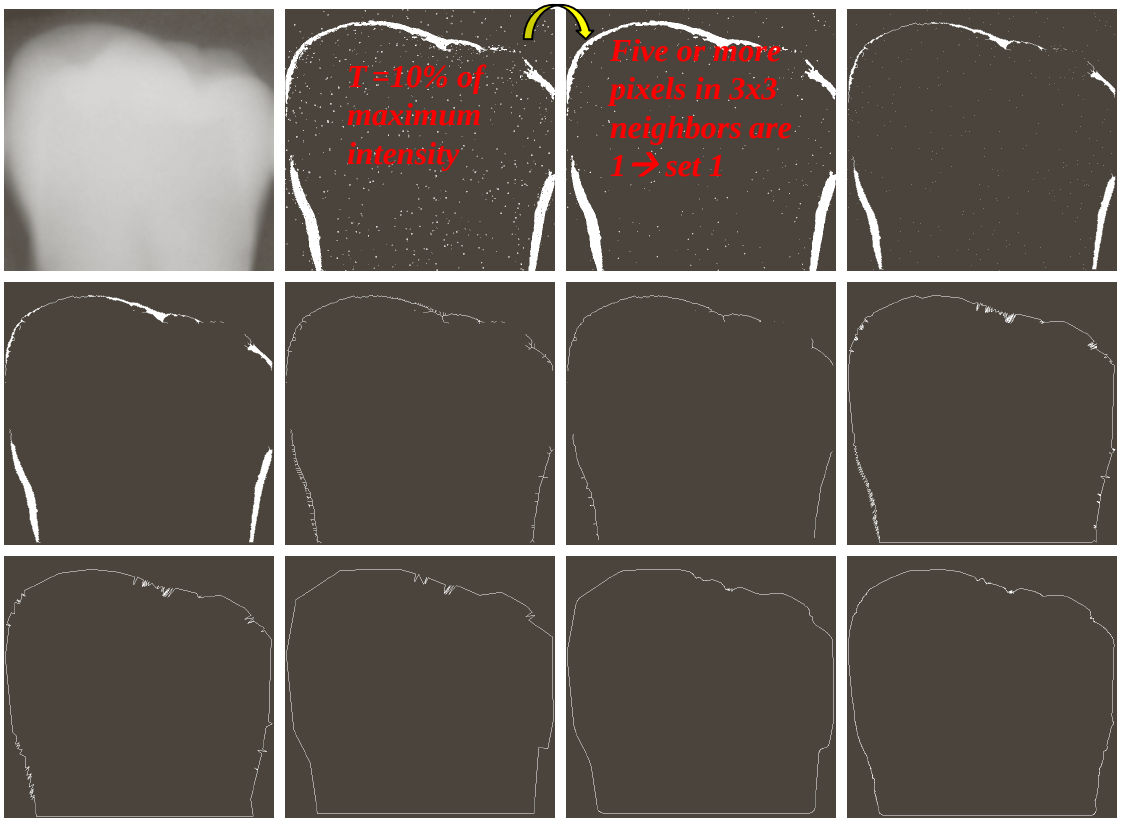


FIGURE 10.29 (a) A set of points in a clockwise path (the points labeled A and B were chosen as the starting vertices). (b) The distance from point C to the line passing through A and B is the largest of all the points between A and B and also passed the threshold test, so C is a new vertex. (d)–(g) Various stages of the algorithm. (h) The final vertices, shown connected with straight lines to form a polygon. Table 10.1 shows step-by-step details.

© 1992–2008 R. C. Gonzalez & R. E. Woods

- **Example of polygon fitting of an image boundary.**



a b c d
e f g h
i j k l

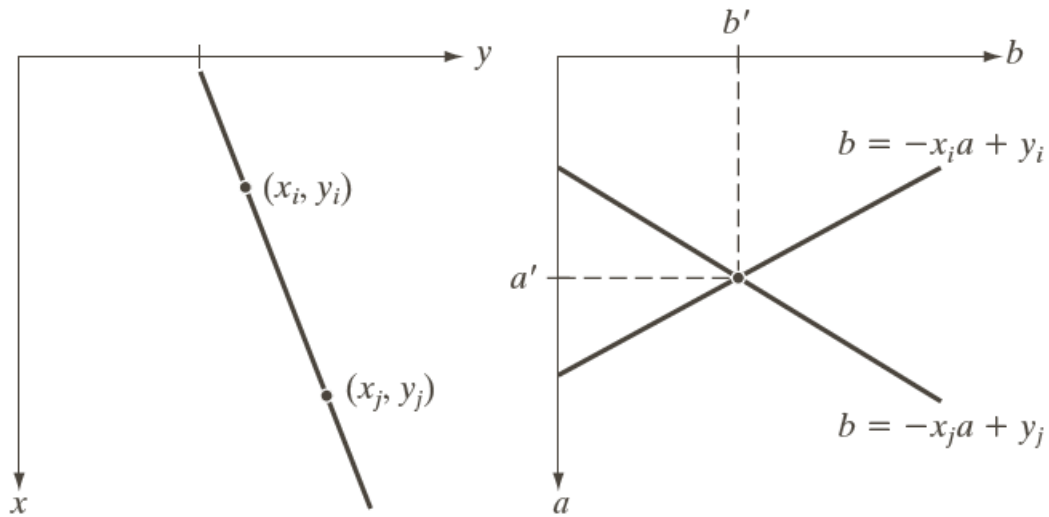
FIGURE 10.30 (a) A 550 × 566 X-ray image of a human tooth. (b) Gradient image. (c) Result of majority filtering. (d) Result of morphological shrinking. (e) Result of morphological cleaning. (f) Skeleton. (g) Spur reduction. (h)–(j) Polygonal fit using thresholds of approximately 0.5%, 1%, and 2% of image width ($T = 3$, 6, and 12). (k) Boundary in (j) smoothed with a 1-D averaging filter of size 1 × 31 (approximately 5% of image width). (l) Boundary in (h) smoothed with the same filter.

© 1992–2008 R. C. Gonzalez & R. E. W

Chapter 10
Segmentation

- **Global Processing using the Hough Transform**
  - xy-plane and parameter space

$$y_i = ax_i + b \quad \Longleftrightarrow \quad b = -x_i a + y_i$$



a b

FIGURE 10.31 (a) *xy*-plane. (b) Parameter space.

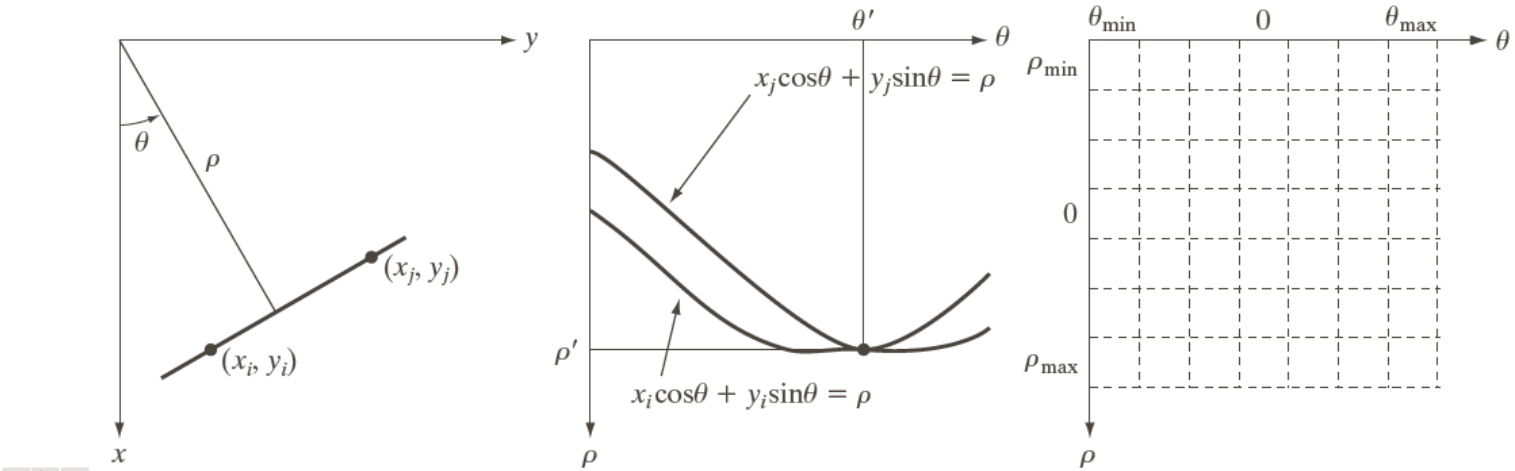
© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Global Processing using the Hough Transform**
  - Normal representation of a line

$$x\cos\theta + y\sin\theta = \rho$$

$$-90^o \le \theta \le 90^o, \quad -D \le \rho \le D$$



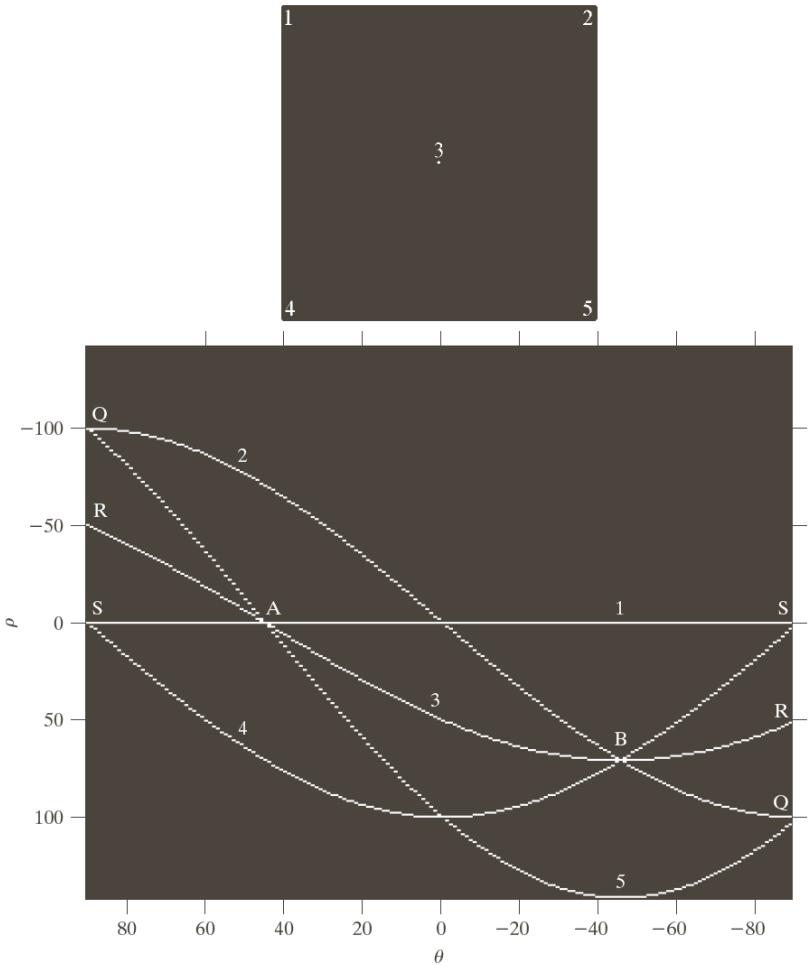
a b c

FIGURE 10.32 (a) (ρ, θ) parameterization of line in the xy-plane. (b) Sinusoidal curves in the $\rho\theta$-plane; the point of intersection (ρ', θ') corresponds to the line passing through points (x_i, y_i) and (x_j, y_j) in the xy-plane. (c) Division of the $\rho\theta$-plane into accumulator cells.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

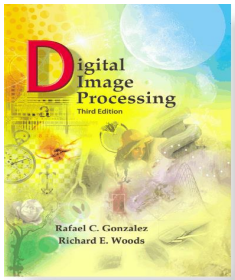
- Illustration of the Hough transform.



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.)

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10
Segmentation

- **Global Processing using the Hough Transform**
  - Hough transformation is applicable to any function of the form g(**v**,**c**)=0, where **v** is a vector of coordinates and **c** is a vector of coefficients. For example, the normal representation of a circle

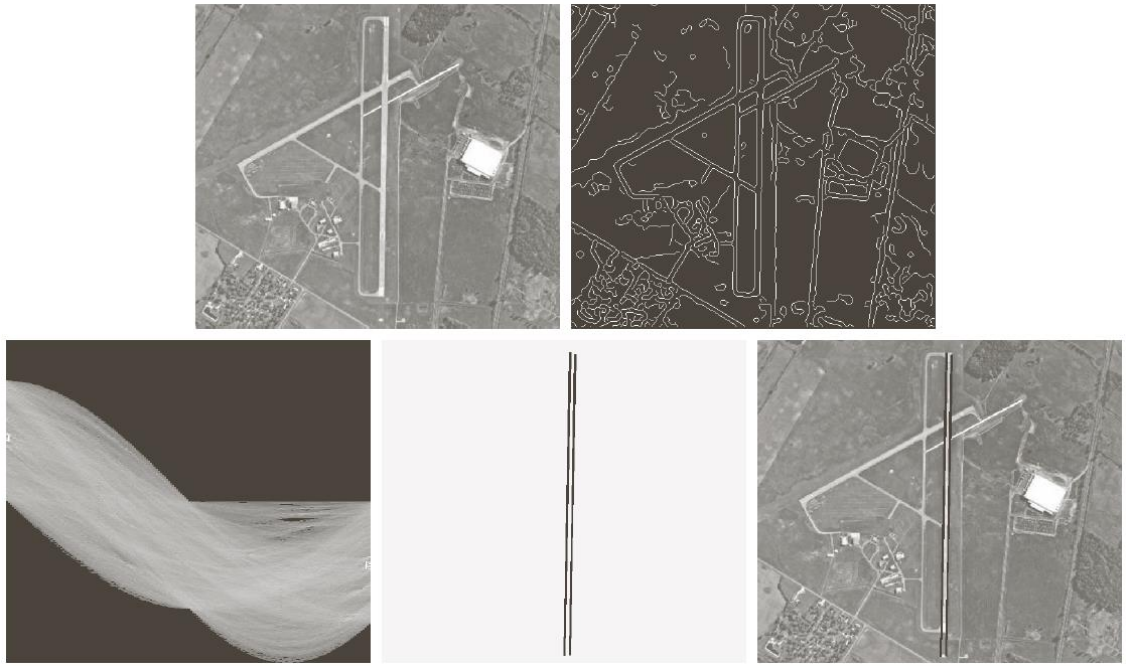
$$(x-c_1)^2+(y-c_2)^2=c_3^2$$

- **Edge-linking based on the Hough transform:**
  1. Obtain a binary image using any of the techniques discussed earlier in this section.
  2. Specify subdivision in the ρθ-plane.
  3. Examine the counts of the accumulator cells for high pixel concentrations.
  4. Examine the relationship (principally for continuity) between pixels in a chosen cell.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

•Example of Global Processing using the Hough Transform

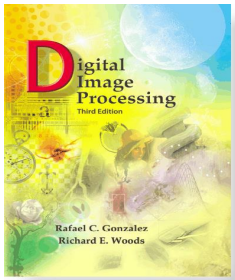


• Joining all gaps not exceeding 20% of the image height (approximately 100 pixels)

a b
c d e

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.

© 1992–2008 R. C. Gonzalez & R. E. Woods



10.3 Thresholding

- **Foundation**

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

$$\text{where } f(x, y) \text{ is the gray level of point } (x, y)$$

$$p(x, y) \text{ denotes some local property of this point}$$

$$T = T[f(x, y)] \Rightarrow \text{global thresholding}$$

$$\boxed{\begin{aligned} &T = T[p(x, y), f(x, y)] \Rightarrow \text{local thresholding} \\ &T = T[x, y] \Rightarrow \text{dynamic or adaptive thresholding} \end{aligned}}$$

variable thresholding

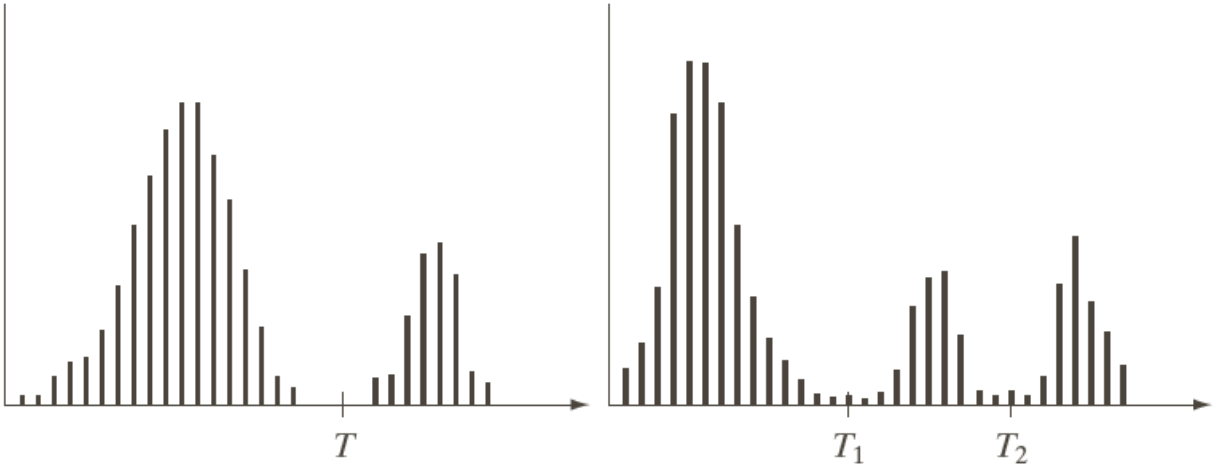
© 1992–2008 R. C. Gonzalez & R. E. Woods

10.3 Thresholding

- **A threshold image g(x,y) is define as**

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

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



a b

FIGURE 10.35 Intensity histograms that can be partitioned (a) by a single threshold, and (b) by dual thresholds.

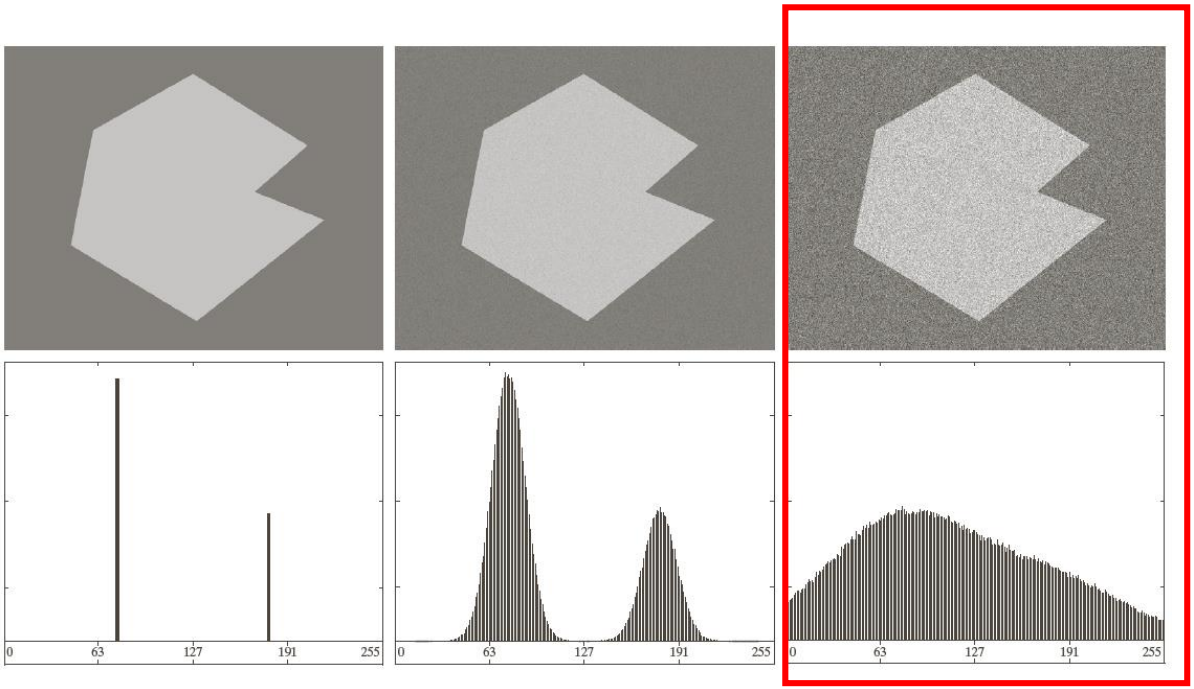
© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **The role of noise in image thresholding**

How to find the suitable threshold ?

It's difficult by observation.



a b c
d e f

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.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **The role of illumination and refectance**

$$f(x, y) = i(x, y)\, r(x, y)$$

$$\begin{aligned} z(x, y) &= \ln f(x, y) \\ &= \ln i(x, y) + \ln r(x, y) \\ &= i'(x, y) + r'(x, y) \end{aligned}$$

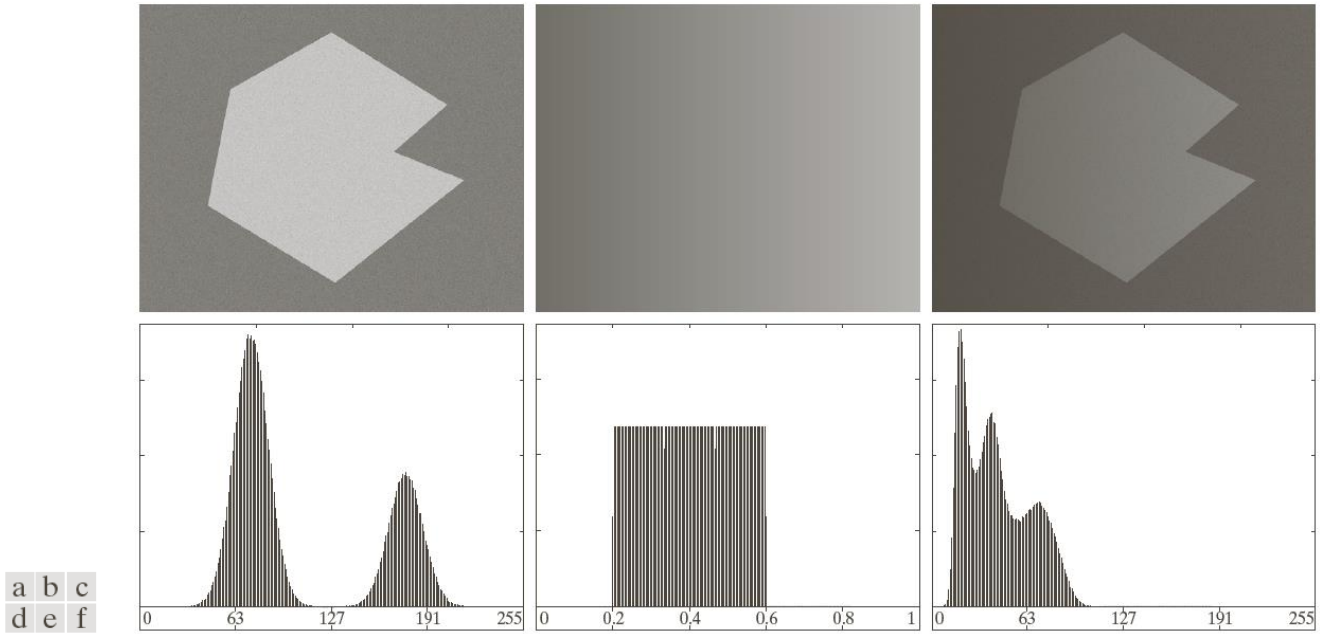
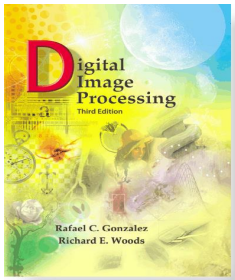


FIGURE 10.37 (a) Noisy image. (b) Intensity ramp in the range [0.2, 0.6]. (c) Product of (a) and (b). (d)–(f) Corresponding histograms.

© 1992–2008 R. C. Gonzalez & R. E. Woods



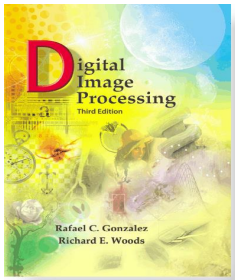
- **The Role of Illumination**
  - When access to the illumination source is available, a solution frequently used in practice to compensate for nonuniformity is to project the illumination pattern onto a constant, white reflective surface.

1. *Yield an image* $g(x,y)=k\, i(x, y)$
2. *For any image* $f(x, y) = i(x, y)\, r(x, y)$ *obtain with the same illumination*
3. *Normalized function*

$$h(x,y)=f(x,y)/g(x,y)=r(x,y)/k$$

4. *if* $r(x,y)$ *can be segmented by using a single threshold* T, *then* $h(x,y)$ *can be segmented by using a single threshold* T/k

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10
Segmentation

- **Basic global thresholding → algorithm to obtain T automatically:**
  1. Select an initial estimate for the global threshold *T*.
  2. Segment the image using T. This will produce two groups of pixels: G_1 consisting of all pixels with gray level values > *T* and G_2 consisting of all pixels with gray level values ≤ *T*.
  3. Compute the average gray level values m_1 and m_2 for the pixels in regions G_1 and G_2 .
  4. Compute a new threshold value

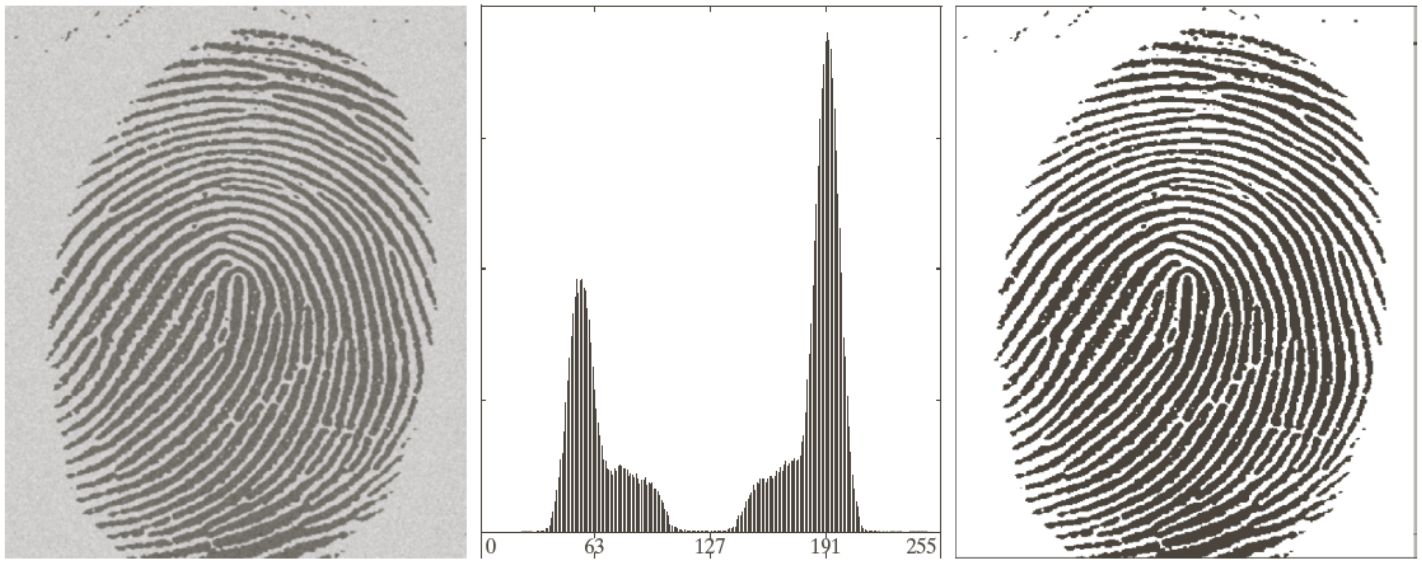
     $$T = 1/2\,(m_1 + m_2)$$

  5. Repeat steps 2 through 4 until the difference between values of T in successive iterations is smaller than a predefined parameter ΔT.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Example: Image segmentation using an estimated global threshold**



a b c

FIGURE 10.38 (a) Noisy fingerprint. (b) Histogram. (c) Segmented result using a global threshold (the border was added for clarity). (Original courtesy of the National Institute of Standards and Technology.)

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Optimal Global Thresholding using Otsu's method**
  - Minimize the average error incurred in assigning pixels to 2 or more groups.
  - Bays decision rule → Otsu's method.

Use $T(k)$ to threshold the input image into 2 classes, C_1 and C_2.
C_1: pixels with intensity values in the range [0, k].
C_2: pixels with intensity values in the range [k+1, L-1].

$$P_1(k) = \sum_{i=0}^{k} p_i$$

$$P_2(k) = \sum_{i=k+1}^{L-1} p_i = 1 - P_1(k)$$

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

$$m_1(k) = \sum_{i=0}^{k} iP(i/C_1) \quad \Rightarrow \quad P(A/B) = P(B/A)P(A)/P(B)$$

$$= \sum_{i=0}^{k} i\,\boxed{P(C_1/i)}\,P(i)/\boxed{P(C_1)} = P_1(k)$$

$$= \frac{1}{P_1(k)} \sum_{i=0}^{k} ip_i$$

=1 : we are dealing only with values of i from class C_1

Similarly

$$m_2(k) = \sum_{i=k+1}^{L-1} iP(i/C_2)$$

$$= \frac{1}{P_2(k)} \sum_{i=k+!}^{L-1} ip_i$$

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

Cumulative mean up to level ***k***

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

Average intensity of the entire image (global mean)

$$m_G(k) = \sum_{i=0}^{L-1} i p_i$$

$$\Rightarrow \quad P_1 \mathrm{m}_1 + P_2 \mathrm{m}_2 = m_G, \quad P_1 + P_2 = 1$$

In order to evaluate the "goodness" of the threshold at level k we use the normalized, dimensionless metric

$$\eta = \frac{\sigma_B^2}{\sigma_G^2}$$

$$\sigma_B^2 = P_1 (m_1 - m_G)^2 + P_2 (m_2 - m_G)^2 \quad \textit{between-class variance}$$

$$\sigma_G^2 = \sum_{i=0}^{L-1} (i - m_G)^2 p_i \quad \textit{global variance}$$

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

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

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

This form is slightly more efficient computationally.

How to determine the threshold value, ***k***, that maximizes the between-class variance ?

$$\sigma_B^2(k) = \frac{\left[m_G P_1(k) - m(k)\right]^2}{P_1(k)\left[1 - P_1(k)\right]} \quad \Rightarrow \quad \sigma_B^2\left(k^*\right) = \max_{0 \le k \le L-1} \sigma_B^2(k)$$

Once k^* has been obtained, the input image ***f(x, y)*** is segmented as

$$g(x,y) = \begin{cases} 1 & if \;\; f(x,y) > k^* \\ 0 & if \;\; f(x,y) \le k^* \end{cases}$$

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

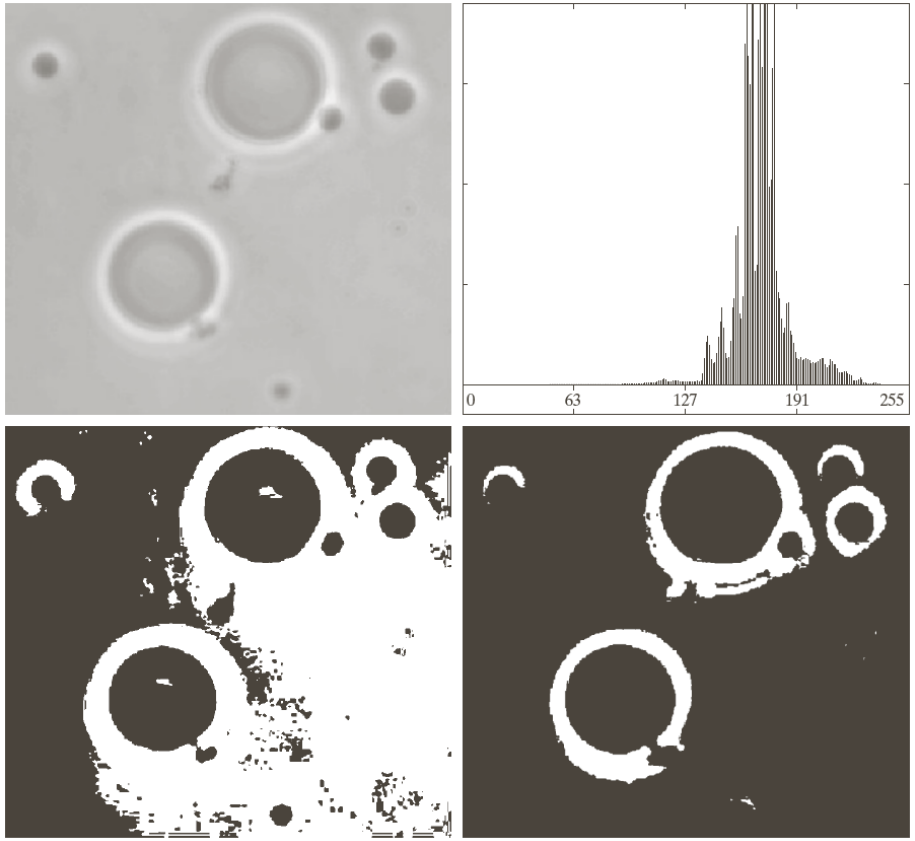
- **Otsu's algorithm may be summarized as follows:**
  1. Compute the normalized histogram of the input image. Denote the components of the histogram by $\boldsymbol{p_i}$, i=0, 1, 2, …, L-1.
  2. Compute the cumulative sums, $\boldsymbol{P_1(k)}$, for k=0, 1, 2, …, L-1.
  3. Compute the cumulative means, $\boldsymbol{m(k)}$, for k=0, 1, 2, …, L-1.
  4. Compute the global intensity mean, $\boldsymbol{m_G}$, for k=0, 1, 2, …, L-1.
  5. Compute the between-class variance, σ_B^2 , for k=0, 1, 2, …, L-1.
  6. Obtain the Otsu threshold, $\boldsymbol{k}^*$, as the value of k for which σ_B^2 is maximum. If the maximum is not unique, obtain $\boldsymbol{k}^*$ by averaging the value of $\boldsymbol{k}$ corresponding to the various maxima detected.
  7. Obtain the separability measure η^* at $\boldsymbol{k}=\boldsymbol{k}^*$.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Example: Optimal global thresholding using Otsu's method**

T=169



T=181

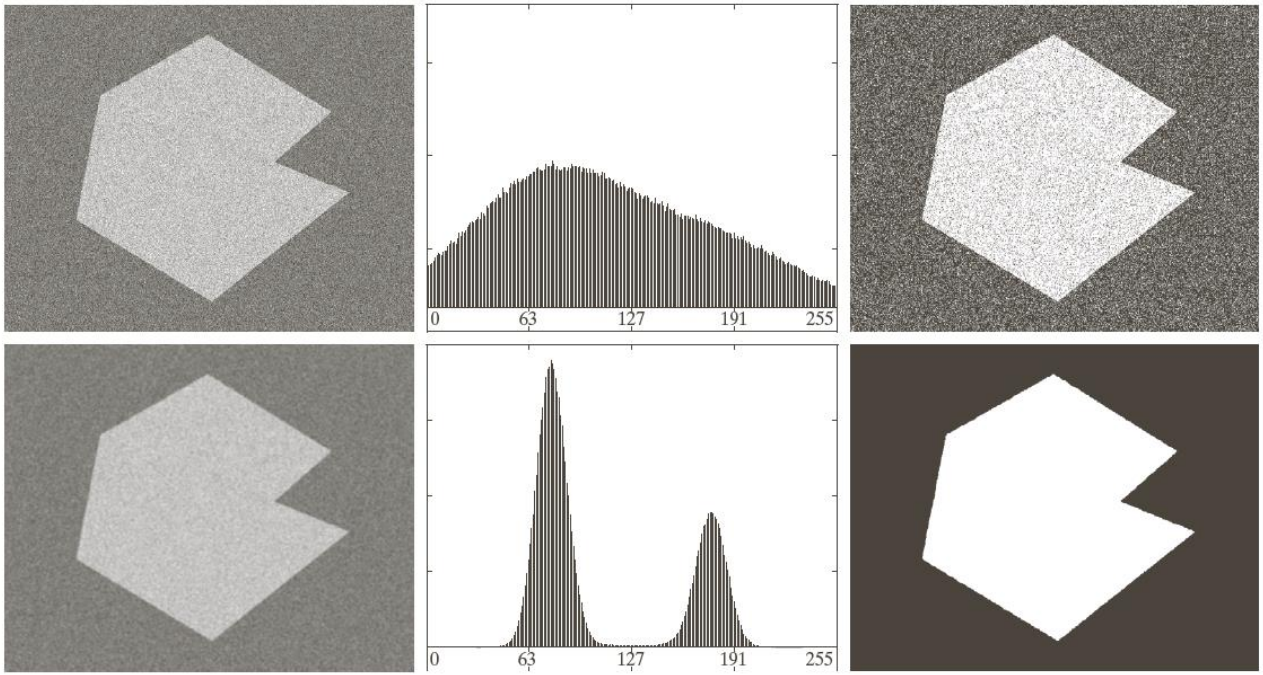
a b
c d

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.)

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

• Using Image Smoothing to Improve Global Thresholding



a b c
d e f

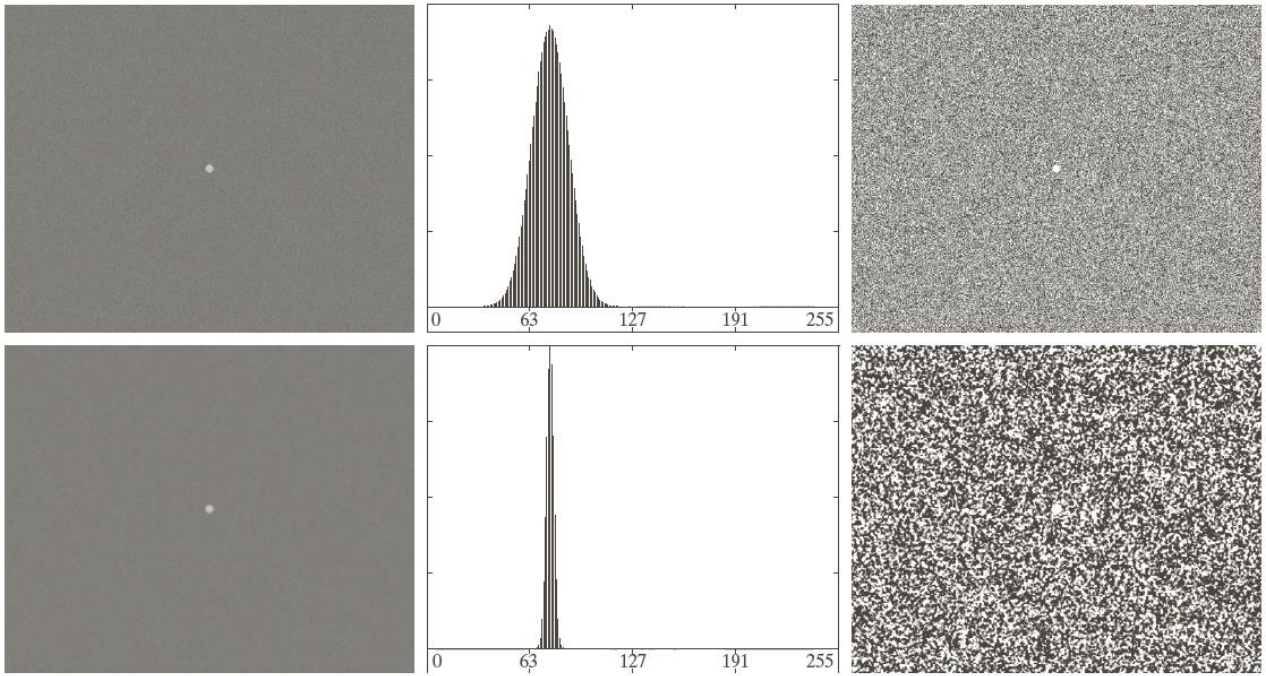
FIGURE 10.40 (a) Noisy image from Fig. 10.36 and (b) its histogram. (c) Result obtained using Otsu's method. (d) Noisy image smoothed using a 5 × 5 averaging mask and (e) its histogram. (f) Result of thresholding using Otsu's method.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

Using Image Smoothing to Improve Global Thresholding

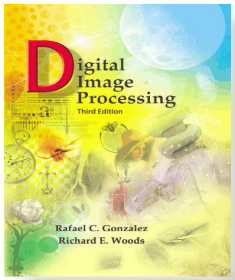
→ **Fail !**



a b c
d e f

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.

© 1992–2008 R. C. Gonzalez & R. E. Woods



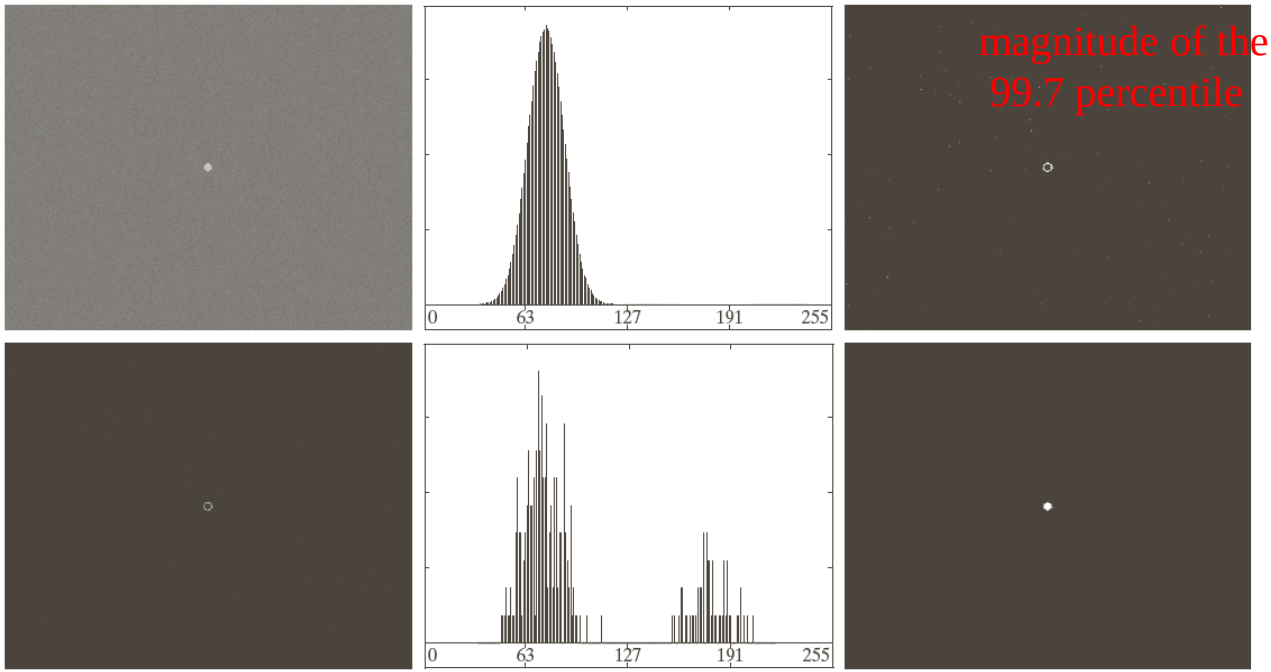
Chapter 10
Segmentation

- **Using Edges to Improve Global Thresholding :**
  1. Compute an edge image as either the magnitude of the gradient, or absolute value of the Laplacian, of $f(x, y)$.
  2. Specify a threshold value, T.
  3. Threshold the image from Step 1 using the threshold from Step 2 to produce a binary image, $g_T(x, y)$. This image is used as mask image in the following step to select pixels from $f(x, y)$ corresponding to "strong " edge pixels.
  4. Compute a histogram using only the pixels in $f(x, y)$ that correspond to the locations of the 1-valued pixels in $g_T(x, y)$.
  5. Use the histogram from Step 4 to segment $f(x, y)$ globally using, for example, Otsu's method.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

• Example 1: Using Edges to Improve Global Thresholding

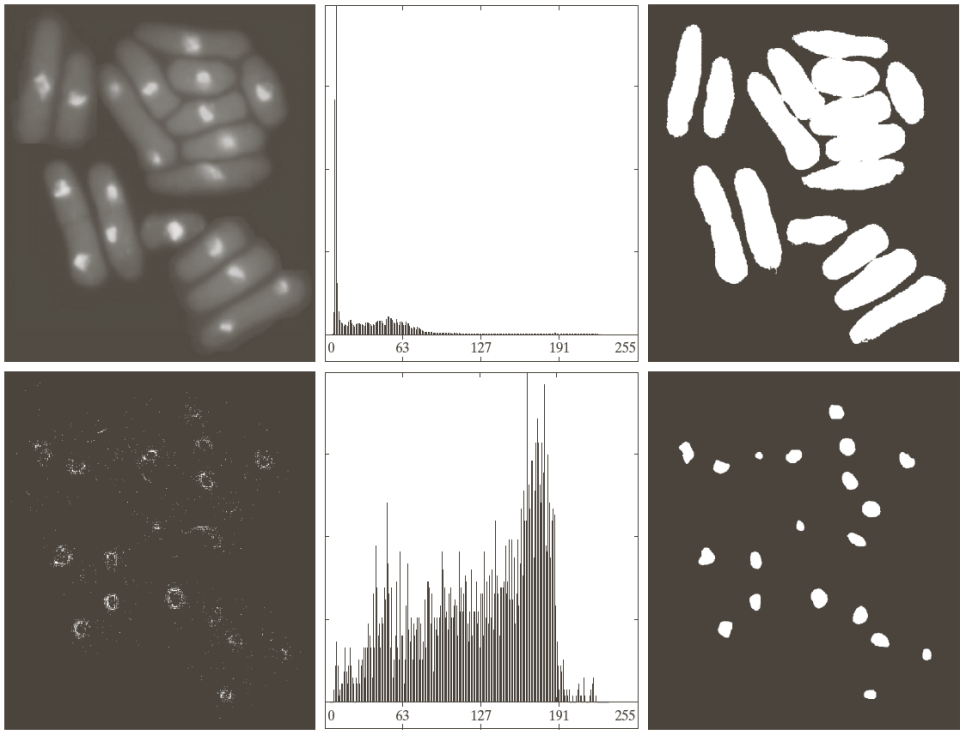


a b c
d e f

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.

© 1992–2008 R. C. Gonzalez & R. E. Woods

- **Example 2: Using Edges to Improve Global Thresholding**



Absolute value of the Laplacian at the 53.9 percentile

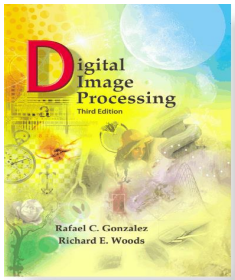
Absolute value of the Laplacian at the 99.5 percentile

a b c
d e f

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.)

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.

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10
Segmentation

- **Multiple Thresholds**
  - In this case of ***K*** classes, $C_1, C_2, \ldots, C_K$, the between-class variance generalized as

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

where

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

How to determine the threshold values to maximizes the between-class variance ?

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

© 1992–2008 R. C. Gonzalez & R. E. Woods

For 3 classes, the between-class variance is given by

$$\sigma_B^2 = P_1\left(m_1 - m_G\right)^2 + P_2\left(m_2 - m_G\right)^2 + P_3\left(m_3 - m_G\right)^2$$

where

$$P_1 = \sum_{i=0}^{k_1} p_i \qquad m_1 = \frac{1}{P_1}\sum_{i=0}^{k_1} p_i$$

$$P_2 = \sum_{i=k_1+1}^{k_2} p_i \qquad m_2 = \frac{1}{P_2}\sum_{i=k_1+1}^{k_2} p_i$$

$$P_3 = \sum_{i=k_2+1}^{L-1} p_i \qquad m_3 = \frac{1}{P_3}\sum_{i=k_2+1}^{L-1} p_i$$

and

$$\Rightarrow \quad P_1 \mathrm{m}_1 + P_2 \mathrm{m}_2 + P_3 \mathrm{m}_3 = m_G, \quad P_1 + P_2 + P_3 = 1$$

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

Find the optimum thresholds by finding

$$\sigma_B^2\left(k_1^*, k_2^*\right) = \max_{0<k_1<k_2<L-1} \sigma_B^2\left(k_1, k_2\right)$$

$$\Rightarrow \quad g(x,y) = \begin{cases} a & \text{if } f(x,y) \le k_1^* \\ b & \text{if } k_1^* < f(x,y) \le k_2^* \\ c & \text{if } f(x,y) > k_2^* \end{cases}$$

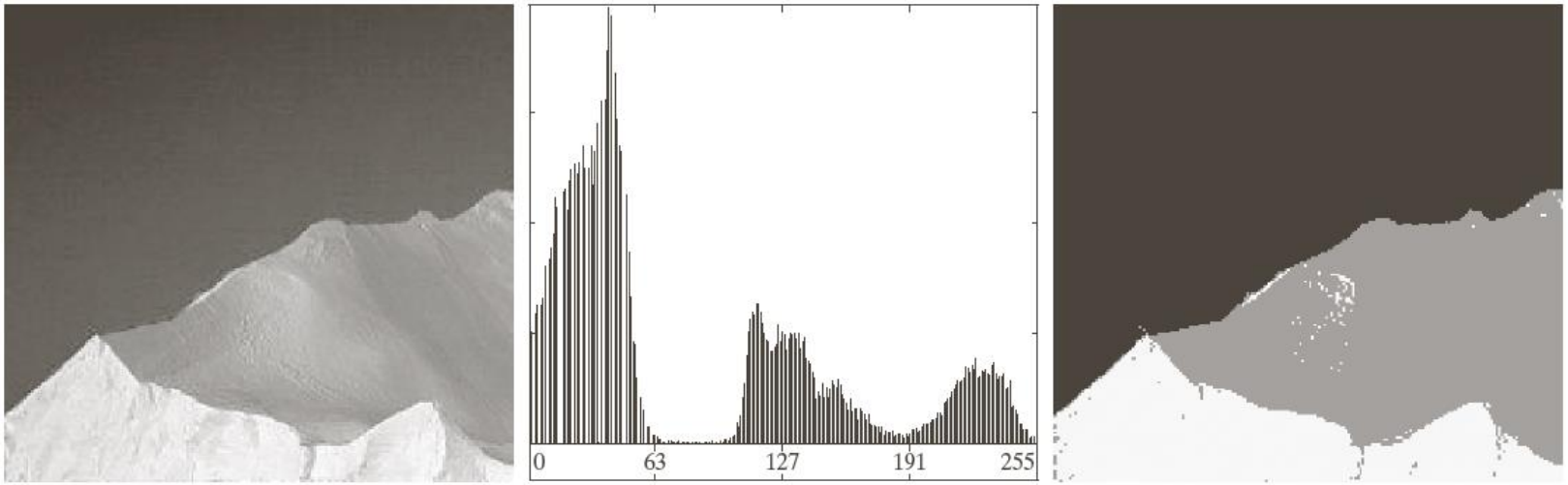
Obtain the separability measure

$$\eta\left(k_1^*, k_2^*\right) = \frac{\sigma_B^2\left(k_1^*, k_2^*\right)}{\sigma_G^2}$$

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Example of multiple global thresholding**



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.)

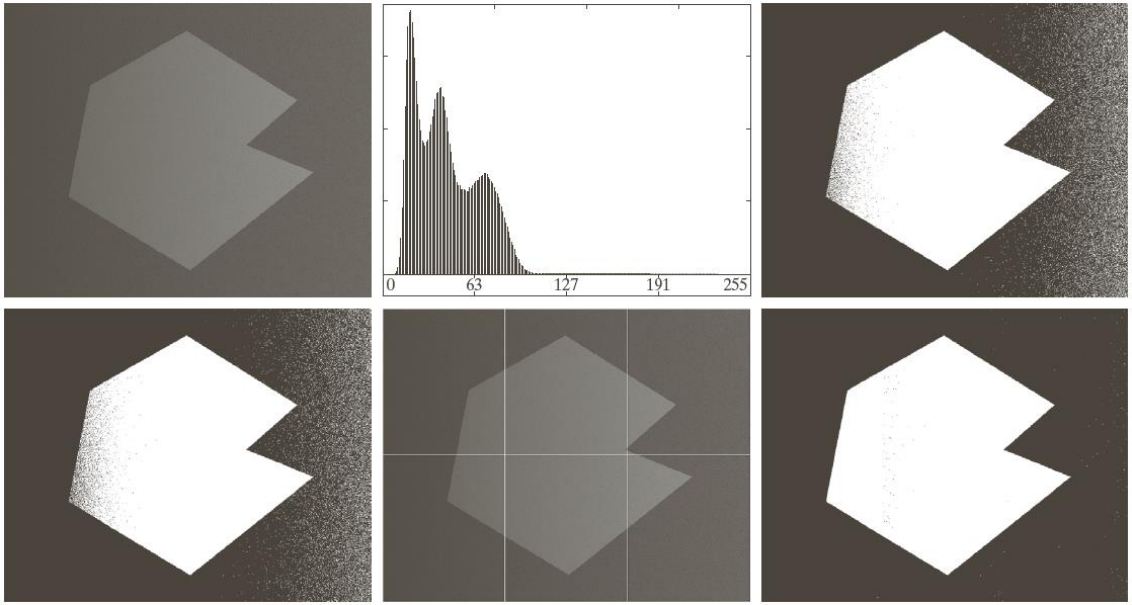
© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Variable thresholding**
  - Image partitioning

Iterative global threshold

Otsu's global threshold



a b c
d e f

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.

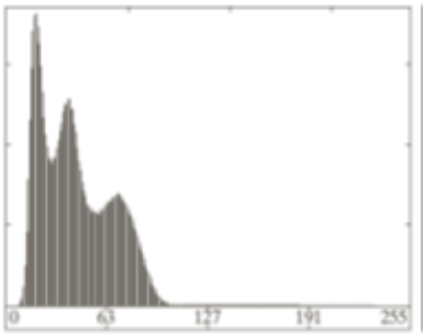
© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Variable thresholding**
  - Image partitioning

subimage histogram

global histogram



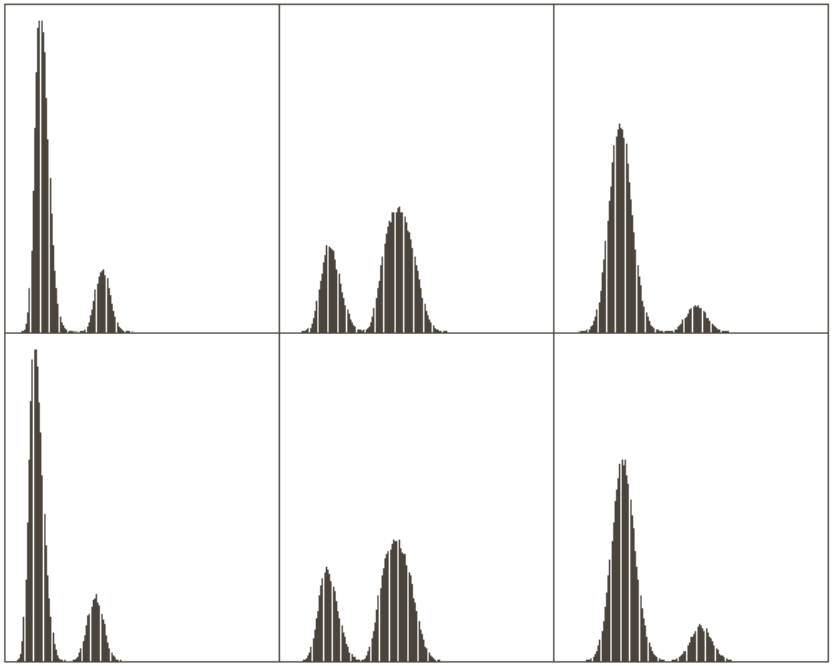


FIGURE 10.47 Histograms of the six subimages in Fig. 10.46(e).

© 1992–2008 R. C. Gonzalez & R. E. Woods

- **Variable thresholding**
  - Local image properties

$$T_{xy} = a\sigma_{xy} + bm_{xy} \longrightarrow$$ ***Mean value of the set of pixels contained in S_{xy}***

$$T_{xy} = a\sigma_{xy} + bm_G \longrightarrow$$ ***Global image mean value***

Standard deviation of the set of pixels contained in S_{xy}

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

$$g(x,y) = \begin{cases} 1 & if \;\; Q(local \;\; parameters) \; is \; true \\ 0 & iif \;\; Q(local \;\; parameters) \; is \;\; false \end{cases}$$

$$g(x,y) = \begin{cases} 1 & if \;\; f(x,y) > a\sigma_{xy} \;\; AND \;\; f(x,y) > bm_{xy} \\ 0 & otherwise \end{cases}$$

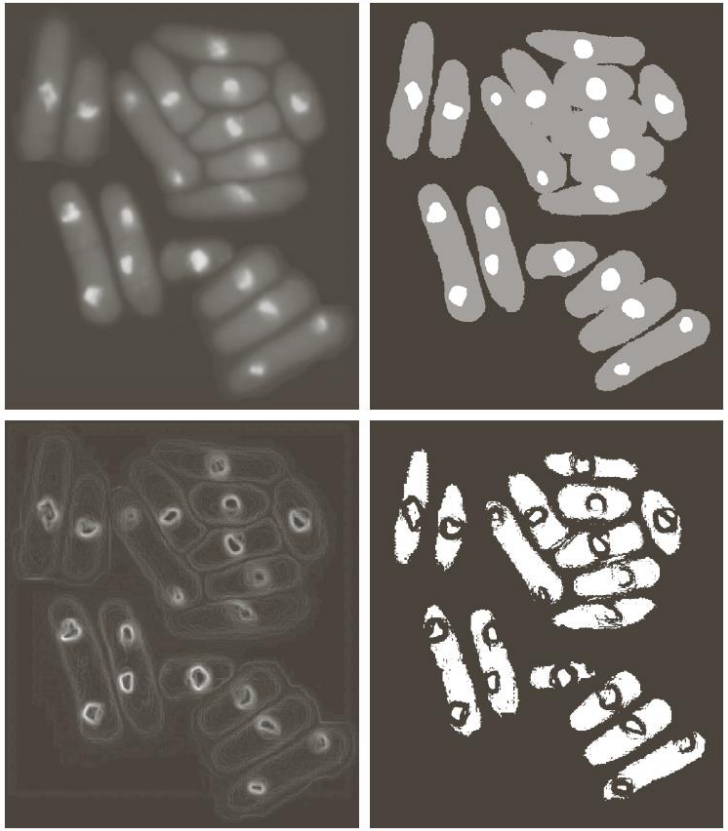
© 1992–2008 R. C. Gonzalez & R. E. Woods

- **Example of Variable thresholding based on local image properties**

$$g(x,y)=\begin{cases}1 & \text{if } f(x,y)>a\sigma_{xy} \text{ AND } f(x,y)>bm_G \\ 0 & \text{otherwise}\end{cases}$$

3x3 window ***a= 30, b=1.5***

Image of σ_{xy}



a b
c d

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.

© 1992–2008 R. C. Gonzalez & R. E. Woods

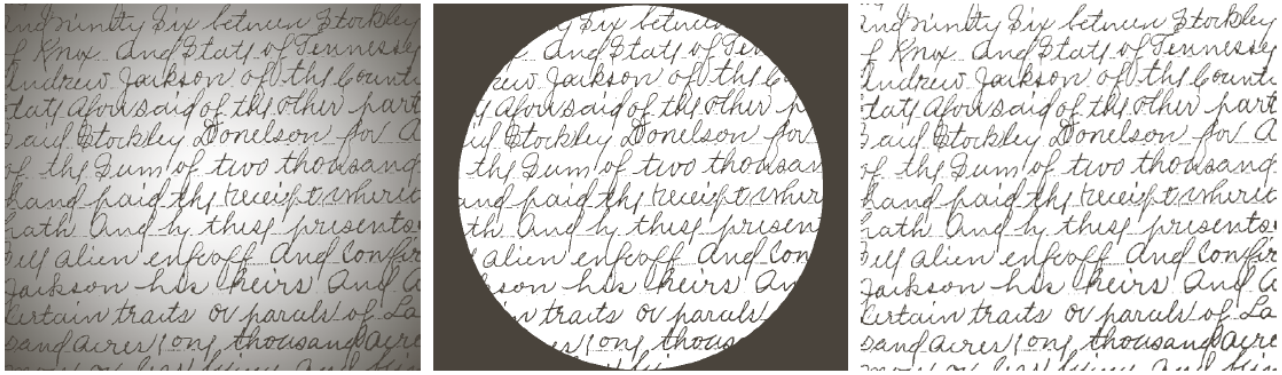
Chapter 10
Segmentation

- **Variable thresholding**
  - Moving average

$$m(k+1) = \frac{1}{n}\sum_{i=k+2-n}^{k+1} z_i = m(k) + \frac{1}{n}\left(z_{k+1} - z_{k-n}\right)$$

$$T_{xy} = b\,m_{xy}$$

- **Example 1: Variable thresholding using moving average**



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.

n=20

b=0.5

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Example 2: Variable thresholding using moving average**

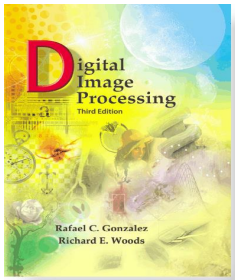
a b c

***n*=20**

***b*=0.5**

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.

© 1992–2008 R. C. Gonzalez & R. E. Woods



10.4 Region-Based Segmentation

- **Basic Formulation (*Remind the basic concept mentioned in 10.1*)**
  - Partition the entire image R into n subimages $R_1, R_2, \ldots, R_n$, such taht

(a) $\bigcup_{i=1}^{n} R_i = R.$

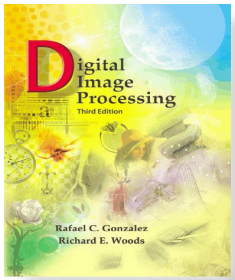
(b) R_i is a connected region, $i = 1, 2, \ldots, n.$

(c) $R_i \cap R_j = \varnothing$ for all i and j, $i \neq j.$

(d) $Q(R_i) = TRUE$ for $i = 1, 2, \ldots, n.$

(e) $Q(R_i \cup R_j) = FALSE$ for any adjacent regions R_i *and* R_j.

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10
Segmentation

- **Region Growing**
  - *f(x, y)*: an input image array
  - *S(x, y)*: a seed array containing 1 at the locations of seed points 0s elsewhere.
  - Q: a predicate to be applied at each location (x, y).

1. Find all connected components in *S(x, y)* and erode each connected component to be one pixel; label all such pixels found as 1. All other pixels is S are labeled 0.
2. Form an image f_Q such that, at coordinates of (x, y), let $f_Q(x, y)=1$ if the input image satisfies the given predicate, *Q*.
3. Let *g* be an image formed by appending to each seed point in *S* all the 1-valued points in f_Q that are 8-connected to that seed point.
4. Label each connected component in *g* with a different region label (e.g., 1, 2, 3, …). This is the segmented image obtained by region growing.

© 1992–2008 R. C. Gonzalez & R. E. Woods

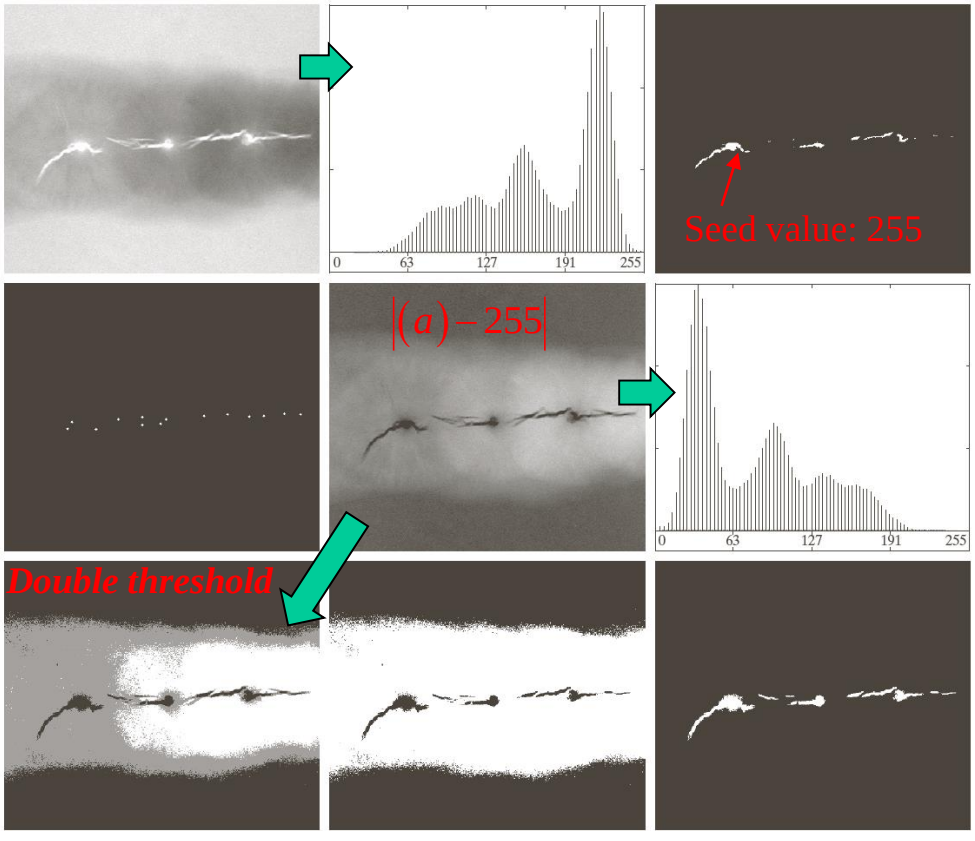
- **Example of region growing in weld inspection**

Intensity value at the 99.9 percentile

$$Q = \begin{cases} 1 & \text{if the absolute difference of the intensities between the seed and the pixel at } (x, y) \text{ is } \le T \\ 0 & \text{otherwise} \end{cases}$$

Morphological erosion
→ single point

$T_1 = 68$, $T_2 = 126$



a b c
d e f
g h i

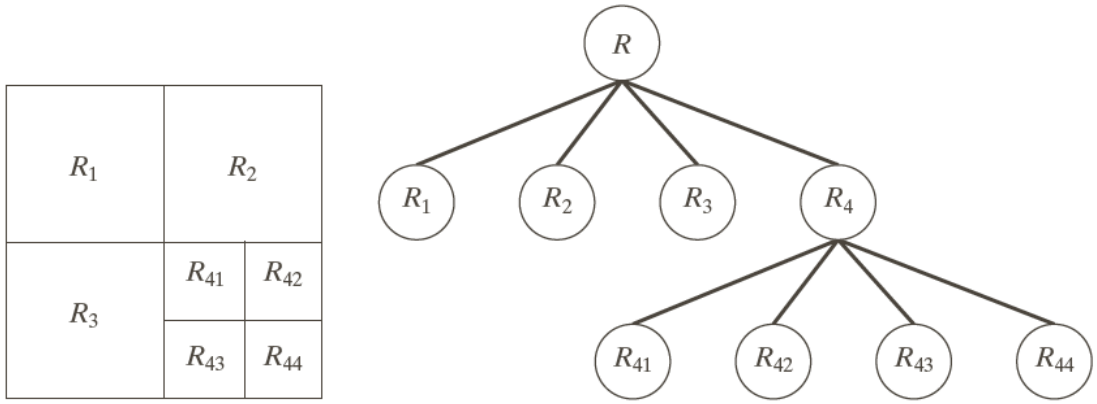
FIGURE 10.51 (a) X-ray image of a defective weld. (b) Histogram. (c) Initial seed image. (d) Final seed image (the points were enlarged for clarity). (e) Absolute value of the difference between (a) and (c). (f) Histogram of (e). (g) Difference image thresholded using dual thresholds. (h) Difference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

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Region Splitting and Merging**
  - The procedure

1. Split into four disjoint quadrants any region R_i for which $Q(R_i)$ = FALSE.
2. When no further splitting is possible, merge any adjacent regions R_j any R_k for which $Q(R_i \cup R_k)$=TRUE.
3. Stop when no further merging is possible.



a b

FIGURE 10.52 (a) Partitioned image. (b) Corresponding quadtree. R represents the entire image region.

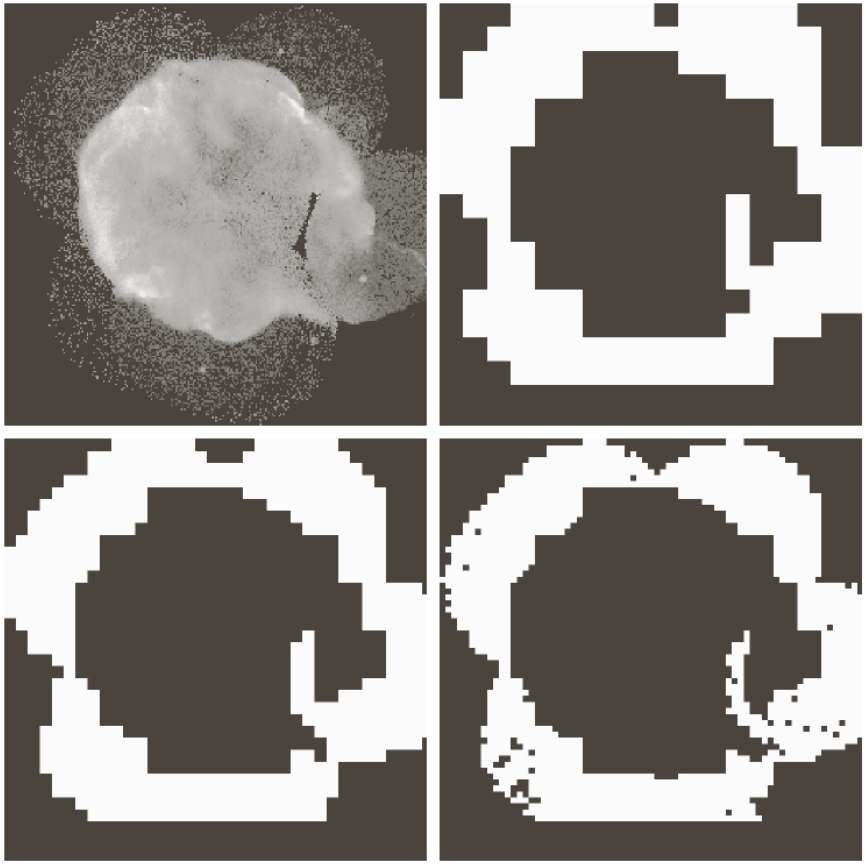
© 1992–2008 R. C. Gonzalez & R. E. Woods

- **Example of region splitting and merging**

$$Q = \begin{cases} 1 & if\ \sigma > a \quad AND\ 0 < m < b \\ 0 & otherwise \end{cases}$$

$\sigma = 10, m = 125$

32 x 32



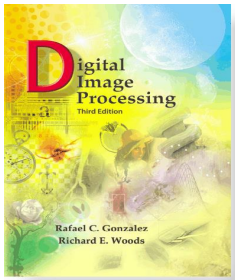
16 x 16

8 x 8

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.)

© 1992–2008 R. C. Gonzalez & R. E. Woods



10.5 Segmentation using Morphological Watersheds

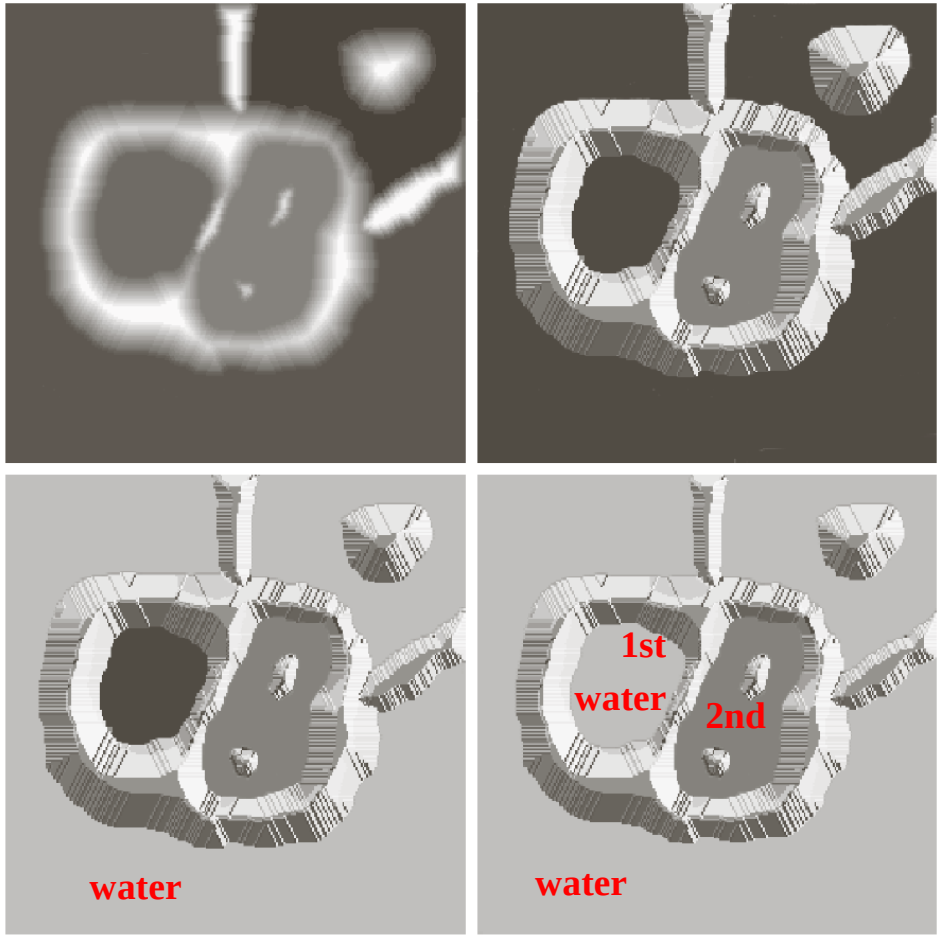
- **Basic Concepts**
  - Three type of points:
    1. regional minimum.
    2. Catchment basin (集水凹地)
    3. watershed line (分水嶺線)
  - Algorithm
    1. a hole is punched in each ***regional minimum***
    2. the entire topography is flooded from below by letting water rise through the holes at a uniform rate
    3. When the rising water in distinct catchment basins is about to merge, a ***dam*** is built to prevent the merging
    4. These ***dam boundaries*** correspond to the divide lines of the **watersheds**.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

a b
c d

FIGURE 10.54 (a) Original image. (b) Topographic view. (c)–(d) Two stages of flooding.



Topographic view

First stage of flooding

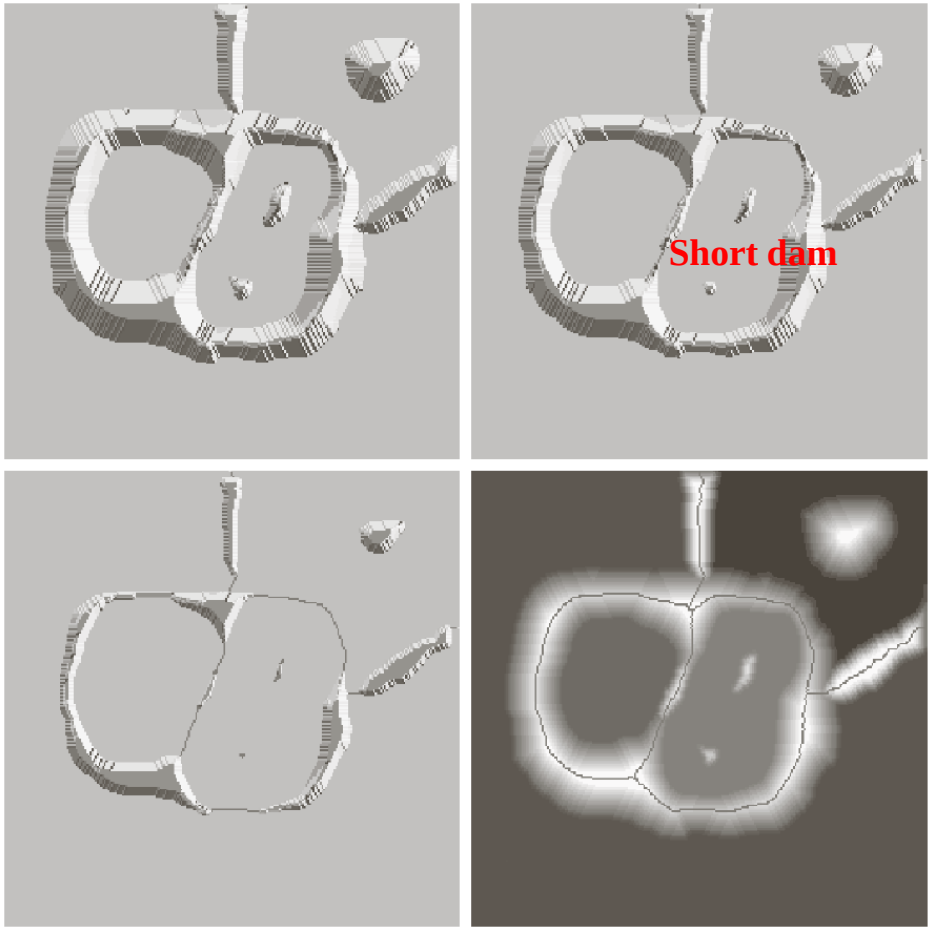
Water has risen into the 1st and 2nd catchment basins, respectively

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

e f
g h

FIGURE 10.54 *(Continued)* (e) Result of further flooding. (f) Beginning of merging of water from two catchment basins (a short dam was built between them). (g) Longer dams. (h) Final watershed (segmentation) lines. (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)



Water overflow from 1st catchment basin into 2nd catchment basin

A short "dam" was built to prevent water from merging at that level of flooding

Final watershed lines

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **Dam construction**
  - Based on binary images
  - Morphological dilation

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

$C_{n-1}(M_1)$ and $C_{n-1}(M_2)$ can be extracted from q by $q \cap C[n-1]$

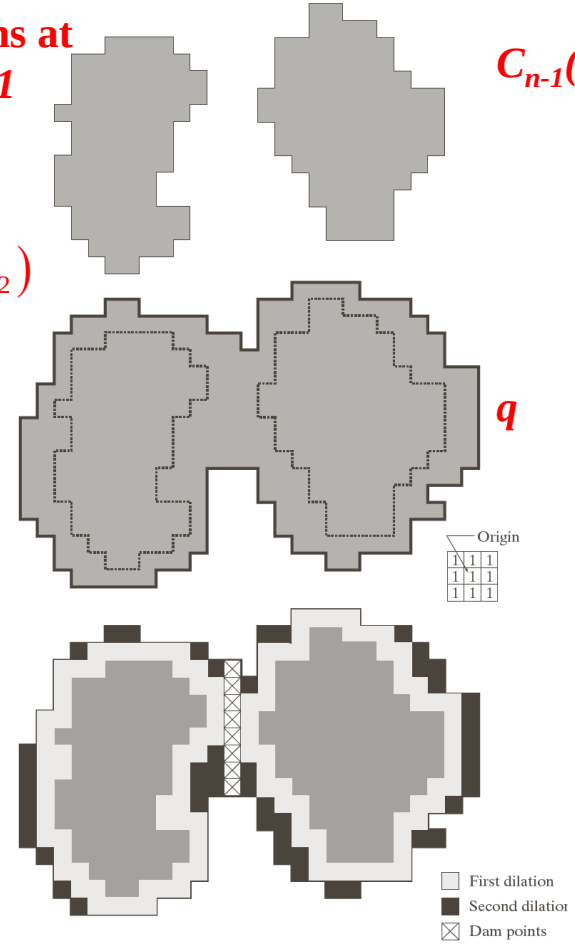
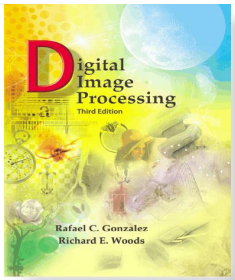


FIGURE 10.55 (a) Two partially flooded catchment basins at stage $n-1$ of flooding. (b) Flooding at stage n, showing that water has spilled between basins. (c) Structuring element used for dilation. (d) Result of dilation and dam construction.

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10
Segmentation

- **Watershed Segmentation Algorithm**
  - Definition

$$\mathrm{C}(\mathrm{M_i}): \textit{coordinates} \text{ of the points in the catchment basin associateed with regional minimum } \mathrm{M_i}$$

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

$$\mathrm{C_n}(\mathrm{M_i}) = \mathrm{C}(\mathrm{M_i}) \cap T[n]$$

$$\rightarrow \textit{coordinates} \text{ of the points in the catchment basin associateed with } \mathrm{M_i} \text{ that are flooded at stage n}$$

$$\Rightarrow \mathrm{C_n}(\mathrm{M_i}) = 1 \; \textit{if} \;\; (\mathrm{x,y}) \in \mathrm{C}(\mathrm{M_i}) \text{ AND } (\mathrm{x,y}) \in T[n]$$

  – union of the flooded catchment basins at stage *n*

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

  – union of all catchment basins

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

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

– process of the algorithm

1. $C[\min+1] = T[\min+1]$

The set of connected components in *T [n]*

2. *Construct* $C[n]$ *from* $C[n\text{-}1]$ *recursively*

$\Rightarrow$ *For* each connected component $q \in Q[n]$, there are 3 possibilities

(a) $q \cap C[n\text{-}1]$ is empty.

$\Rightarrow$ q is incorporated into $C[n\text{-}1]$ *to form* $C[n]$

(b) $q \cap C[n\text{-}1]$ contains one connected component of $C[n\text{-}1]$.

$\Rightarrow$ q is incorporated into $C[n\text{-}1]$ *to form* $C[n]$

(c) $q \cap C[n\text{-}1]$ contains more than one connected component of $C[n\text{-}1]$.

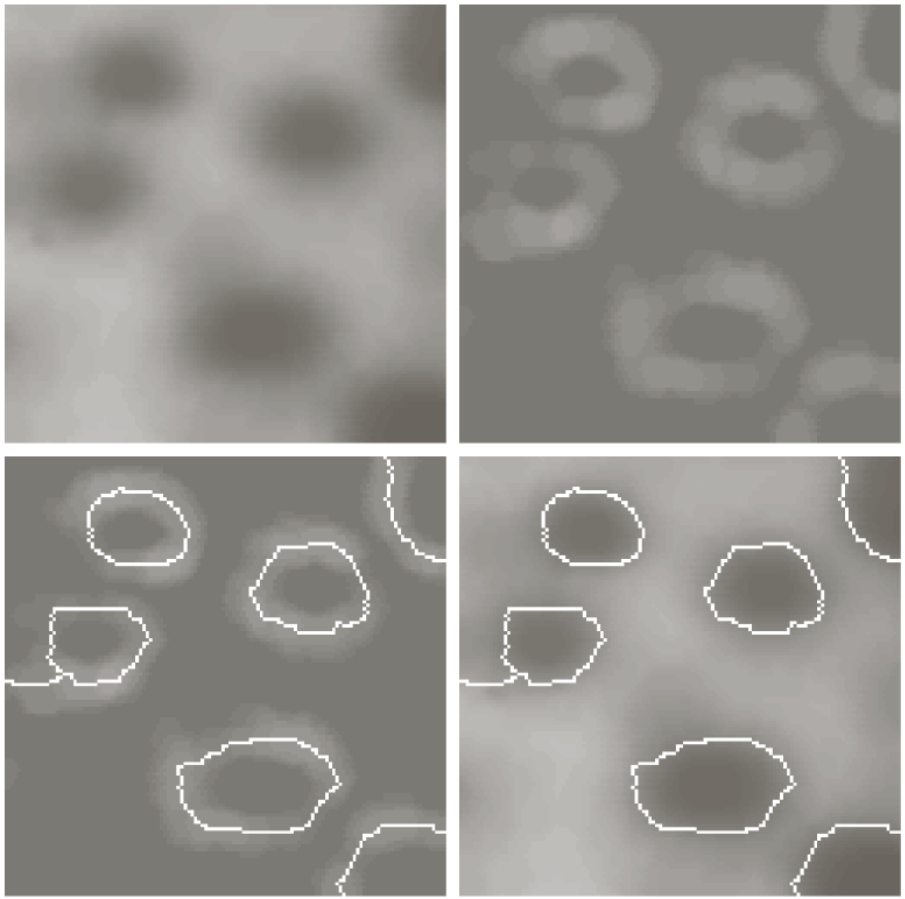
$\Rightarrow$ *a* dam must be built within q

3. *Construct dam* (or dams) and record the watershed lines.

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

Image gradient



a b
c d

FIGURE 10.56 (a) Image of blobs. (b) Image gradient. (c) Watershed lines. (d) Watershed lines superimposed on original image. (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)

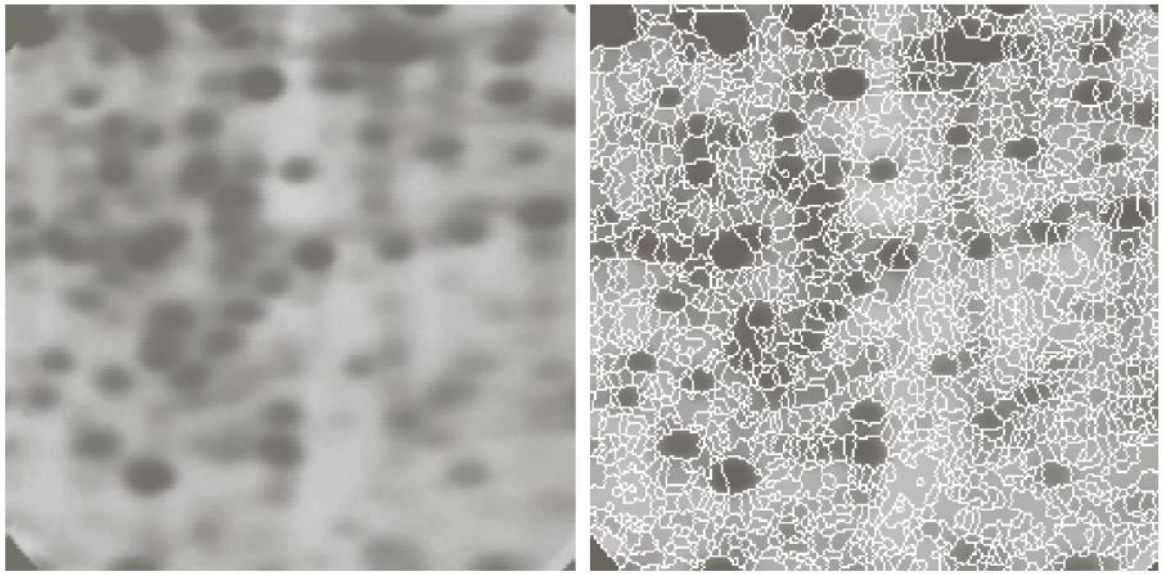
© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

– over segmentation

➡ Marker

- **Principal steps for marker selection :**
  1. Preprocessing: filtering the image with smoothing filter.
  2. Definition of a set of criteria that markers must satisfy.



a b

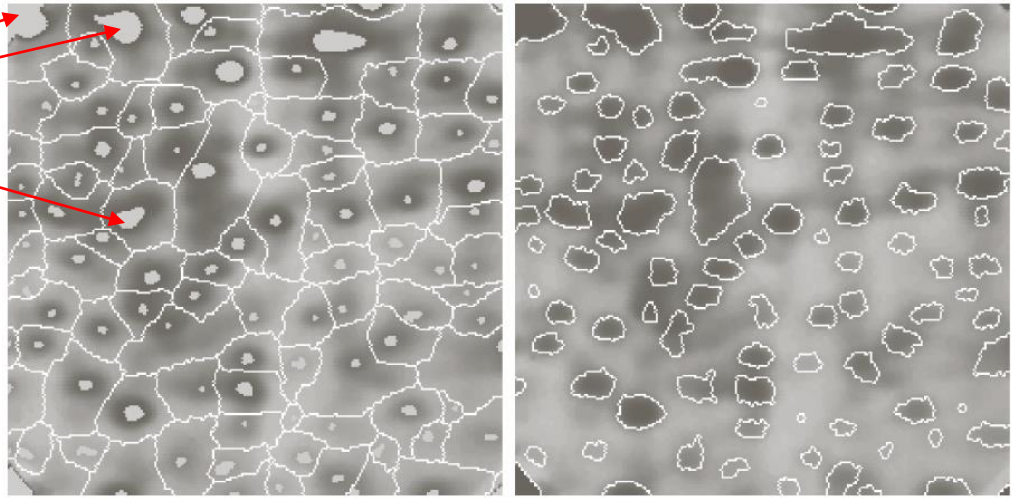
FIGURE 10.57 (a) Electrophoresis image. (b) Result of applying the watershed segmentation algorithm to the gradient image. Oversegmentation is evident. (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- **The Use of Markers**
  - internal markers: object
  - external markers: background
- **Define internal markers as**
  1. A region that is surrounded by points of higher "altitude".
  2. The points in the region form a connected component.
  3. All the points in the connected component have the same intensity value.

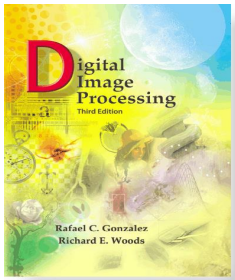
Markers



a b

FIGURE 10.58 (a) Image showing internal markers (light gray regions) and external markers (watershed lines). (b) Result of segmentation. Note the improvement over Fig. 10.47(b). (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)

© 1992–2008 R. C. Gonzalez & R. E. Woods



10.6 The Use of Motion in Segmentation

- **Spatial Techniques**
  - Basic approach

$$d_{ij}(x,y)=\begin{cases}1 & \text{if } \left|f(x,y,t_i)-f(x,y,t_j)\right|>T\\ 0 & \text{otherwise}\end{cases}$$

Difference image between 2 images taken at time t_i and t_j.

  - Accumulative differences

Reference image.

$$A_k(x,y)=\begin{cases}A_{k-1}(x,y)+1 & \text{if } \left|R(x,y)-f(x,y,k)\right|>T\\ A_{k-1}(x,y) & \text{otherwise}\end{cases}$$

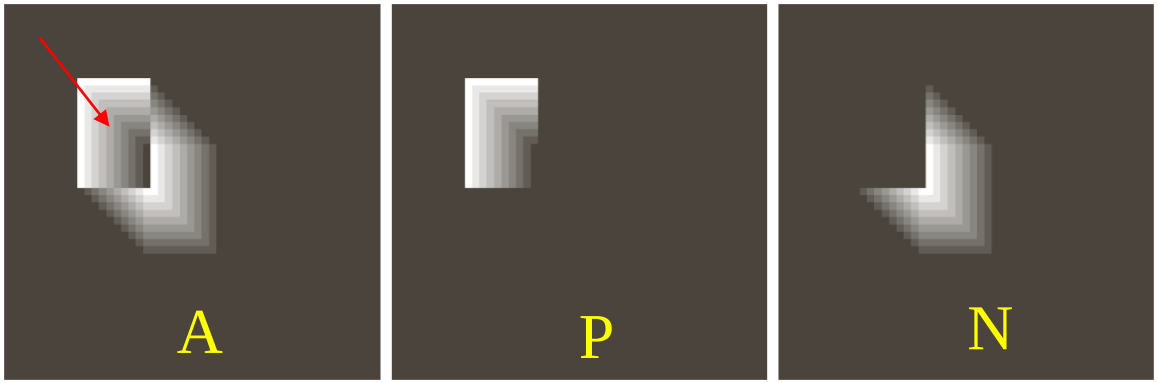
$$P_k(x,y)=\begin{cases}P_{k-1}(x,y)+1 & \text{if } \left[R(x,y)-f(x,y,k)\right]>T\\ P_{k-1}(x,y) & \text{otherwise}\end{cases}$$

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

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

- The Procedure:
  1. The nonzero area of the positive ADI is equal to the size of the moving object.
  2. The location of positive ADI corresponds to the location of the moving object in the reference frame.
  3. The number of counts in the positive ADI stops increasing when the moving object is displaced completely with respect to the same object in the reference frame.
  4. The absolute ADI contains the regions of the positive and negative ADIs.
  5. The direction and speed of the moving object can be determined from the entries in the absolute and negative ADIs.



a b c

FIGURE 10.59 ADIs of a rectangular object moving in a southeasterly direction. (a) Absolute ADI. (b) Positive ADI. (c) Negative ADI.

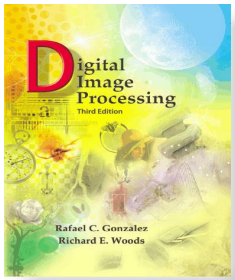
© 1992–2008 R. C. Gonzalez & R. E. Woods

– Establishing a reference image

a b c

FIGURE 10.60 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.)

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10
Segmentation

- **Frequency Domain Techniques**
  - For a sequence of *K* digital images of size *MxN*, the sum of the weighted projections onto the *x* and *y* axis at any integer instant of time is

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

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

  - 1-D Fourier transforms (to time)

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

$$G_y(u_2, a_2) = \sum_{t=0}^{K-1} g_y(t, a_2) e^{-j2\pi u_2 t/K} \quad u_2 = 0, 1, 2, \ldots, K-1$$

© 1992–2008 R. C. Gonzalez & R. E. Woods

$$u_1 = a_1 V_1$$
$$u_2 = a_2 V_2$$

Example:

V_1=10 → a motion of 10 pixels in K frames

If K=30, the frame rate is two images per second, and the distance between pixels is 0.5 m , then the actual physical speed in the x-direction is

$$V_1 = (10\ pixels)(0.5\, m/\ pixel)(2\ \ frames/s)/(30\ \ frames) = 1/3\, m/\sec$$

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

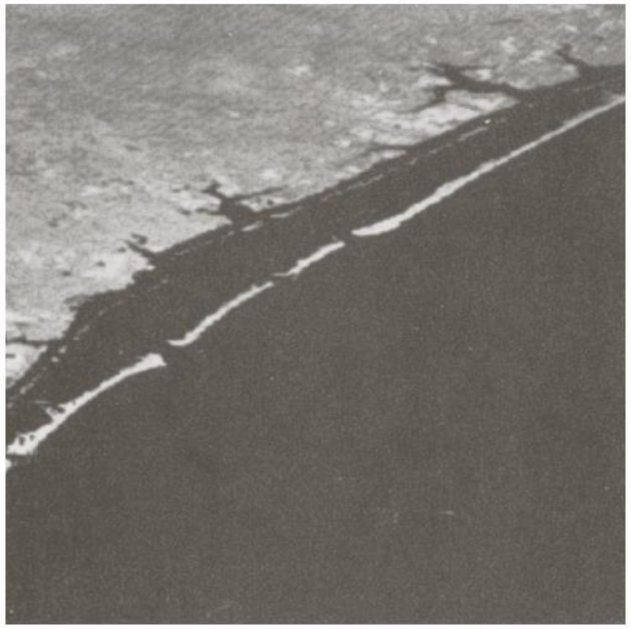
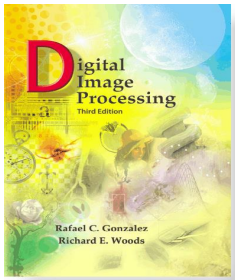


FIGURE 10.61 LANDSAT frame. (Cowart, Snyder, and Ruedger.)

- 32-frame sequence of LANDSAT images
- Containing a superimposed target
  - Moving at 0.5 pixel/frame in the x-direction
  - Moving at 1 pixel/frame in the y-direction

It is not easily discernible by eye !

© 1992–2008 R. C. Gonzalez & R. E. Woods



Chapter 10
Segmentation

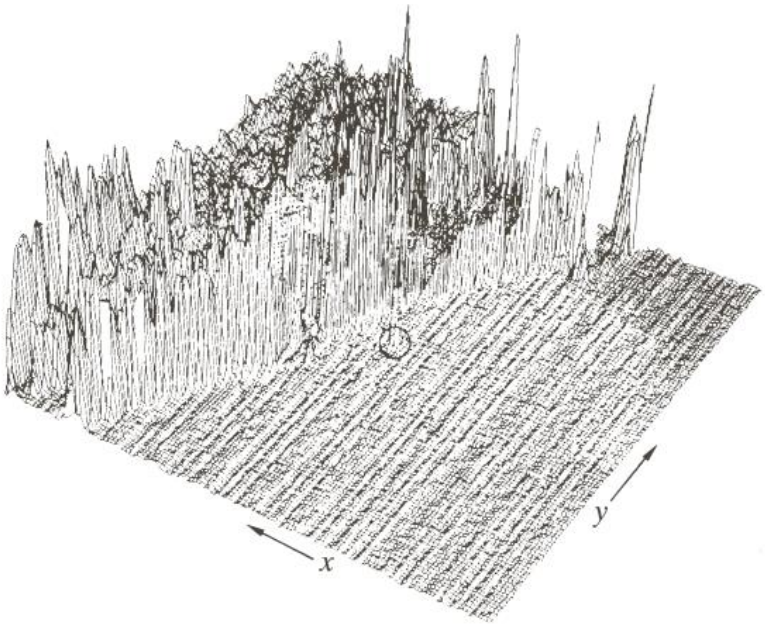
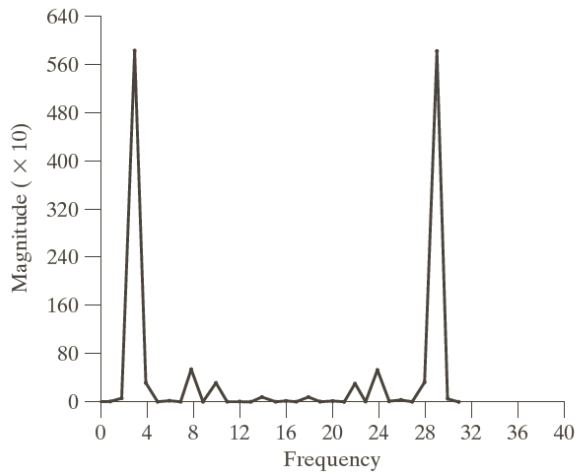


FIGURE 10.62 Intensity plot of the image in Fig. 10.61, with the target circled. (Rajala, Riddle, and Snyder.)

© 1992–2008 R. C. Gonzalez & R. E. Woods

Chapter 10
Segmentation

FIGURE 10.64 Spectrum of Eq. (10.6-9) showing a peak at $u_2 = 4$. (Rajala, Riddle, and Snyder.)



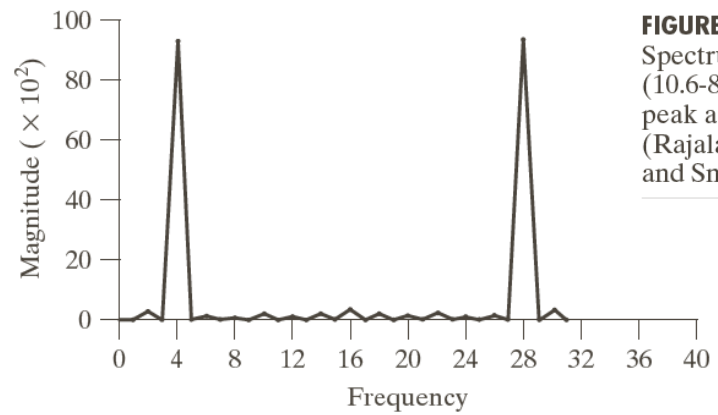


FIGURE 10.63 Spectrum of Eq. (10.6-8) showing a peak at $u_1 = 3$. (Rajala, Riddle, and Snyder.)

$$u_1 = 3,\ a_1 = 6 \quad \Rightarrow V_1 = 0.5$$
$$u_2 = 4,\ a_2 = 4 \quad \Rightarrow V_2 = 1$$

- **Selection of *a*:**

$$a_1 = u_{\max} / V_{\max}$$

$u_{\max}$: the aliasing frequency limitation

$V_{\max}$: the maximum expected object velocity

© 1992–2008 R. C. Gonzalez & R. E. Woods