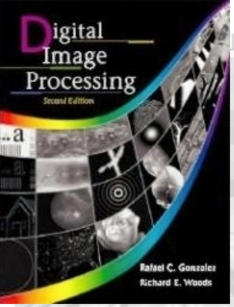


Chapter 10
Image Segmentation

- Segmentation subdivides an image into its constituent ***regions*** or ***objects***.
- Segmentation based on the discontinuity and similarity.
- Discontinuity: abrupt changes in intensity, such as edges.
- Similarity: partitioned into regions similar according to a set of predefined criteria, such as thresholding region growing, region splitting and merging.

10.1 Detection of Discontinuity

- The response of the mask at any point in the image is given by $R = w_1 z_1 + w_2 z_2 + \ldots + w_9 z_9$

FIGURE 10.1 A general 3×3 mask.

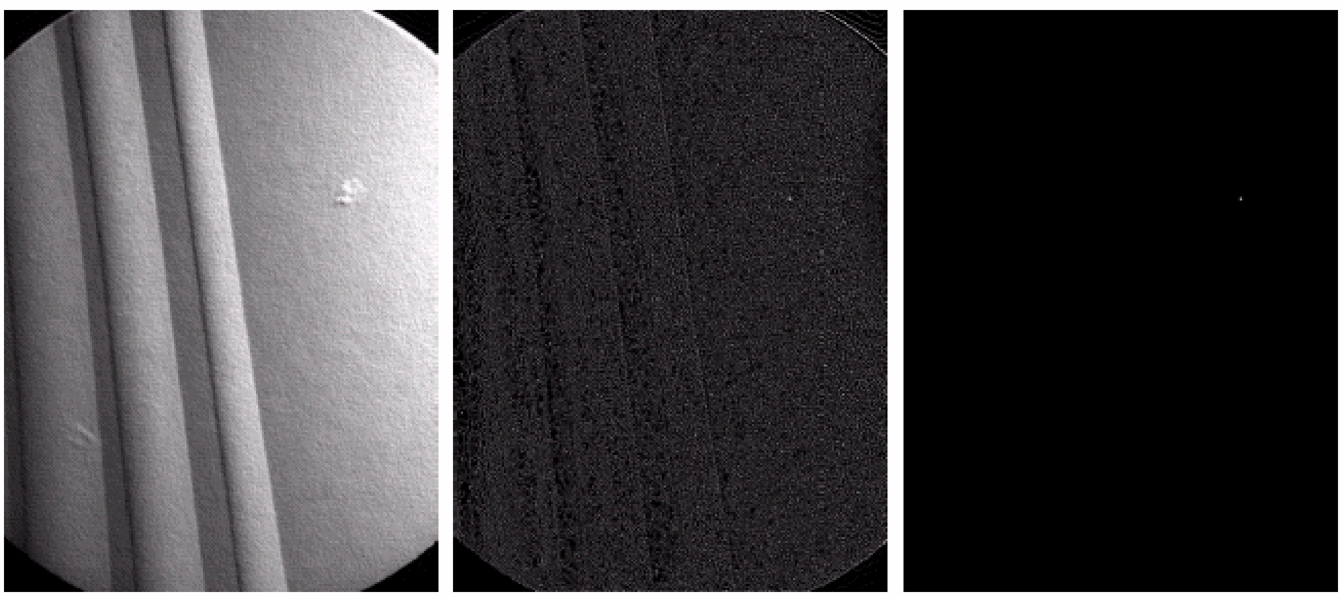
w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

10.1 Detection of Discontinuities

- Point detection: detect isolated point.
- A point is detected if $|R| \geq T$

Laplacian operator →

−1	−1	−1
−1	8	−1
−1	−1	−1



a
b c d

FIGURE 10.2 (a) Point detection mask. (b) X-ray image of a turbine blade with a porosity. (c) Result of point detection. (d) Result of using Eq. (10.1-2). (Original image courtesy of X-TEK Systems Ltd.)

10.1 Detection of Discontinuities

Line detection:

FIGURE 10.3 Line masks.

Horizontal

−1	−1	−1
2	2	2
−1	−1	−1

+45°

−1	−1	2
−1	2	−1
2	−1	−1

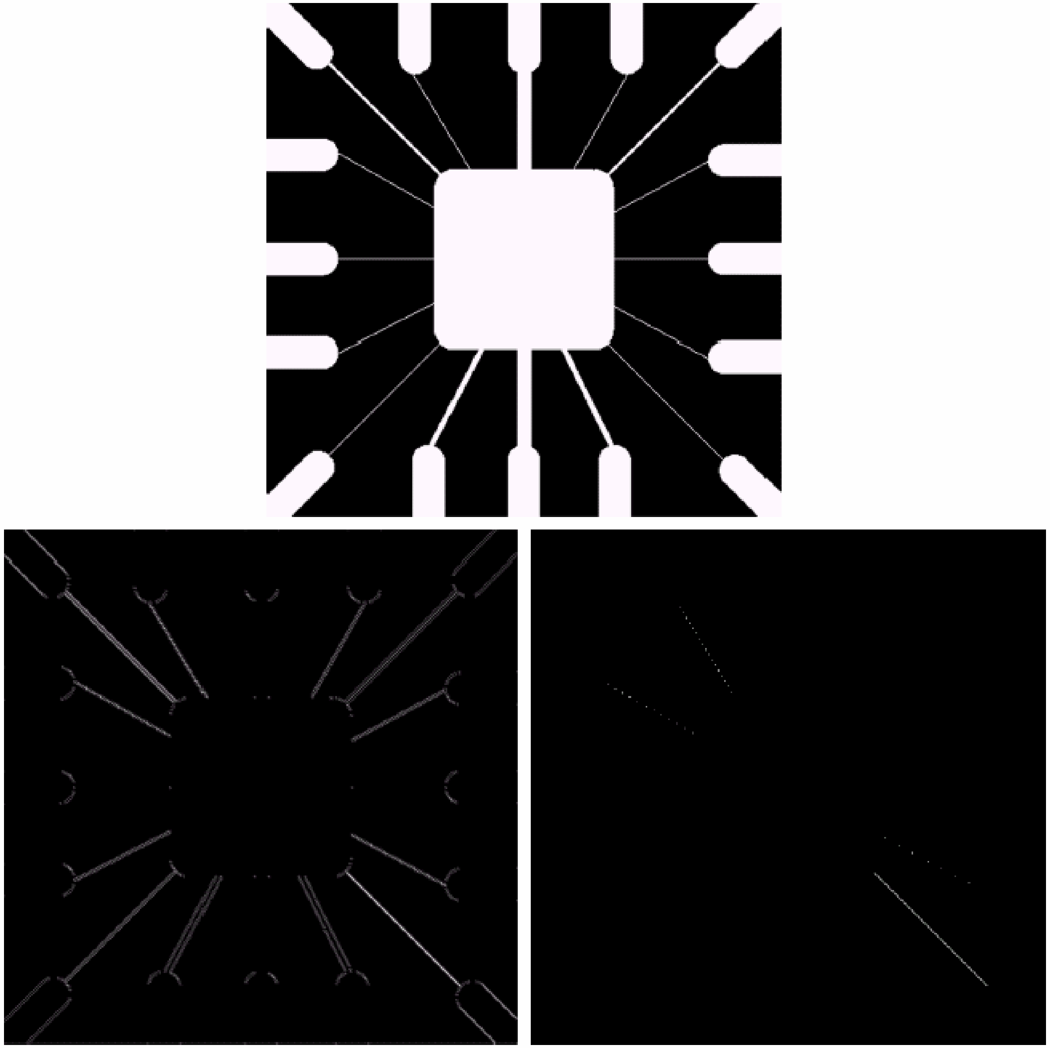
Vertical

−1	2	−1
−1	2	−1
−1	2	−1

−45°

2	−1	−1
−1	2	−1
−1	−1	2

10.1 Detection of Discontinuities

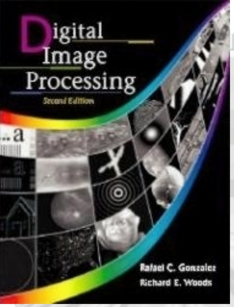


a
b c

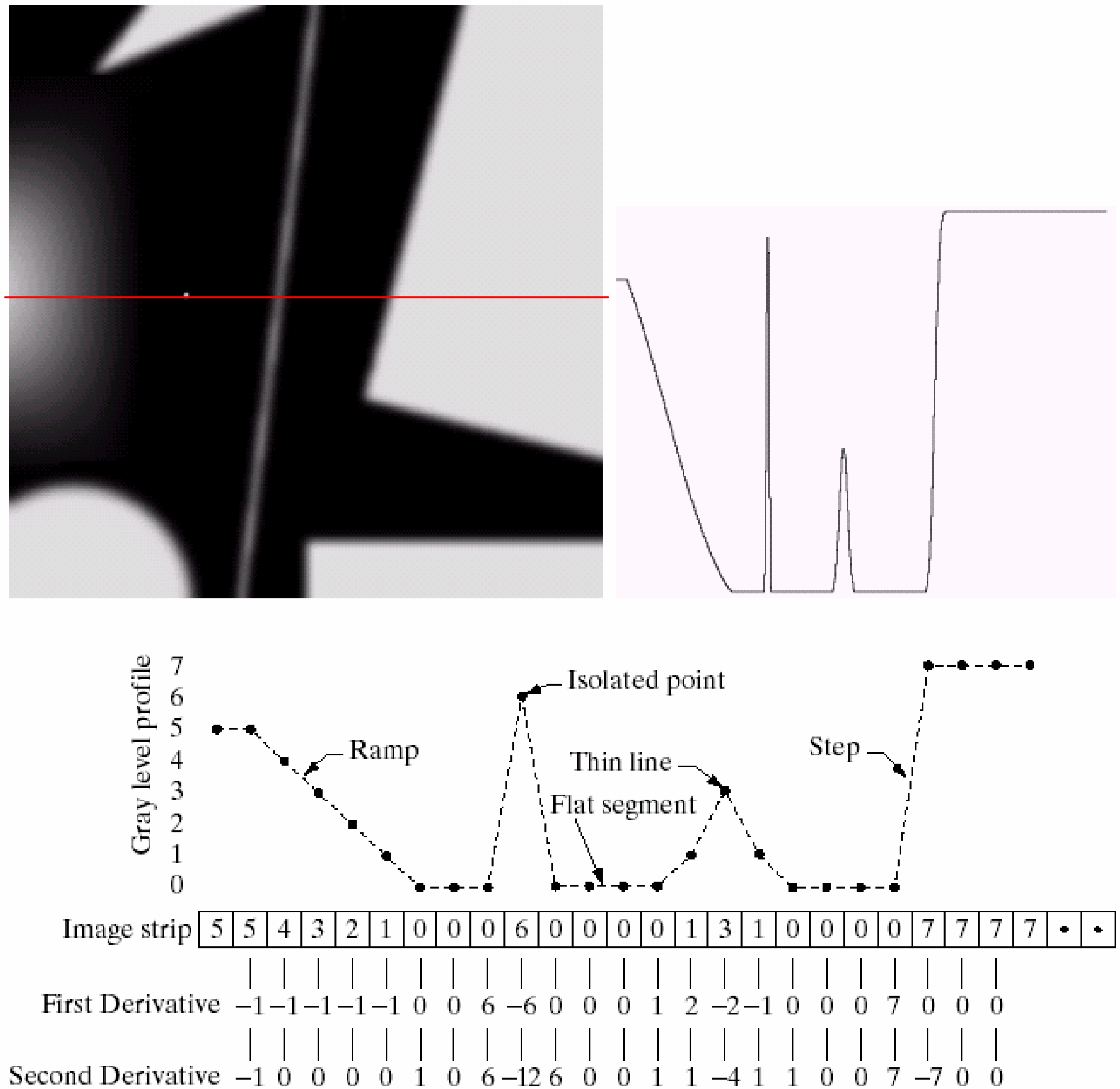
FIGURE 10.4 Illustration of line detection. (a) Binary wire-bond mask. (b) Absolute value of result after processing with −45° line detector. (c) Result of thresholding image (b).

10.1 Detection of Discontinuities -Edge Detection

- An edge is modeled as a “meaningful” transitions in gray-levels.
- Ideal (or Step) edge
- Ramp edge
- Roof edge
- First derivative and second derivative on the edge profile.
- “Zero-crossing” property is used to identify the location of edge.



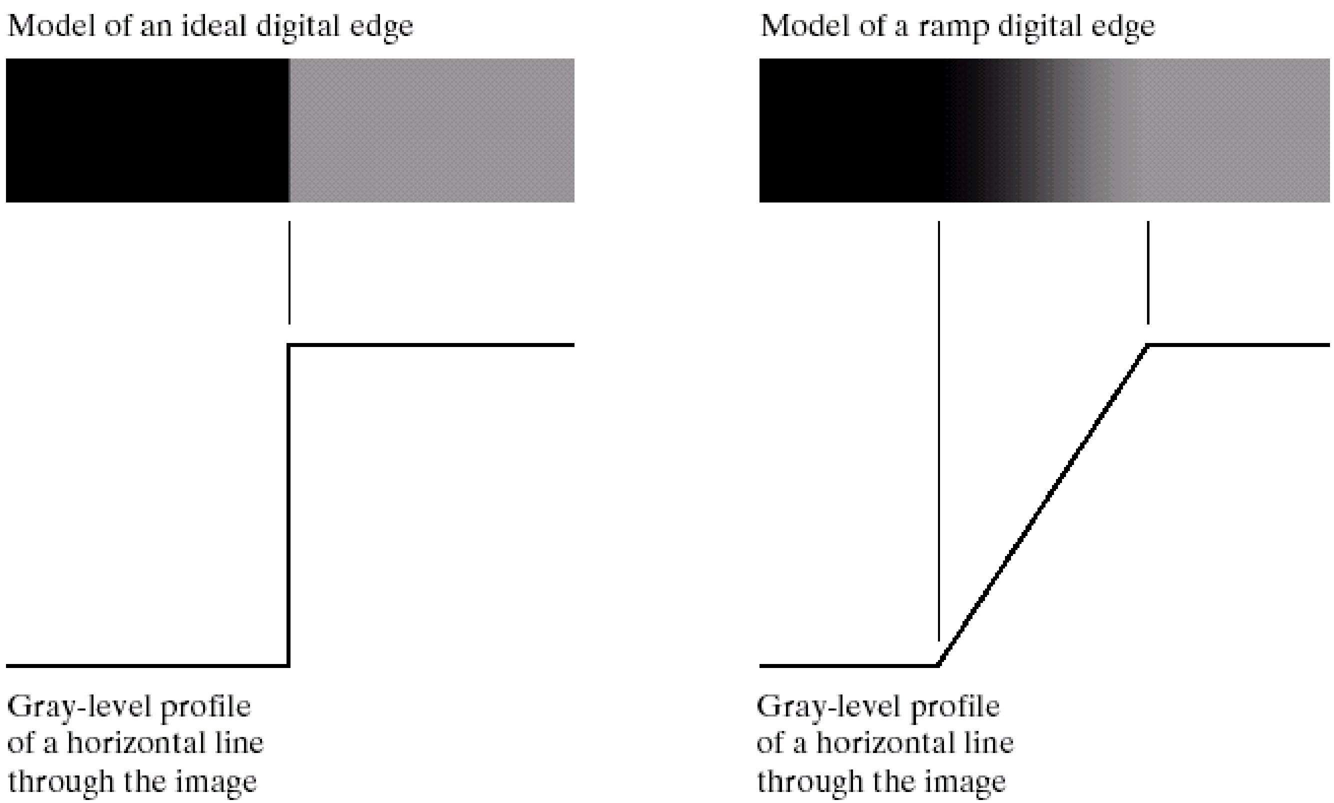
10.1 Detection of Discontinuities - Edge types



a b
c

FIGURE 3.38 (a) A simple image. (b) 1-D horizontal gray-level profile along the center of the image and including the isolated noise point. (c) Simplified profile (the points are joined by dashed lines to simplify interpretation).

10.1 Detection of Discontinuities



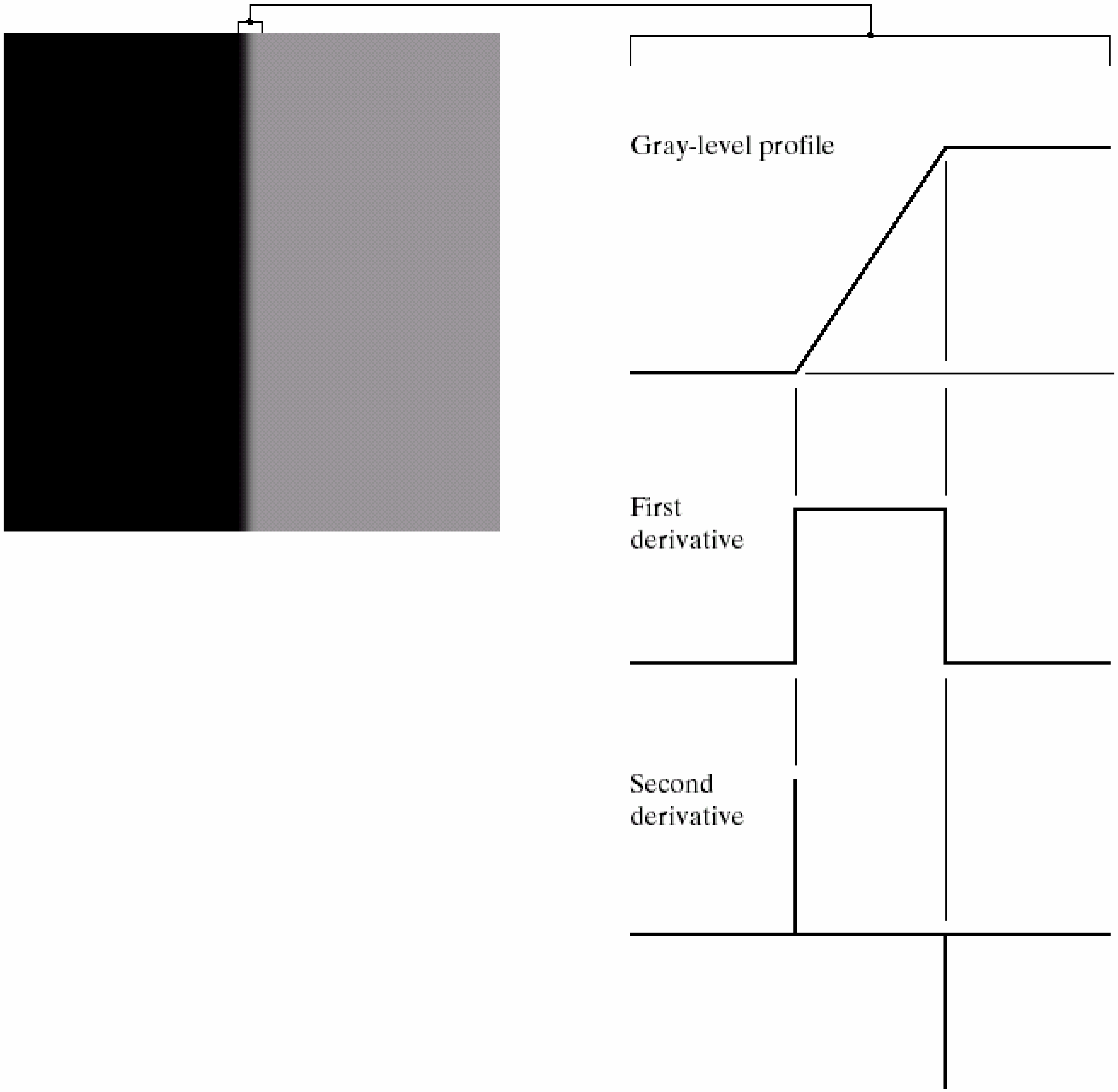
a b

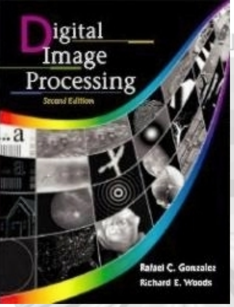
FIGURE 10.5 (a) Model of an ideal digital edge. (b) Model of a ramp edge. The slope of the ramp is proportional to the degree of blurring in the edge.

10.1 Detection of Discontinuities

a b

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





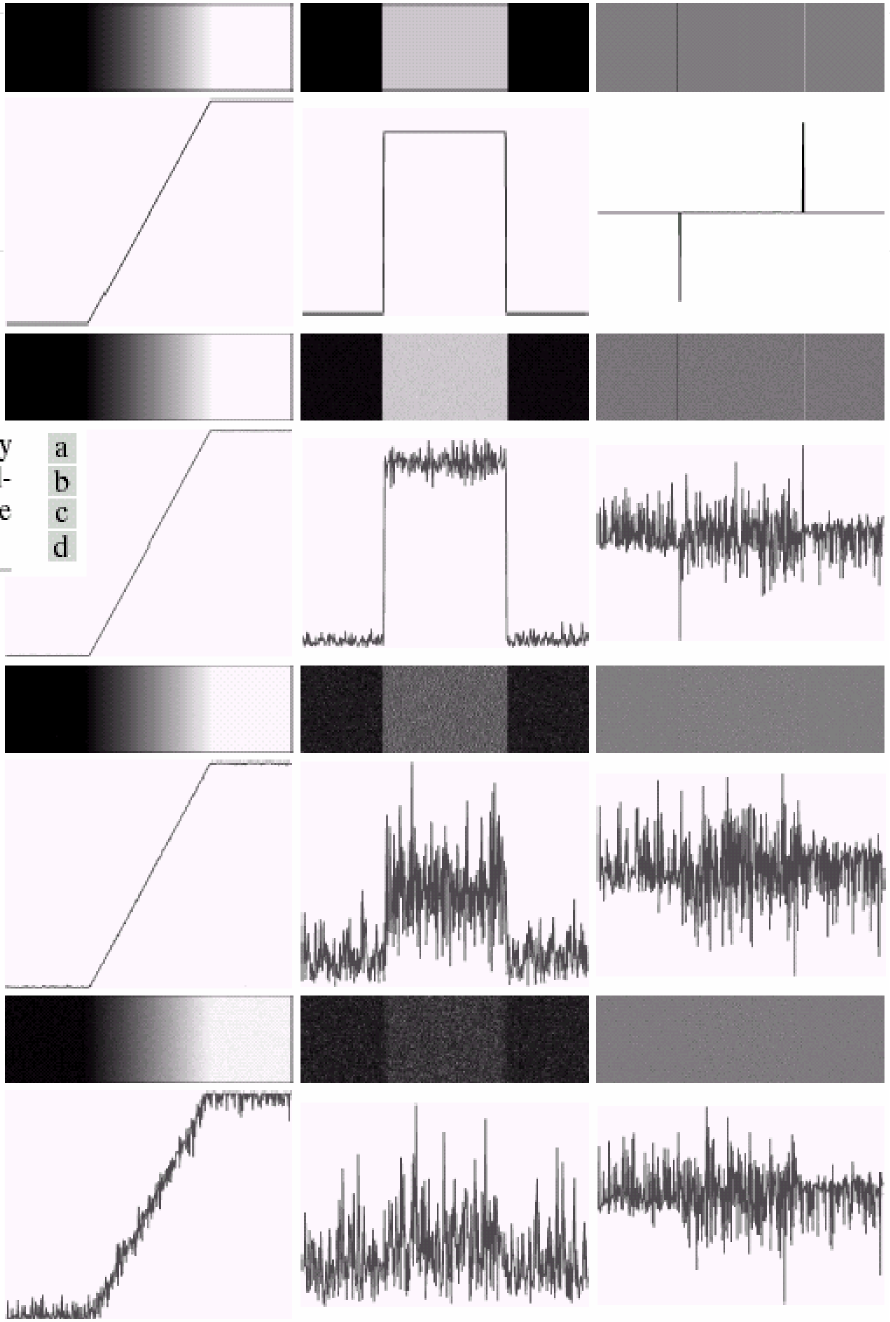
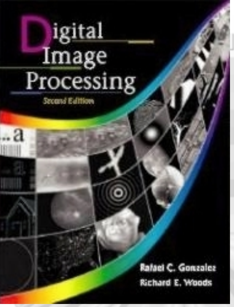


FIGURE 10.7 First column: images and gray-level profiles of a ramp edge corrupted by random Gaussian noise of mean 0 and $\sigma = 0.0, 0.1, 1.0$, and 10.0, respectively. Second column: first-derivative images and gray-level profiles. Third column: second-derivative images and gray-level profiles.

a
b
c
d

10.1 Detection of Discontinuities



10.1 Detection of Discontinuities -Edge Detection

- Apply gradient operators on an image $f(x, y)$ at location (x, y) to obtain a 2-D gradient defined as:

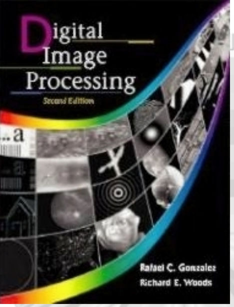
$$\nabla\mathbf{f}=[G_x, G_y]=[\partial \mathrm{f}/\partial \mathrm{x}, \partial \mathrm{f}/\partial \mathrm{y}]$$

- The magnitude of this vector is

$$\nabla \mathrm{f}=\mathrm{mag}(\nabla\mathbf{f})= [G_x^{\,2} , G_y^{\,2}]^{1/2}$$

- The direction is

$$\alpha(\mathrm{x}, \mathrm{y})=\tan^{-1}(G_x/G_y)$$



10.1 Detection of Discontinuities -Edge Detection

- Robert cross-gradient operators:

  $$G_x=(z_9-z_5) \text{ and } G_y=(z_8-z_6)$$

- It does not have clear center.
- Prewitt 3x3 operators

  $$G_x=(z_7+z_8+z_9)-(z_1+z_2+z_3) \text{ and}$$

  $$G_y=(z_3+z_6+z_9)-(z_1+z_4+z_{37})$$

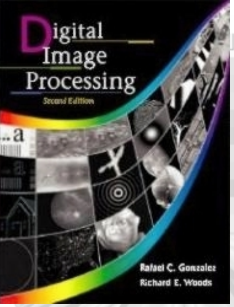
- Weighted Prewitt 3x3 operators

  $$G_x=(z_7+2z_8+z_9)-(z_1+2z_2+z_3) \text{ and}$$

  $$G_y=(z_3+2z_6+z_9)-(z_1+2z_4+z_{37})$$

- The gradient is $\nabla f \cong |G_x|+|G_y|$

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9



a
b c
d e
f g

FIGURE 10.8 A 3 × 3 region of an image (the z's are gray-level values) and various masks used to compute the gradient at point labeled z_5.

10.1 Detection of Discontinuities

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
0	0	0
1	1	1

−1	0	1
−1	0	1
−1	0	1

Prewitt

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

Sobel

10.1 Detection of Discontinuities

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

a b
c d

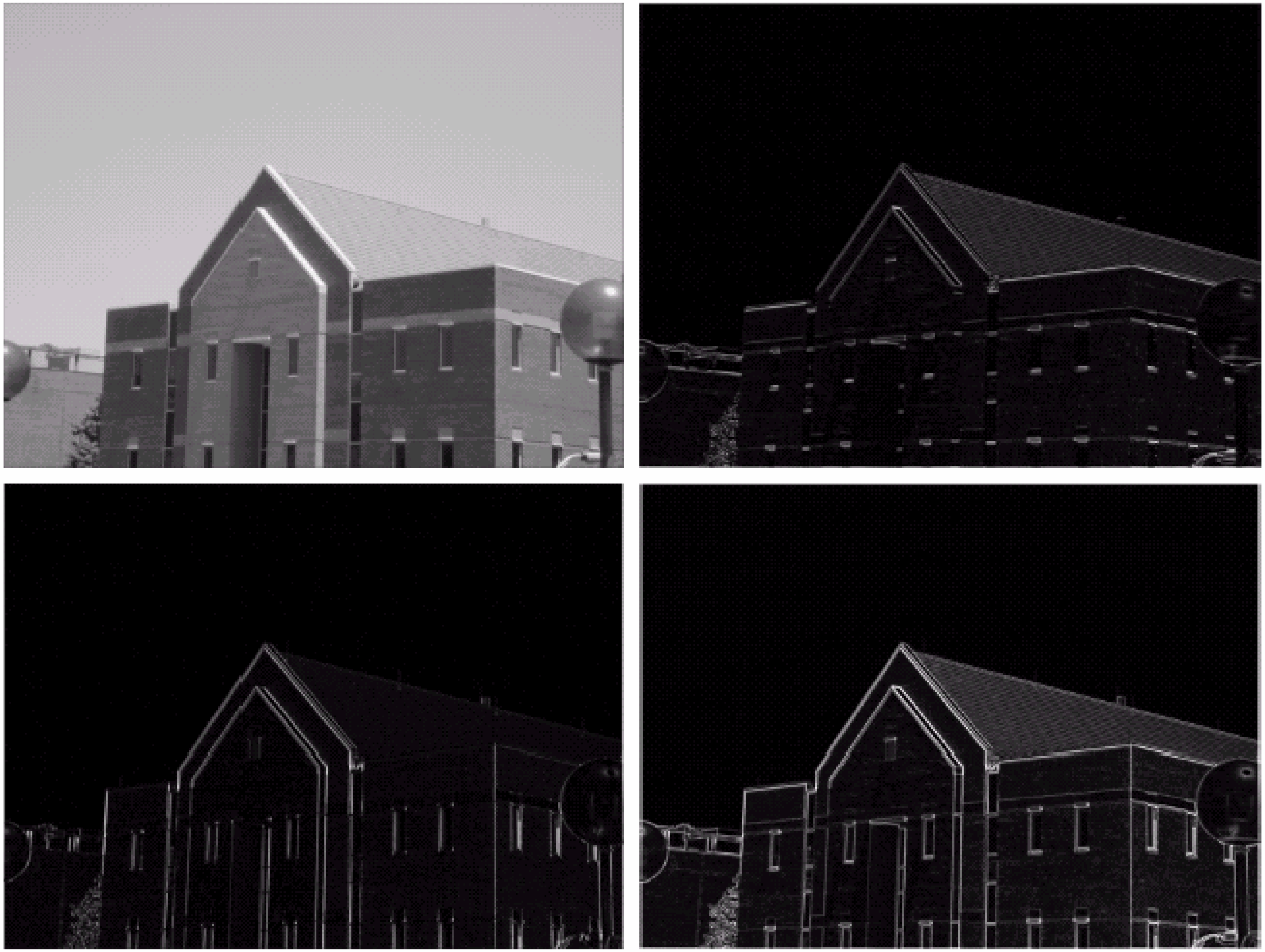
FIGURE 10.9 Prewitt and Sobel masks for detecting diagonal edges.

10.1 Detection of Discontinuities

a b
c d

FIGURE 10.10 (a) Original image. (b) $|G_x|$, component of the gradient in the x-direction. (c) $|G_y|$, component in the y-direction. (d) Gradient image, $|G_x| + |G_y|$.

10.1 Detection of Discontinuities



a b
c d

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

10.1 Detection of Discontinuities

a b

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

10.1 Detection of Discontinuities -Edge Detection

- The Laplacian operator:

  $\nabla^2 f = [\partial^2 f/\partial x^2, \partial f^2/\partial y^2]$

- Digital approximation for 3x3 region is

  $\nabla^2 f = 4z_5 - (z_2+z_4+z_6+z_8)$

or $\nabla^2 f = 8z_5 - (z_2+z_4+z_6+z_8+z_1+z_3+z_7+z_9)$

- The Laplacian is very sensitive to the noise.

10.1 Detection of Discontinuities

FIGURE 10.13 Laplacian masks used to implement Eqs. (10.1-14) and (10.1-15), respectively.

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

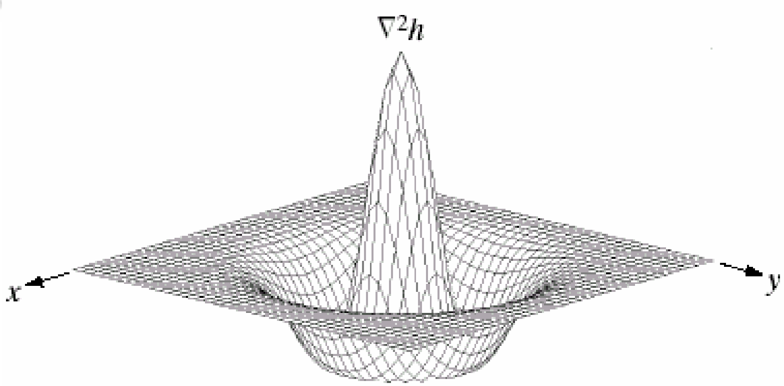
10.1 Detection of Discontinuities -Edge Detection

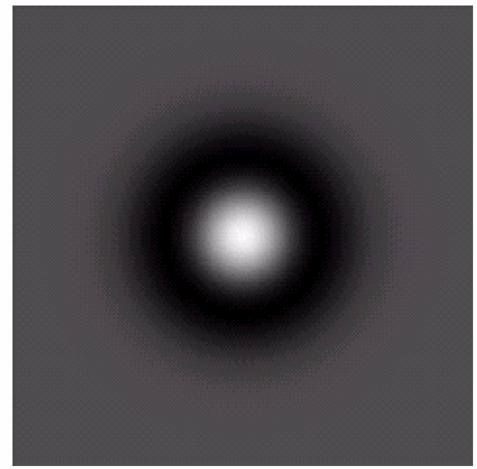
- Laplacian plus Gaussian (smoothing)

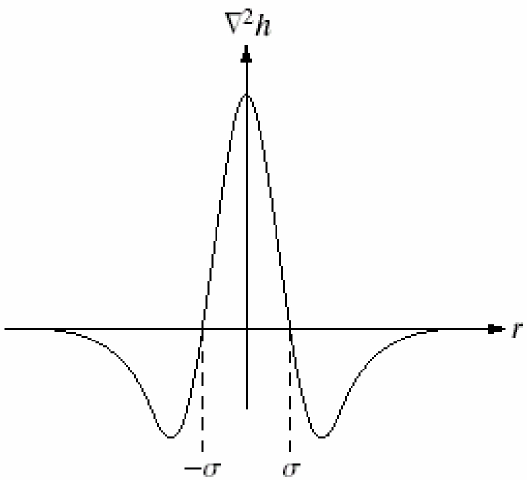
  Gaussian: $h(r)=-e^{-r^2/2\sigma^2}$ where $r^2=x^2+y^2$

- Convolving the Gaussian with the image will blur the image.
- The Laplacian of $h(r)$ is $\nabla^2 h(r) = -\left[\frac{r^2-\sigma^2}{\sigma^4}\right]e^{-\frac{r^2}{2\sigma^2}}$
- It is called the Lapalcian of a Gaussian (LoG) which is also called ***Mexican hat function***

10.1 Detection of Discontinuities





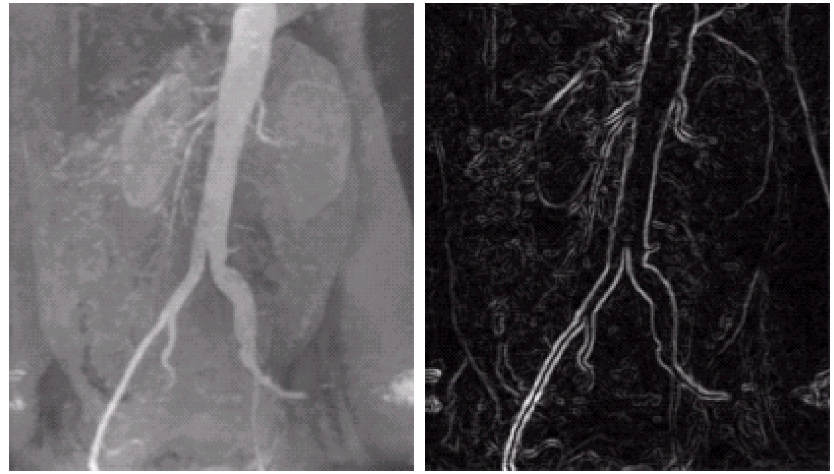


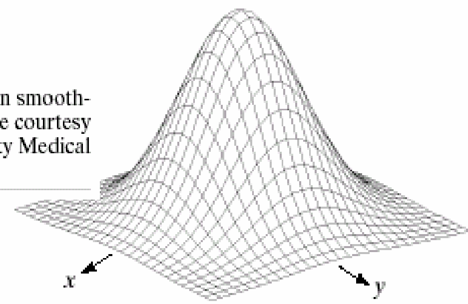
0	0	−1	0	0
0	−1	−2	−1	0
−1	−2	16	−2	−1
0	−1	−2	−1	0
0	0	−1	0	0

a b
c d

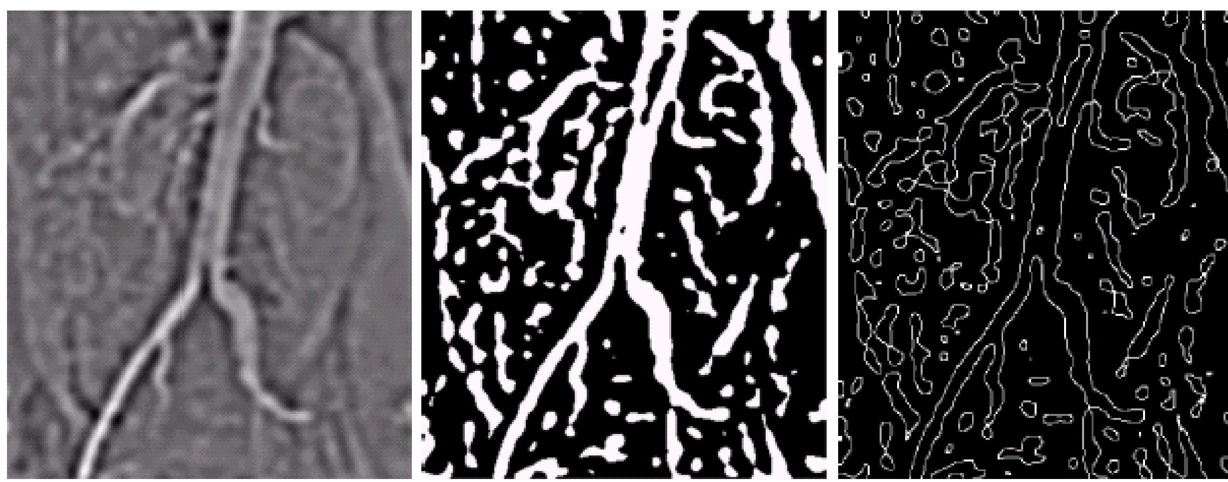
FIGURE 10.14 Laplacian of a Gaussian (LoG). (a) 3-D plot. (b) Image (black is negative, gray is the zero plane, and white is positive). (c) Cross section showing zero crossings. (d) 5 × 5 mask approximation to the shape of (a).

10.1 Detection of Discontinuities





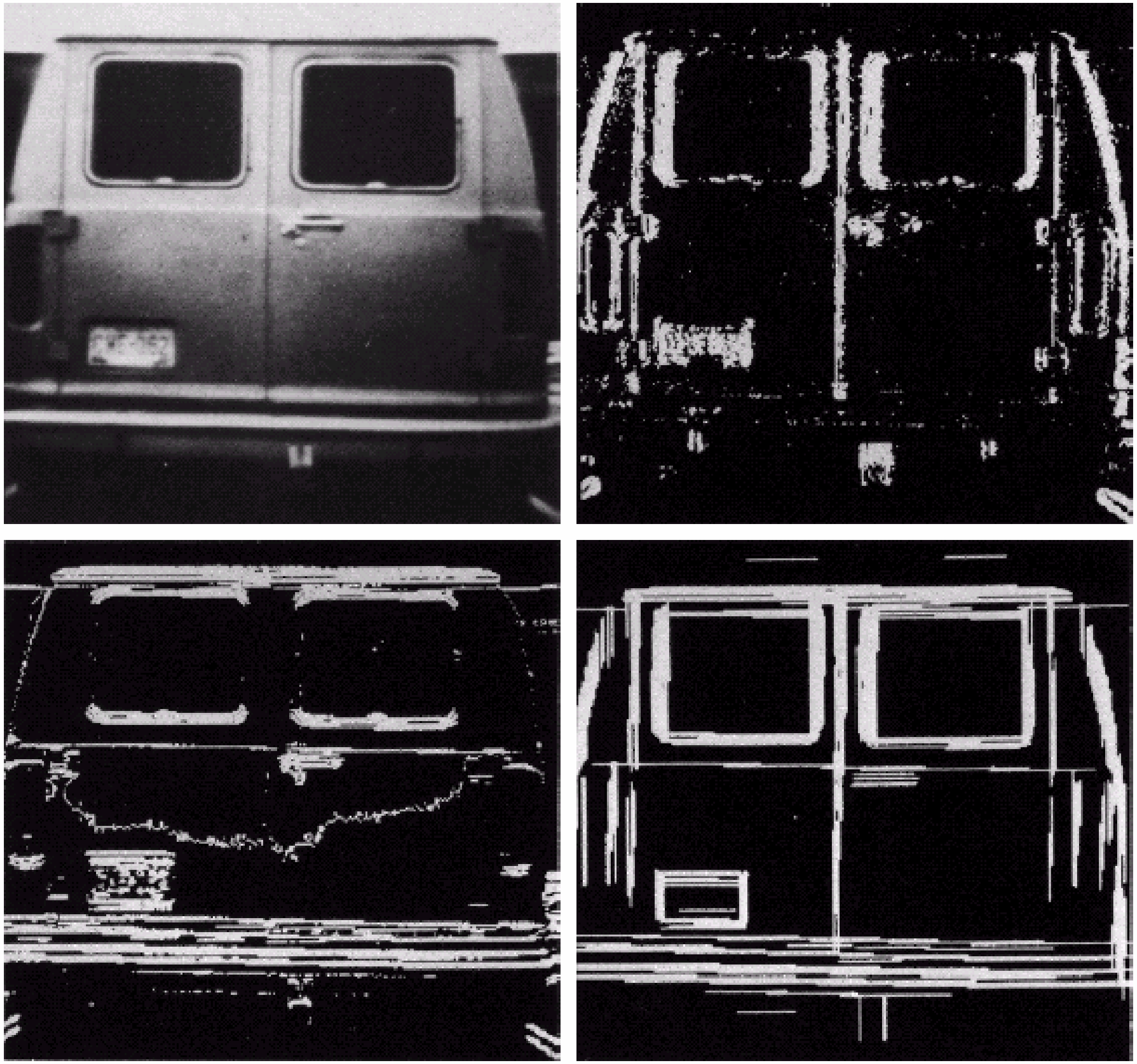
−1	−1	−1
−1	8	−1
−1	−1	−1



a b
c d
e f g

FIGURE 10.15 (a) Original image. (b) Sobel gradient (shown for comparison). (c) Spatial Gaussian smoothing function. (d) Laplacian mask. (e) LoG. (f) Thresholded LoG. (g) Zero crossings. (Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

10.2 Edge Linking and Boundary detection



a b
c d

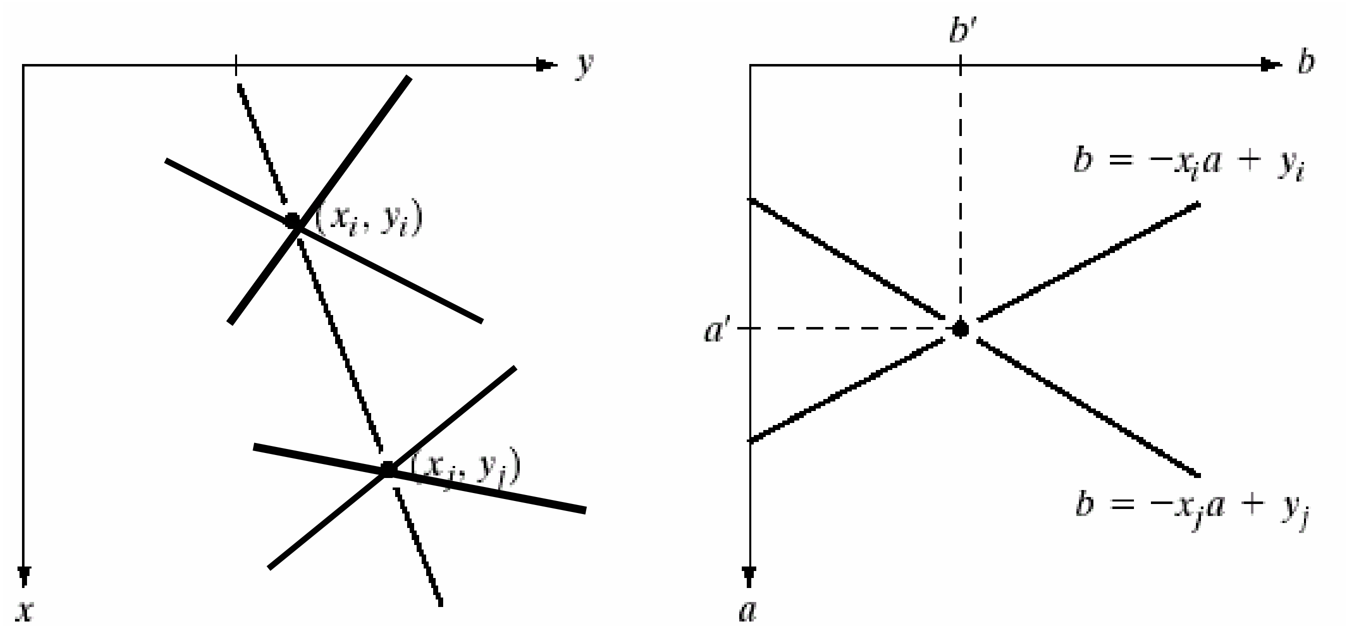
FIGURE 10.16
(a) Input image.
(b) G_y component of the gradient.
(c) G_x component of the gradient.
(d) Result of edge linking. (Courtesy of Perceptics Corporation.)

10.2 Edge Linking and Boundary detection- Global processing via Hough transform

- Given a point (x_i, y_i) --- many lines pass through this point as $y_i=ax_i+b$ with different a and b.
- A point (x_i, y_i) in image space is mapped to many points $\{(a, b)\}$ in parameter space which are on line

  $b=-ax_i+y_i$.
- The collinear point (x_j, y_j) -many lines pass through this point as $y_i=ax_i+b$ with different a and b.
- The collinear point (x_j, y_j) in image space is mapped to many points $\{(a, b)\}$ in parameter space which are on line: $b=-ax_j+y_j$.
- These two lines in parameter space intersect at (a', b')

10.2 Edge Linking and Boundary detection



a b

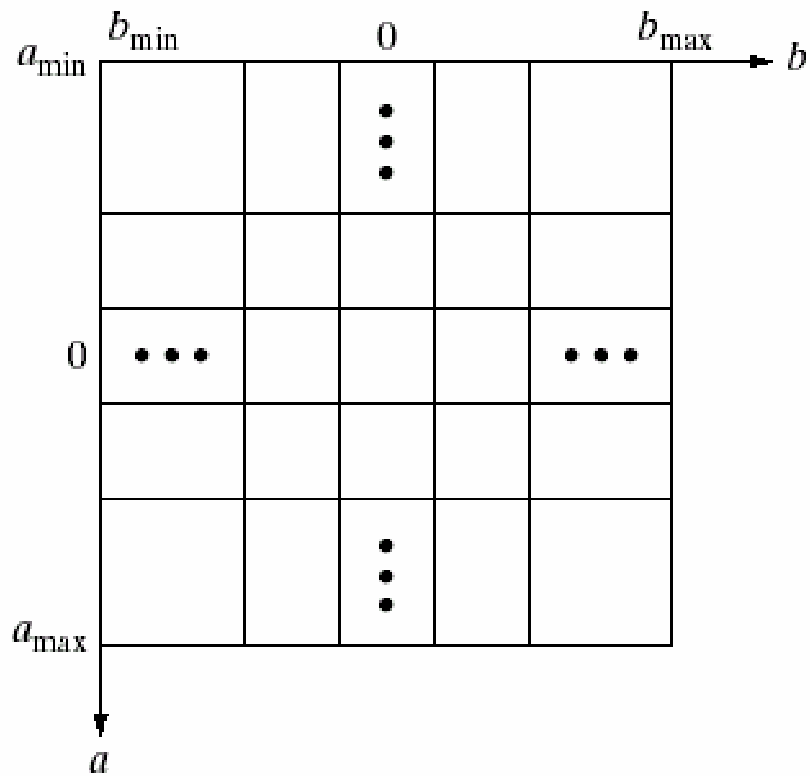
FIGURE 10.17 (a) *xy*-plane. (b) Parameter space.

a' is slope, *b'* is the intercept of the line passing through (x_i, y_i) and (x_j, y_j)

$$(a', b') \in \{(a, b)\}$$

10.2 Edge Linking and Boundary detection Hough Transform

FIGURE 10.18 Subdivision of the parameter plane for use in the Hough transform.



Accumulator Cell *A(i ,j)*:

$(x_i, y_i) \rightarrow \{(a, b)\}$

$a \rightarrow a_p$ and $b \rightarrow b_q$

$A(p, q) = A(p, q) + 1$

Problem: slope "a" may approach infinity for detecting a vertical line

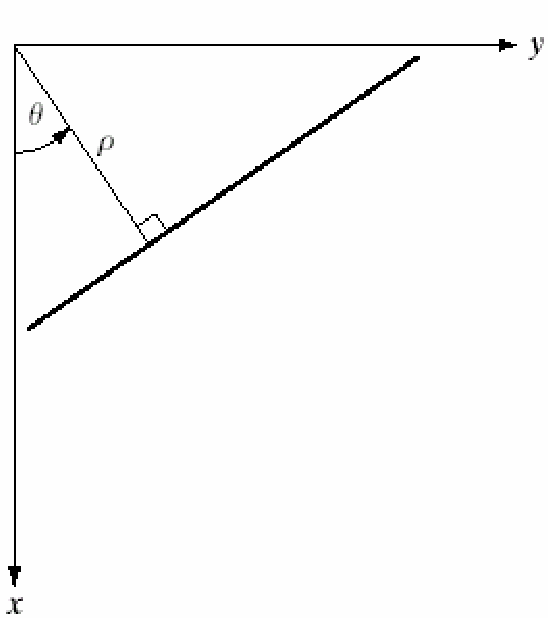
10.2 Edge Linking and Boundary detection
Hough Transform

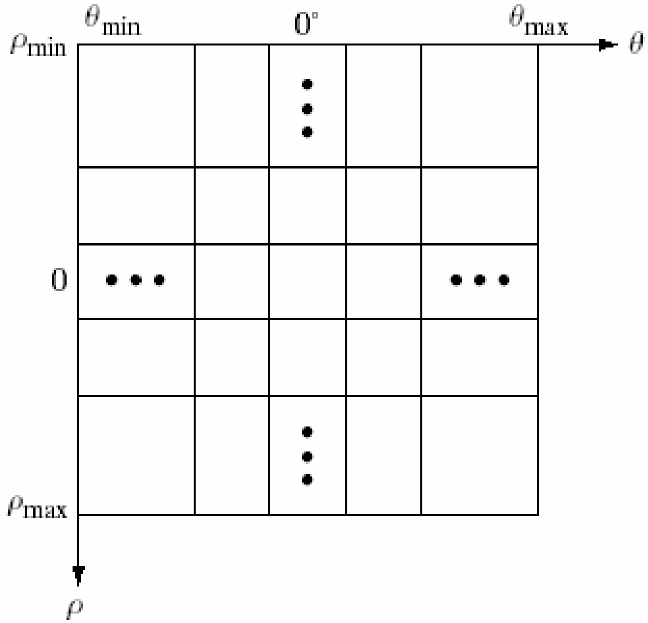
- Given a point $(x_i, y_i) \rightarrow$ many lines pass through this point as $x_i \cos\theta + y_i \sin\theta = \rho$ with different θ and ρ
- A point (x_i, y_i) in image space is mapped a set of points $\{(\theta, \rho)\}$ in parameter space
- The point $(x_j, y_j) \rightarrow$ many lines pass through this point as $x_j \cos\theta + y_j \sin\theta = \rho$ with different θ and ρ.
- The collinear point (x_j, y_j) in image space is mapped to another set of points $\{(\theta, \rho)'\}$ in parameter space.
- These two sets in parameter space intersect at (θ_j, ρ_i)
- Collinear points $(x_i, y_i) \in \{(x, y)\}$ lies on a line:

$$x \cos\theta_j + y \sin\theta_j = \rho_i$$

10.2 Edge Linking and Boundary detection

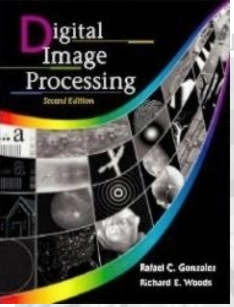
Representing a line as $x\cos\theta + y\sin\theta = \rho$





a b

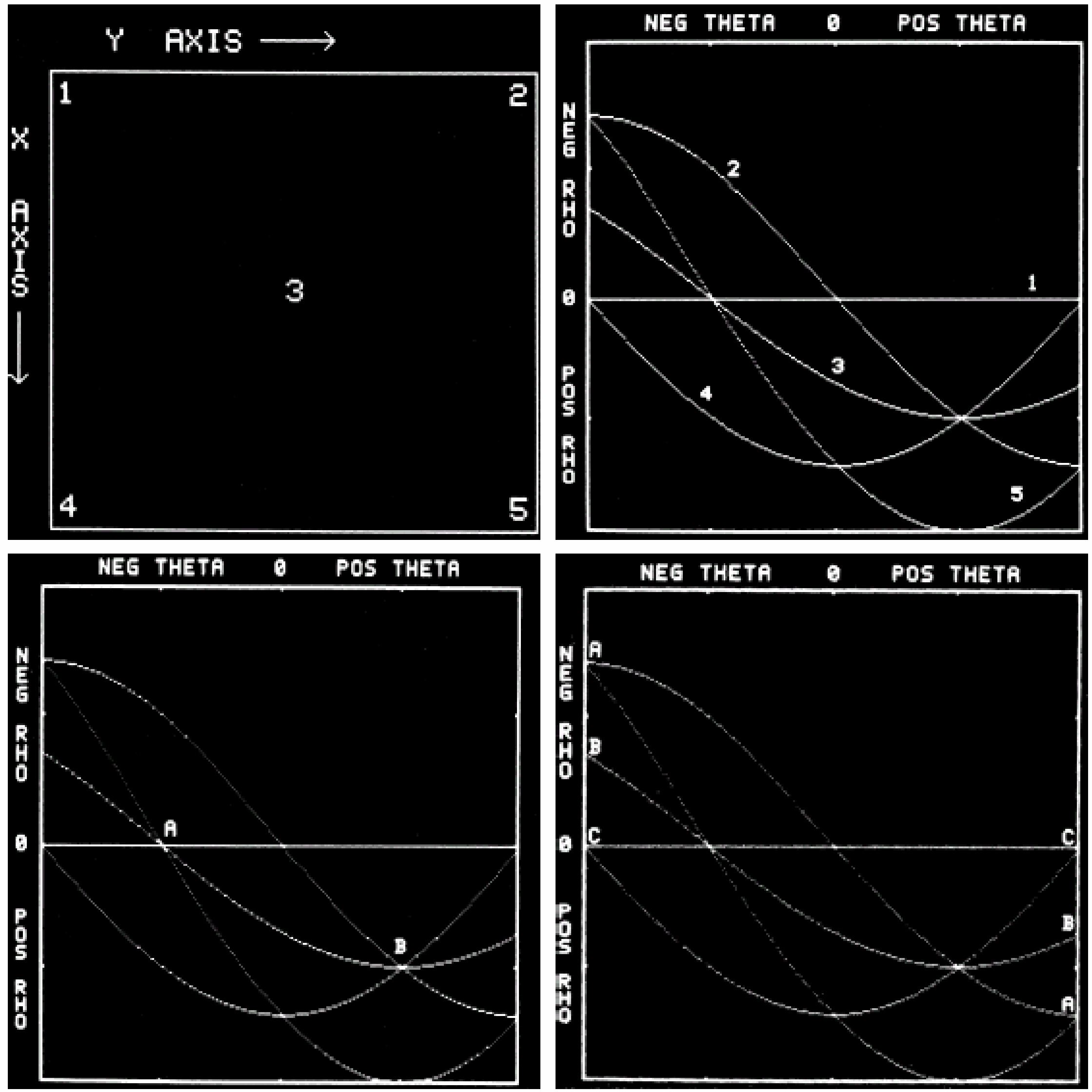
FIGURE 10.19 (a) Normal representation of a line. (b) Subdivision of the $\rho\theta$-plane into cells.



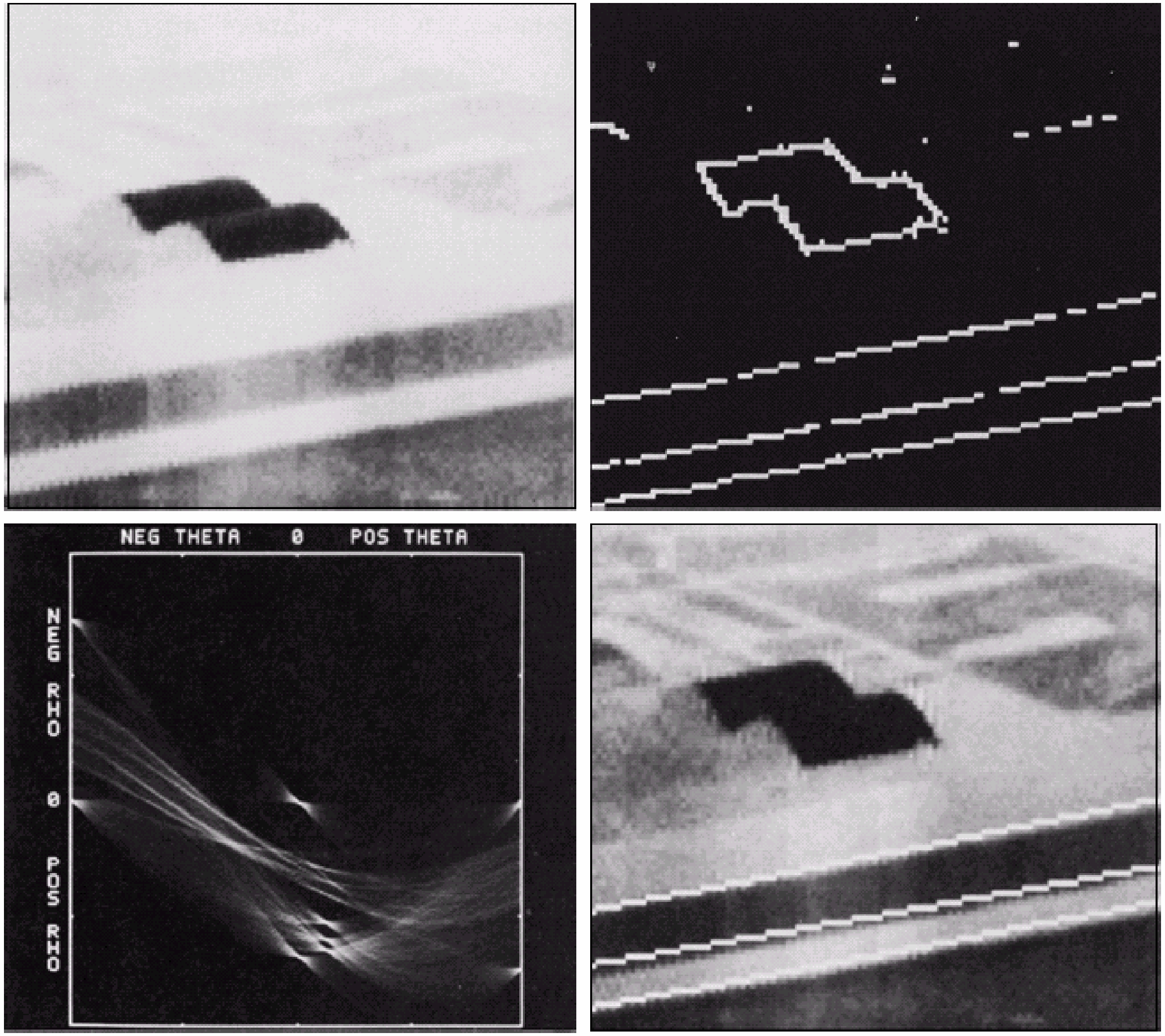
10.2 Edge Linking and Boundary detection Hough Transform

a b
c d

FIGURE 10.20 Illustration of the Hough transform. (Courtesy of Mr. D. R. Cate, Texas Instruments, Inc.)



10.2 Edge Linking and Boundary detection
Hough Transform



a b
c d

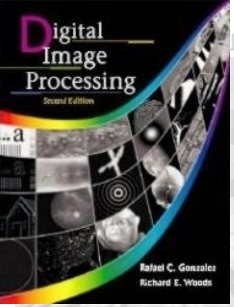
FIGURE 10.21 (a) Infrared image. (b) Thresholded gradient image. (c) Hough transform. (d) Linked pixels. (Courtesy of Mr. D. R. Cate, Texas Instruments, Inc.)

10.2 Edge Linking and Boundary detection
Hough Transform

- The Hough transform can be applied to any function of the form $g(\mathbf{v}, \mathbf{c})=0$ where $\mathbf{v}$ is the vector of coordinate and $\mathbf{c}$ is the vector of coefficients
- For example to detect a circle:

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

  where $\mathbf{v}=(x, y)$, $\mathbf{c}=(c_1, c_2, c_3)$
- Accumulator $A(i, j, k)=A(c_1, c_2, c_3)$



10.2 Edge Linking and Boundary detection
Hough Transform

- ***Circle detection***

1. Let V_p denote the set of point $\{\mathbf{v}|\mathrm{p}(\mathbf{v}) \neq 0\}$
2. For each image point $p(\mathbf{v})$, there is a set of a set of circles passing through $\mathbf{v}$. Let C_v denote the set of circles.
3. Find the center (c_1, c_2) and the radius c_3 of each member in C_v under the constraint that $\mathbf{v} \in V_p$.
4. For each member of $\{\mathbf{c}|\mathbf{v} \in V_p.\}$ an ***accumulator*** at (c_1, c_2, c_3) in $\mathbf{c}$ space is incremented by 1.

10.2 Edge Linking and Boundary detection Graph-theoretic Technique

- Representing the edge segments in the form of graph.
- Edge detection and linking based on searching the graph for low cost paths that correspond to the significant edges.
- Graph G=(N, U) with a set of nodes N, and a set U of unordered pairs of distinct elements of N.
- Each pair (n_i, n_j) of U is called an arc.
- An arc is directed from node n_i to node n_j , n_i, is the *parent;* n_j is the *successor*.
- Graph traveling starts with the start(root) node and ends with the goal node.
- A cost c(n_i, n_j) is associated with every arc (n_i, n_j) .

10.2 Edge Linking and Boundary detection Graph-theoretic Technique

- A sequence of node (n_1, n_2..., n_k) indicates a path from n_1 *to* n_k .
- The cost of entire path is $C = \sum_{i=2}^{K} c(n_{i-1}, n_i)$
- An edge element is boundary between two pixels *p* and *q*.
- An edge is a sequence of connected edge elements.

10.2 Edge Linking and Boundary detection Graph-theoretic Technique

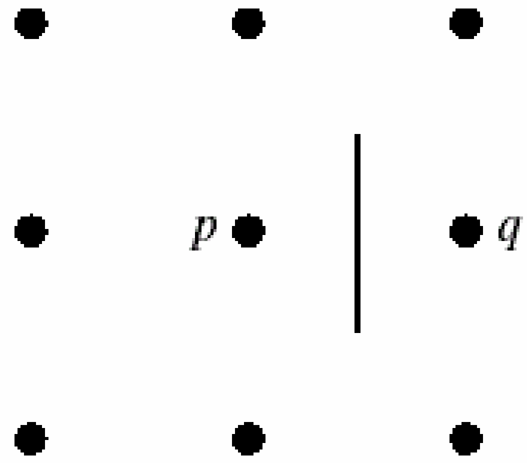


FIGURE 10.22 Edge element between pixels p and q.

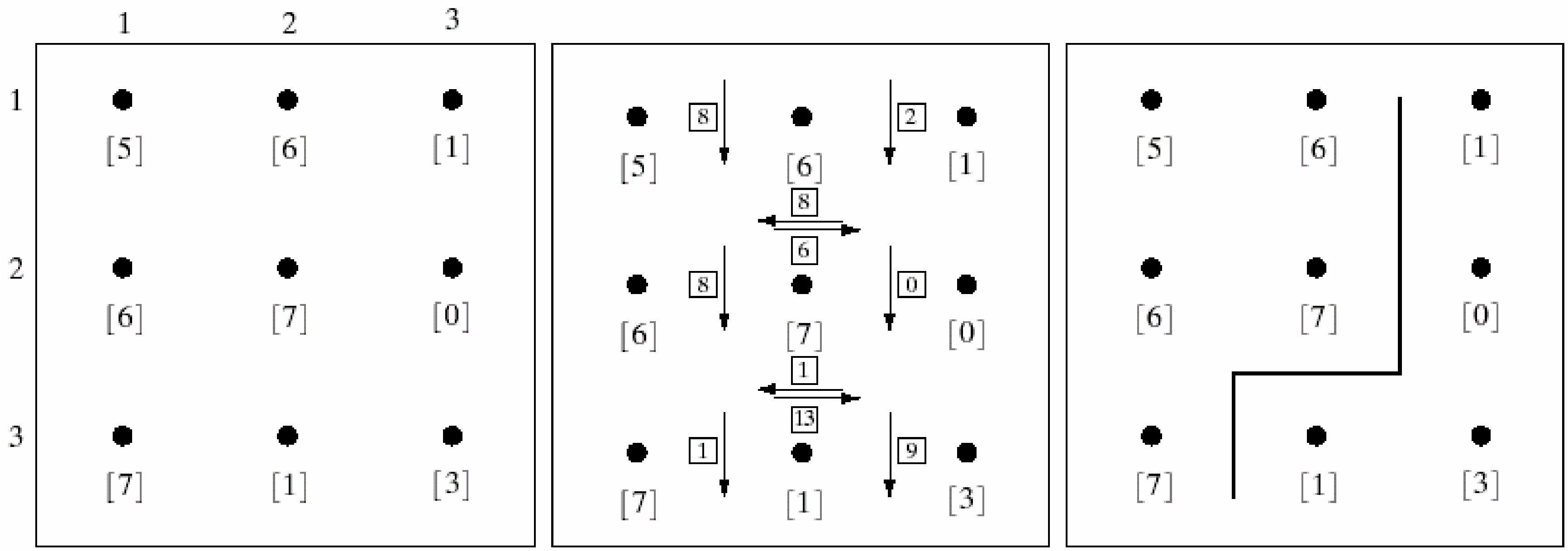
10.2 Edge Linking and Boundary detection Graph-theoretic Technique

- Each element define by pixels p and q has an associated cost defined as

$$c(p, q)=H-[f(p)-f(q)]$$

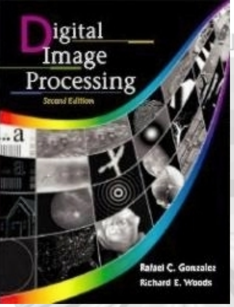
where H is the highest gray-level value in the image (*i.e.*, H=7), and $f(p)$ and $f(q)$ is the gray level of pixels p and q.

10.2 Edge Linking and Boundary detection Graph-theoretic Technique



a b c

FIGURE 10.23 (a) A 3 × 3 image region. (b) Edge segments and their costs. (c) Edge corresponding to the lowest-cost path in the graph shown in Fig. 10.24.



10.2 Edge Linking and Boundary detection
Graph-theoretic Technique

- Let $r(n)$ be an estimate of the cost of a minimum-cost path from ***s*** to ***n*** plus an estimate of the path from ***n*** to a ***goal*** node, that is $r(n)=g(n)+h(n)$, where $g(n)$ can be chosen as the ***lowest-cost path*** from ***s*** to ***n*** found ***so far***, and $h(n)$ is obtained by using any desirable heuristic information.

Graph search algorithm :

- Step 1: Mark the start node OPEN and set $g(s)=0$.
- Step 2: If no node is OPEN exit with failure, otherwise continue.
- Step 3: Mark CLOSED the OPEN node n whose estimate $r(n)$ is the smallest.
- Step 4: If n is a goal node, exit with solution path obtained by tracing back through the pointers; otherwise continue.

10.2 Edge Linking and Boundary detection Graph-theoretic Technique

- Step 5: Expand node n, generating all of its successors (if no successor go to step 2).
- Step 6: If a successor n_i is not marked, set $g(n_i)=g(n)+c(n, n_i)$, then mark it OPEN, and direct pointer from it back to n.
- Step 7: If a successor n_i is marked CLOSED or OPEN, update its value by letting

$$g'(n_i)=min[g(n_i), g(n)+c(n, n_i)].$$

Mark OPEN those CLOSED successors whose g' values were thus ***lowered*** and redirect to n, the pointers from all nodes whose g' values were lowered. Go to step 2.

- This algorithm depends on the use of heuristic function $h(n)$ which is a lower bound on the cost of the minimal-cost path.
- If h=0 then it is reduced to *uniform-cost algorithm*.

10.2 Edge Linking and Boundary detection Graph-theoretic Technique

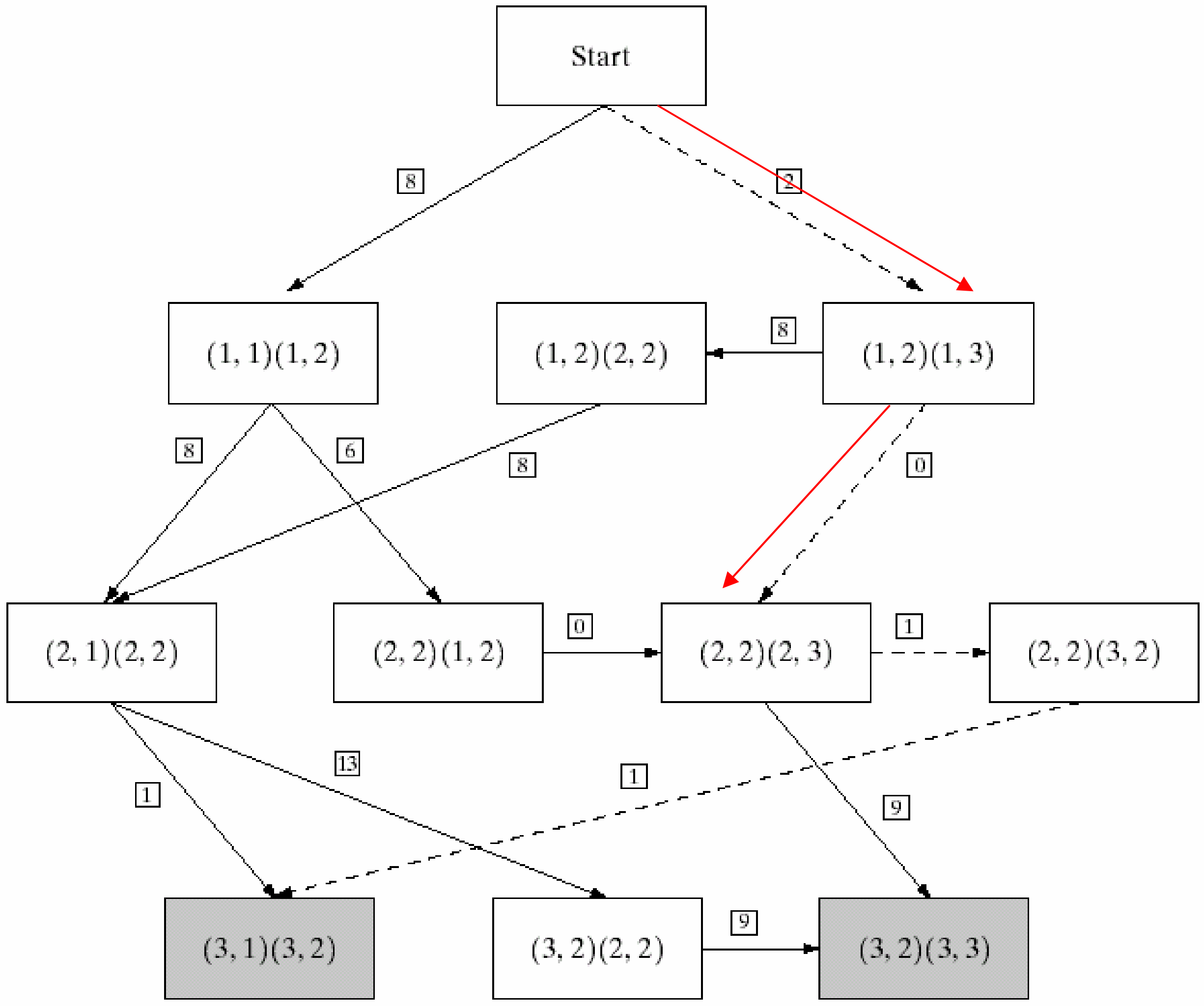


FIGURE 10.24 Graph for the image in Fig. 10.23(a). The lowest-cost path is shown dashed.

10.2 Edge Linking and Boundary detection Graph-theoretic Technique

FIGURE 10.25 Image of noisy chromosome silhouette and edge boundary (in white) determined by graph search.

10.3 Thresholding

- To extract an object form the background is to select a threshold T that separate the object pixels from background pixels.
- Any point (x, y) with $f(x, y)>T$ is called an object point; otherwise, the point is called a background point.
- For multilevel thresholding classifies a point (x, y) as belongs to one object class if $T_1<f(x, y)\leq T_2$, and to the other object class if $f(x, y)>T_2$.

10.3 Thresholding

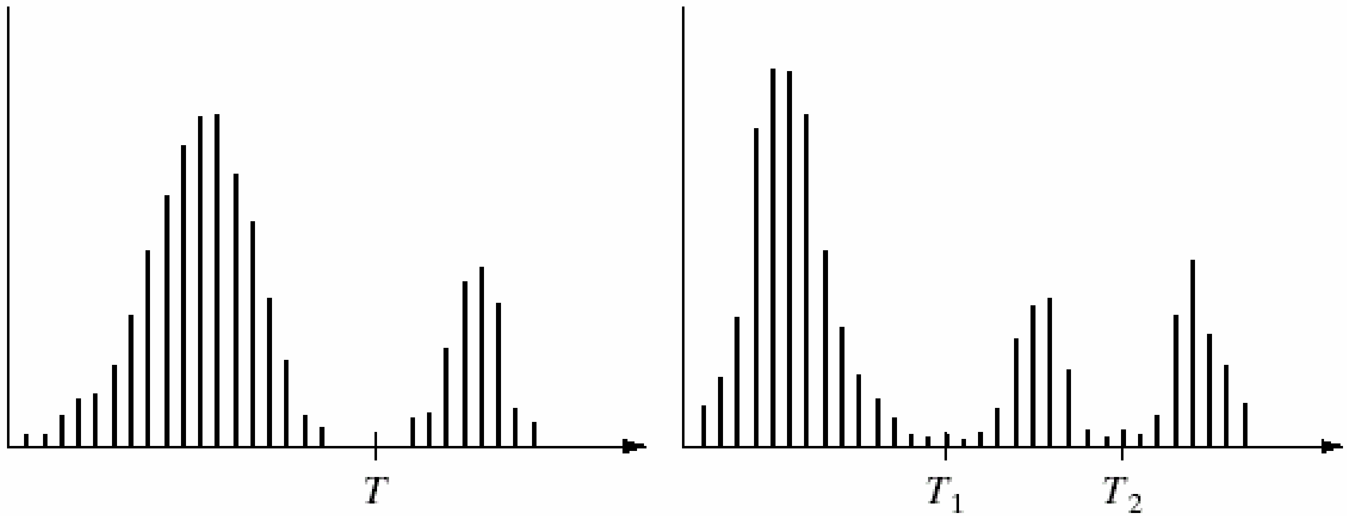


FIGURE 10.26 (a) Gray-level histograms that can be partitioned by (a) a single threshold, and (b) multiple thresholds.

10.3 Thresholding

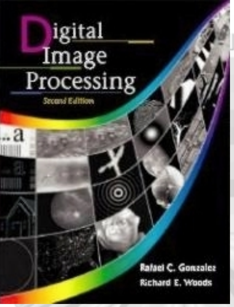
- The threshold T is determined as

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

  where $p(x, y)$ denotes some ***local property*** of this point (x, y), *i.e.*, the average level of a neighborhood centered on (x, y).
- A thresholded image is defined as:

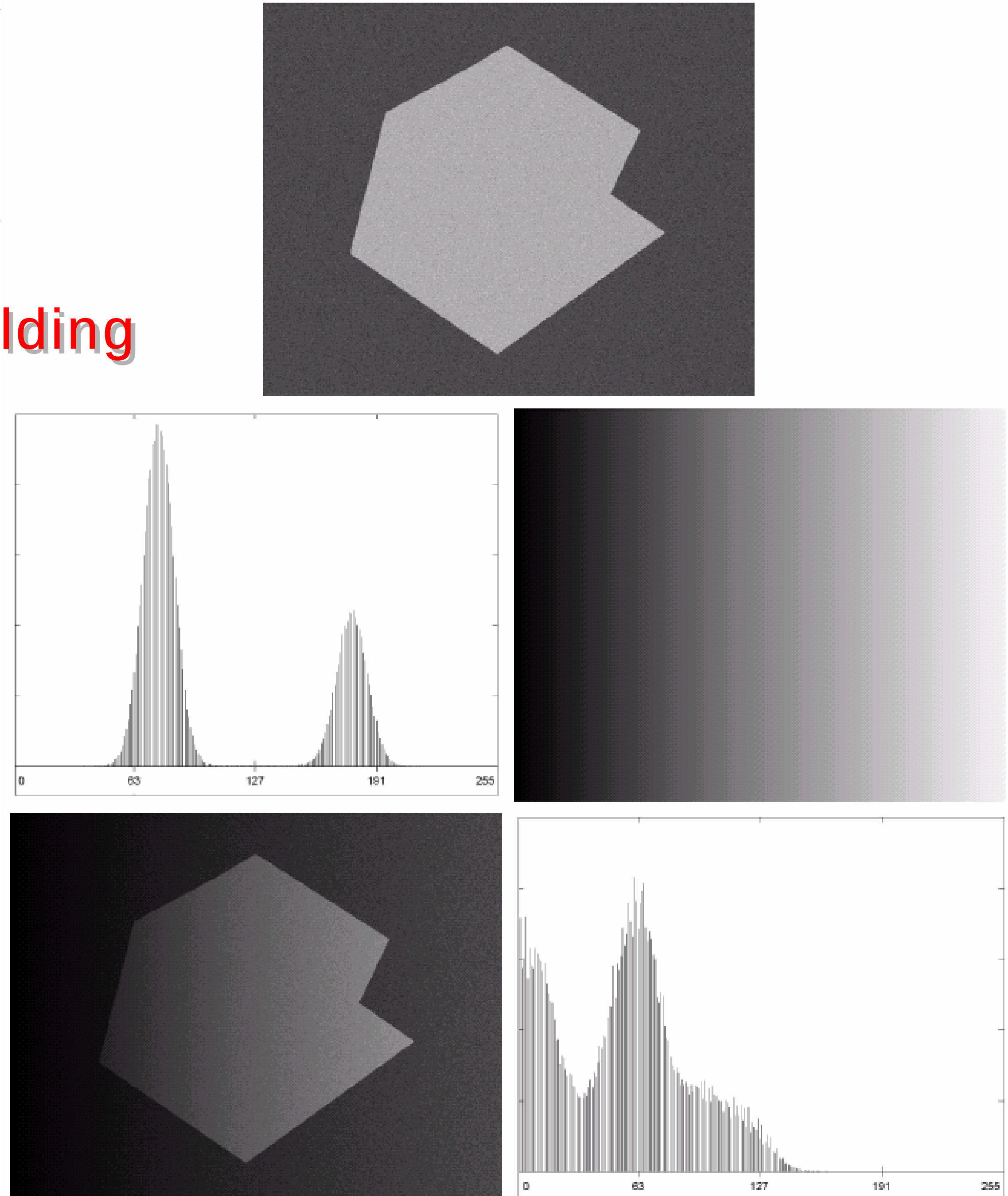
  $$g(x, y) = 1 \text{ if } f(x, y) \geq T$$

  $$g(x, y) = 0 \text{ if } f(x, y) < T$$
- If T does not depend on $p(x, y)$ then the threshold is called *global* threshold, otherwise it is called *local or adaptive* threshold.



10.3 Thresholding

Histogram distortion due to non-uniform illumination



a
b c
d e

FIGURE 10.27 (a) Computer generated reflectance function. (b) Histogram of reflectance function. (c) Computer generated illumination function. (d) Product of (a) and (c). (e) Histogram of product image.

10.3 Thresholding -illumination

- $f(x, y)=i(x, y)r(x, y)$
- $z(x, y)=ln\{f(x, y)\}=ln\{i(x, y)\}+ln\{r(x,y)\}$

  $=i'(x, y)+r'(x, y)$
- If $i'(x, y)$ and $r'(x, y)$ are ***independent random variables***, the histogram of $z(x, y)$ is given by the ***convolution*** of the histogram of $i'(x, y)$ and $r'(x, y)$.
- If $i'(x, y)$=constant and $i'(x, y)$=constant (its histogram is an impulse), then the histogram of $z(x, y) \approx r'(x, y)$ is unchanged.
- If the $i'(x, y)$ is a broader histogram (nonuniform illumination), the convolution process may smear the histogram of $r'(x, y)$ and the shape of the histogram of $z(x, y)$ will be quite different from $r'(x, y)$.
- The degree of distortion depends on the broadness of the histogram of $i'(x, y)$.

10.3 Thresholding -illumination

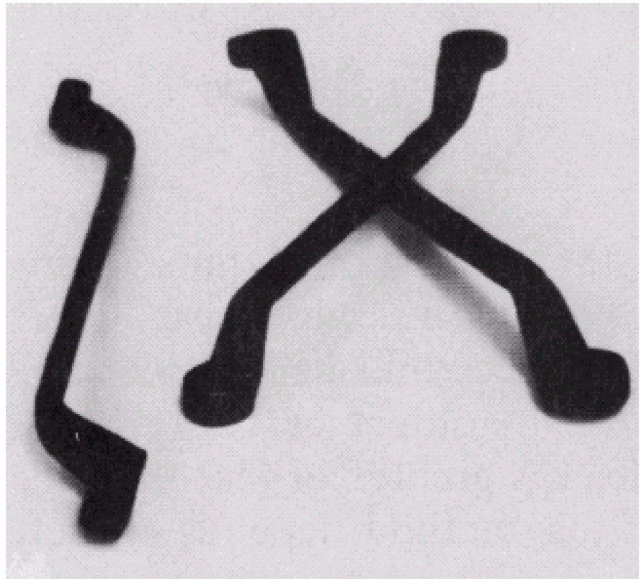
- If the illumination source $i(x, y)$ is available, compensating the non-uniformity by projecting the illumination on a constant white reflective surface ($i.e.$, $r(x, y)=k$) . This yields a new image as

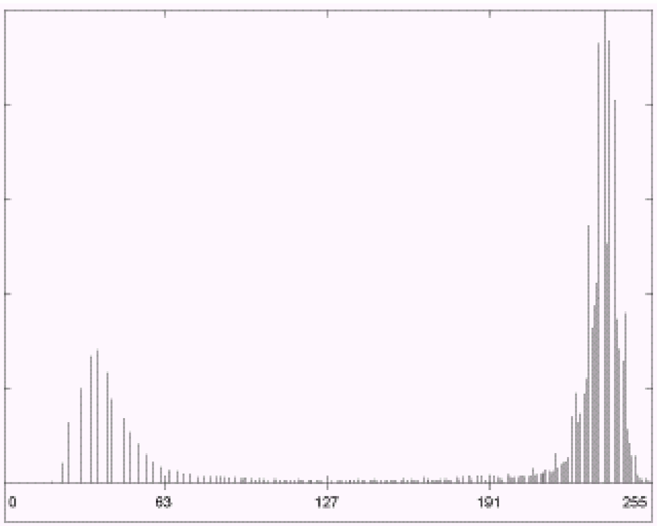
  $g(x, y)=ki(x, y)$ where k is a constant
- For any image $f(x, y)=i(x, y)r(x, y)$, we have a normalized image as $h(x, y)=f(x, y)/g(x, y)=r(x, y)/k$
- If $r(x, y)$ can be segmented by threshold T then $h(x, y)$ can be segmented by threshold T/k.

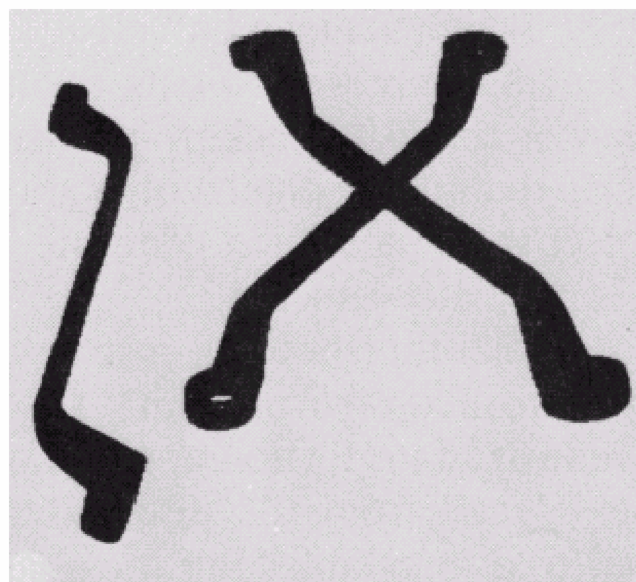
10.3 Thresholding - Global Thresholding

1. Partition the image using a single threshold T

2. How to find the best T ?







a
b c

FIGURE 10.28 (a) Original image. (b) Image histogram. (c) Result of global thresholding with T midway between the maximum and minimum gray levels.

10.3 Thresholding –Global Thresholding

- Assume that the background and the object occupy comparable areas in the image, a good initial value of T is the average gray level of the image.

1. Select an initial estimate for T.
2. Segment the image into two group of pixels G1 and G2 using T.
3. Compute the average gray level values of G1 and G2 are μ1 and μ2.
4. Compute a new threshold value as T_{new}=(μ1+μ2)/2,
5. Compare if $|T_{new}-T|>z_0$ (predefined thredhold z_0) then $T=T_{new}$ and go to step 2, else stop

10.3 Thresholding – Otsu Thresholding

- ***Minimizing within group variance***
- Let P(i) denote the probability distribution of the gray level $i=1,\ldots I$ of a picture.
- Let t be the ***threshold*** that separate the image pixels into two groups, $\{1,\ldots.t\}$ and $\{t+1,\ldots.I\}$
- q_1(t) be the probability for group with values less than or equal to t, *i.e.*,

$$q_1(t) = \sum_{i=1}^{t} P(i)$$

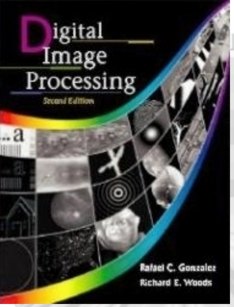
- q_2(t) be the probability for group with values greater than t, *i.e.*,

$$q_2(t) = \sum_{i=t+1}^{I} P(i)$$

10.3 Thresholding – Otsu Thresholding

- Let $\mu_1(t)$ and $\mu_2(t)$ be the mean for group 1 and group 2 as $\mu_1(t)=\sum_{i=1}^{t} iP(i)/q_1(t)$ $\mu_2(t)=\sum_{i=t+1}^{I} iP(i)/q_2(t)$
- Let $\sigma_1(t)$ and $\sigma_2(t)$ be the variance for group 1 and group 2 as $\sigma_1^2(t)=\sum_{i=1}^{t}[i-\mu_1(t)]^2 P(i)/q_1(t)$ and $\sigma_2^2(t)=\sum_{i=1}^{t}[i-\mu_2(t)]^2 P(i)/q_2(t)$
- Let σ_W be the weighted sum of group variance (or ***within group variance***) , *i.e.*,

$$\sigma_w^2(t)=q_1(t)\sigma_1^2(t)+q_2(t)\sigma_2^2(t)$$



10.3 Thresholding – Otsu Thresholding

- The relationship between the σ and σ_W

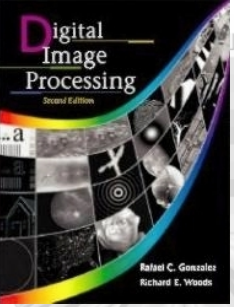
$$\sigma^2(t) = \sum_{i=1}^{I} [i - \mu]^2 P(i)$$

$$= \sum_{i=1}^{t} [i - \mu_1(t) + \mu_1(t) - \mu]^2 P(i) + \sum_{i=t+1}^{I} [i - \mu_2(t) + \mu_2(t) - \mu]^2 P(i)$$

$$= \sigma_w^2(t) + q_1(t)[1 - q_1(t)][\mu_1(t) - \mu_2(t)] = \sigma_w^2(t) + \sigma_B^2(t)$$

- σ_B is the ***between group variance***
- Minimize σ_W = maximize σ_B
- There is a relation ship between the value of computed *t* and that computed for next *t*: *t+1*.

$$q_1(t+1) = q_1(t) + P(t+1) \quad \text{and} \quad q_2(t+1) = q_2(t) - P(t+1)$$



10.3 Thresholding – Otsu Thresholding

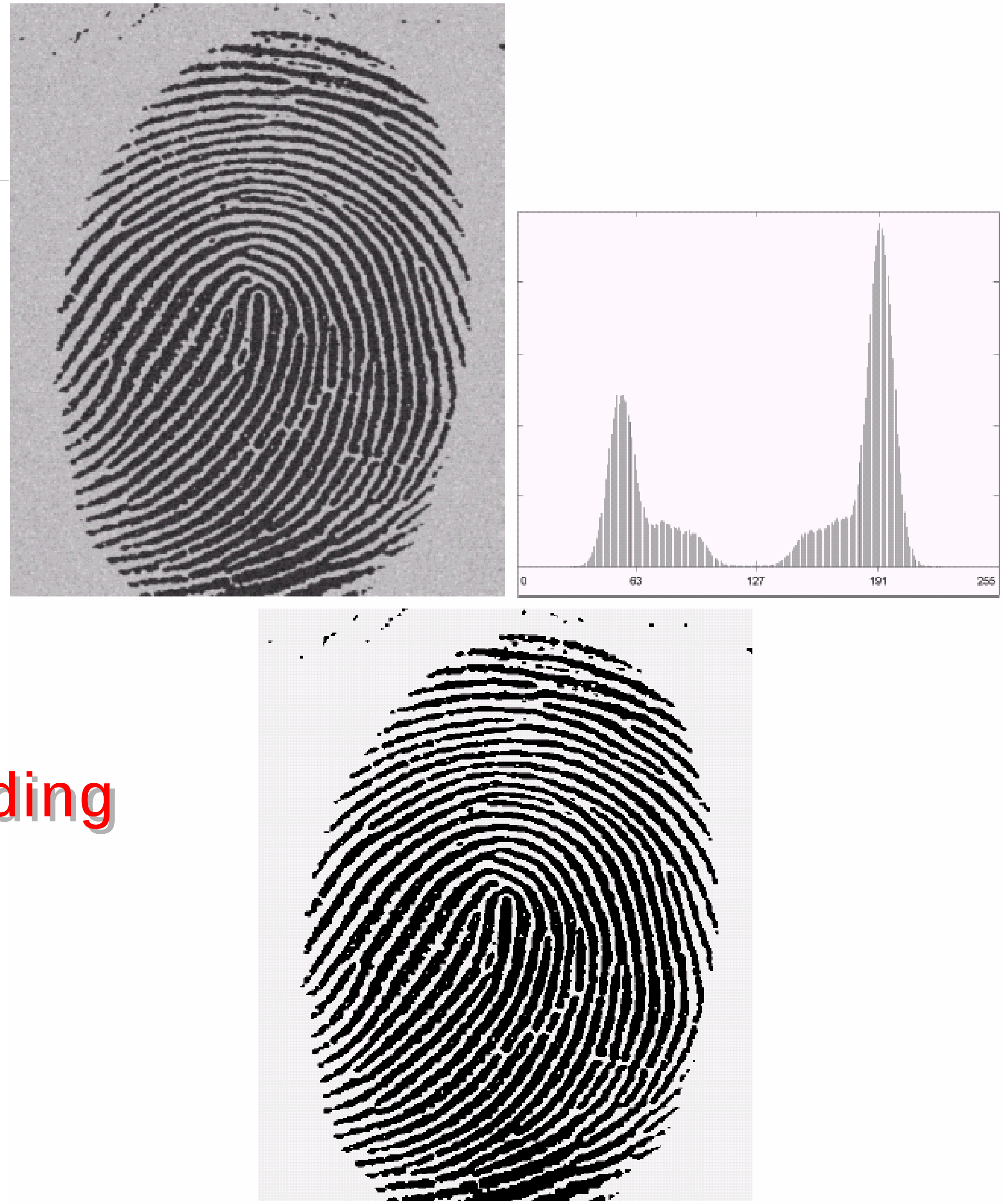
- We can obtain the recursive relation

$$\mu_1(t+1) = \frac{\sum_{i=1}^{t} iP(i) + (t+1)P(t+1)}{q_1(t+1)} = \frac{q_1(t)\mu_1(t) + (t+1)P(t+1)}{q_1(t+1)}$$

$$\mu_2(t+1) = \frac{q_2(t)\mu_2(t) - (t+1)P(t+1)}{q_2(t+1)} = \frac{\mu - q_1(t)\mu_1(t) - (t+1)P(t+1)}{1 - q_1(t+1)}$$

$$= \frac{\mu - q_1(t+1)\mu_1(t+1)}{1 - q_1(t+1)}$$

10.3 Thresholding



a b
c

FIGURE 10.29 (a) Original image. (b) Image histogram. (c) Result of segmentation with the threshold estimated by iteration. (Original courtesy of the National Institute of Standards and Technology.)

10.3 Thresholding- Adaptive threshold

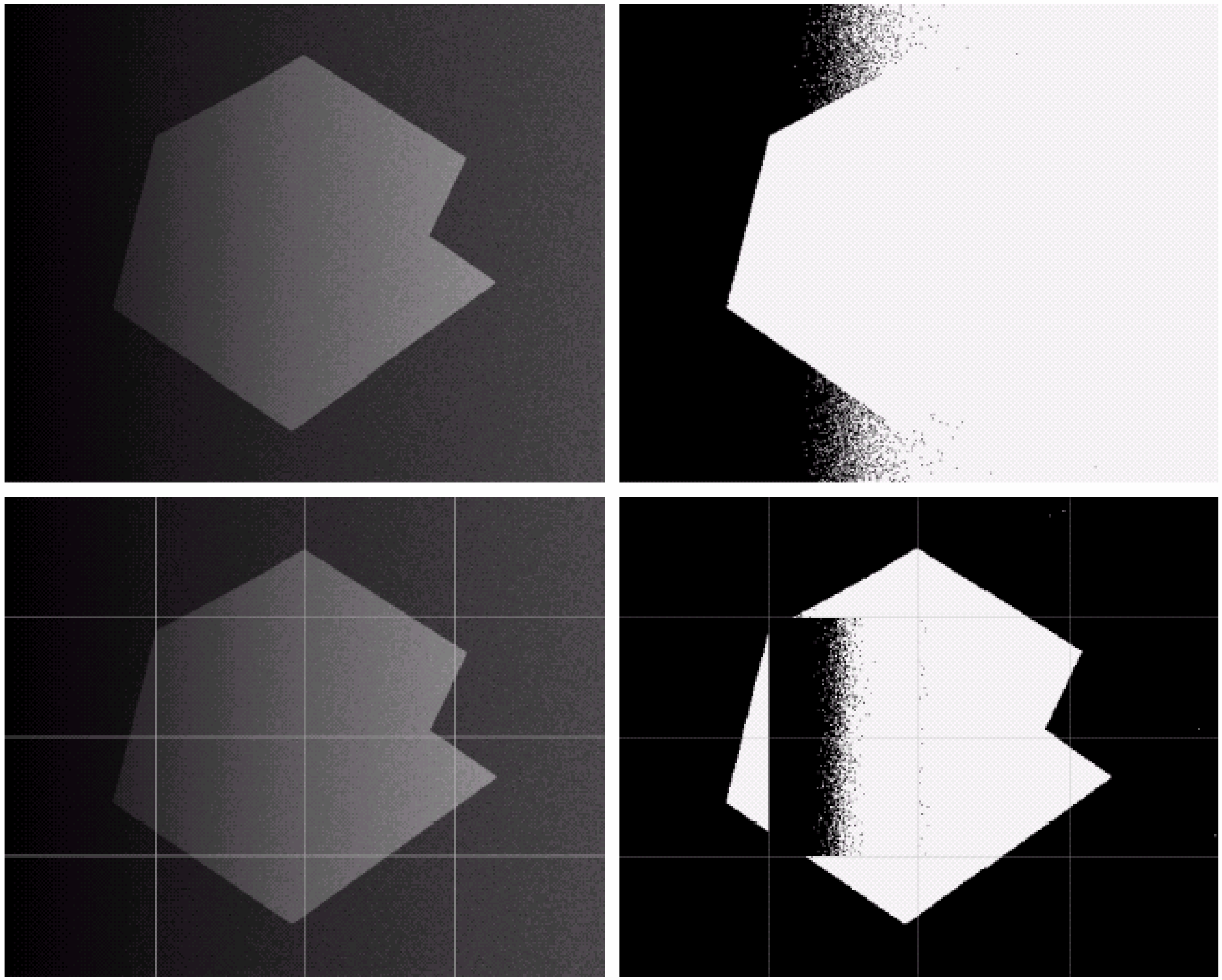
- Divide the image into sub-images.
- All sub-images containing ***boundaries***,

  its variances>100, else its variance <75.
- Each sub-image with variance > 100 are segmented with a threshold computed for that specific sub-image.
- All sub-images with variance <100 are treated as one ***composite image*** which is segmented with a single threshold.

10.3 Thresholding

a b
c d

FIGURE 10.30 (a) Original image. (b) Result of global thresholding. (c) Image subdivided into individual subimages. (d) Result of adaptive thresholding.



10.3 Thresholding

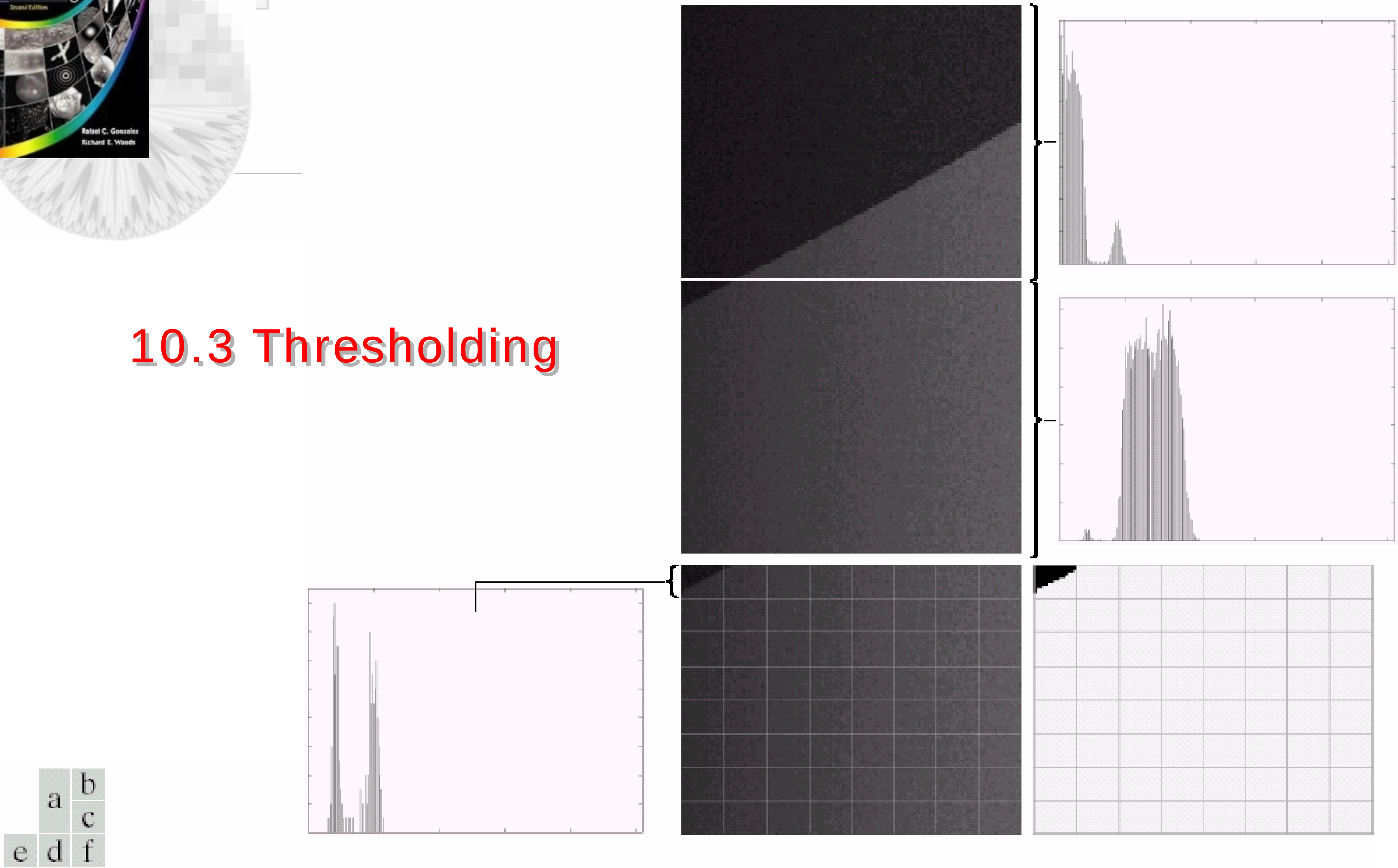
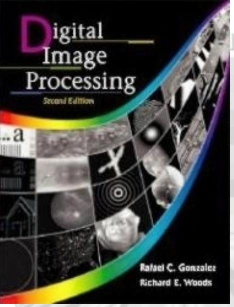


FIGURE 10.31 (a) Properly and improperly segmented subimages from Fig. 10.30. (b)–(c) Corresponding histograms. (d) Further subdivision of the improperly segmented subimage. (e) Histogram of small subimage at top, left. (f) Result of adaptively segmenting (d).



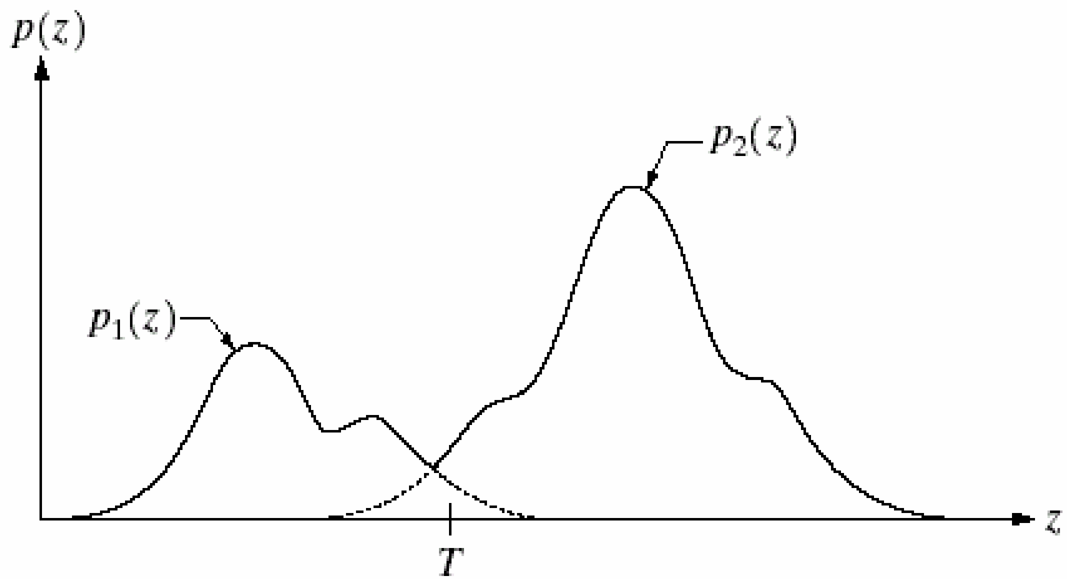
10.3 Thresholding -Optimal Global and Adaptive Thresholding

- Suppose an image contains only two principal gray-level regions: ***objects*** and ***background.***
- Let *z* denote the gray level, which can be treated as a random variable.
- The histogram of the image may be treated as the probability density function(PDF) $p(z)$.
- There are two PDFs, one for the objects, $p_1(z)$, and one for the background $p_2(z)$.
- The mixture probability density function describing the overall gray-level variation in the image as

$$p(z)=P_1p_1(z)+P_2p_2(z)$$

10.3 Thresholding
-Optimal Global and Adaptive Thresholding

FIGURE 10.32 Gray-level probability density functions of two regions in an image.



10.3 Thresholding
-Optimal Global and Adaptive Thresholding

- P_1 and P_2 are the probability of occurrence of the two classes of pixels.
- P_1 is the probability that the pixel is an object pixel and P_2 is the probability that the pixel is a background pixel, and $P_1+P_2=1$.
- The threshold is T, the pixel with $z>T$ is classified as background pixel, and vise versa.
- The probability of erroneously classifying a background pixel as an object pixel is

$$E_1(T)=\int_{-\infty}^{T} p_2(z)dz$$

10.3 Thresholding -Optimal Global and Adaptive Thresholding

- The probability of erroneously classifying a object pixel as an background pixel is

$$E_2(T) = \int_T^{\infty} p_1(z)dz$$

- The overall probability of error is

$$E(T)=P_2E_1(T)+P_1E_2(T)$$

- To find the T that minimize $E(T)$ by $dE(T)/dT=0$, and we have $P_1p_1(T)=P_2p_2(T)$
- If $P_1=P_2$ then we find the T at $p_1(T)=p_2(T)$
- Assume $p_1(z)$and $p_2(z)$ are Gaussian distribution with the mean and standard variation as (μ_1, σ_1^2) and (μ_2, σ_2^2).

10.3 Thresholding
-Optimal Global and Adaptive Thresholding

- The threshold T is found by the following equation: $AT^2+BT+C=0$

where $A=\sigma_1^2-\sigma_2^2$

$B=2(\mu_1\sigma_2^2-\mu_2\sigma_1^2)$

$C=\sigma_1^2\mu_2^2-\sigma_2^2\mu_1^2+2\sigma_1^2 2\sigma_2^2 \ln(\sigma_2 P_1/\sigma_1 P_2)$

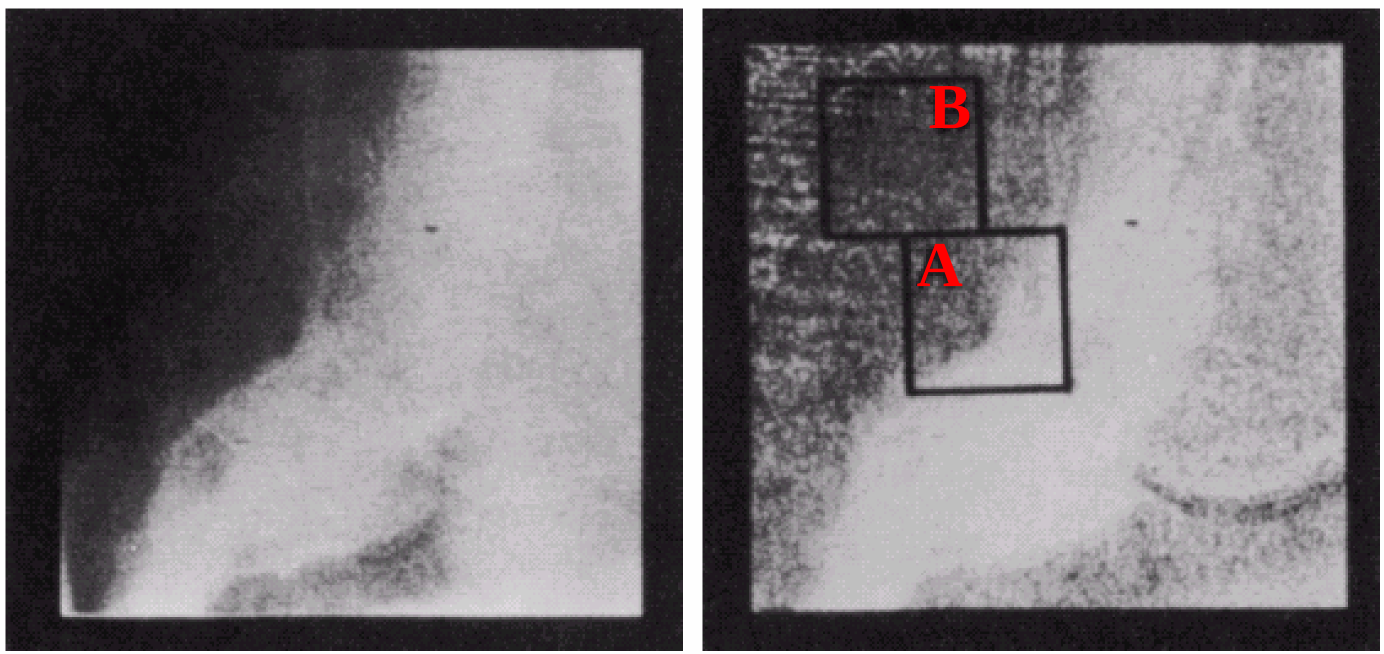
- If the variances are equal $\sigma^2=\sigma_1{}^2=\sigma_2{}^2$ *then*

$$T=\frac{\mu_1+\mu_2}{2}+\frac{\sigma^2}{\mu_1-\mu_2}\ln\left(\frac{P_2}{P_1}\right)$$

10.3 Thresholding-example

Example: To outlines the boundary of heart ventricles in cardioangiograms(X-ray image)

After preprocessing



a b

FIGURE 10.33 A cardioangiogram before and after preprocessing. (Chow and Kaneko.)

10.3 Thresholding-example

- ***Preprocessing:***

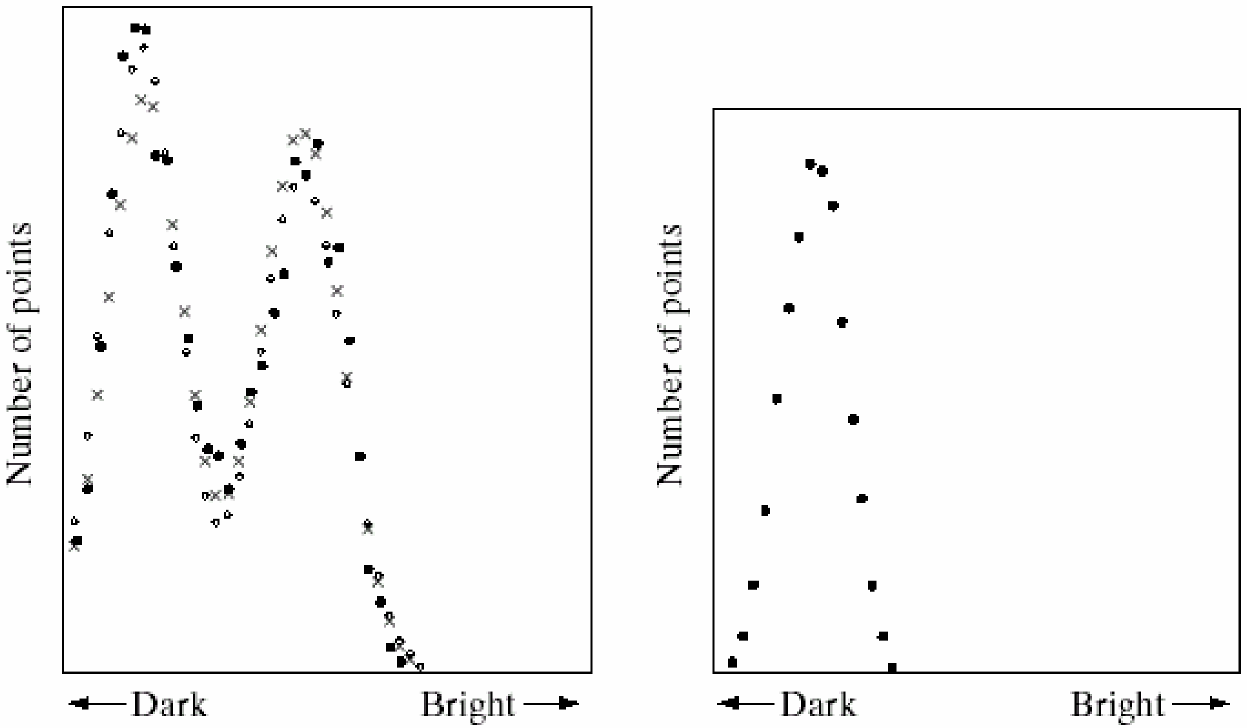
1. Each pixel is mapped with a log function, *i.e.*, $s=c\cdot\log(1+r)$, where c is a constant,

   to counter exponential effect of radio absorption.
2. Image (after radioactive absorption) subtracts Image (before radioactive absorption).
3. Several angiograms are summed to reduce the noise.

10.3 Thresholding -example

- ***Segmentation***:
- Image (256x256) is subdivided into 49 overlapped regions (64x64).
- All 49 histograms are computed.
- Test of bimodality and thresholding (Fig. 10.34(a)) is found by fitting the bimodal Gaussian density curve (region A)
- The threshold for the remaining regions (region B) were obtained by interpolating these thresholds.
- Every point (x, y) in the image will be assigned a threshold T_{xy}.
- $f(x, y)=1$ if $f(x, y)>T_{xy}$, otherwise $f(x, y)=0$

10.3 Thresholding



a b

FIGURE 10.34 Histograms (black dots) of (a) region *A*, and (b) region *B* in Fig. 10.33(b). (Chow and Kaneko.)

10.3 Thresholding

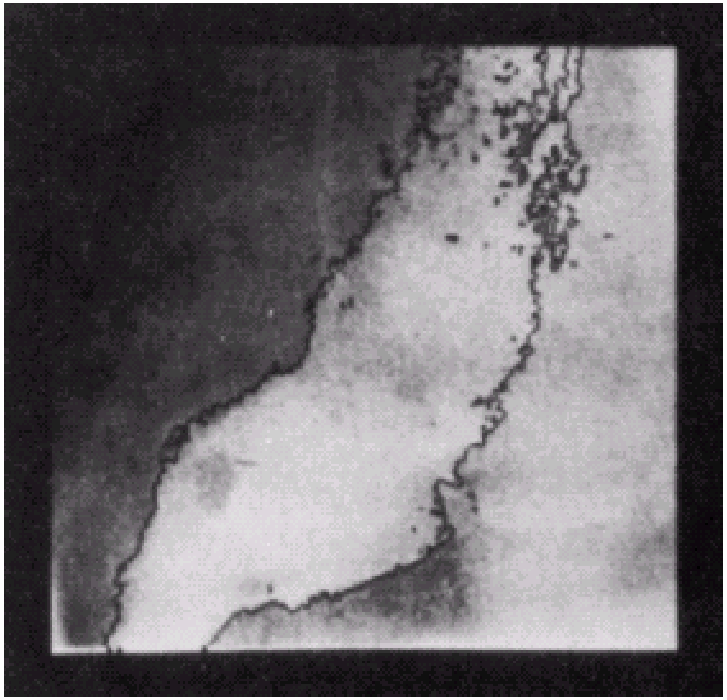


FIGURE 10.35 Cardioangiogram showing superimposed boundaries. (Chow and Kaneko.)

10.3 Thresholding - Histogram improvement and Local thresholding

- Good histogram shape →good thresholding
- To improve the shape of histograms is to consider only those pixels that lie on or near the edges between objects and background.
- The histogram will be less dependent on relative size of the objects and background.
- Use gradient to find the pixel on an edge or not
- Laplacian can yield information regarding whether a give pixel lies on the dark or light side of an edge.

10.3 Thresholding - Histogram improvement and Local thresholding

- Consider only those pixels that lie on or near the edges between objects and background
- Three-level image

$$s(x,y)=\begin{cases} 0 & if \quad \nabla f < T \\ + & if \nabla f \geq T \; and \; \nabla^2 f \geq 0 \\ - & if \nabla f \geq T \; and \; \nabla^2 f < 0 \end{cases}$$

- For pixels not on edges, $s(x, y)=0$
- For pixels on dark side of edges, $s(x, y)=+$
- For pixels not on light side of edges, $s(x, y)= -$
- Fig. 10.36 shows a dark object on light background.

10.3 Thresholding

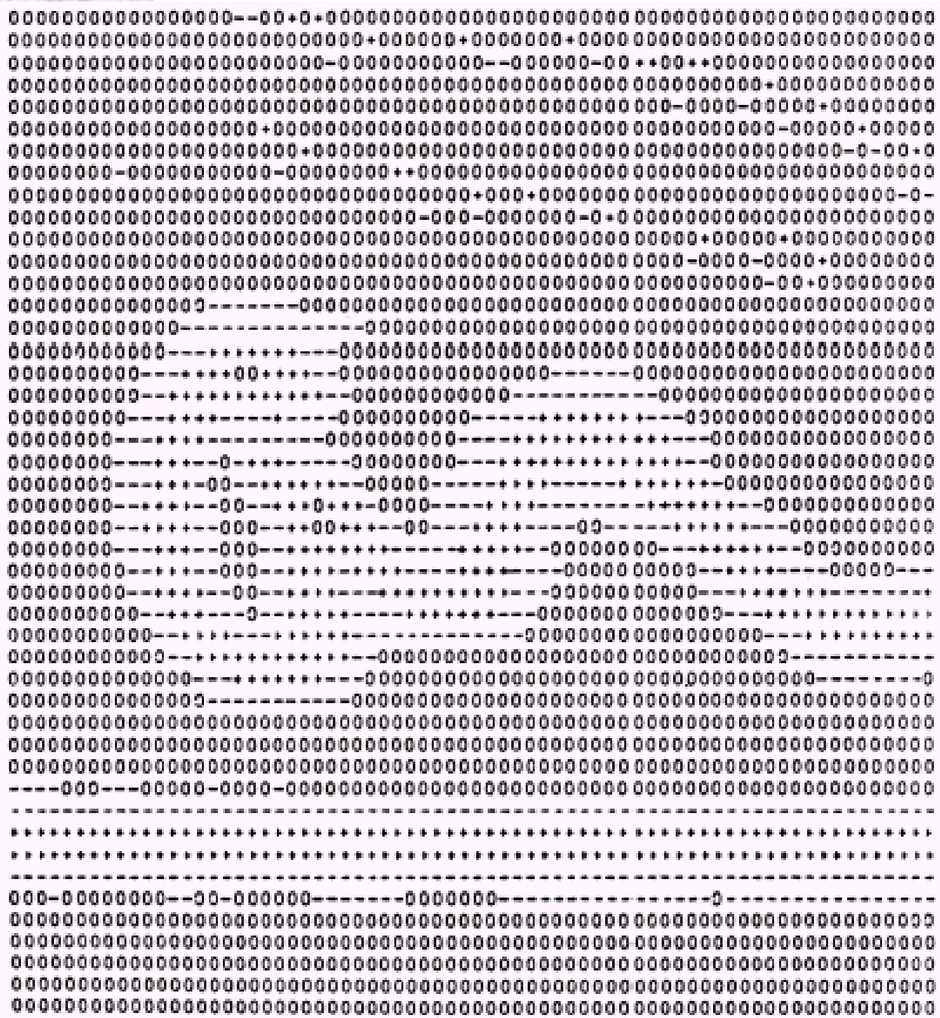


FIGURE 10.36 Image of a handwritten stroke coded by using Eq. (10.3-16). (Courtesy of IBM Corporation.)

10.3 Thresholding

- For binary image, the transition from the ***light background*** to ***dark object*** is characterized by the occurrence of a "–" followed by a "+".
- The ***interior*** of the object is characterized as "+" or "0".
- The transition from the object back to the background is characterized by the occurrence of a "+" followed by a "–".
- The vertical or horizontal scan line containing a section of an object has the following structure:

  (…..)(–, +)(+ or 0)(+, –)(….)
- (…..) represents any combination of +, –, and 0.

10.3 Thresholding -example

a
b

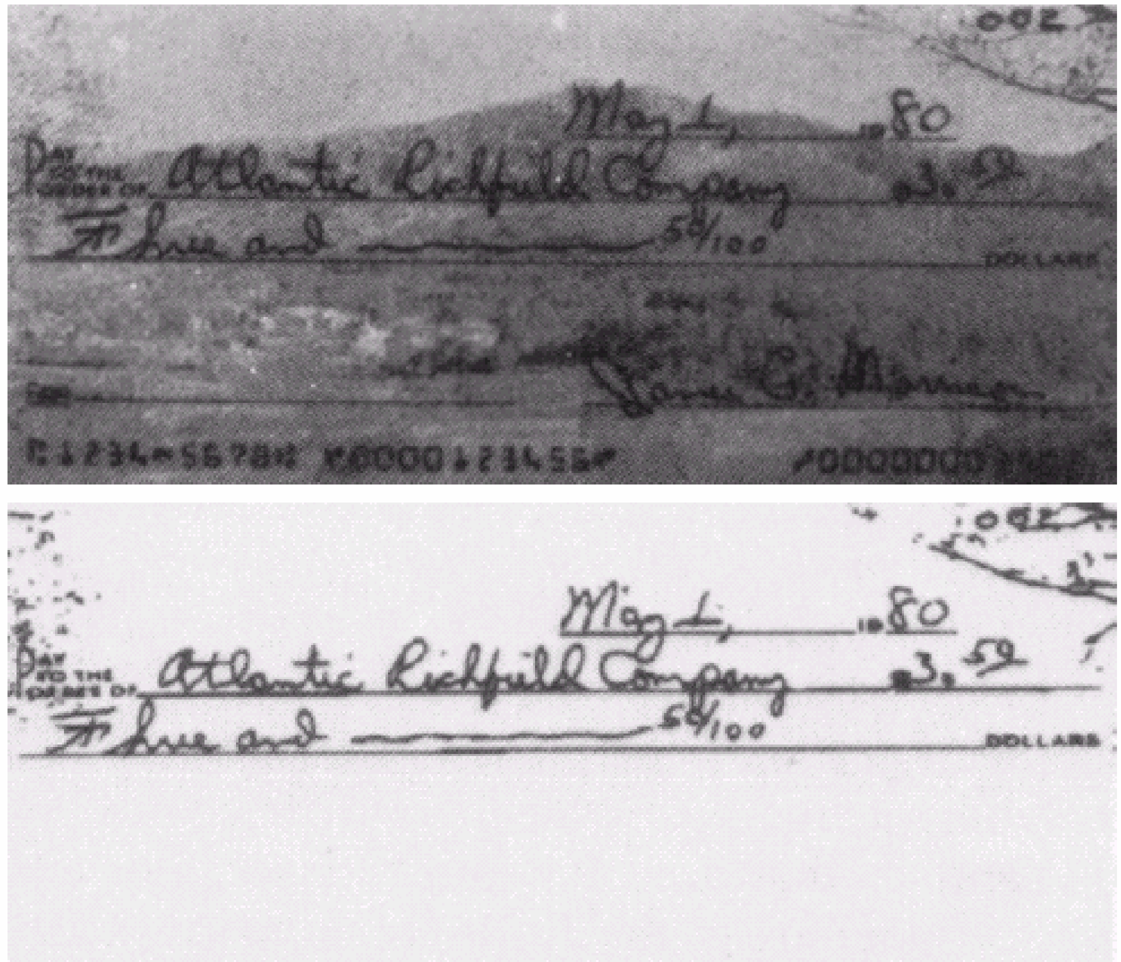
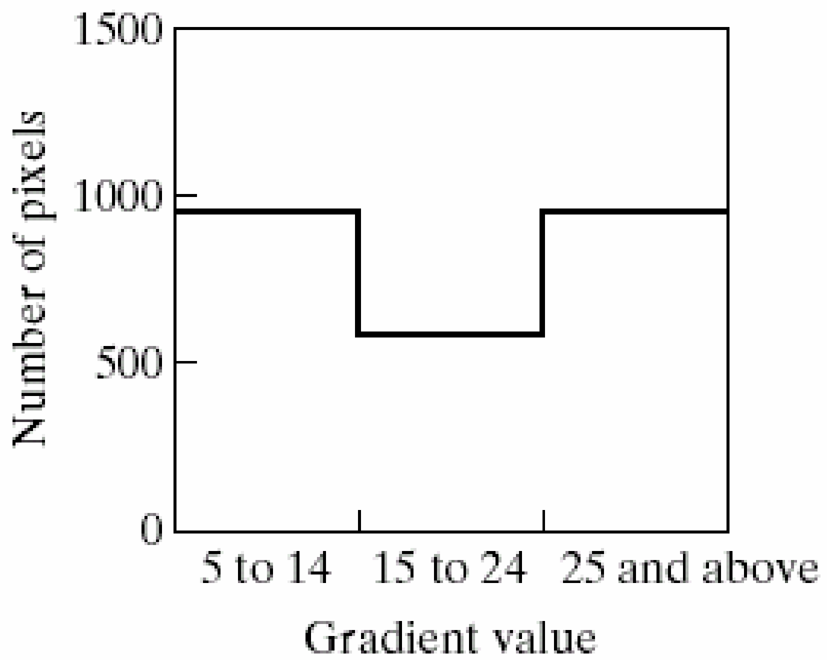


FIGURE 10.37 (a) Original image. (b) Image segmented by local thresholding. (Courtesy of IBM Corporation.)

10.3 Thresholding-example

FIGURE 10.38 Histogram of pixels with gradients greater than 5. (Courtesy of IBM Corporation.)



10.3 Thresholding- Based on several variables

- Multi-spectral thresholding for RGB color image
- Finding clusters of points in 3-D space.
- Image segmentation: if the pixel value is close to one cluster then assign one value (cluster centroid) to the pixel.

a b c

FIGURE 10.39 (a) Original color image shown as a monochrome picture. (b) Segmentation of pixels with colors close to facial tones. (c) Segmentation of red components.

10.4 Region-based Segmentation

- Segmentation is accomplished by finding the region directly.
- Segmentation is to partition the image into sub-regions: R_1, R_2…..R_n ,where

(a) R_i is a connected region

(b) $R_1 \cup R_2 \ldots\ldots \cup R_n = R$

(c) $R_i \cap R_j = \varnothing$ for $i \neq j$

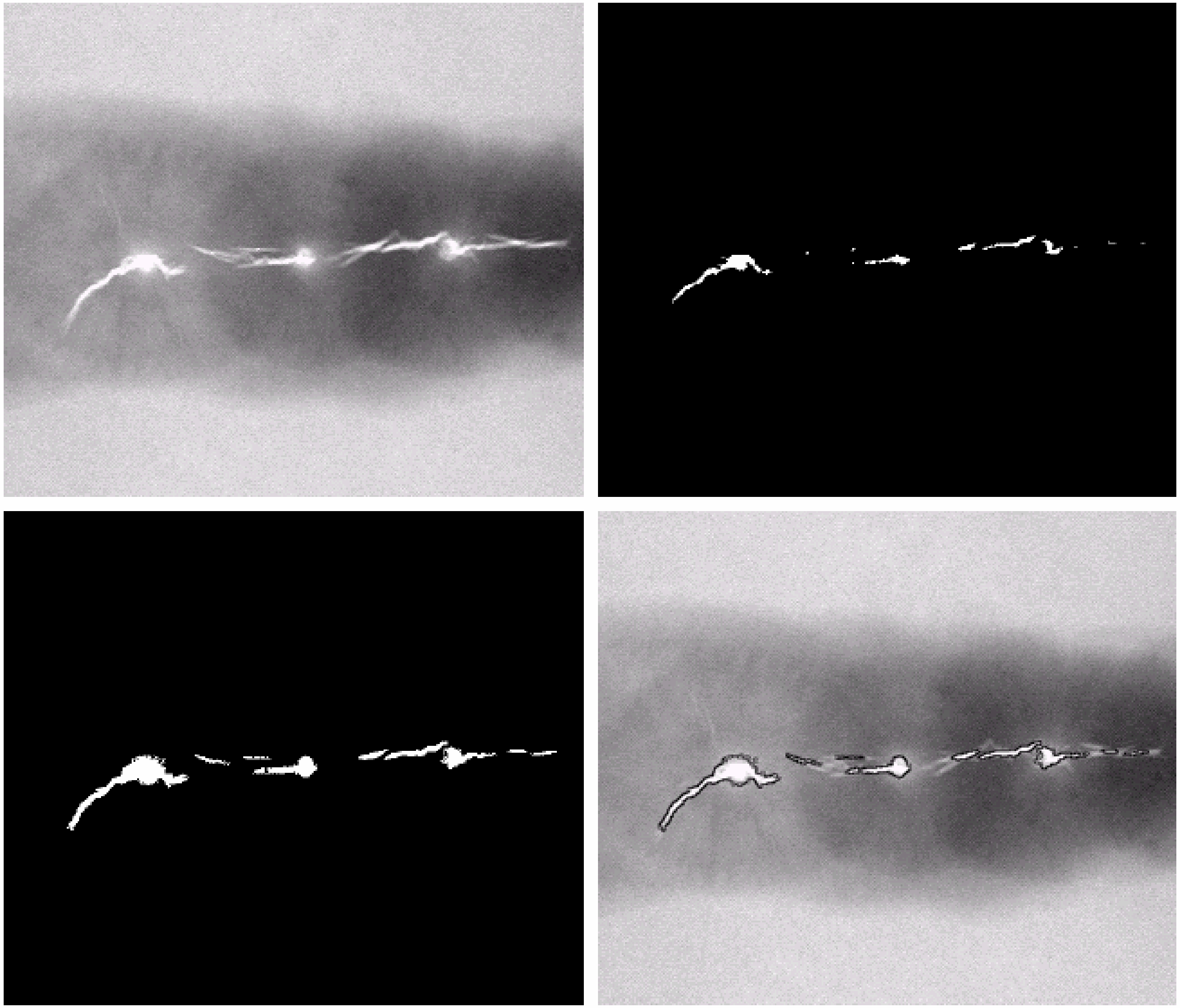
(d) $P(R_i)$=TRUE, all pixel in R_i have ***the same gray level or texture***.

(e) $P(R_i \cup R_j)$= FALSE for $i \neq j$

10.4 Region-based Segmentation –Region groing

- Region growing is a procedure that groups pixels or subregions into larger regions based on predefined criteria.
- It starts with a set of"seed" points and from these grow regions by appending to each seed those neighboring pixels that have properties similar to the seed (such as specified gray-level or color)

10.4 Region-based Segmentation



a b
c d

FIGURE 10.40 (a) Image showing defective welds. (b) Seed points. (c) Result of region growing. (d) Boundaries of segmented defective welds (in black). (Original image courtesy of X-TEK Systems, Ltd.).

10.4 Region-based Segmentation

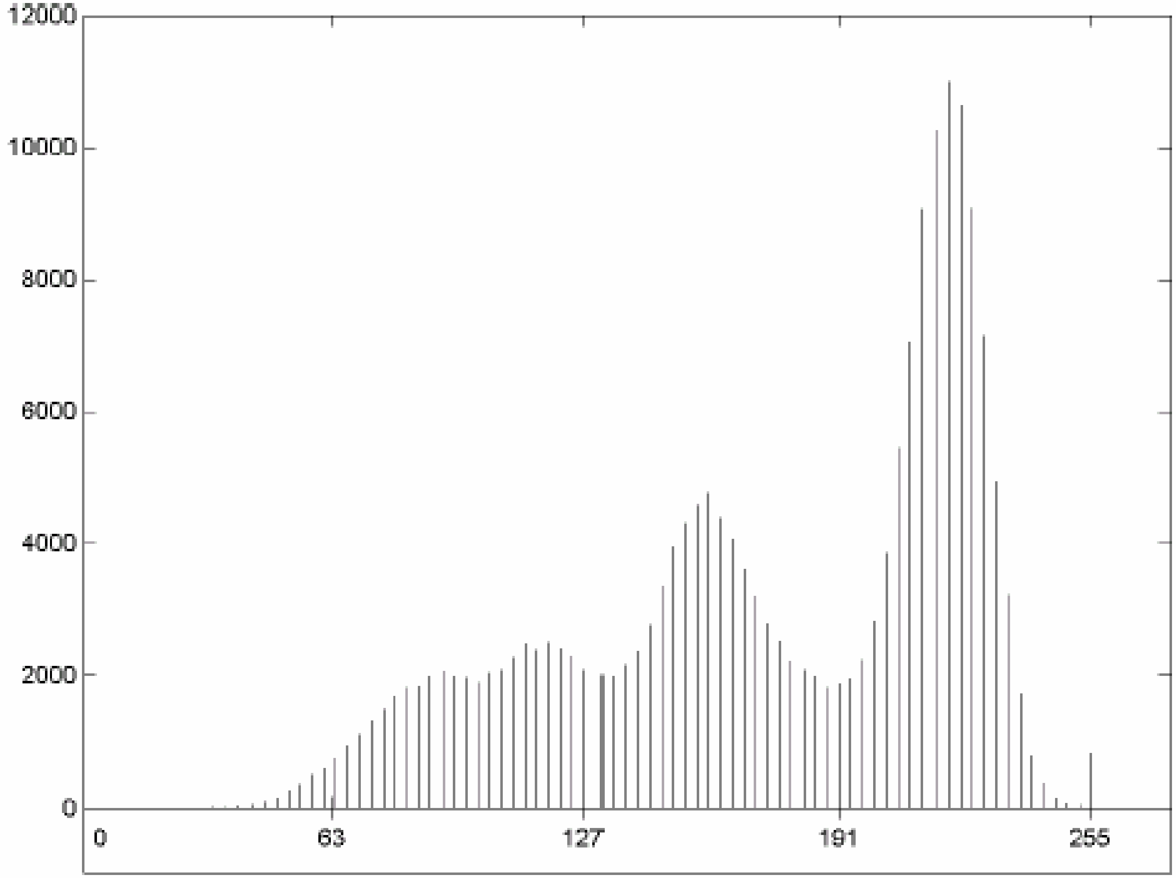


FIGURE 10.41 Histogram of Fig. 10.40(a).

10.4 Region-based Segmentation-Region splitting and merging

- Subdivide an image initially into a set of arbitrary disjointed regions and then merge and/or split the regions in an attempt to satisfy the conditions of regions.
- Two adjacent regions R_i and R_j are merged only if $P(R_i \cup R_j) = \text{TRUE}$.

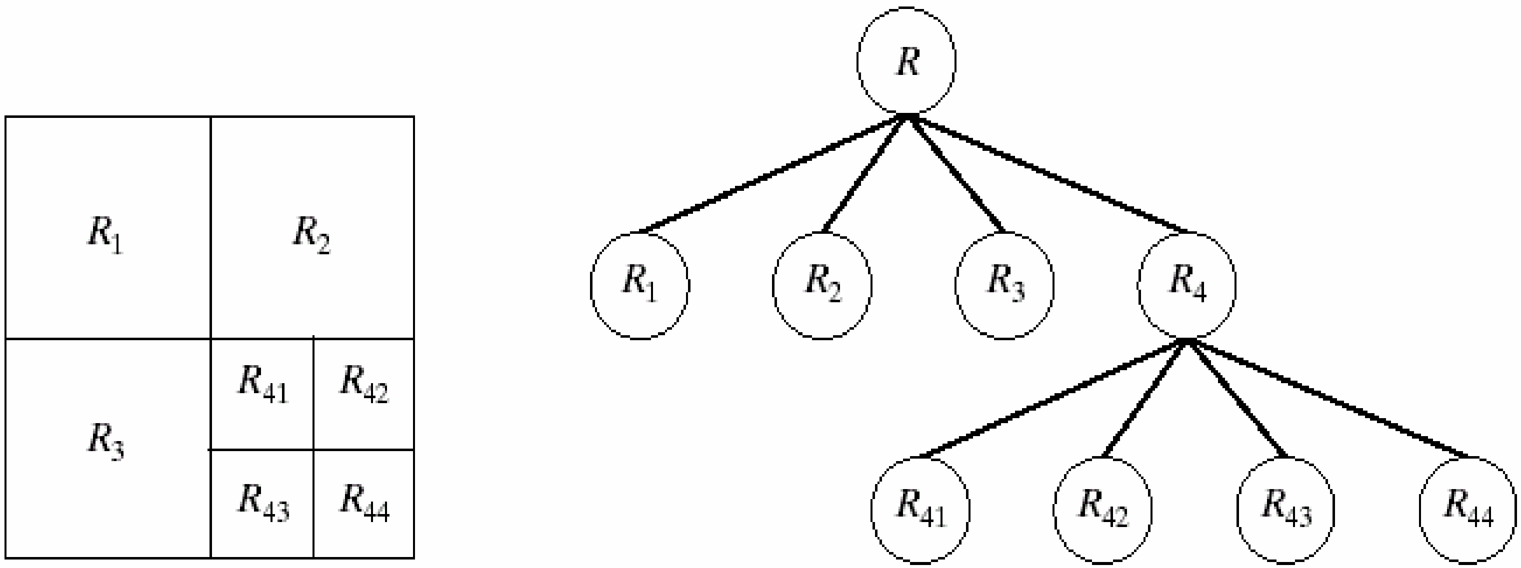
  The split and merge algorithm is mentioned as follows:

  1. Split into four disjoint quadrants any region R_i for which $P(R_i) = \text{FALSE}$
  2. Merge any adjacent regions R_i and R_j, for which $P(R_i \cup R_j) = \text{TRUE}$
  3. Stop when no further merging or splitting is possible.

10.4 Region-based Segmentation
-Region splitting and merging

a b

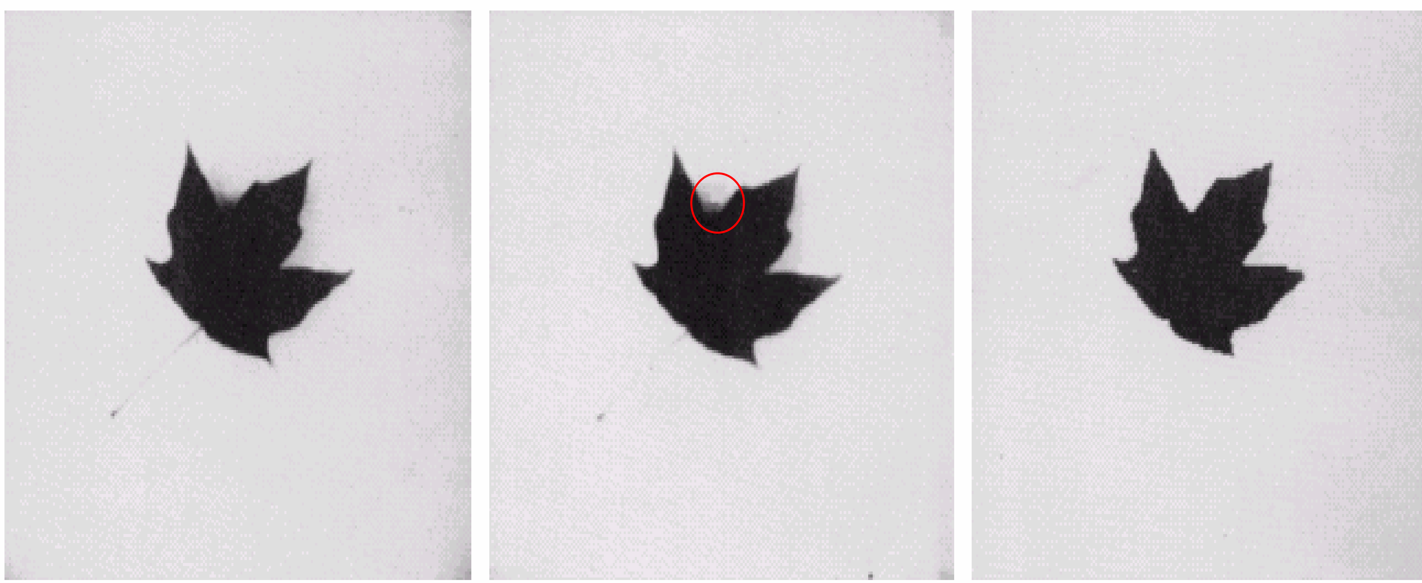
FIGURE 10.42
(a) Partitioned image.
(b) Corresponding quadtree.



10.4 Region-Based Segmentation -Region splitting and merging

1. Define $P(R_i)$=True if at least 80% of the pixel in R_i have the property $|z_i - m_i| \leq 2\sigma_i$, m_i is the mean, σ_i is the standard deviation.
2. If $P(R_i)$=True , the value of all pixels in R_i are set to m_i

The shaded area is errorneously removed.



a b c

FIGURE 10.43 (a) Original image. (b) Result of split and merge procedure. (c) Result of thresholding (a).

10.5 Segmentation by Morphological Watersheds

- Segmentation based on (1) detection of discontinuity (2) thresholding, and (3) region processing.
- Segmentation by Morphological Watersheds embodies the concepts of the three approaches.
- Produce more stable segmentation results, *i.e.*, continuous segmentation boundary.
- Incorporate knowledge-based based constraints in the segmentation process.

10.5.1 Segmentation by Morphological Watersheds- basic

- Visual image in 3-D (coordinate and gray-level) and consider three types of points:

  1) points belong to regional minimum.

  2) points at which a drop of water, if placed at the location of any of these points, would fall with certainty to ***a single minimum***. A set of such points is called *catchment basin*（盆地）or *watershed* of that minimum.

  3) points at which water would be equally likely to fall ***more than one*** such minimum. A set of such points is called *divide lines* or *watershed lines*（分水嶺）.

10.5.1 Segmentation by Morphological Watersheds

- ***Segmentation*** → to find the watershed lines.
- The entire topography is flooded from below by letting the water rise at a uniform rate.
- The rising water in distinct catchment basins is about to merge, a ***dam*** is built to prevent this merging.
- The flooding will reach a stage when only the tops of the dam are visible above the water line.
- The dam boundaries correspond to the divide lines of the watersheds.

10.5 Segmentation by Morphological Watersheds

a b
c d

FIGURE 10.44 (a) Original image. (b) Topographic view. (c)–(d) Two stages of flooding.

10.5 Segmentation by Morphological Watersheds

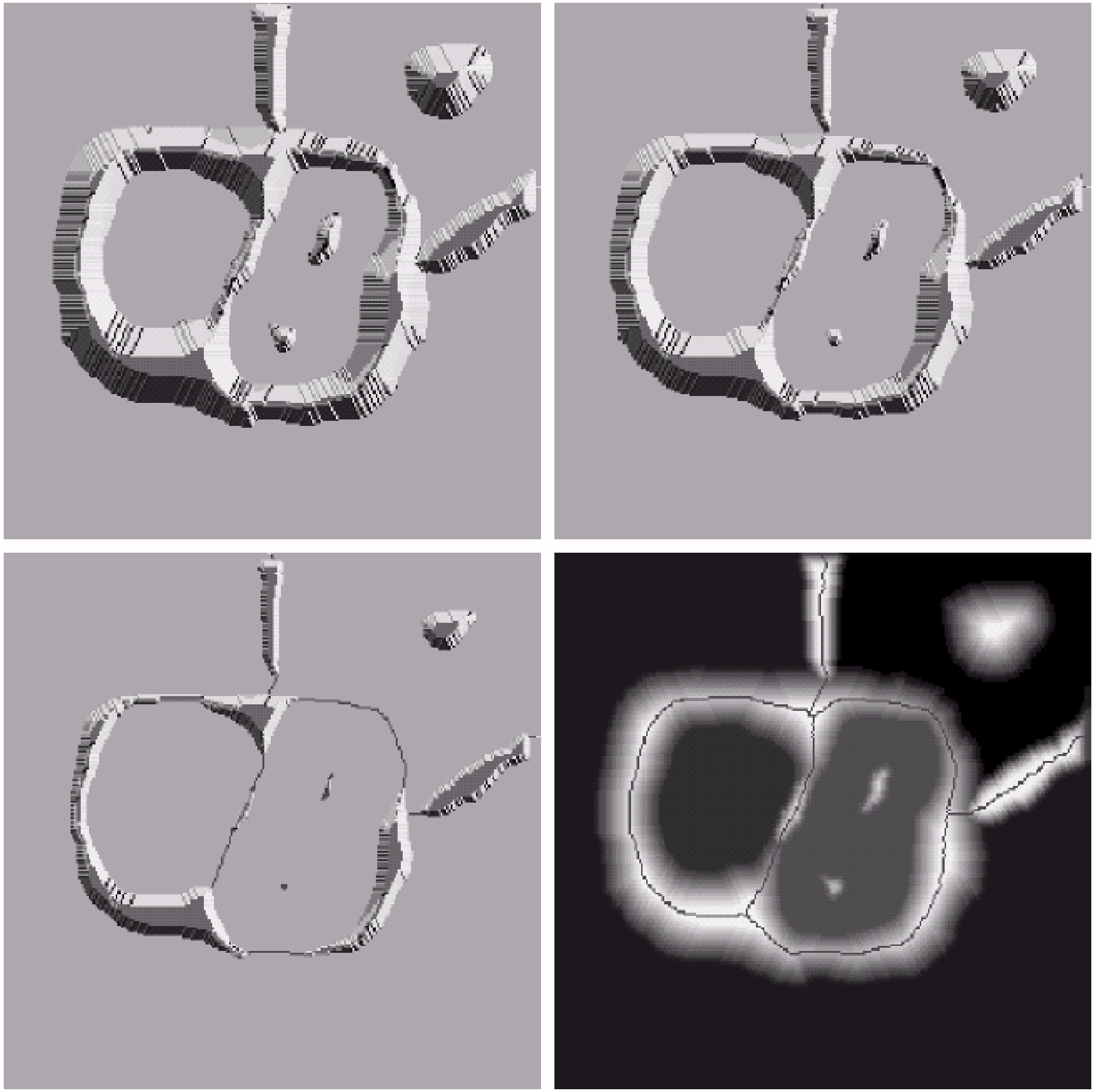
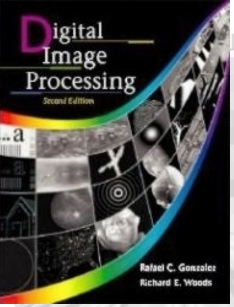


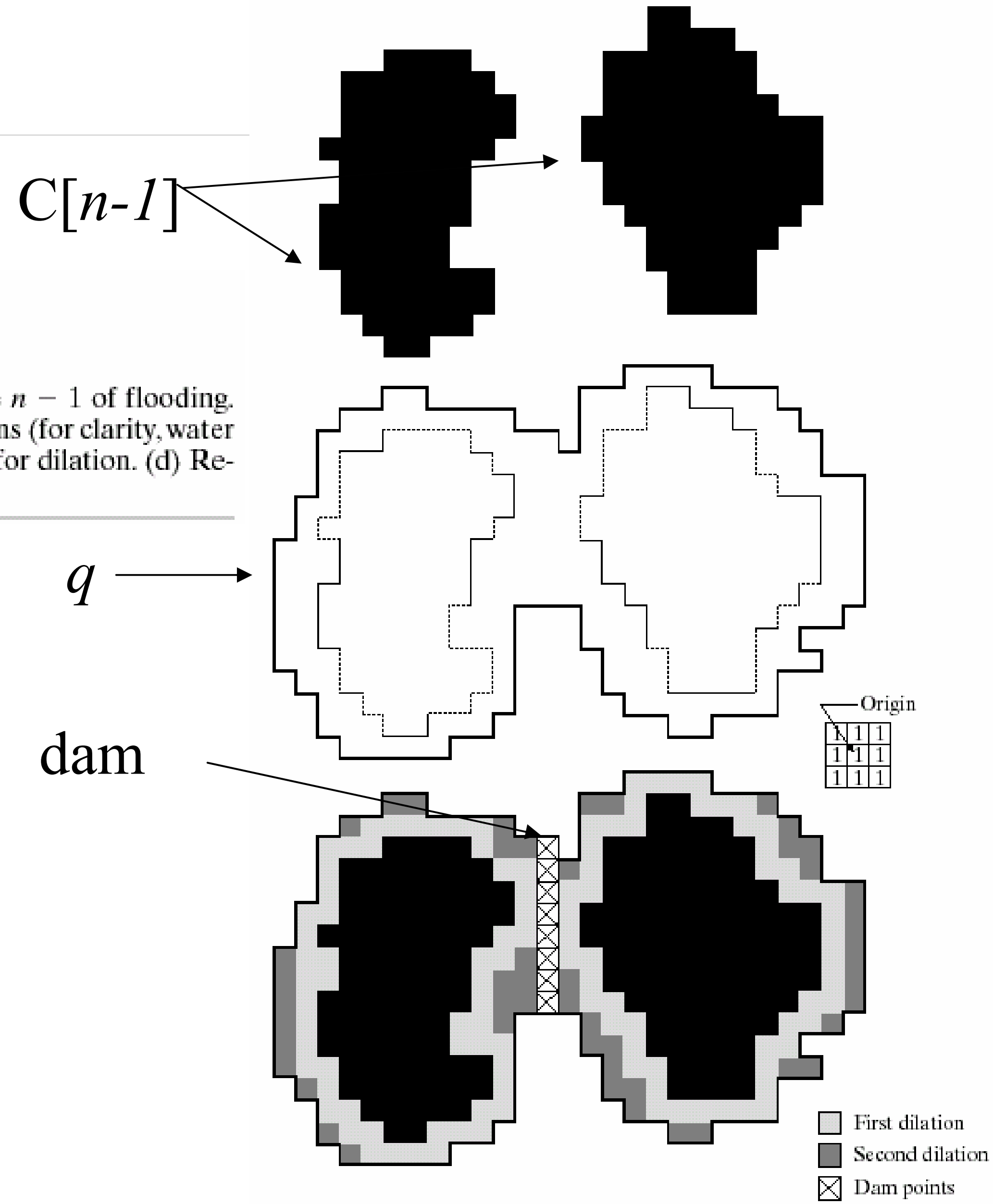
FIGURE 10.44 *(Continued)* (e) Result of further flooding. (f) Beginning of merging of water from two catchment basins (a short dam was built between them). (g) Longer dams. (h) Final watershed (segmentation) lines. (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)



10.5.2 Segmentation by Morphological Watersheds -dam construction

- Use ***morphological dilation*** to construct ***dam***.
- Let M_1 and M_2 denote the set of coordinates of points in two regional minima.
- Let the set of coordinates of points in the *catchment basin* associated with the two minima at stage *n-1* of flooding be denoted by $C_{n-1}(M_1)$ and $C_{n-1}(M_2)$.
- Let the union of the two sets be $C[n-1]$.
- The two components merge when the water between the two *catchment basins* has merged at the flooding step *n*
- Let this ***connected component*** (Figure 10.45 (b)) be denoted as *q*.
- The two components from step *n-1* can be extracted from *q* by the following AND operation: $q \cap C[n-1]$.

10.5 Segmentation by Morphological Watersheds



a
b c
d

FIGURE 10.45 (a) Two partially flooded catchment basins at stage $n-1$ of flooding. (b) Flooding at stage n, showing that water has spilled between basins (for clarity, water is shown in white rather than black). (c) Structuring element used for dilation. (d) Result of dilation and dam construction.

10.5.2 Segmentation by Morphological Watersheds
Dam construction

- Fig. 10.45(a) is dilated by the structure element in fig. 10.45(c), subject to two conditions:
  1) *The dilation has to be constrained to q.*
  2) *The dilation cannot be performed on points that would cause sets being dilated to merge.*
- During the 1st dilation, cond. (1) is satisfied only.
- During the 2nd dilation, cond. (2) is consider only, it results in broken perimeter.
- The only points in *q* that satisfy the ***two conditions*** under consideration describe the one-pixel-thick connected path shown by ***cross-hatched points.***
- The path constitutes the desired separation ***dam*** at stage *n* of flooding. (*n*：水位)

10.5.3 Segmentation by Morphological Watersheds

Watershed Segmentation Algorithm

- Let M_1, M_2 ....M_R be sets denoting the coordinates of the points in the regional minima of an image $g(x, y)$.
- *Let* $C(M_i)$ be a set denoting the coordinates of the points in the *catchment basin* associated with *regional minimum* M_i .
- Let $T[n]$ represent the set of coordinates (s, t) for which $g(s, t)<n$. i.e., $T[n]=\{(g, s)|g(s, t)<n\}$
- The topology will be flooded in *integer* flood increments from $n=min+1$ to $n=max+1$ where *min* and *max* are the minimum and maximum value of $g(x, y)$.

10.5.3 Segmentation by Morphological Watersheds

Watershed Segmentation Algorithm

- *Let* $C_n(M_i)$ denote the set of ***coordinates of points*** in the catchment basin associated with minimum M_i that are flooded at stage n.
- $C_n(M_i)$ can be viewed as a ***binary image*** given by $C_n(M_i) = C(M_i) \cap T[n]$
- $C_n(M_i)=1$ at location (x, y) if $(x, y) \in C(M_i)$ AND $(x, y) \in T[n]$, otherwise $C_n(M_i)=0$
- Let C[n] denote the union of the flooded catchment basins portion at stage n as:

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

10.5.3 Segmentation by Morphological Watersheds

Watershed Segmentation Algorithm

- Then C[*max*+1] is the union of all catchment basins as:

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

- C[*n-1*] is subset of C[*n*].
- C[*n*] is a subset of T[*n*].
- The algorithm for finding the watershed lines is initialized with $C[min+1] = T[min+1]$.
- The algorithm proceeds recursively assuming that at step *n*, $C[n-1]$ has been constructed.

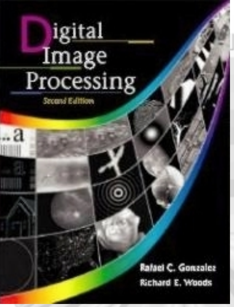
10.5.3 Segmentation by Morphological Watersheds

- Obtain $C[n]$ from $C[n\text{-}1]$ as follows:
  - Let Q denote the set of connected components in $T[n]$.
  - For each connected component $q \in Q[n]$: $q \cap C[n\text{-}1]$ may be

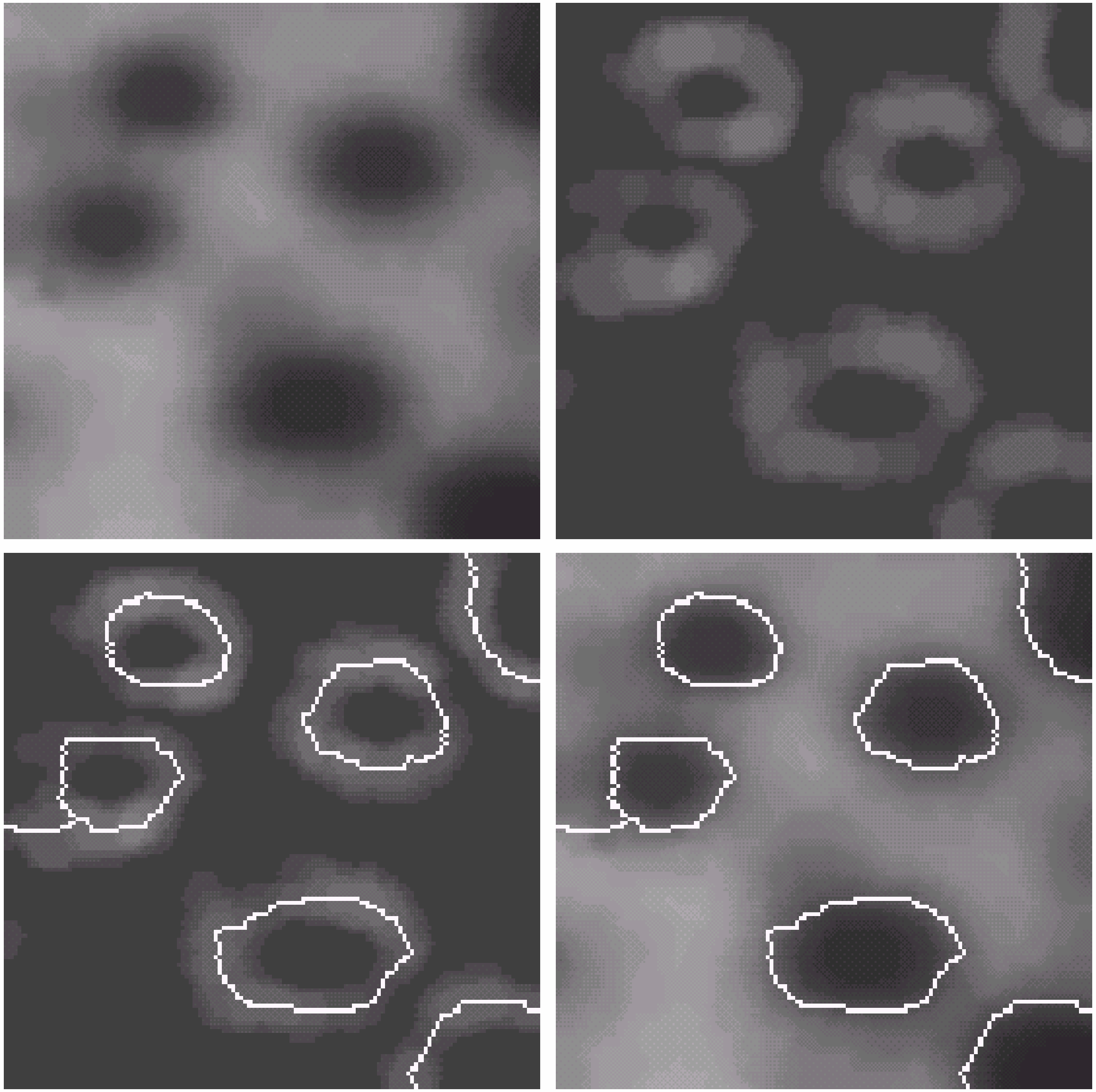
    (a) ***empty*** *:* when ***a new minimum is encountered***, in which case connected component q is incorporated into $C[n\text{-}1]$ to form $C[n]$

    (b) ***one connected component***: when q lies within the *catchment basin* of some regional minimum in which case q is incorporated into $C[n\text{-}1]$ to form $C[n]$

    (c) ***more than one component***: when all or part of a ridge separating two or more *catchment basins* is encountered. Further flooding would cause water level in these catchment basin to merge, therefore, a ***dam*** must built within q to prevent overflow between the catchment basins



10.5 Segmentation by Morphological Watersheds



a b
c d

FIGURE 10.46 (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.)

10.5 Segmentation by Morphological Watersheds
The use of markers

- Over-segmentation due to noise or other irregularities of the gradient - use the markers
- A marker is a connected component belonging to an image.
- Internal markers: object of interests.
- External markers: background.

10.5.4 Segmentation by Morphological Watersheds The use of markers

- ***Marker selection:***

  1) Preprocessing: Use smooth filtering to remove small spatial detail.

  2) A set of criteria that markers must satisfy

     1)A region that is surrounded by points of higher "altitude"

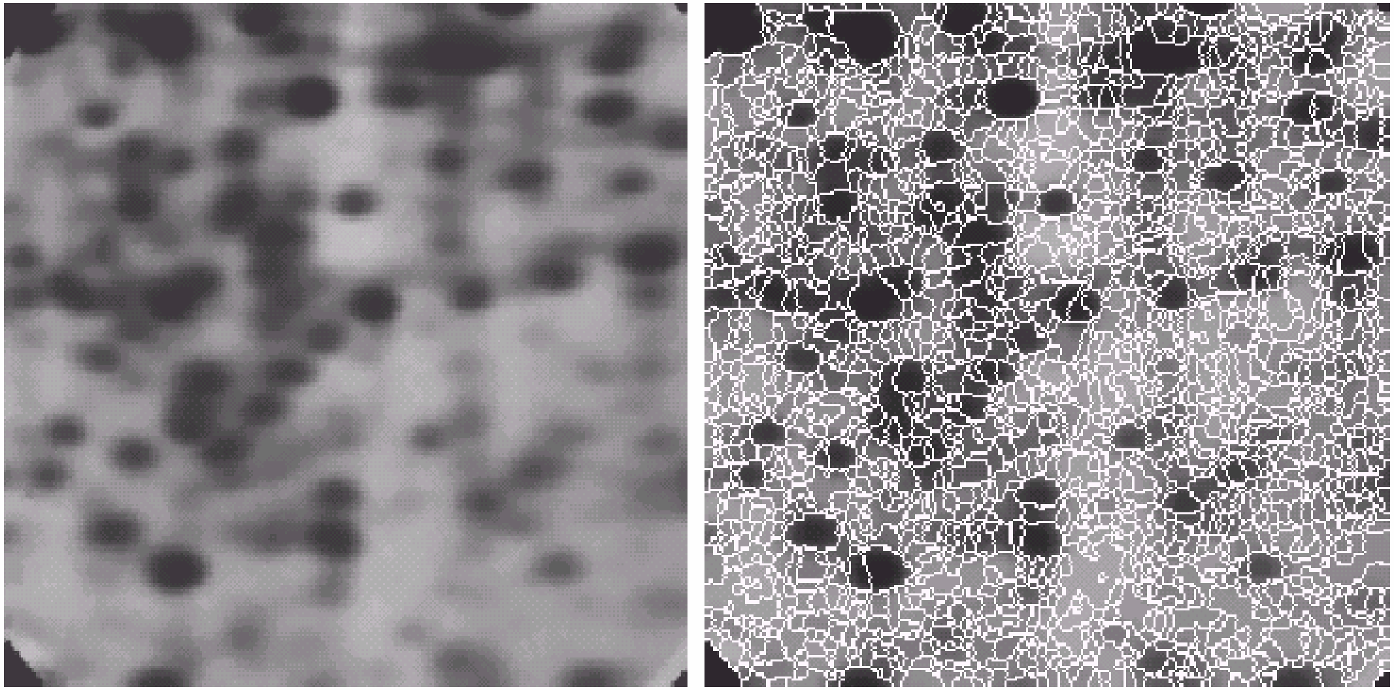
     2)The points in the regions form a connected component.

     3)All the points in the connected region have the same gray-level value.

     4)After image is smoothed, the ***internal markers*** are shown as light gray, blob like region

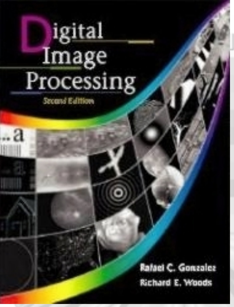
     5)Watershed algorithm is applied and the resulting water shed lines are defined as the ***external markers*** (Figure 10.48(a)).

10.5 Segmentation by Morphological Watersheds

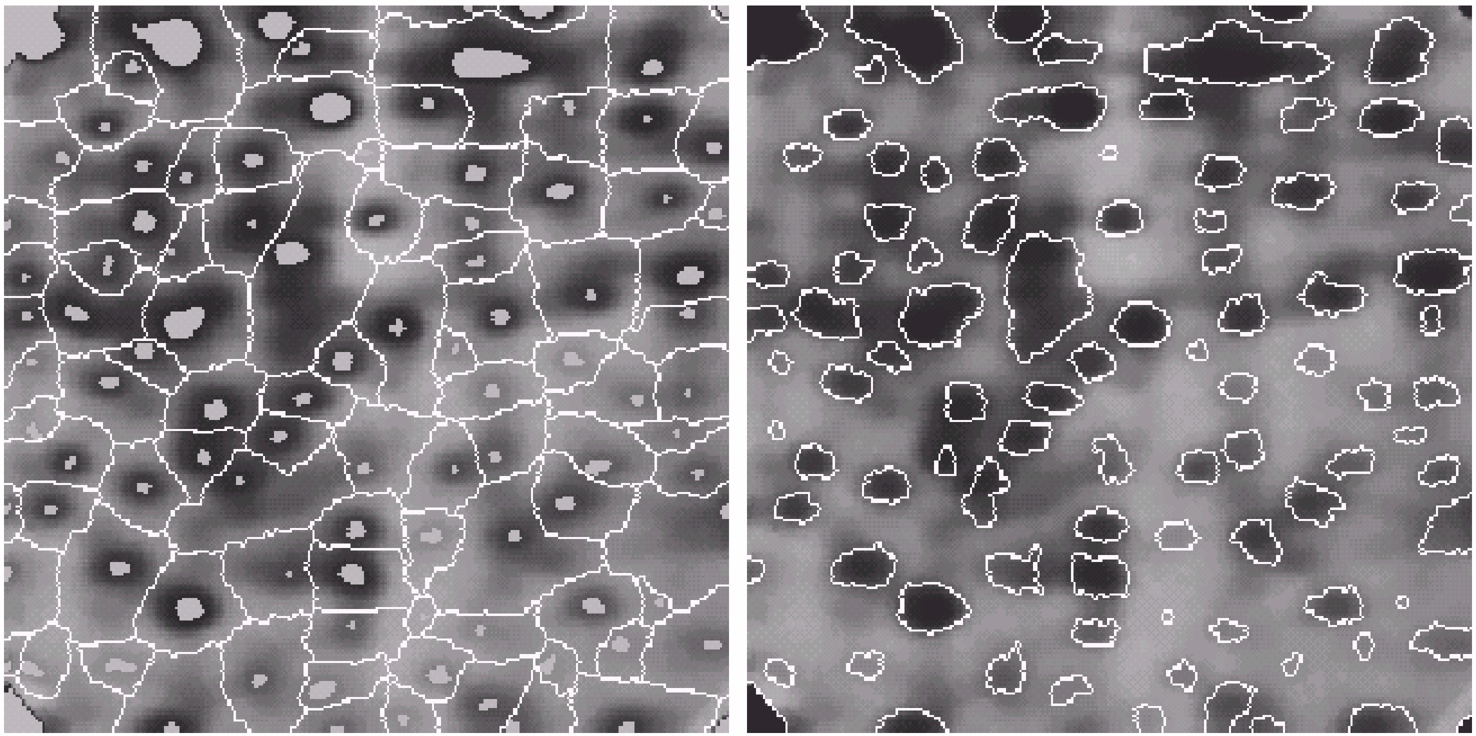


a b

FIGURE 10.47 (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.)



10.5 Segmentation by Morphological Watersheds The use of markers



a b

FIGURE 10.48 (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.)

10.5 Segmentation by Morphological Watersheds
The use of markers

- In Fig. 10.48, the image is partitioned into regions, each containing a singer internal marker and part of the background.
- Simplify the problem as partition each region into a ***single object*** and ***its background***.
- Marker selection can be based on gray-level value and connectivity, and more complex description involving size, shape, location, texture content, and so on.
- Apply the watershed segmentation on each region with ***internal marker***.

10.6 Use of Motion in Segmentation -spatial domain

- ***Spatial Domain Technique***

1) Detect changes between two image frames $f(x, y, t_i)$ *and* $f(x, y, t_j)$ taken at time t_i and t_j

2) Form a difference image $d_{ij}(x, y)$ defined as

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

3) Accumulate differences: consider a sequence of image frame $f(x, y, t_1)\ldots f(x, y, t_n)$ and $f(x, y, t_1)$ is the reference image, an ***accumulative difference image*** (***ADI***) is formed by compare this reference with every subsequence image.

4) A counter for each pixel location is increased every time a difference occurs at that location.

10.6 Use of Motion in Segmentation

- Three ***ADIs***: ***absolute, positive*** and ***negative***.
- Let $R(x, y)= f(x, y, t_1)$ and $f(x, y, k)=f(x, y, t_k)$, then for any $k>1$ three *ADIs* are ***counters*** defined as follows:

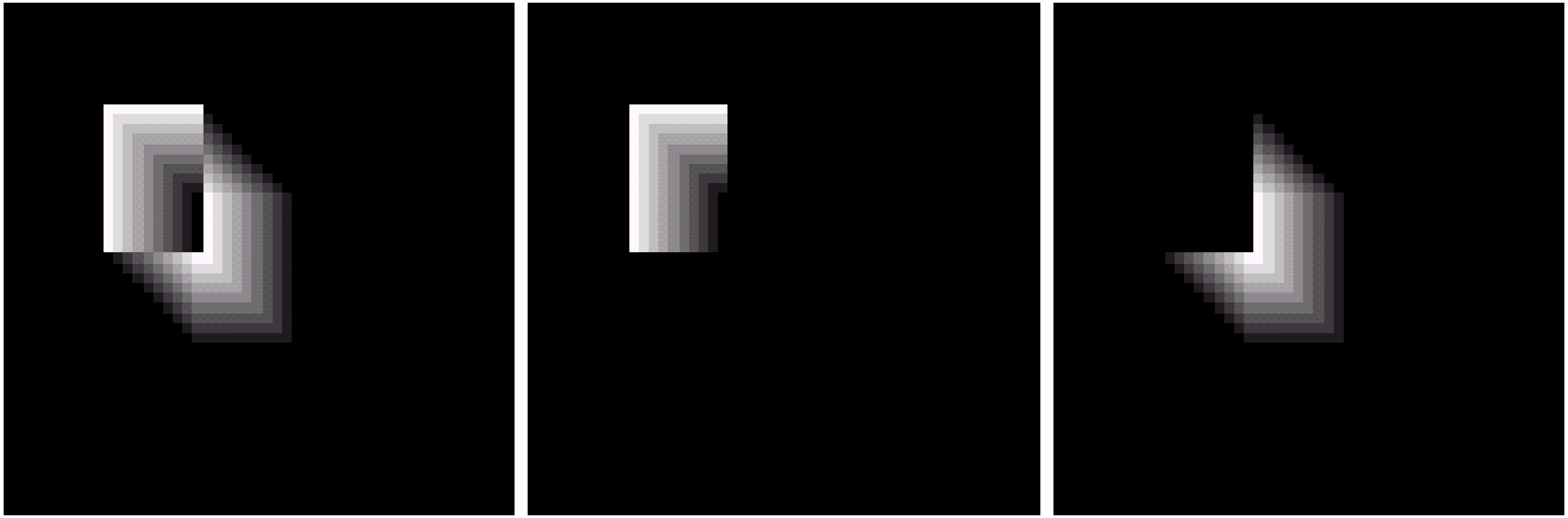
$$A_k(x,y)=\begin{cases} A_{k-1}(x,y)+1 & if \ \left|R(x,y)-f(x,y,k)\right|>T \\ A_{k-1}(x,y) & otherwise \end{cases}$$

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

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

10.6 Use of Motion in Segmentation- example

(1) The non-zero area of the positive ADI equals to the size of moving object. (2) The positive ADI stops increasing when the moving object displaced complete away from the same object in the reference image.



a b c

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

10.6 Use of Motion in Segmentation - spatial domain

- To generate a reference frame with only stationary elements is as follows:

  1. Consider the first image as the reference image

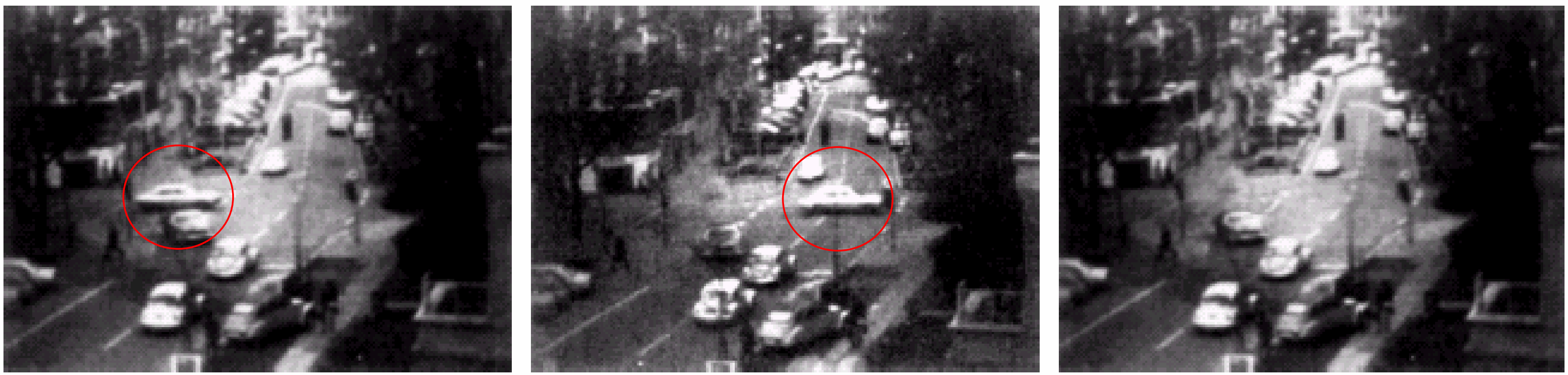
  2.When a non-stationary component move completely out of its position in the reference frame, the corresponding background can be duplicated in the location originally occupied by the object.

  3. Similar process can be done for other moving objects

  4. Object displacement can be established by monitoring the changes in the positive ADI.

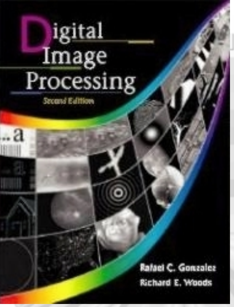
10.6 Use of Motion in Segmentation - example

To remove the principal moving objects (the car at the intersection moving from left to right) in the reference image to create a static image. By monitoring the changes in the positive ADI, we may find the position of a moving object.



a b c

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



10.6 Use of Motion in Segmentation - Frequency domain

- A video sequence (frame size MxN) can be represented as a **space time function** $f(x, y, t)$, $t=0, 1, 2, \ldots K\text{-}1$.
- All frames have homogeneous background of zero intensity.
- The image plane is **projected onto the *x*-axis** yields a 1-D array with M entry that are 0, except at the location where the object is projected.
- Multiply the components of the array by $exp[j2\pi a_1 x\Delta t]$, $x=0,1,\ldots M\text{-}1$, with the object located at (x', y'), it produces a sum equal to $exp[j2\pi a_1 x'\Delta t]$

10.6 Use of Motion in Segmentation - Frequency domain

- If the object moves *one-pixel* per frame, then at any instance of time t, we have $exp[j2\pi a_1(x'+t)\Delta t]$
- This procedure yields a complex sinusoid with frequency a_1.
- If the object moves with v_1 pixels between frames then the sinusoid would have frequency $v_1 a_1$.
- The Fourier transform of complex sinusoid has two peaks, one at $v_1 a_1$ and the other at $K-v_1 a_1$.

10.6.2 Frequency domain Technique

- For a sequence of *K* digital images of size MxN, the sum of weighted projections onto the *x* axis and y axis at any integer instance of time are

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

where $t = 1, 2, \ldots K-1.$

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

- The 1-D Fourier transform of $g_x(t, a_1)$ and $g_y(t, a_2)$ are

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

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

10.6.2 Frequency domain Technique

- The ***frequency-velocity relationship*** is

$$u_1 = v_1 a_1 \quad and \quad u_2 = v_2 a_2.$$

Example:

v_1=10 pixels in K frames, K=30, frame rate=2 image/sec, the distance between pixel=0.5m

Actual speed is

$$v_1 = (10 pixels)(0.5 m/pixel)(2 frame/secs)/(30 frames)$$

$$= 1/3\ m/sec$$

10.6.2 Frequency domain Technique

- The sign of x-component of the velocity is obtained by computing

$$S_{1x} = \frac{d^2 \operatorname{Re}\{g_x(t, a_1)\}}{dt^2}\bigg|_{t=n}$$

$$S_{2x} = \frac{d^2 \operatorname{Im}\{g_x(t, a_1)\}}{dt^2}\bigg|_{t=n}$$

- g_x is sinusoidal, if v_1 is positive then S_{1x} and S_{2x} have the same sign, otherwise, they have opposite signs.

10.6.2 Frequency domain Technique

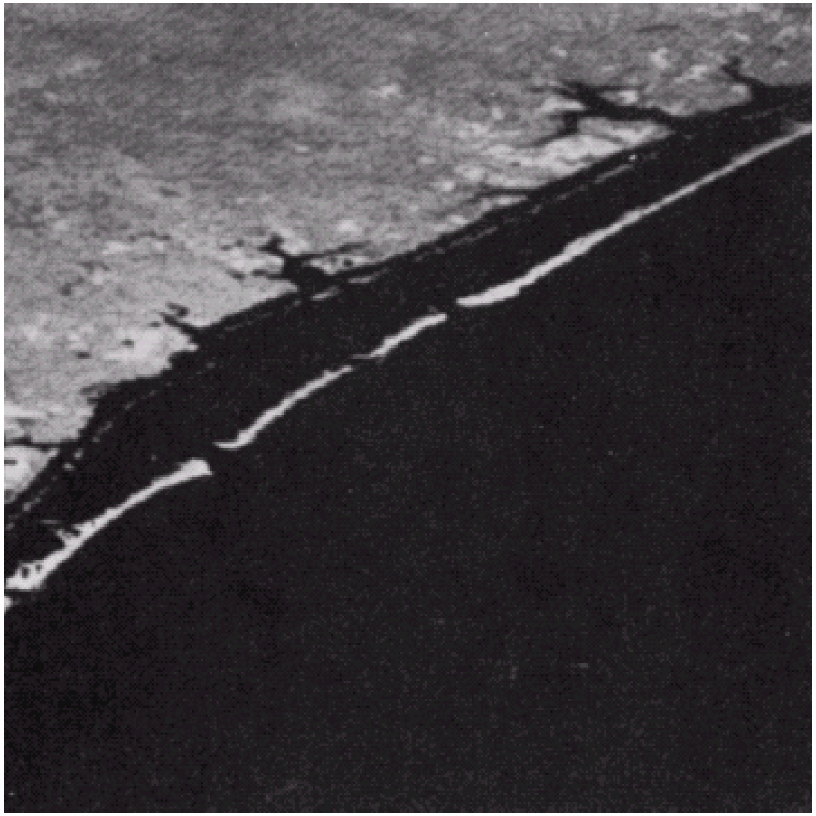


FIGURE 10.51 LANDSAT frame. (Cowart, Snyder, and Ruedger.)

Vx=0.5 pixel/frame

Vy=1 pixel/frame

a1=6 and a2=4

u1=3, v1=0.5, v2=1.0

10.6.2 Frequency domain Technique

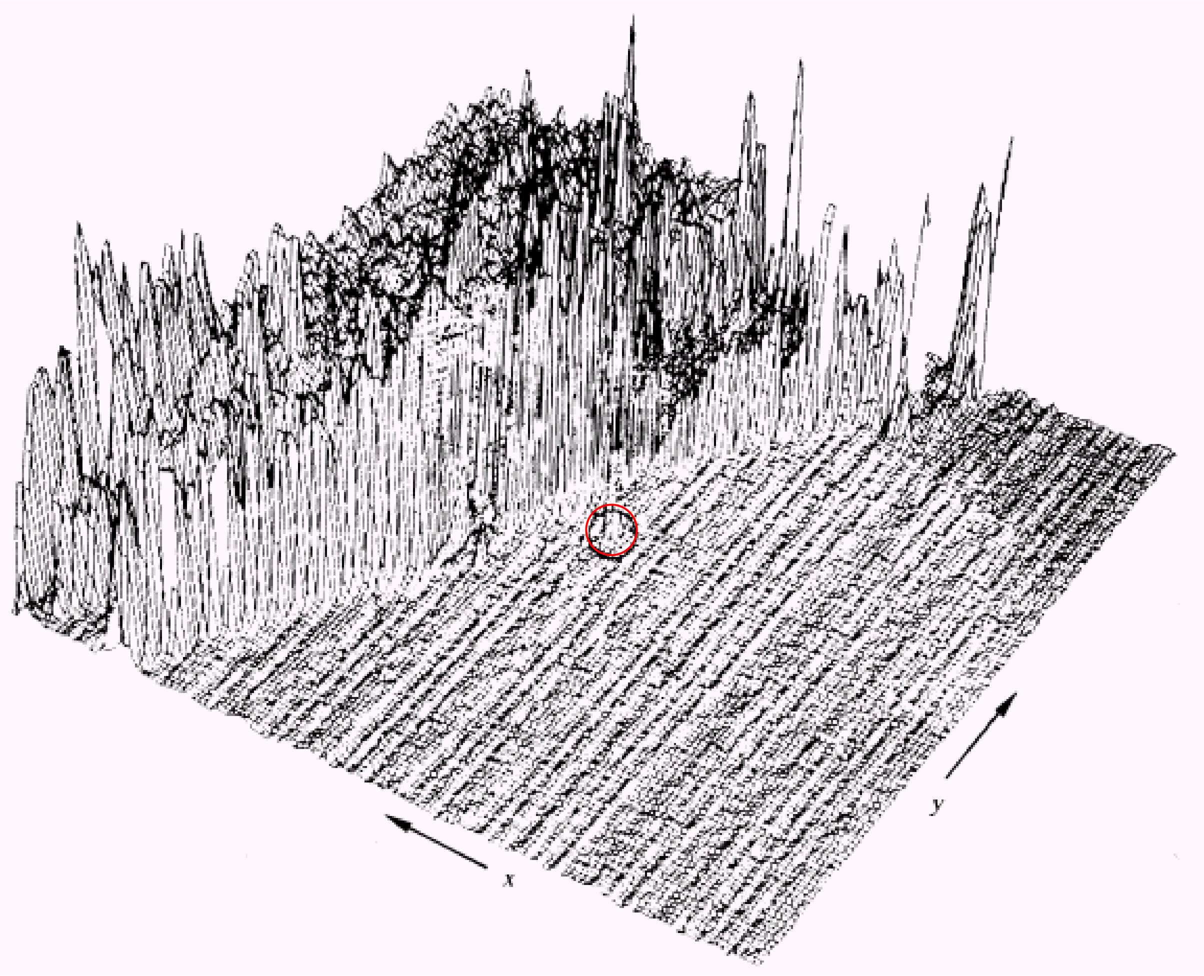


FIGURE 10.52 Intensity plot of the image in Fig. 10.51, with the target circled. (Rajala, Riddle, and Snyder.)

10.6.2 Frequency domain Technique

a1=6 and a2=15

u1=15, 17

v1=1.0

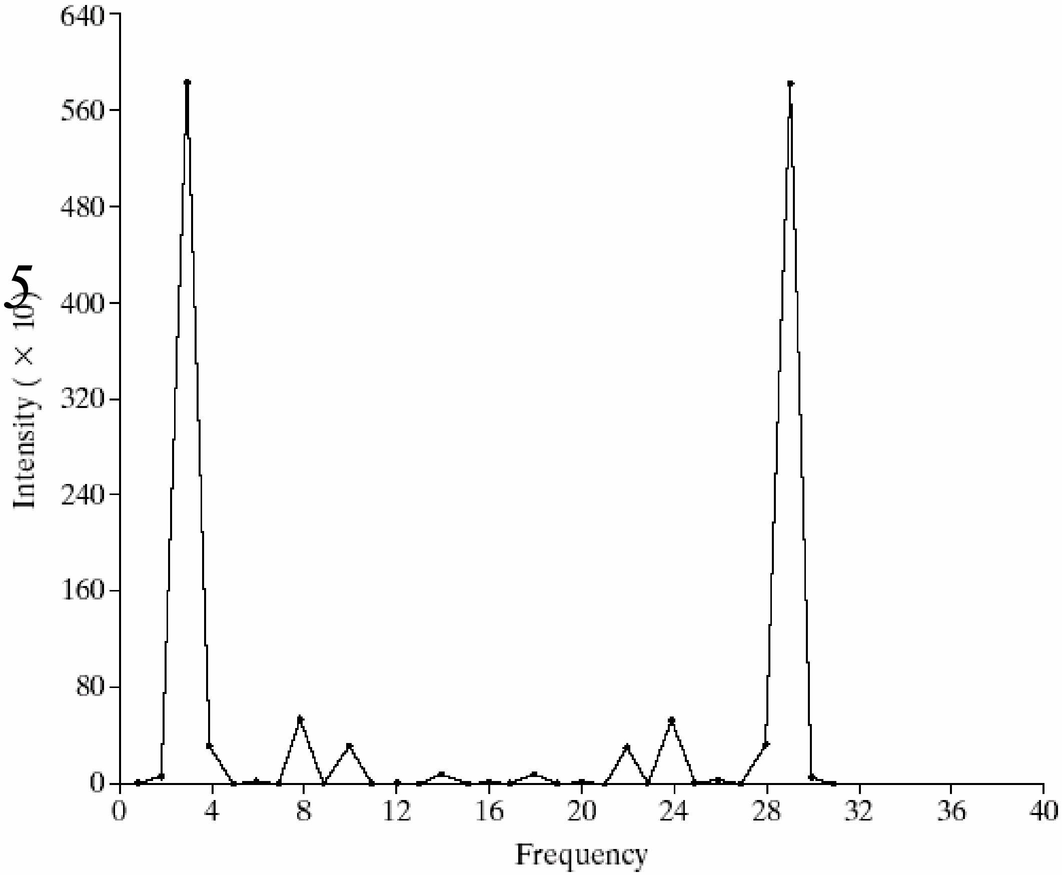
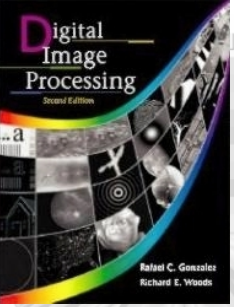


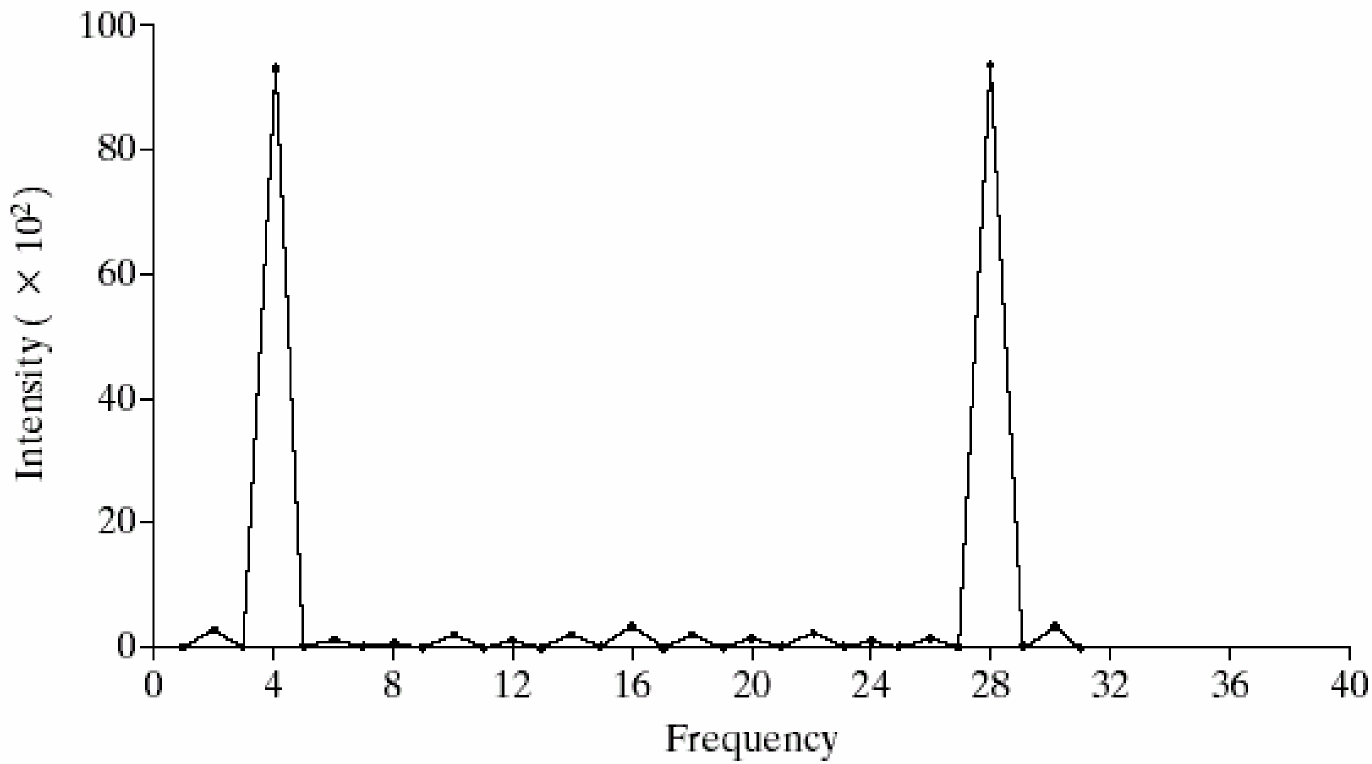
FIGURE 10.53 Spectrum of Eq. (10.6-8) showing a peak at $u_1 = 3$. (Rajala, Riddle, and Snyder.)

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



10.6.2 Frequency domain Technique

FIGURE 10.54 Spectrum of Eq. (10.6-9) showing a peak at $u_2 = 4$. (Rajala, Riddle, and Snyder.)



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