Image Segmentation
Part I

Figures are provided by the Pearson Education

Use of segmentation

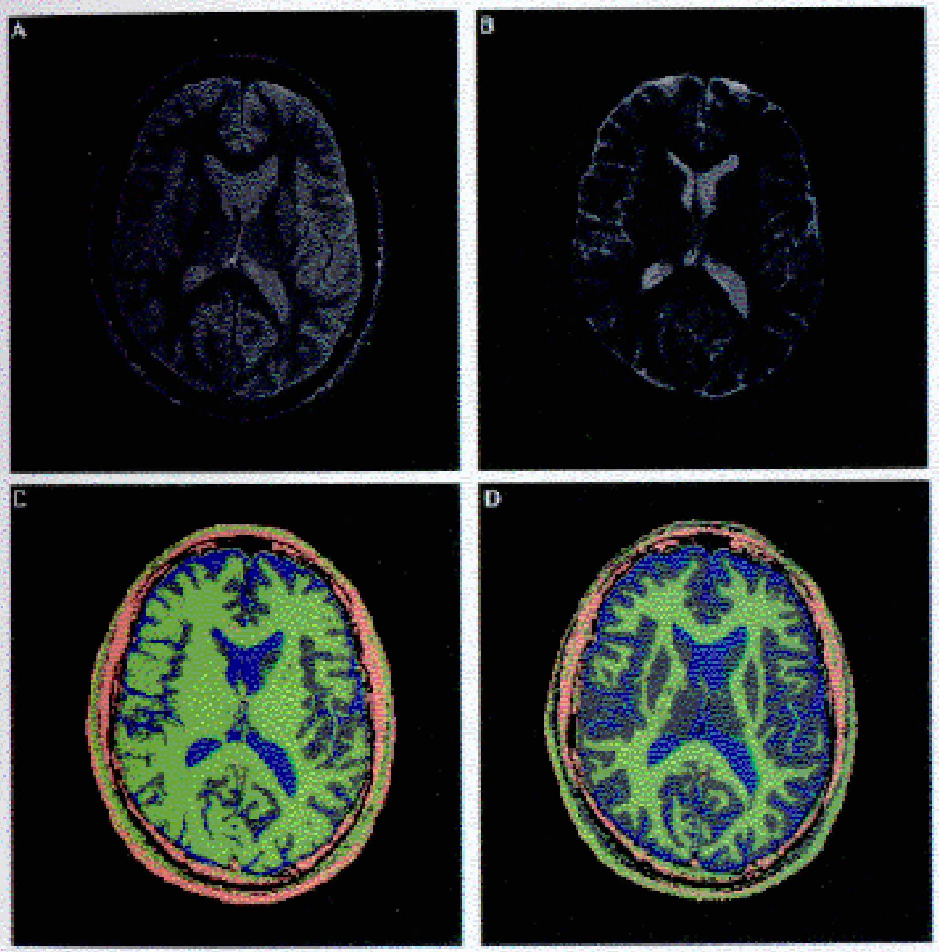


PLATE 4 The results of adaptive segmentation applied to dual-echo images of the brain. (A) Original *T2*-weighted image, (B) original proton-density weighted image, (C) result of conventional statistical classification, (D) result of EM segmentation. The tissue classes are represented by colors: blue, CSF; green, white matter; gray, gray matter; pink, fat; black, background. (Courtesy of Dr. W. M. Wells III, Surgical Planning Lab, Department of Radiology, Brigham and Women's Hospital, Boston.)

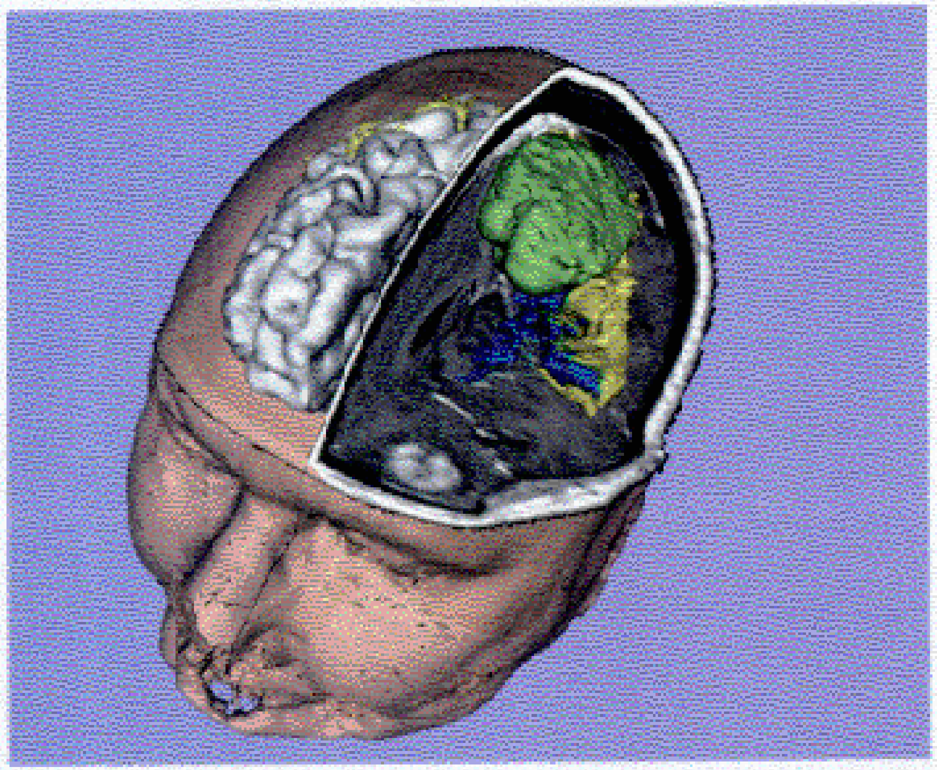


PLATE 6 Rendering of 3D anatomical models and 2D MRI cross-sections of a patient with a meningioma. The models of the skin surface, the brain, and the tumor (green) are based on automatically segmented 3D MRI data. The precentral gyrus (yellow) and the corticospinal tract (blue) are based on a previously aligned digital brain atlas [61]. (Courtesy of Drs. Ron Kikinis, Michael Kaus, and Simon Warfield, Surgical Planning Lab, Department of Radiology, Brigham and Women's Hospital, Boston.)

Handbook of Medical Imaging, Plate 6

Image Segmentation

http://en.wikipedia.org/wiki/Segmentation_(image_processing)

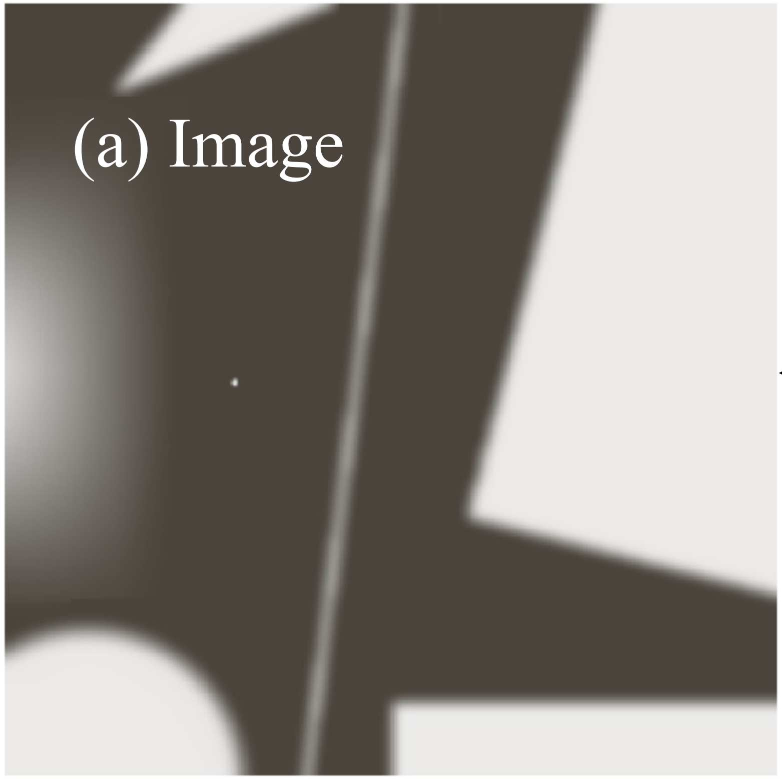
1. Segmentation subdivides an image into its constituent regions or objects.
2. Segmentation accuracy determines the eventual success or failure of computerised analysis procedures.
3. In general, image segmentation algorithms are based on (a) discontinuity and (b) similarity.
4. Discontinuity in intensity value (sudden/abrupt change of intensity values), e.g., edges.
5. Similarity in intensity value, e.g., regions.

Gradient based segmentation

1. Discontinuity = large gradient magnitude = edge of the boundary.

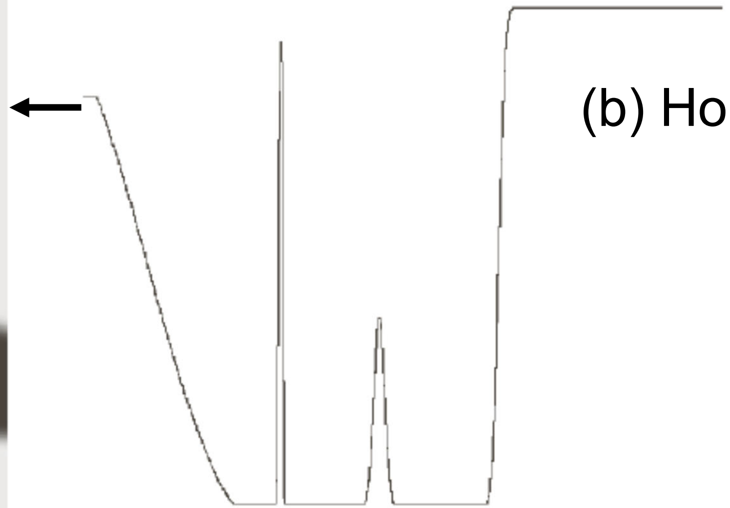
2. Boundary detection for segmentation based on detected edge elements
   a. (Local) Edge linking based on edge strength, direction and distance.
   b. Hough transforms.

Where are intensity discontinuities?

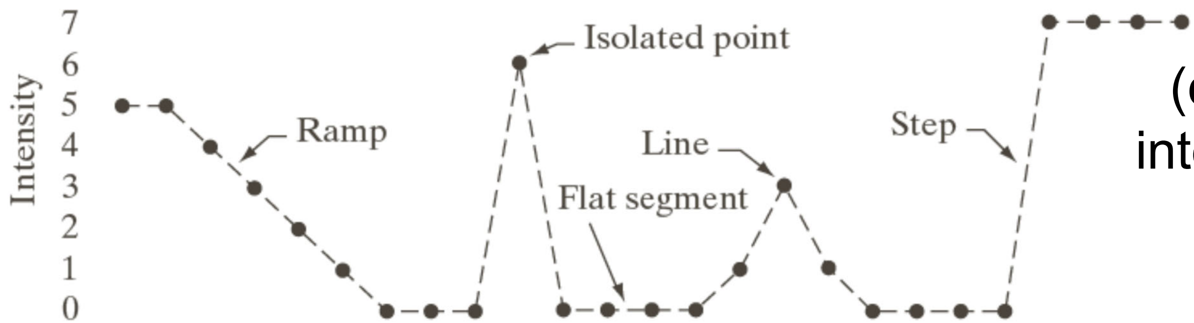


a b
c

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).



(b) Horizontal intensity profile



(c) Different kinds of intensity discontinuities

Image strip	5	5	4	3	2	1	0	0	0	6	0	0	0	0	1	3	1	0	0	0	0	7	7	7	7
First derivative		−1	−1	−1	−1	−1	0	0	6	−6	0	0	0	1	2	−2	−1	0	0	0	7	0	0	0	
Second derivative		−1	0	0	0	0	1	0	6	−12	6	0	0	1	1	−4	1	1	0	0	7	−7	0	0	

[Gonzalez and Woods]

Detection of Intensity Discontinuities

1. There are three different types of intensity discontinuities in a digital image:
   a. Point (Isolated Point)
   b. Line
   c. Edge (Ideal, Ramp and Roof)
2. Intensity discontinuity is detected based on the mask response R within a pre-defined window, e.g., 3x3 window.

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.

$$\text{Mask Response } R = \sum_{i=1}^{9} w_i z_i$$

where w_i represent weights within a pre-defined window; z_i represent intensity values.

[Gonzalez and Woods]

An image is represented by intensity values z_i

w_i represent weights within a 3x3 pre-defined window

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Mask response at a pixel, $R = \sum_{i=1}^{9} w_i z_i$

Working principle: For each pixel, x, in an image, if the absolute value of the mask response, $|R|$, is larger than or equal to some threshold T, $|R| \geq T$, then an intensity discontinuity is detected and located at the pixel.

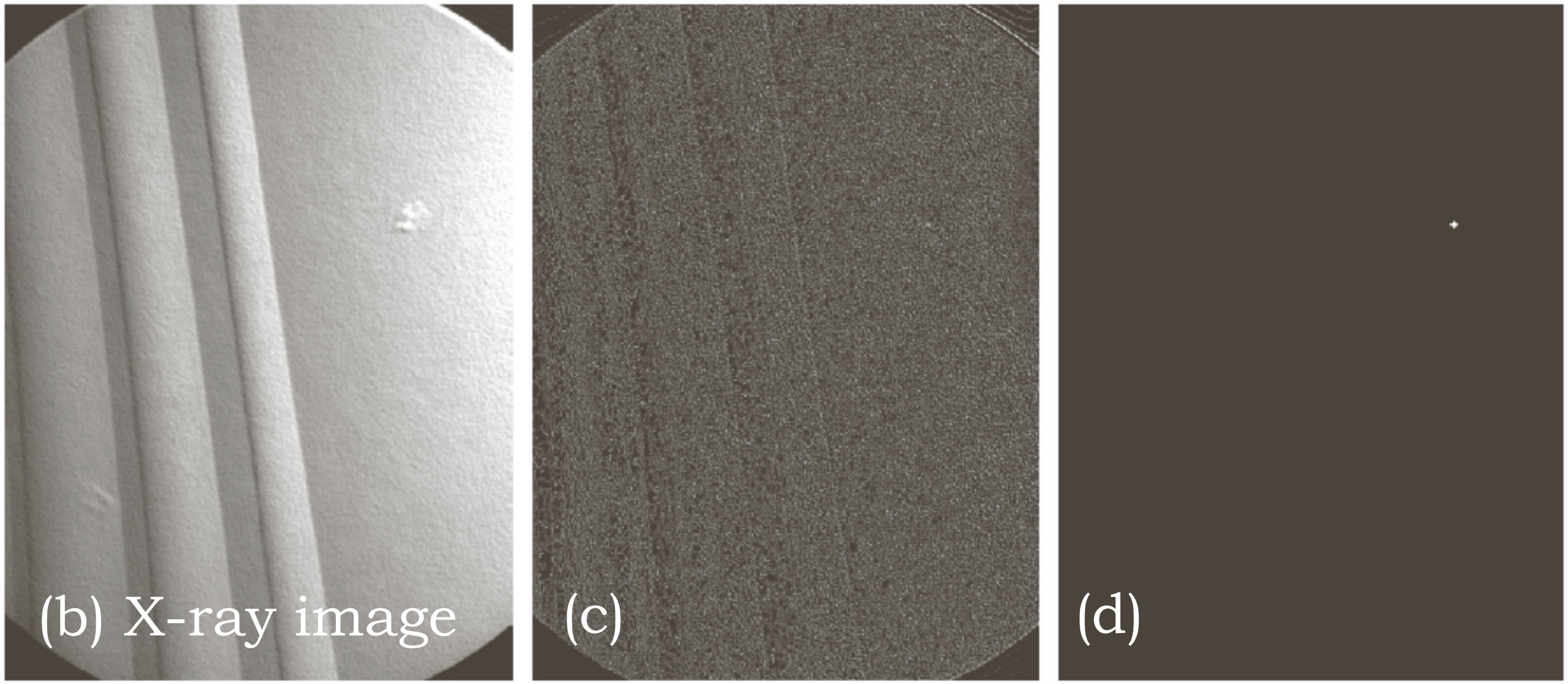
[Gonzalez and Woods]

Point Detection

1. If $|R| \geq T$, then a point has been detected. This point is the location on which the mask is centred, where T is a non-negative threshold.
2. For example, an isolated point is detected if it is different from its surroundings.

1	1	1
1	−8	1
1	1	1

(a)



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.)

[Gonzalez and Woods]

Point Detection

3. This mask is the Laplacian mask for detecting point. Sum of all weights is zero to make sure that there is no response at a flat region (constant intensity region).
4. The Laplacian is given as (f = input image)

1	1	1
1	−8	1
1	1	1

Laplacian mask

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

where

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

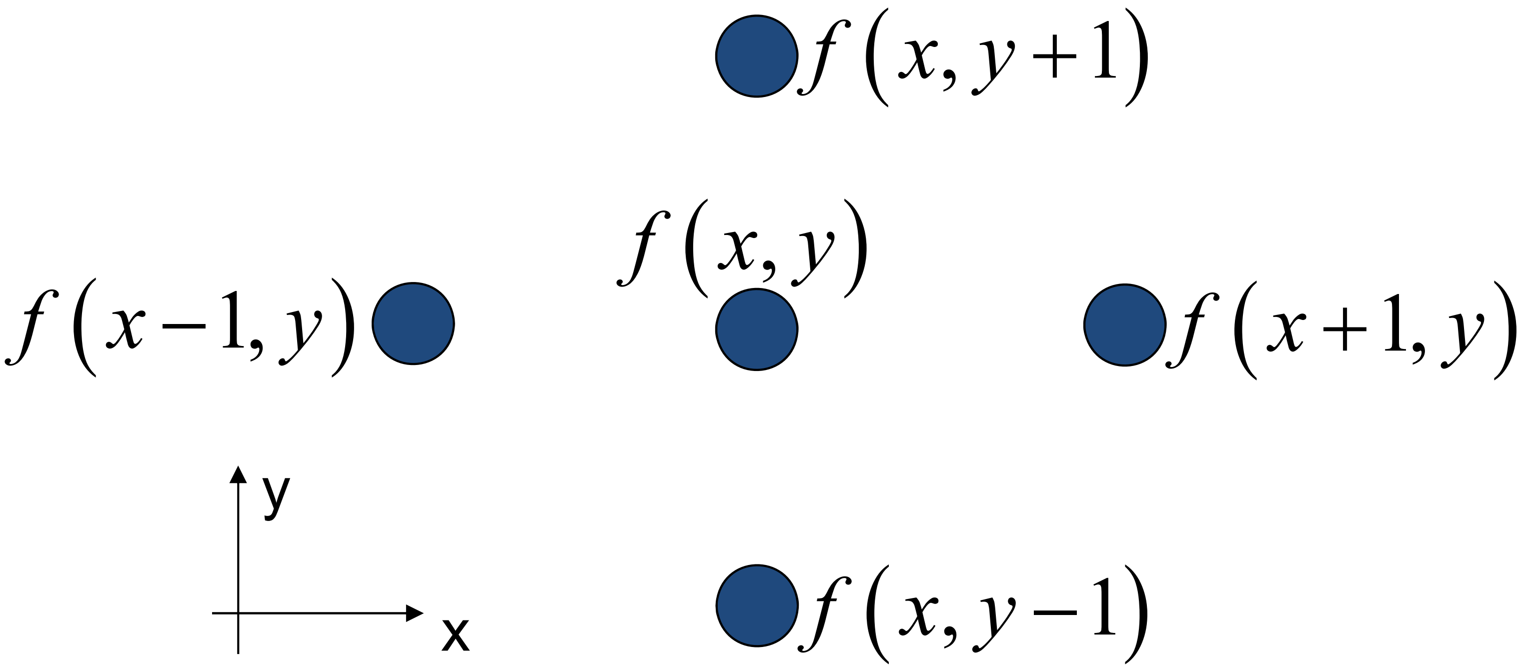
$$\frac{\partial^2 f}{\partial y^2} = f(x,y+1) - 2f(x,y) + f(x,y-1)$$

[Gonzalez and Woods]

Point Detection

5. The discrete implementation of the Laplacian operator is given as (f = input image)

$$\nabla^2 f(x,y) = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)$$



[Gonzalez and Woods]

Point Detection

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

a b
c d

FIGURE 3.37
(a) Filter mask used to implement Eq. (3.6-6). (b) Mask used to implement an extension of this equation that includes the diagonal terms. (c) and (d) Two other implementations of the Laplacian found frequently in practice.

[Gonzalez and Woods]

Point Detection

6. Point detection is implemented in MATLAB using function `imfilter`, with a mask on an image. Regarding the mask, the important requirements are that the strongest response of a mask must be obtained when the mask is centered on an isolated point, and that the response be 0 in areas of constant intensity.
7. The following command implements the point-detection approach.

```
>> g = abs(imfilter(f,w)) >= T;
```

[Gonzalez, Woods and Eddins]

Point Detection

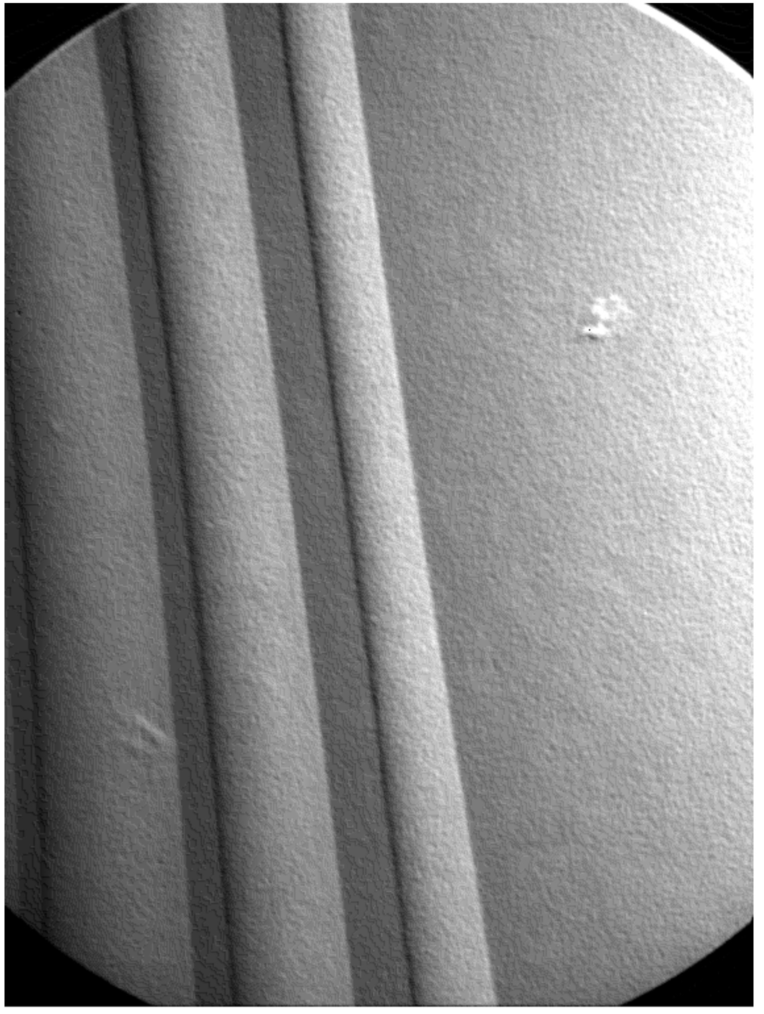


Fig1004(b)(turbine_blade_black_dot).tif

From: http://www.imageprocessingplace.com/DIP-3E/dip3e_book_images_downloads.htm

Point Detection

8. A MATLAB example (filename: `point_detection.m`)

```
clear all % clear all variables, globals, functions and MEX links.
close all % close all figures

% read an image
f = imread('Fig1004(b)(turbine_blade_black_dot).tif');
imshow(f, [min(min(f)) max(max(f))]);

w = [1 1 1; 1 -8 1; 1 1 1]; % mask
g = abs(imfilter(double(f),w)); % mask absolute responses
figure; imshow(g, [min(min(g)) max(max(g))]);

T = max(max(g)); % threshold
h = (g >= T); % thresholding
h = imdilate(h, [0 1 0; 1 1 1; 0 1 0]); % image dilation
figure; imshow(h, [0 1]);
```

[Gonzalez, Woods and Eddins]

Line Detection

1. A line is detected when more than one aligned, connected points are detected;
2. Or, the response of line mask is greater than some threshold, e.g.,

Horizontal

−1	−1	−1
2	2	2
−1	−1	−1

+45°

2	−1	−1
−1	2	−1
−1	−1	2

Vertical

−1	2	−1
−1	2	−1
−1	2	−1

−45°

−1	−1	2
−1	2	−1
2	−1	−1

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

The above are line masks for detecting lines (1 pixel thick) in 4 different specific directions.

Centerline detection: http://cvlab.epfl.ch/software/centerline-detection

[Gonzalez and Woods]

Line Detection

3. If we want to detect a line in a specified direction, then we should use the mask associated with that direction and threshold its output responses.

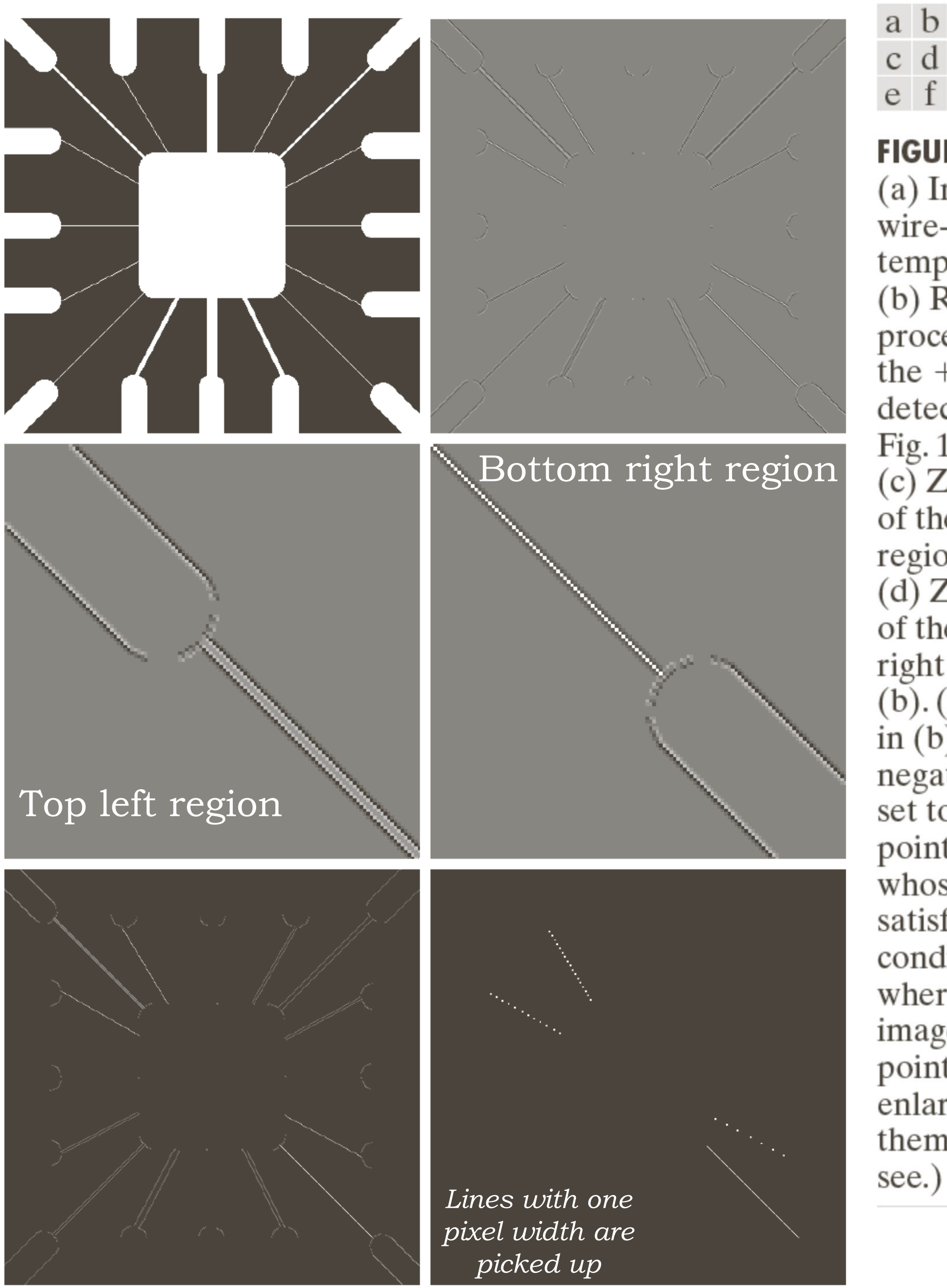


FIGURE 10.7 (a) Image of a wire-bond template. (b) Result of processing with the +45° line detector mask in Fig. 10.6. (c) Zoomed view of the top left region of (b). (d) Zoomed view of the bottom right region of (b). (e) The image in (b) with all negative values set to zero. (f) All points (in white) 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.)

[Gonzalez and Woods]

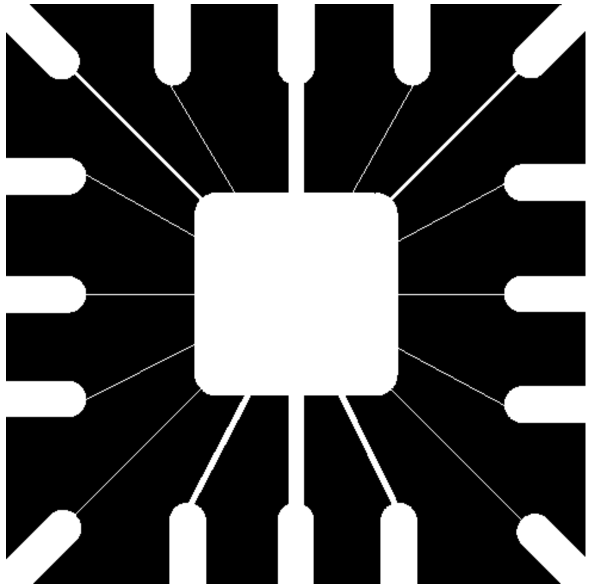
Line Detection

4. If 4 line masks are used, then the final response is equal to the largest response among the masks.

$$R = \max\left(\left|R_{horizontal}\right|, \left|R_{45^o}\right|, \left|R_{vertical}\right|, \left|R_{-45^o}\right|\right)$$

5. An example image

Fig1007(a)(wirebond_mask).tif



From: http://www.imageprocessingplace.com/DIP-3E/dip3e_book_images_downloads.htm

[Gonzalez and Woods] [Gonzalez, Woods and Eddins]

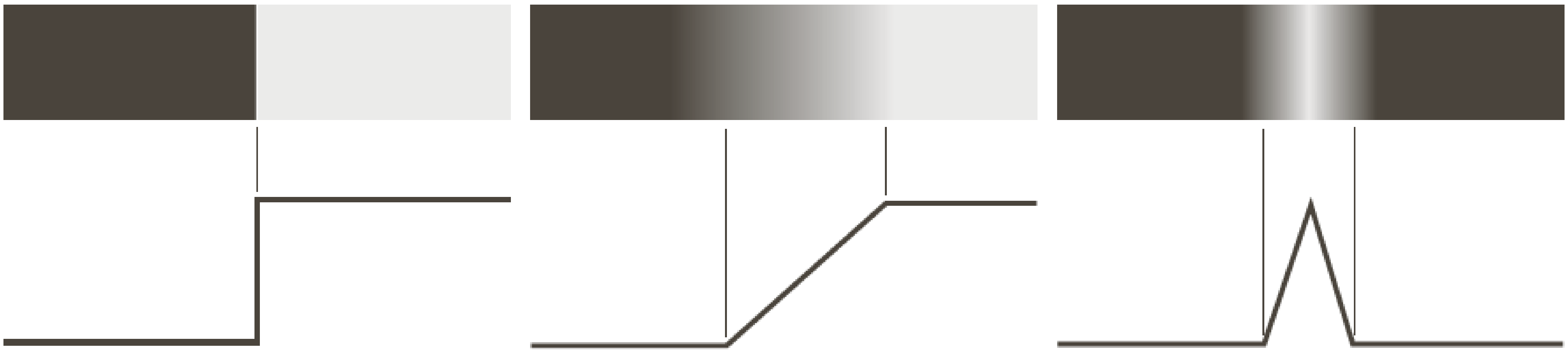
6. MATLAB example (filename: `line_detection.m`)

```
clear all % clear all variables, globals, functions and MEX links.
close all % close all figures

% read an image
f = imread('Fig1007(a)(wirebond_mask).tif');
imshow(f, [min(min(f)) max(max(f))]);

w = [2 -1 -1; -1 2 -1; -1 -1 2]; % mask
g = imfilter(double(f),w); % mask responses
figure; imshow(g, [min(min(g)) max(max(g))]);

[m,n]=size(g); m_top = round(m/4); n_top = round(n/4);
gtop = g(1:m_top, 1:n_top); % top left region of the responses
figure; imshow(gtop, [min(min(gtop)) max(max(gtop))]);

m_bot = round(3*m/4); n_bot = round(3*n/4);
gbot = g(m_bot:m, n_bot:n); % bottom right region of the responses
figure; imshow(gbot, [min(min(gbot)) max(max(gbot))]);

g = abs(g); % mask absolute responses
T = max(max(g)); % threshold
h = (g >= T); % thresholding
h = imdilate(h, [0 1 0; 1 1 1; 0 1 0]); % image dilation
figure; imshow(h, [0 1]);
```

[Gonzalez and Woods] [Gonzalez, Woods and Eddins]

Edge Detection

1. Edge is the boundary of regions. The boundary has discontinuities in grey intensity level.
2. Three types of edges: ideal edge, ramp edge and roof edge.



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.

[Gonzalez and Woods]

Edge Detection

http://en.wikipedia.org/wiki/Edge_detection

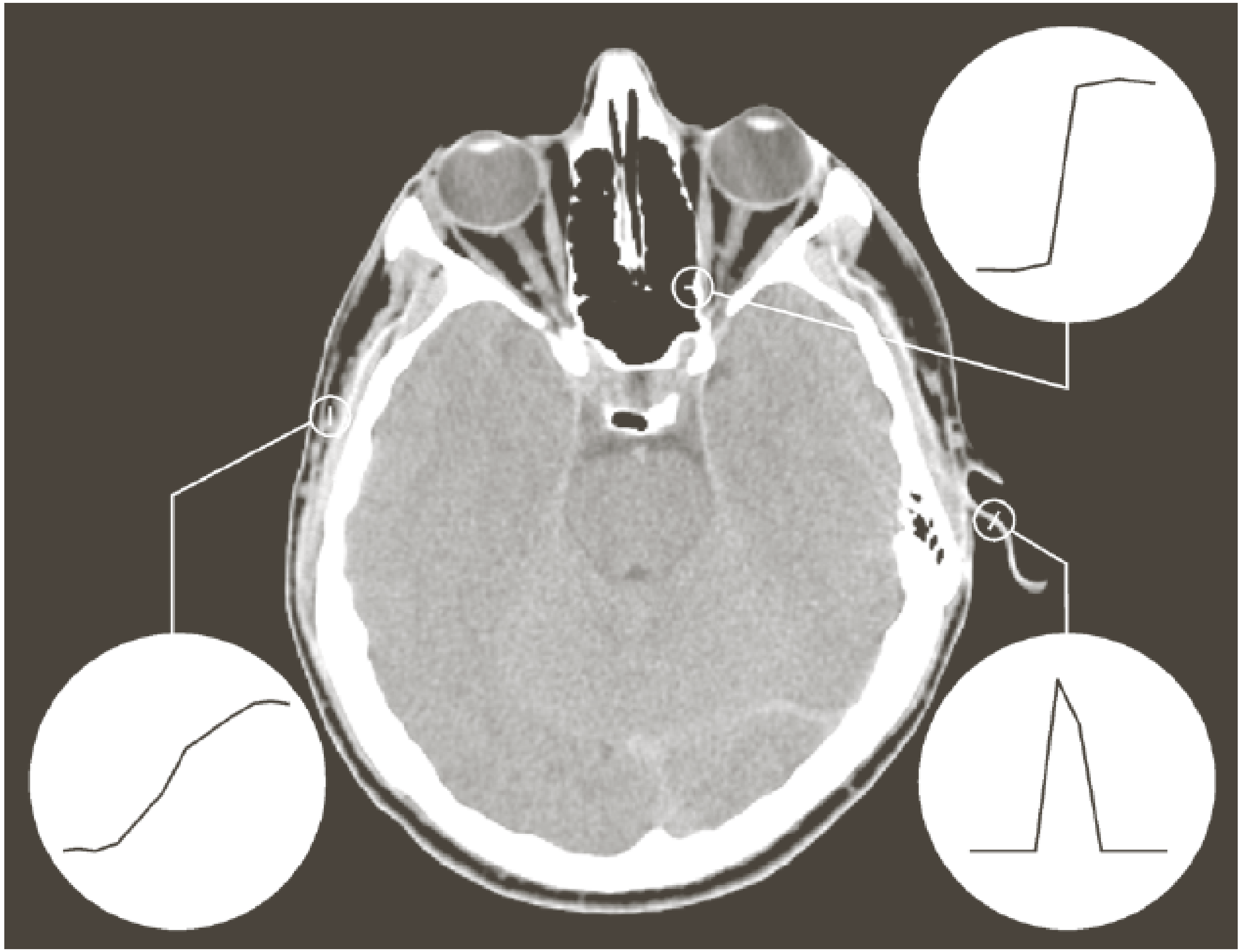


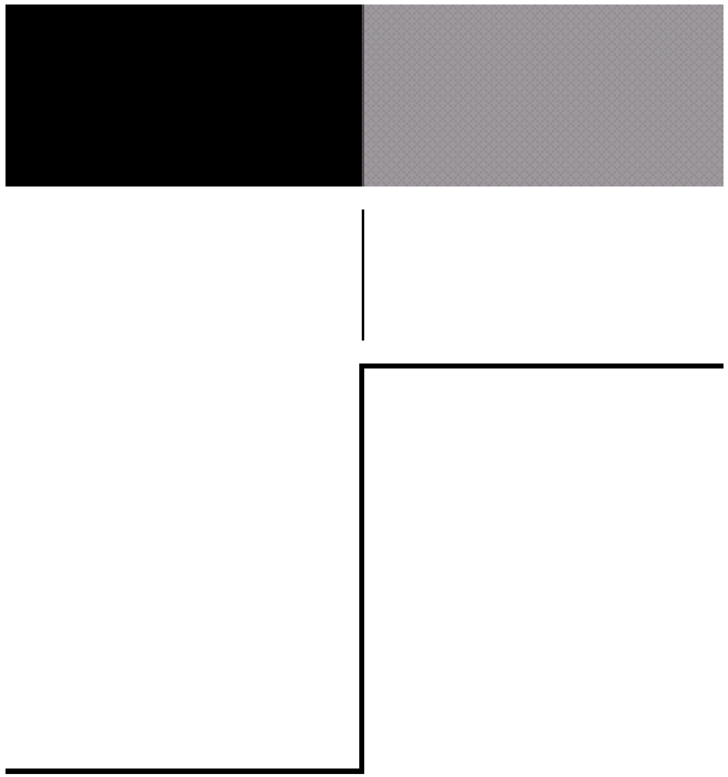
FIGURE 10.9 A 1508 × 1970 image showing (zoomed) actual ramp (bottom, left), step (top, right), and roof edge profiles. The profiles are from dark to light, in the areas indicated by the short line segments shown in the small circles. The ramp and "step" profiles span 9 pixels and 2 pixels, respectively.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

[Gonzalez and Woods]

Edge Detection

3. For an ideal edge (step edge), an edge is a collection of connected pixels on the region boundary. Ideal edges can occur over the distance of 1 pixel.

Model of an ideal digital edge



Gray-level profile
of a horizontal line
through the image

[Gonzalez and Woods]

Edge Detection

4. Roof edges are models of lines through a region, with the base (width) of a roof edge being determined by the thickness and sharpness of the line.

5. In the limit, when its base is 1 pixel wide, a roof edge becomes a 1 pixel thick line running through a region in an image.

6. Roof edges can represent thin features, e.g., roads, line drawings, etc.

[Gonzalez and Woods]

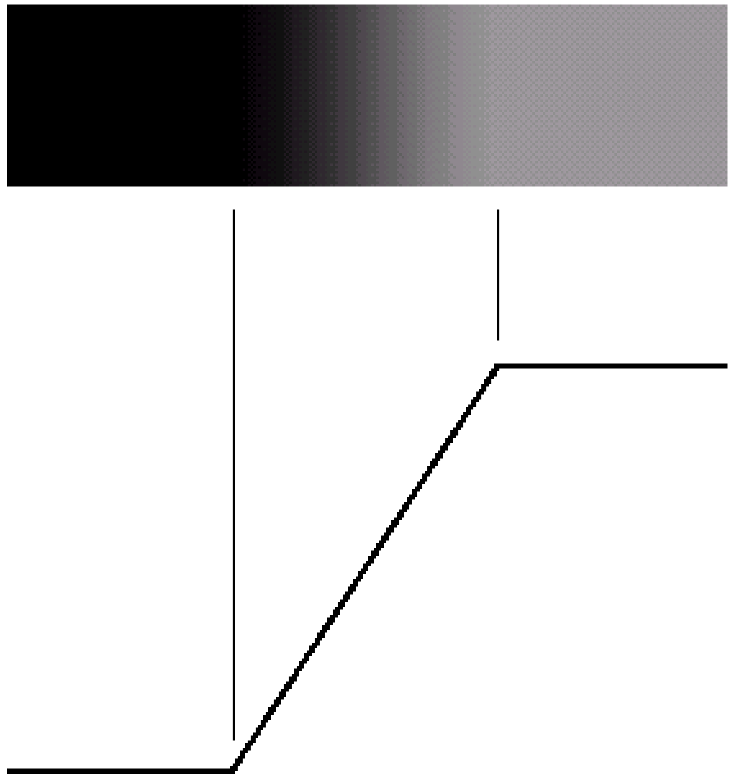
Edge Detection

7. For a ramp edge,
   a. edge point is any point contained in the ramp.
   b. edge length is determined by the length of the ramp.
   c. the slope of the ramp is inversely proportional to the degree of blurring in the edge.

   d. the first derivative of the intensity profile is positive at the points of transition into and out of the ramp (we move from left to right).

   e. the second derivative of the intensity profile is positive at the transition associated with the dark side of the edge, and negative at the transition associated with the light side of the edge.

Model of a ramp digital edge



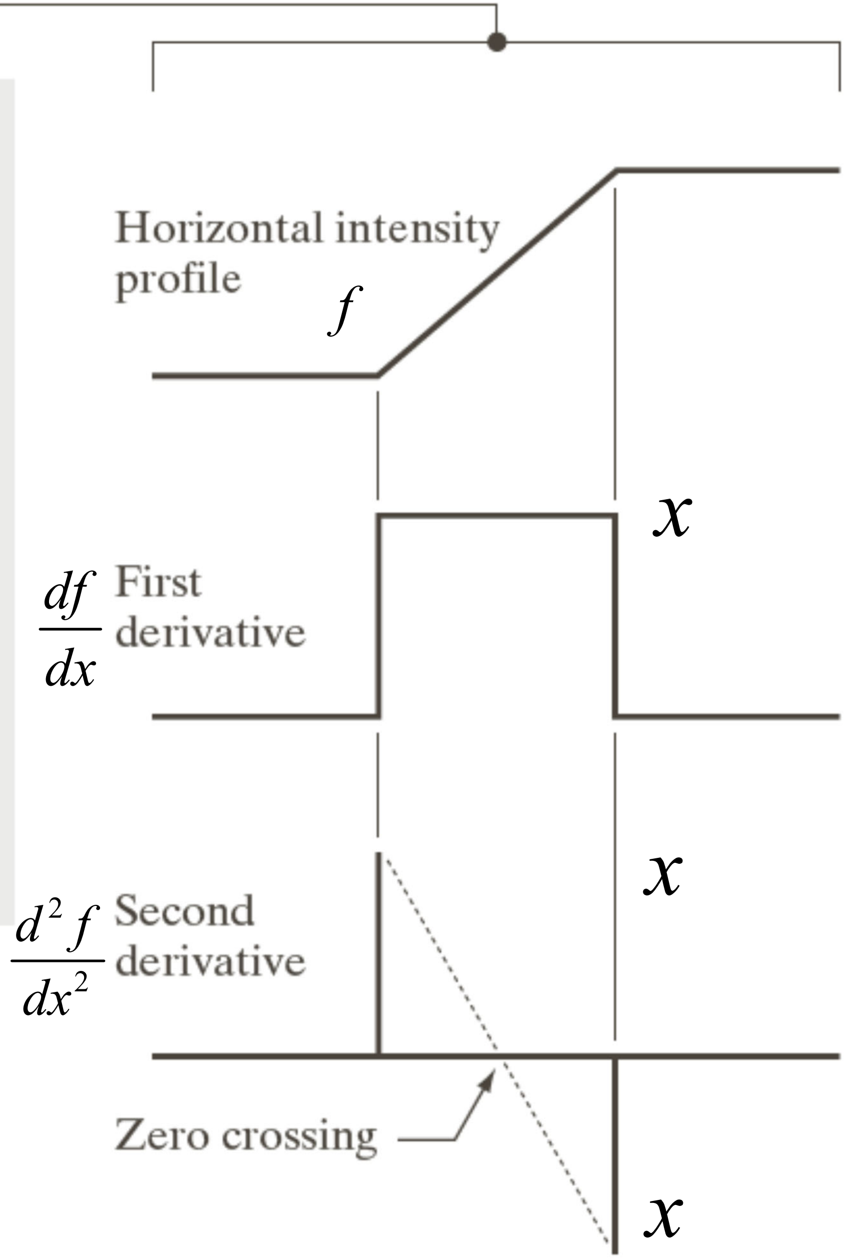
Gray-level profile of a horizontal line through the image

[Gonzalez and Woods]

Edge Detection

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.



Finite difference approximations

$$\frac{df}{dx} \approx f[x+1] - f[x]$$

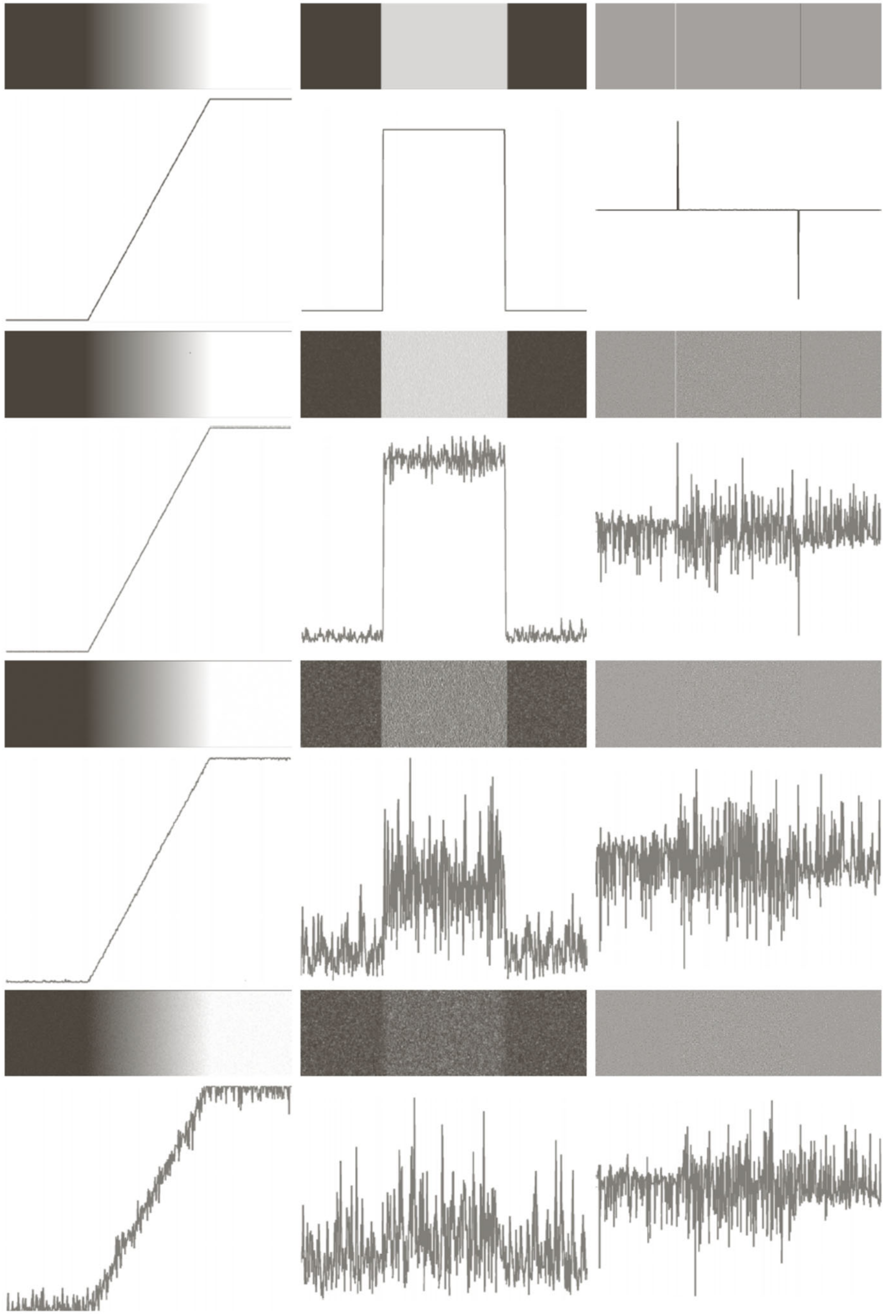
$$\frac{d^2 f}{dx^2} \approx f[x+1] - 2f[x] + f[x-1]$$

[Gonzalez and Woods]

Edge Detection

8. The magnitude of the first derivative can be used to detect the presence of an edge.
9. The sign of the second derivative can be used to determine whether an edge pixel lies on the dark or light side of an edge.
10. The *zero-crossing* property of the second derivative is very useful for locating the centres of thick edge.
11. However, fairly little noise can have a significant impact on the first and second derivatives used for edge detection in images.
12. Image smoothing is commonly used prior to the edge detection so that the estimations of the two derivatives can be more accurate.

[Gonzalez and Woods]



1. Noise free image.
2. The first and second derivatives are estimated correctly.

1. Noisy image.
2. The estimations of the first and second derivatives are corrupted.

FIGURE 10.11 First column: Images and intensity profiles of a ramp edge corrupted by random Gaussian noise of zero mean and standard deviations of 0.0, 0.1, 1.0, and 10.0 intensity levels, respectively. Second column: First-derivative images and intensity profiles. Third column: Second-derivative images and intensity profiles.

[Gonzalez and Woods]

Gradient operator

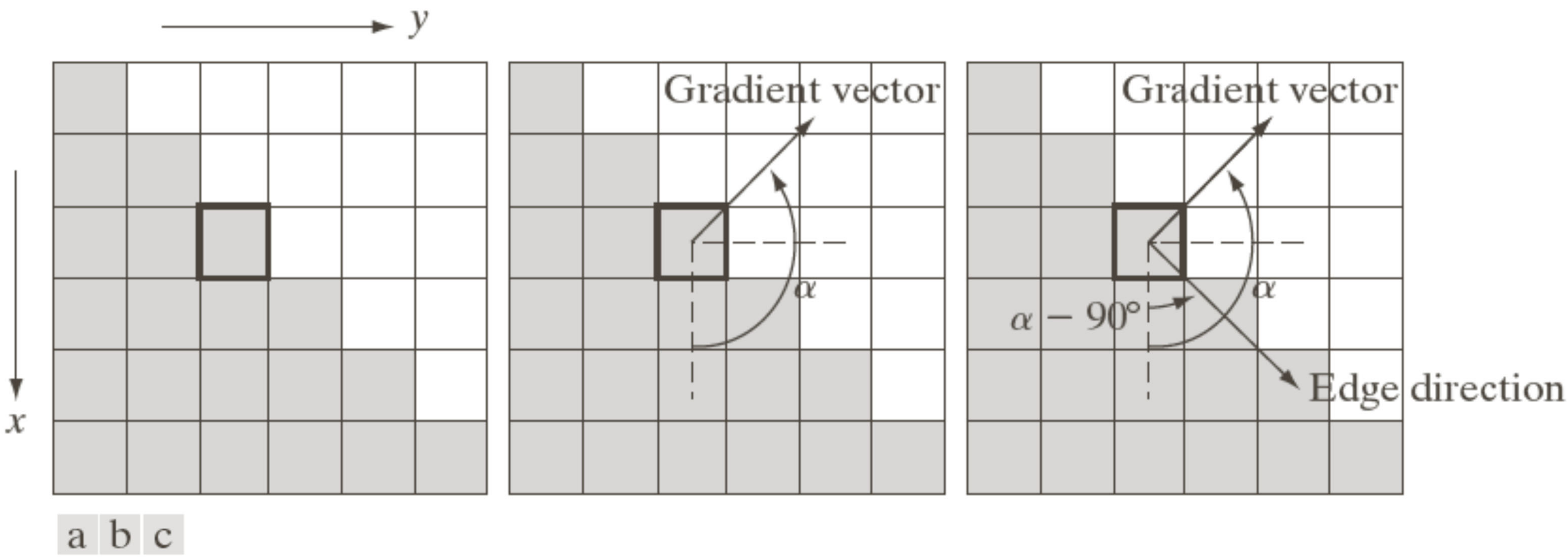


FIGURE 10.12 Using the gradient to determine edge strength and direction at a point. Note that the edge is perpendicular to the direction of the gradient vector at the point where the gradient is computed. Each square in the figure represents one pixel.

[Gonzalez and Woods]

Gradient operator

13. The gradient of an image *f*(*x,y*) at location (*x,y*) is defined as

$$\nabla f = \text{grad}(f) = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix} \text{ where } \begin{aligned} \frac{\partial f}{\partial x} &\approx f[x+1, y] - f[x, y] \\ \frac{\partial f}{\partial y} &\approx f[x, y+1] - f[x, y] \end{aligned}$$

14. Gradient Magnitude

$$|\nabla f| = \text{mag}(\nabla f) = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

[Gonzalez and Woods]

Gradient operator

15. Gradient direction

$$\tan^{-1}\left(\frac{G_y}{G_x}\right)$$

16. The computation of the gradient of an image is based on obtaining the partial derivatives $\partial f/\partial x$ and $\partial f/\partial y$ at every pixel location (x,y).

[Gonzalez and Woods]

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

where z_i represent intensity values.

−1	0
0	1

0	−1
1	0

Roberts

For horizontal edge detection

−1	−1	−1
0	0	0
1	1	1

For vertical edge detection

−1	0	1
−1	0	1
−1	0	1

Prewitt

For horizontal edge detection

−1	−2	−1
0	0	0
1	2	1

For vertical edge detection

−1	0	1
−2	0	2
−1	0	1

Sobel

[Gonzalez and Woods]

Gradient operator

17. Roberts cross-gradient operators

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

$$G_x = \left(z_9 - z_5\right)$$

$$G_y = \left(z_8 - z_6\right)$$

where z_i represent intensity values.

18. Prewitt operators

$$G_x = \left(z_7 + z_8 + z_9\right) - \left(z_1 + z_2 + z_3\right)$$

$$G_y = \left(z_3 + z_6 + z_9\right) - \left(z_1 + z_4 + z_7\right)$$

where z_i represent intensity values.

19. Sobel operators

$$G_x = \left(z_7 + 2z_8 + z_9\right) - \left(z_1 + 2z_2 + z_3\right)$$

$$G_y = \left(z_3 + 2z_6 + z_9\right) - \left(z_1 + 2z_4 + z_7\right)$$

where z_i represent intensity values.

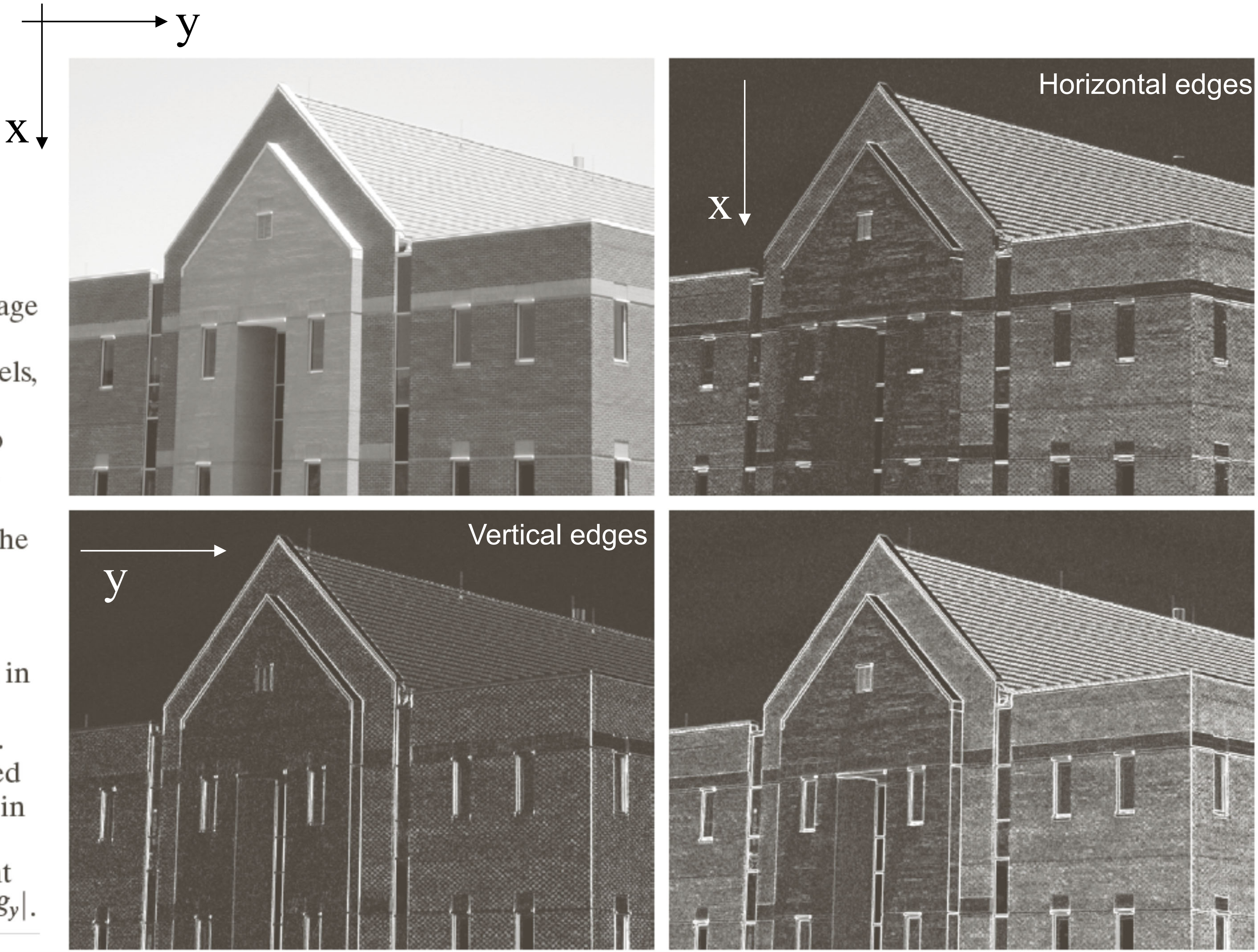
[Gonzalez and Woods]

Gradient operator

20. Approximation

$$\left|\nabla f\right| \approx \left|G_x\right| + \left|G_y\right|$$

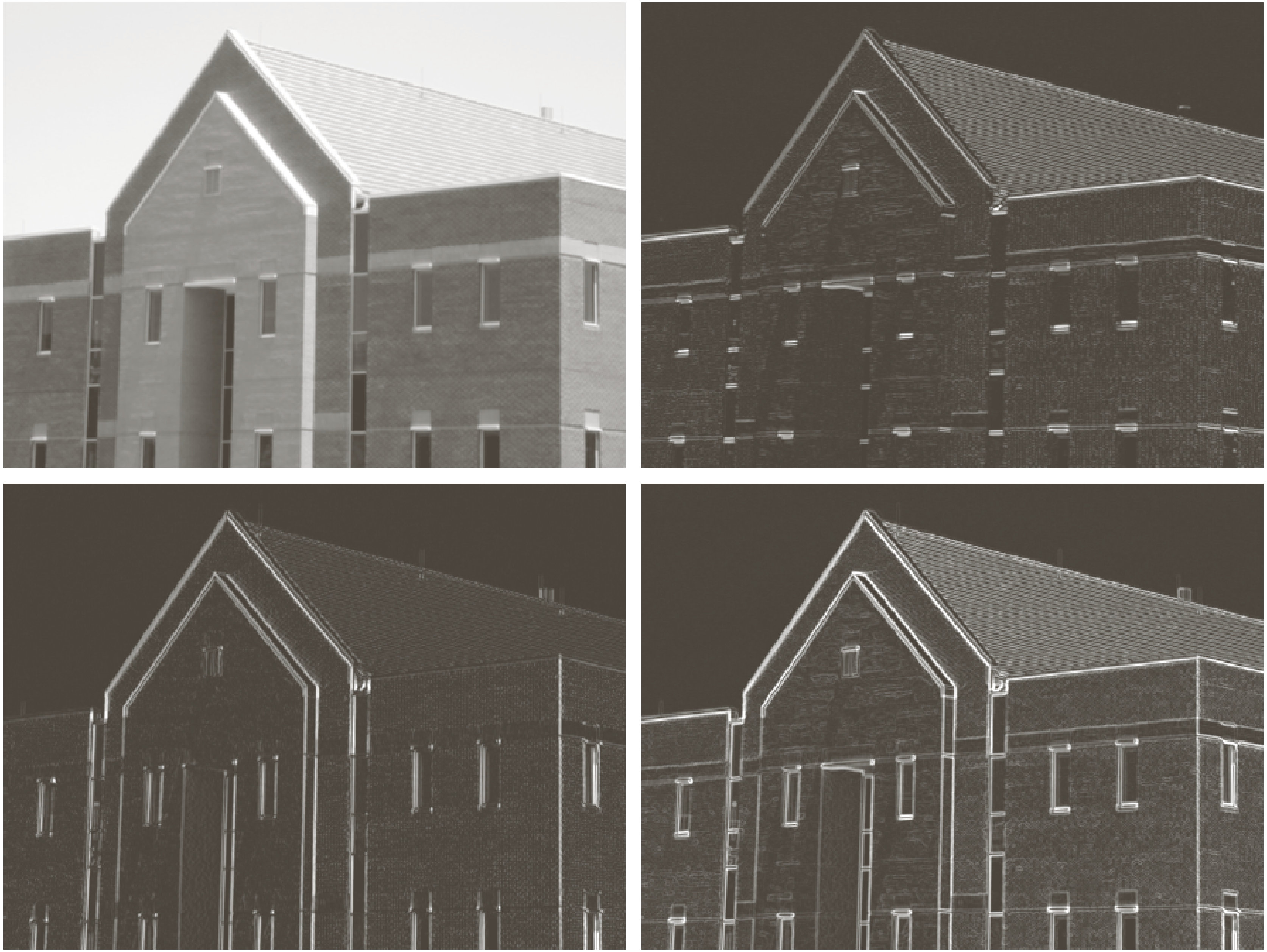
[Gonzalez and Woods]



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, $|g_x| + |g_y|$.

[Gonzalez and Woods]



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.

[Gonzalez and Woods]

Gradient operator

23. Diagonal edge masks for detecting discontinuities in 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

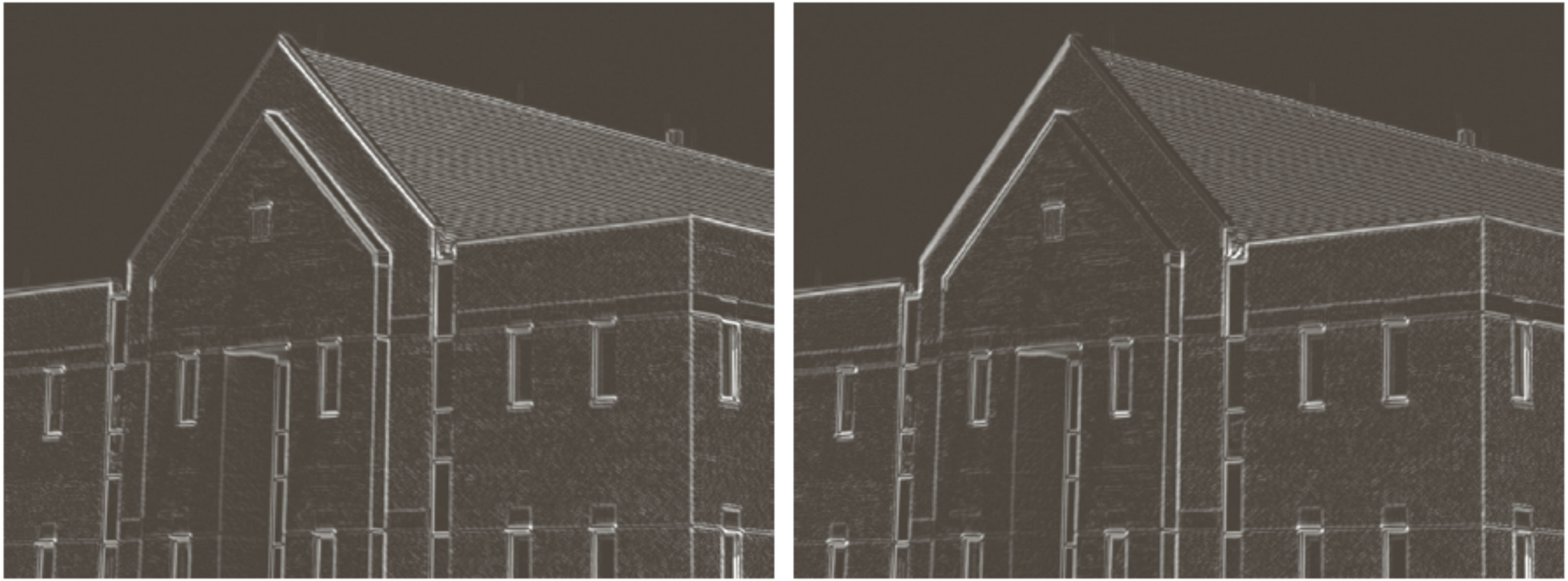
Sobel

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

a	b
c	d

FIGURE 10.15 Prewitt and Sobel masks for detecting diagonal edges.

[Gonzalez and Woods]

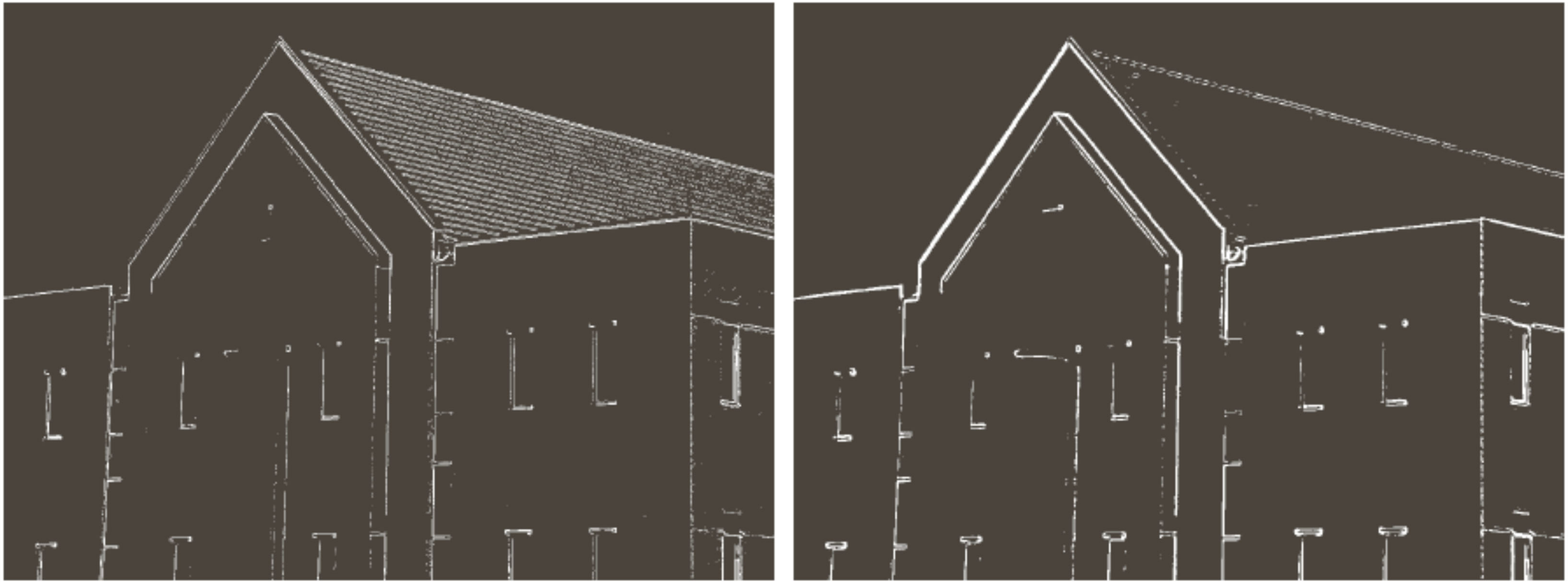


a b

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).

[Gonzalez and Woods]

Combining the gradient with thresholding



a b

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.

[Gonzalez and Woods]

Gradient operator

Fig1016(a)(building_original).tif

From: http://www.imageprocessingplace.com/DIP-3E/dip3e_book_images_downloads.htm

24. MATLAB example (filename: `edge_detection.m`)

```
% read an image
f = imread('Fig1016(a)(building_original).tif');
imshow(f, [min(min(f)) max(max(f))]);
title('Original Input Image');

[g, t]= edge(f, 'sobel', 0.1, 'both');
imshow(g, [0 1]);
title('Binary edge Image (Sobel)');
```

More information about the **`edge`** function:
http://www.mathworks.com/help/images/ref/edge.html

[Gonzalez and Woods] [Gonzalez, Woods and Eddins]

The Laplacian

1. The Laplacian of an image $f(x,y)$ at location (x,y) is defined as

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

2. Two approximations

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

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

The Laplacian

3. Two masks for the approximations

0	−1	0		−1	−1	−1
−1	4	−1		−1	8	−1
0	−1	0		−1	−1	−1

4. The Laplacian generally is not used in its original form for edge detection (based on zero-crossing property) because it is unacceptably sensitive to noise.
5. Solution: smooth the image by using a Gaussian blurring function $h(r)$ before we apply the Laplacian operator.

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

The Laplacian

6. *The Laplacian of a Gaussian* (LoG) operator

$$LoG(f) = \nabla^2 (f * h)$$

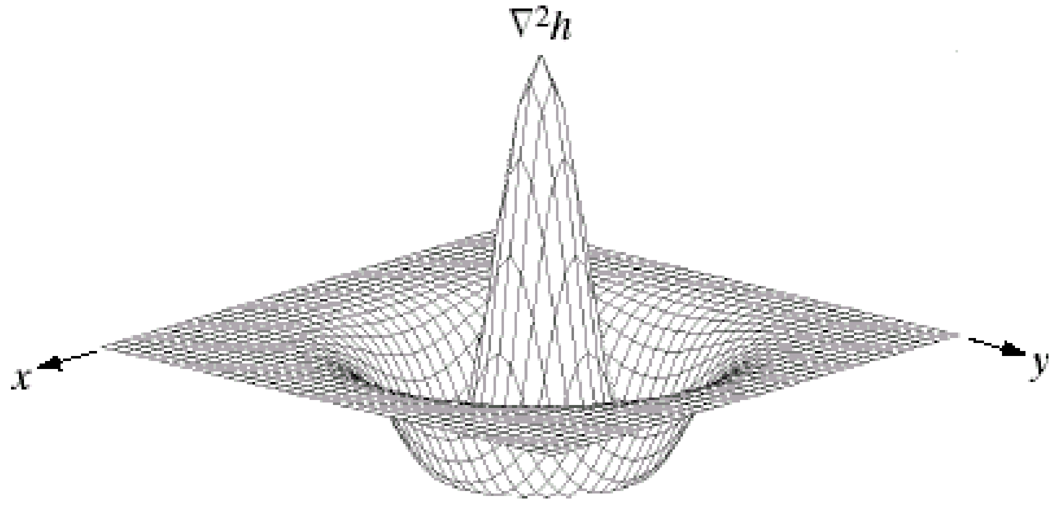
7. Another definition

$$LoG(f) = f * (\nabla^2 h)$$

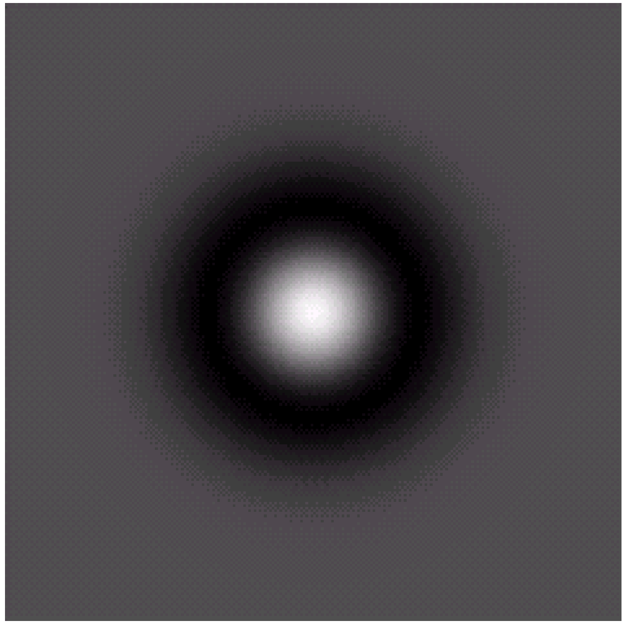
where

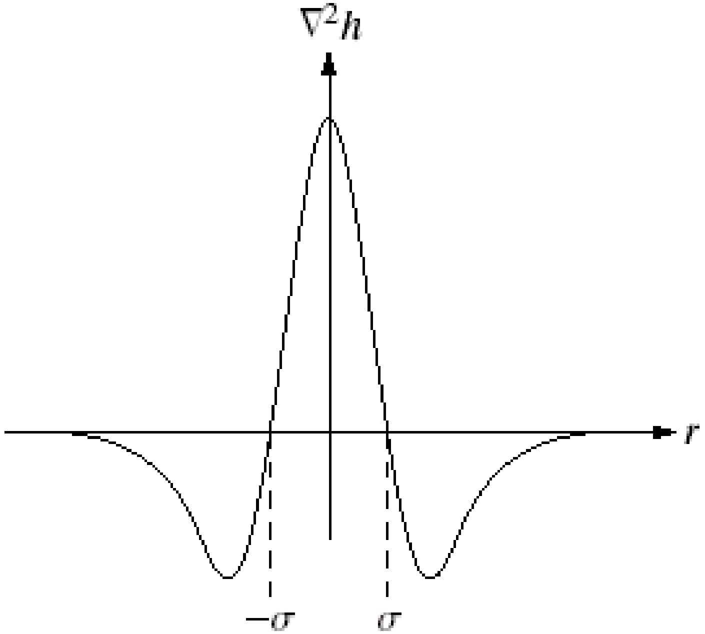
$$\nabla^2 h = -\left[\frac{r^2 - 2\sigma^2}{\sigma^4}\right] e^{\frac{-r^2}{2\sigma^2}}$$

8. Using the LoG, the location of edges can be detected reliably based on the zero-crossing values. It is because noise on the image is reduced by the Gaussian function.



Mexican hat function





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).

Original image

Sobel gradient magnitudes

Gaussian function

Laplacian mask

Thresholded LoG

Zero-crossings

Laplacian of Gaussian (LoG) filter responses

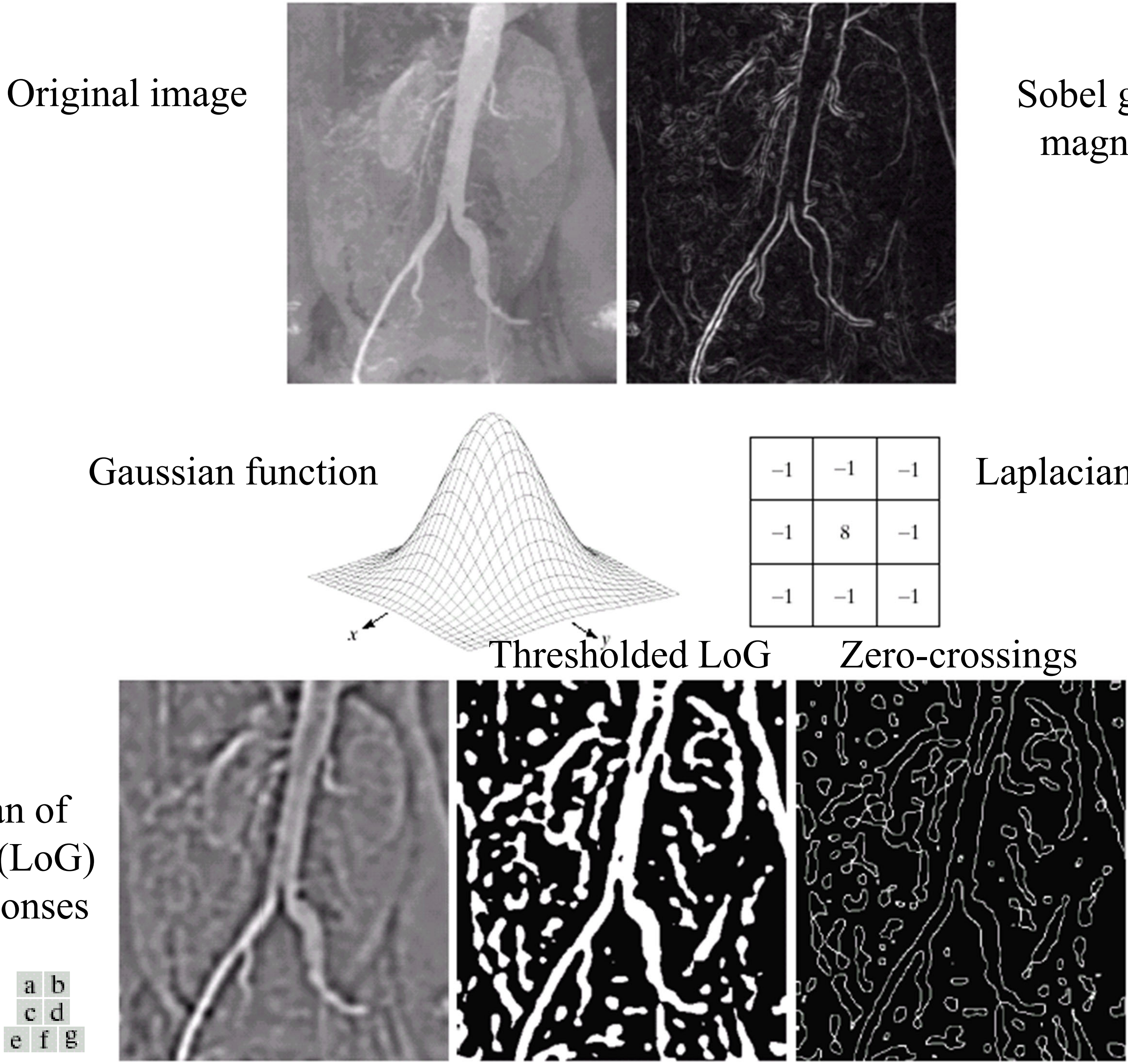


FIGURE 10.15 (a) Original image. (b) Sobel gradient (shown for comparison). (c) Spatial Gaussian smoothing function. (d) Laplacian mask. (e) LoG. (f) Thresholded LoG. (g) Zero crossings. (Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University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

The Laplacian in frequency domain

1. The Laplacian of an image $f(x,y)$ at location (x,y) is defined as

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

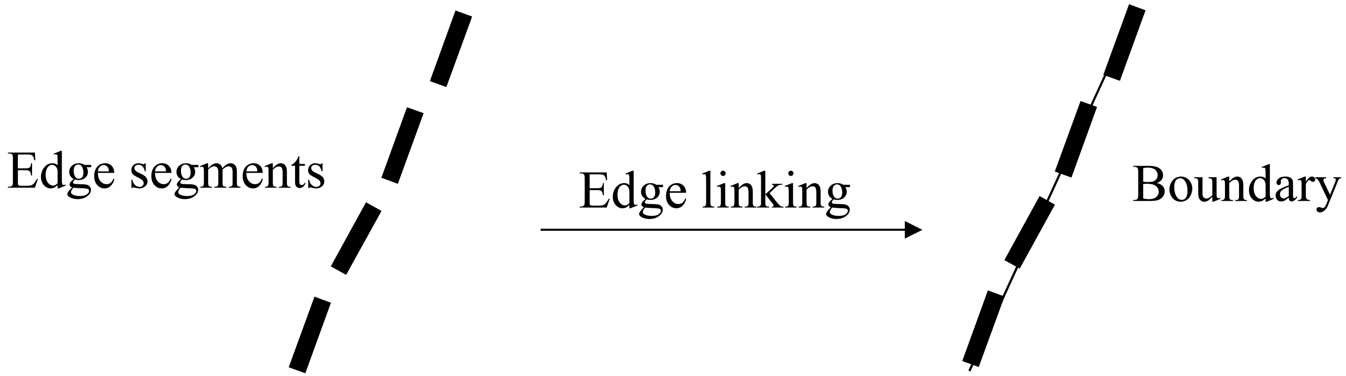
2. Frequency representation

$$\Im\left\{\nabla^2 f(x,y)\right\} = -\left(u^2 + v^2\right) F(u,v)$$

3. LoG in frequency domain?

Edge linking and boundary detection

1. Edge points/pixels seldom characterise an edge completely.
2. Reason: noise, breaks in the edge from non-uniform illumination, etc.
3. Solutions: edge points are linked to become a meaningful boundary by using either
   a. Local processing, or
   b. Global processing (based on Hough Transform).



Local processing

1. Idea: all edge points that are similar according to a set of predefined criteria are linked/connected.
2. Criterion 1: similar strength of the response of the gradient operator, which is used to produce an edge pixel.

$$\left| \left| \nabla f(x, y) \right| - \left| \nabla f(x_0, y_0) \right| \right| \leq E$$

where (x_0, y_0) represents current pixel coordinates, (x,y) represents neighbour pixel coordinates, E is a non-negative magnitude threshold.

3. Criterion 2: similar direction of the gradient operator.

$$\left| \alpha(x, y) - \alpha(x_0, y_0) \right| \leq A$$

where A is a non-negative angle threshold.

Local processing

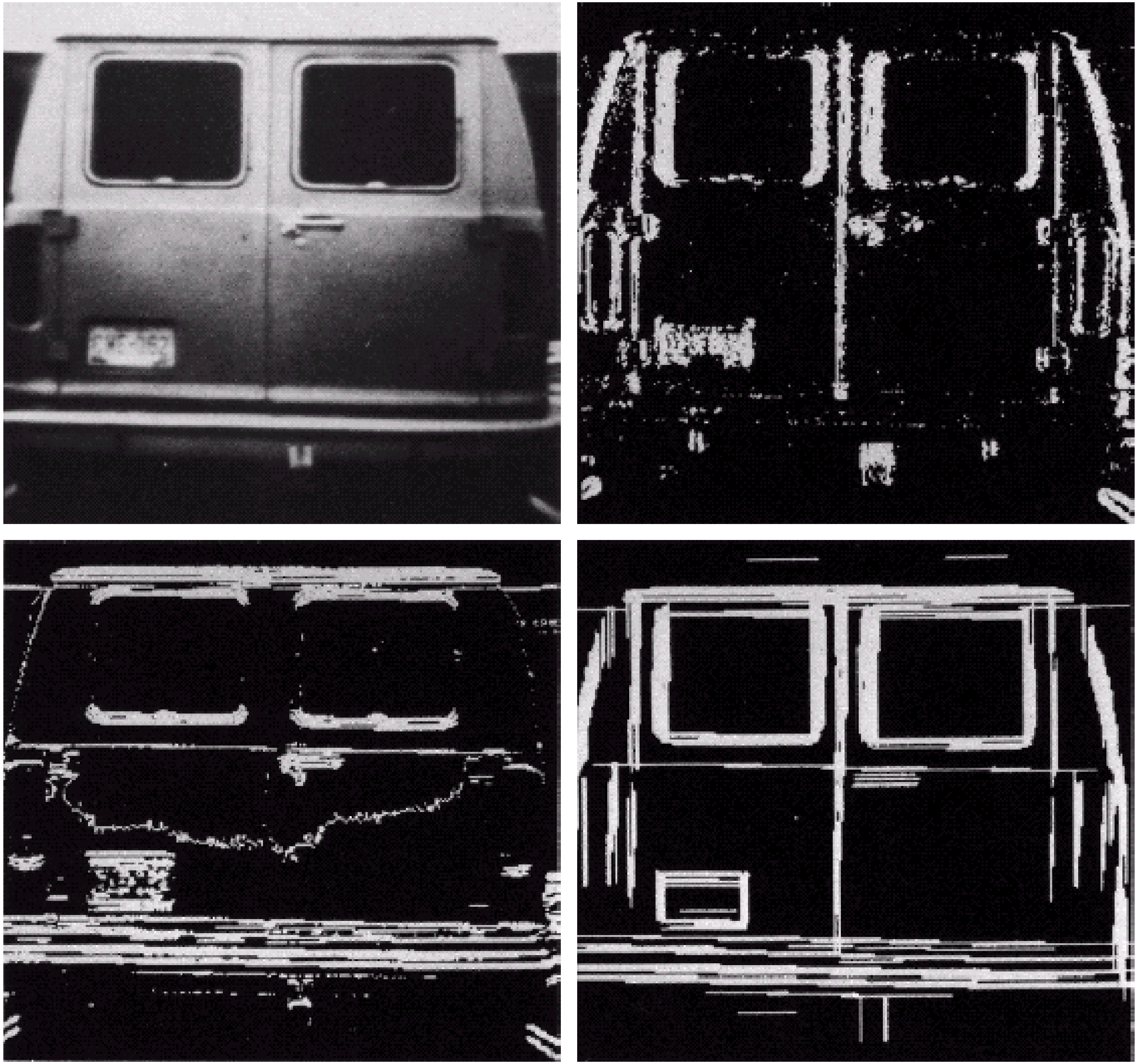
4. Criterion 3: close distance between current and neighbour pixel points.

$$\sqrt{\left(x-x_0\right)^2+\left(y-y_0\right)^2} \leq D$$

where (x_0,y_0) represents current pixel coordinates, (x,y) represents neighbour pixel coordinates, E is a non-negative distance threshold.

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.)

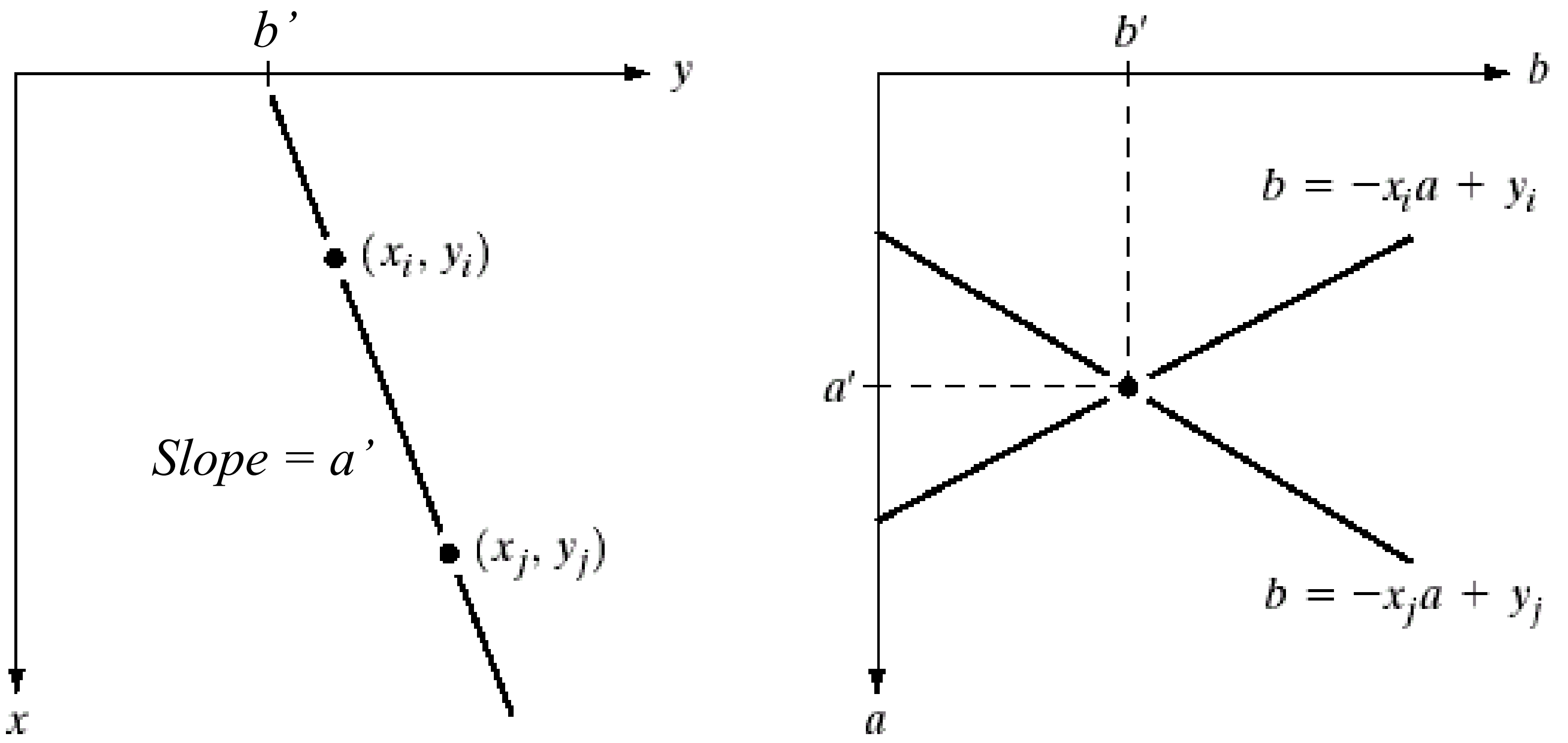


$E = 25,$

$A = 15^{o}$

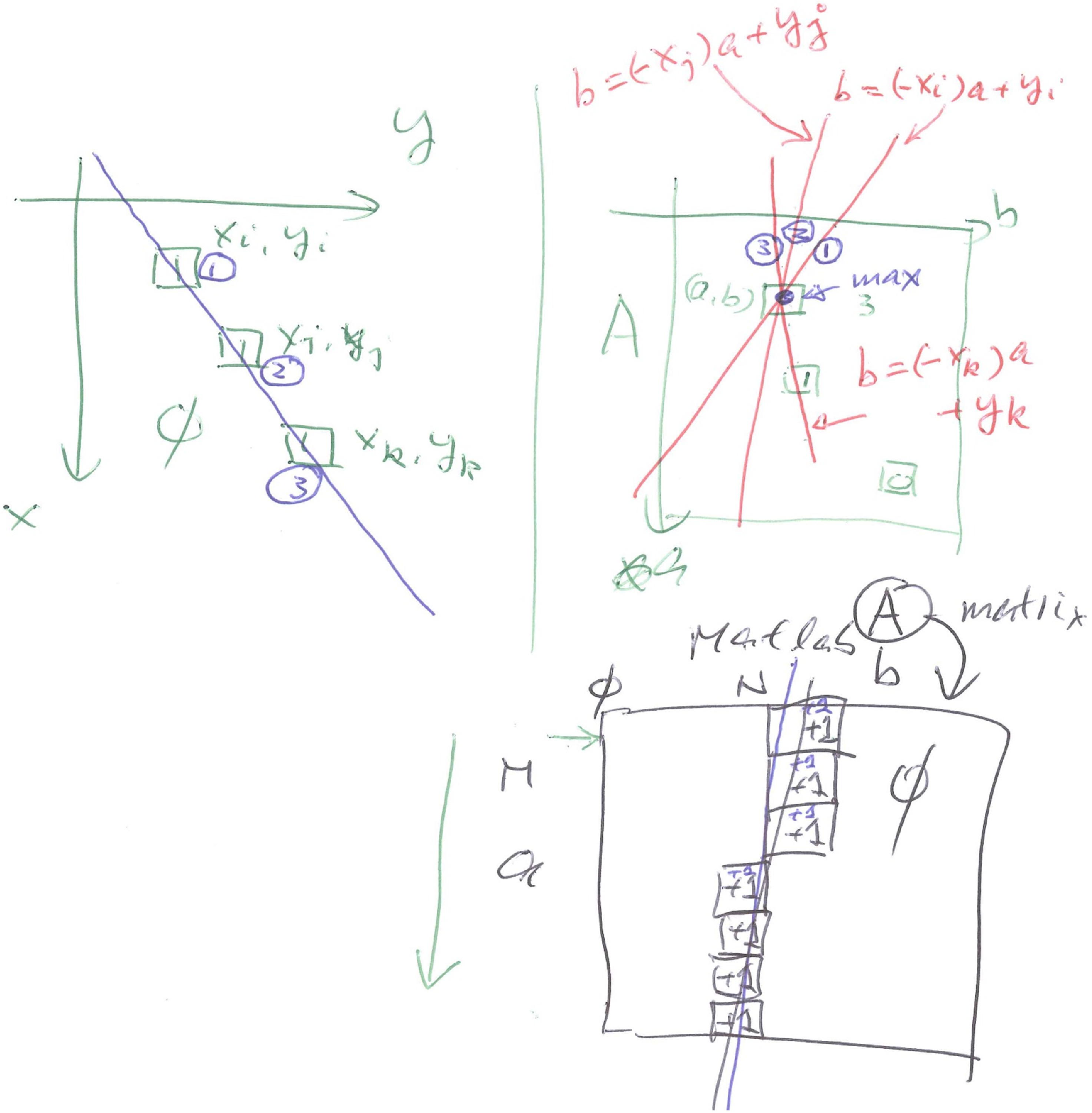
Global processing via Hough transform

1. Idea based on Hough transform: if two points lie on the same straight line, then they should have the same values of slope and y-intercepts.

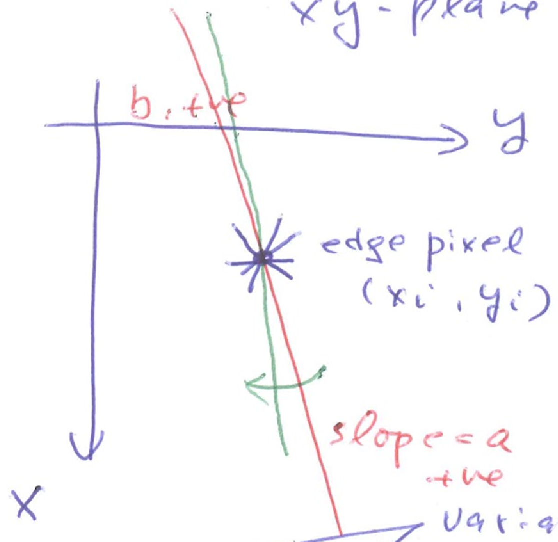


a b

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



y
x
x_i, y_i
x_j, y_j
x_k, y_k
$b=(-x_j)a+y_j$
$b=(-x_i)a+y_i$
$b=(-x_k)a+y_k$
b
A
(a,b)
max
3
Matlab
A
matrix
N
b
M
a
+1



variables

$$y = ax + b$$

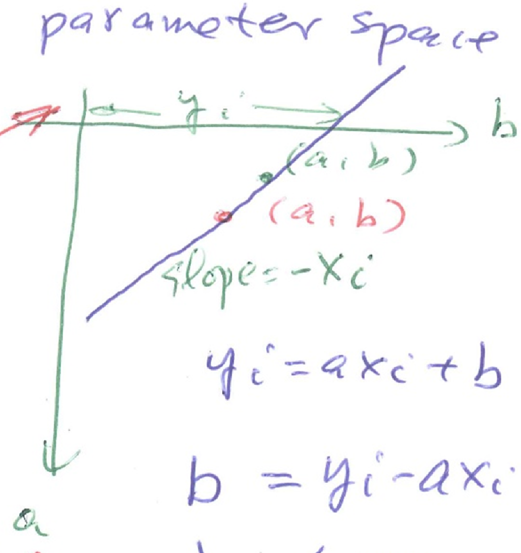
slope (constant)

y-intercept (constant)

$$y_i = ax_i + b$$

constant

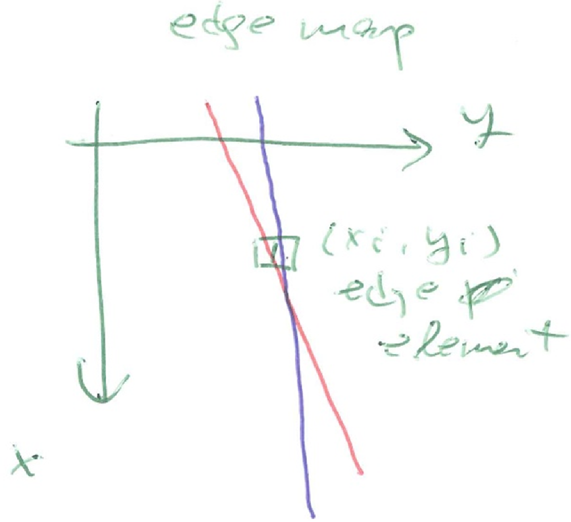
variables

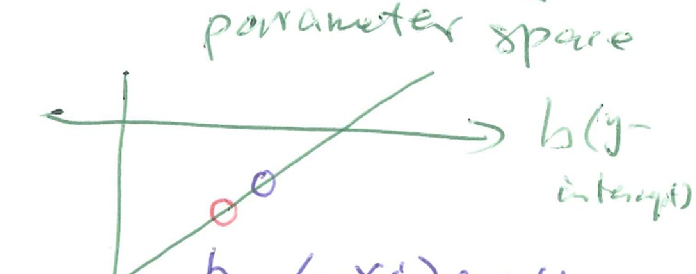


$$y_i = ax_i + b$$

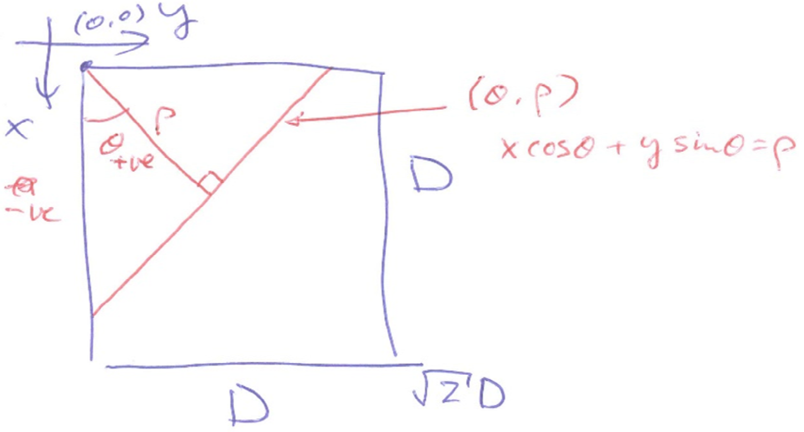
$$b = y_i - ax_i$$

$$b = \underbrace{(-x_i)}_{\text{slope}} a + y_i$$

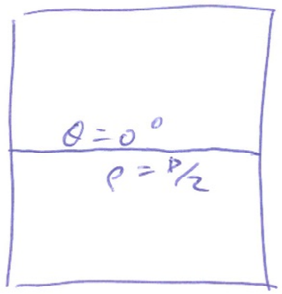




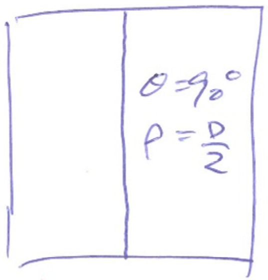
$$b = (-x_i)a + y_i$$



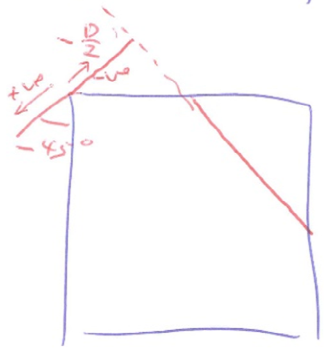
$\theta = 0°, \rho = \frac{D}{2}$

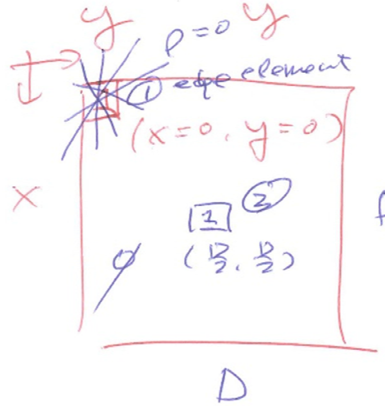


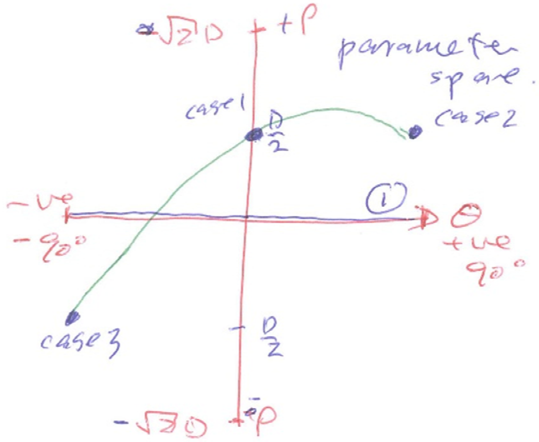
$\theta = 90°, \rho = \frac{D}{2}$



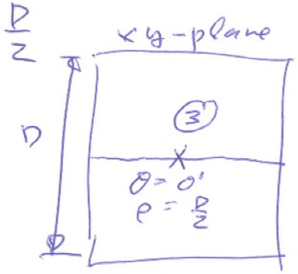
$\theta = -45°, \rho = -D/2$



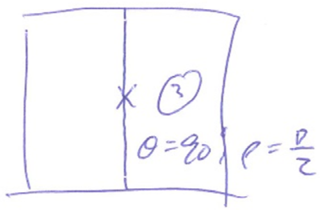




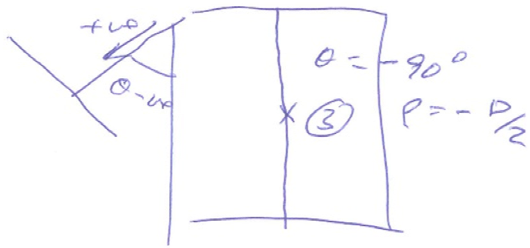
case 1, $\theta = 0°, \rho = \frac{D}{2}$



case 2. $\theta = 90°, \rho = \frac{D}{2}$



case 3, $\theta = -90°, \rho = -\frac{D}{2}$



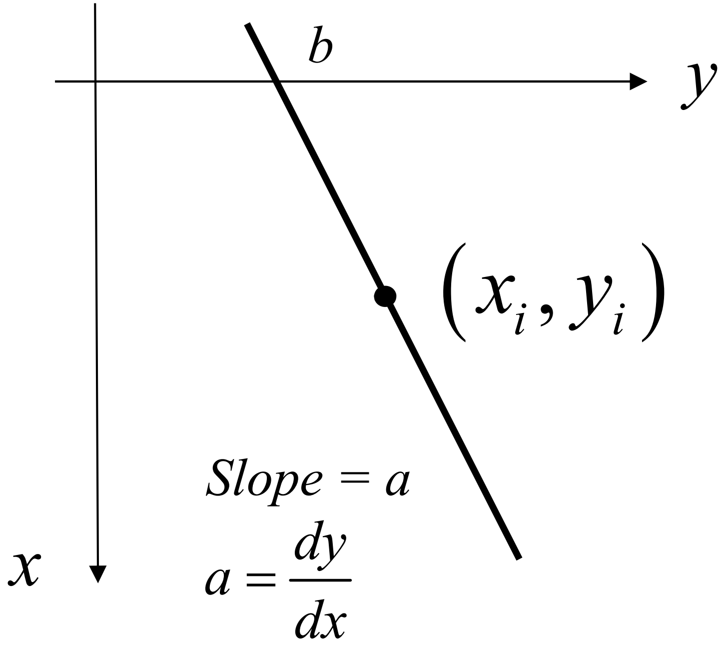
Global processing via Hough transform

2. Concepts and procedures:
   a. For a point (x_i,y_i), we set up a straight line equation.

   $$y_i = ax_i + b \Leftrightarrow b = (-x_i)a + y_i$$

   where a = slope (variable), b = y-intercept (variable), x_i and y_i are known.

   b. We subdivide the a axis into K increments between $[a_{min}, a_{max}]$. For each increment of a, we evaluate the value of b.
   c. A relationship between a and b can be plotted in a parameter space.
   d. We partition the parameter space into a number of bins (accumulator cells), and increment the corresponding bin $A(a,b)$ by 1 (b is rounded into the nearest interger).



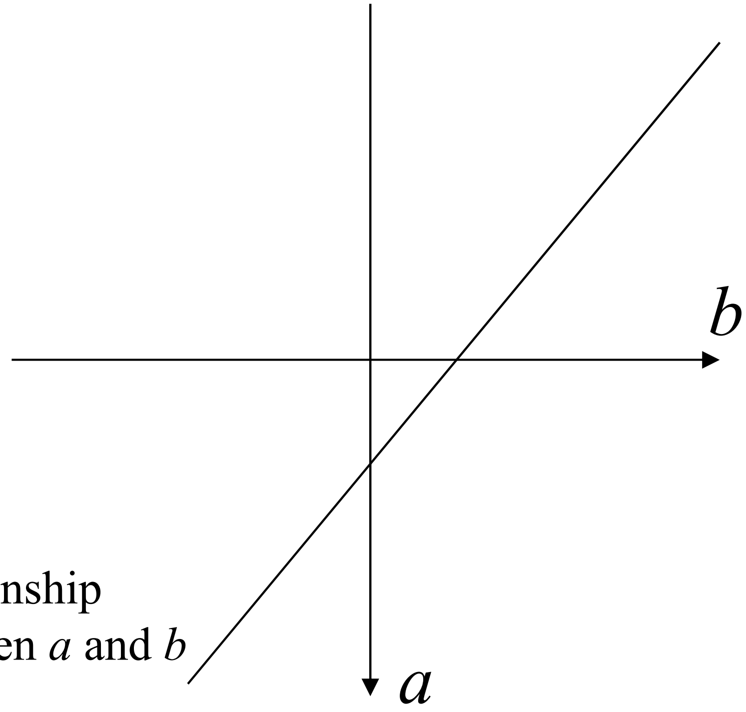
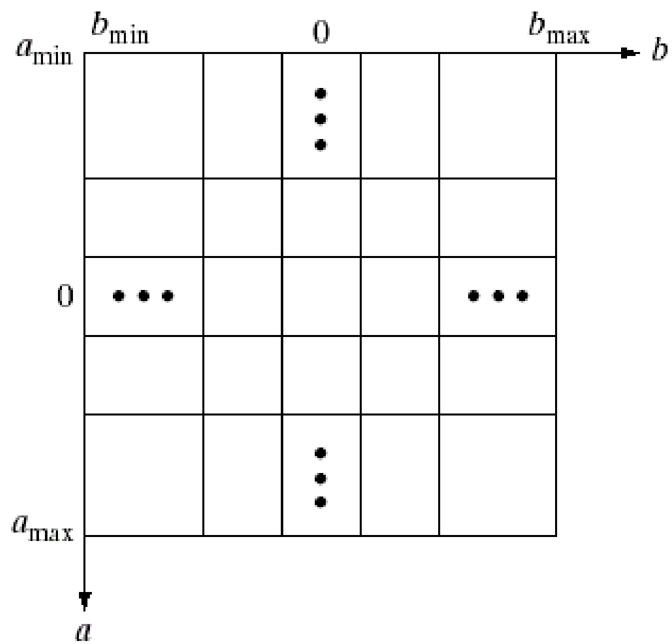


FIGURE 10.18 Subdivision of the parameter plane for use in the Hough transform.

$A(a,b)$



Global processing via Hough transform

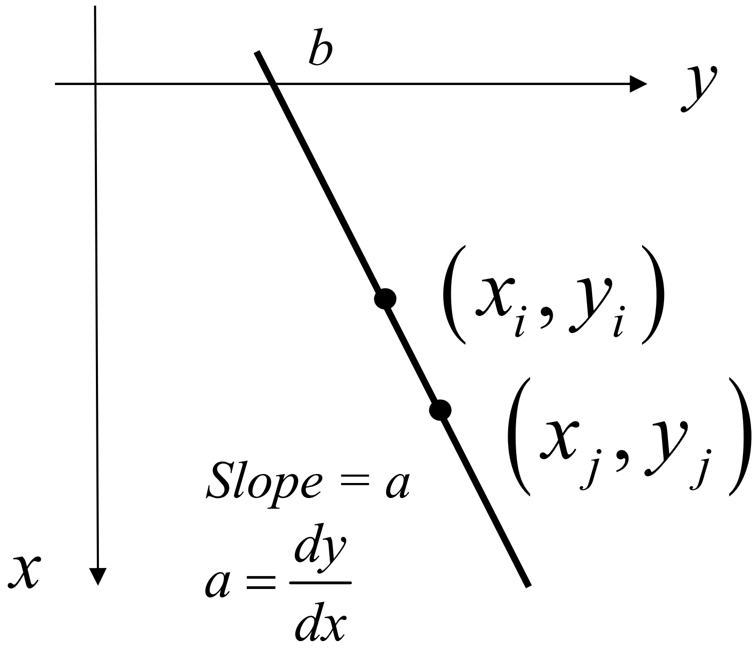
2. Concepts and procedures:

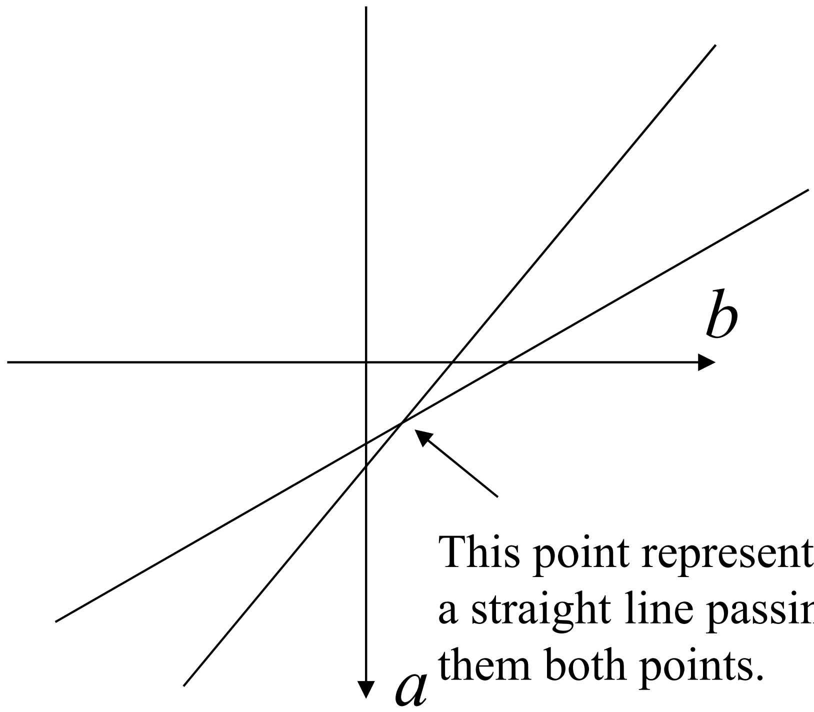
   e. For another point (x_j, y_j), we set up another straight line equation.

   $$y_j = ax_j + b \Leftrightarrow b = \left(-x_j\right)a + y_j$$

   f. Similarly, we subdivide the a axis into K increments between $[a_{min}, a_{max}]$. For each increment of a, we evaluate the value of b. We plot the relationship between a and b in the same parameter space, and update bin values in the discrete parameter space.

   g. The bin $A(a,b)$ having the highest count corresponds to the straight line passing through the points (x_i, y_i) and (x_j, y_j).


b
y
(x_i, y_i)
(x_j, y_j)
$Slope = a$
$a = \frac{dy}{dx}$
x


b
a
This point represents
a straight line passing
them both points.

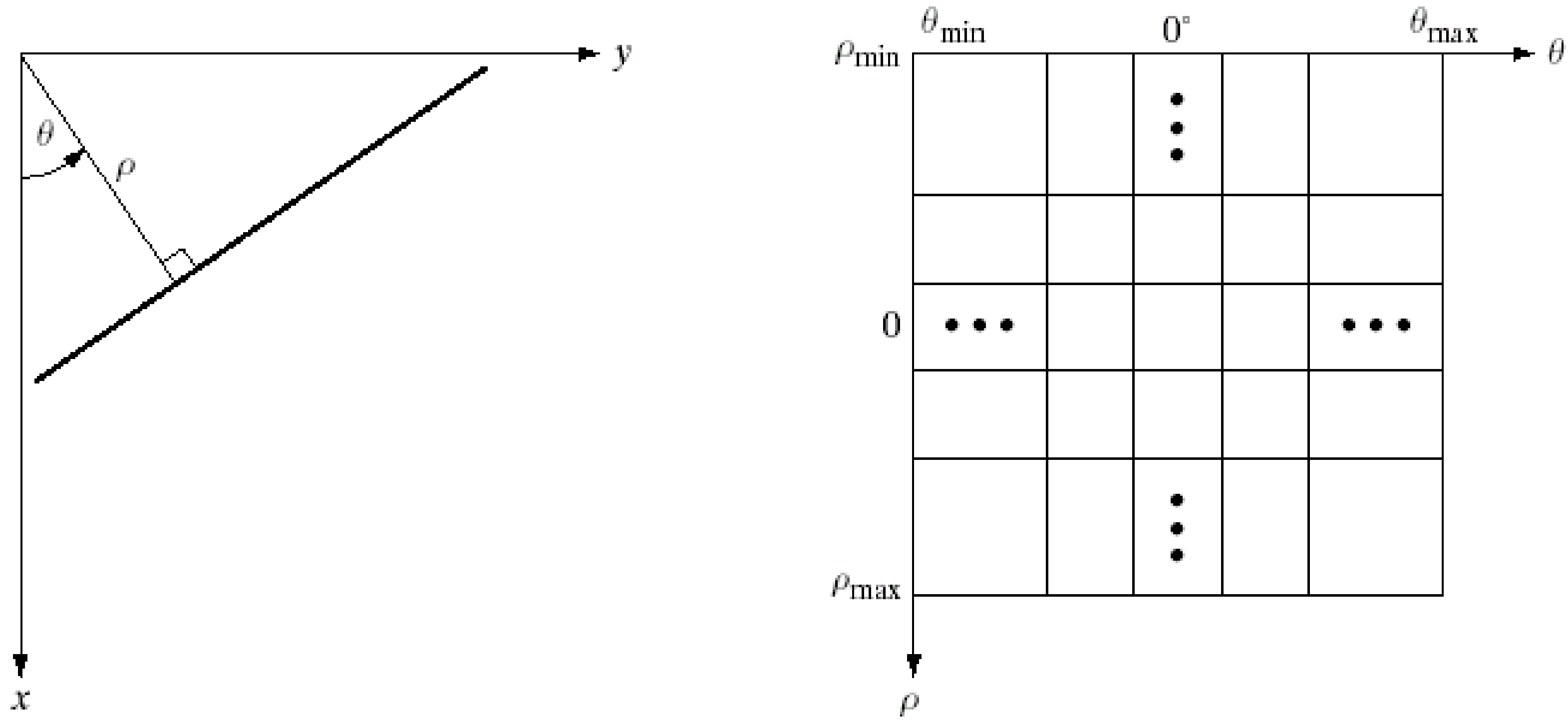
Global processing via Hough transform

2. Concepts and procedures:
   h. The same procedure can be applied to all points. The bin $A(a,b)$ having the highest count corresponds to the straight line passing through (or passing near) the largest number of points.

3. Problem? Values of a and b run from negative infinity to positive infinity. We need infinite number of bins!

Global processing via Hough transform

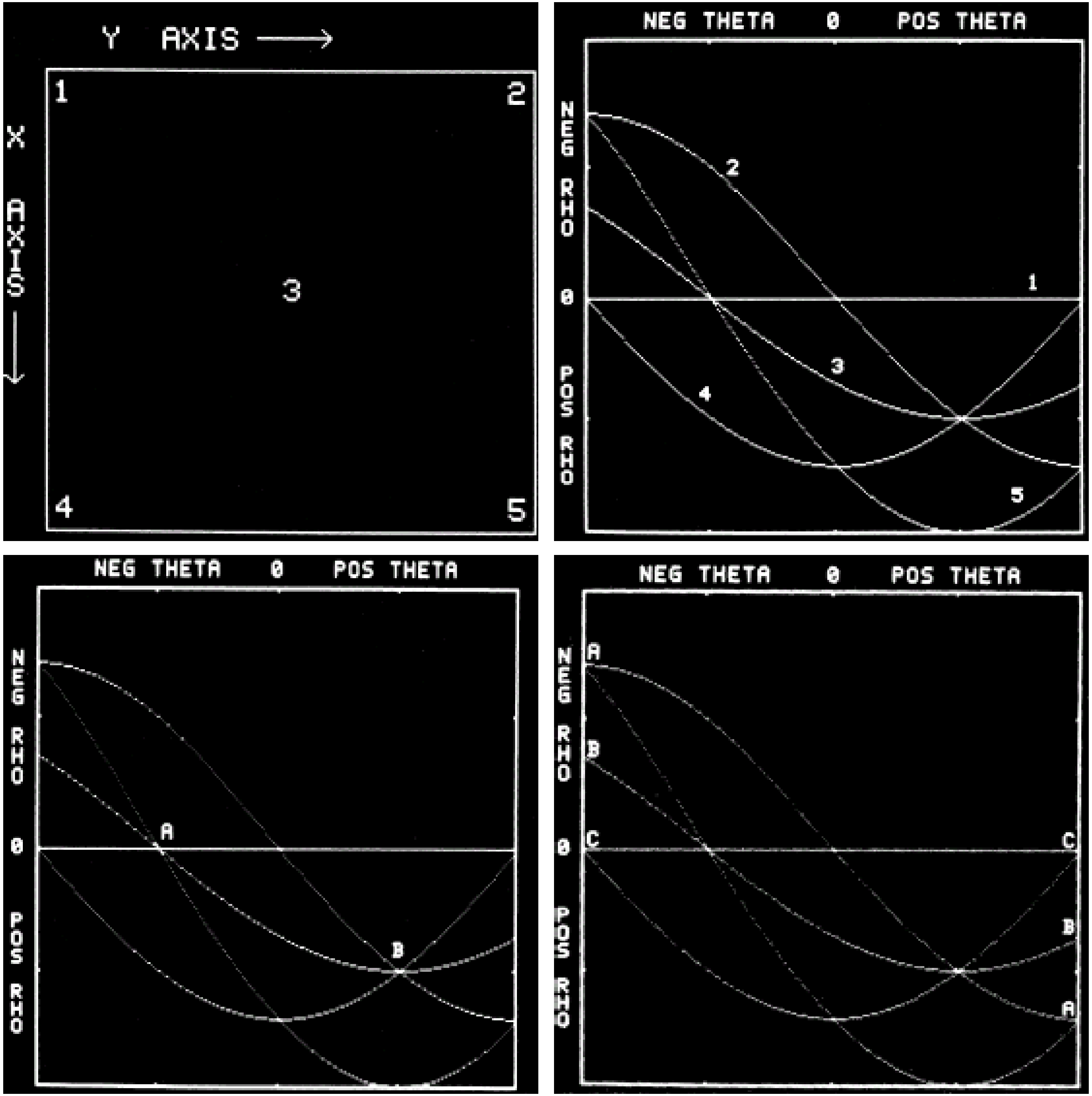
4. Solution: use normal representation of a line.



a b

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

5. Theta runs from –90° to 90°. Rho runs from $-\sqrt{2}D$ to $\sqrt{2}D$, where D is the distance between corners in the image (length and width).



a b
c d

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

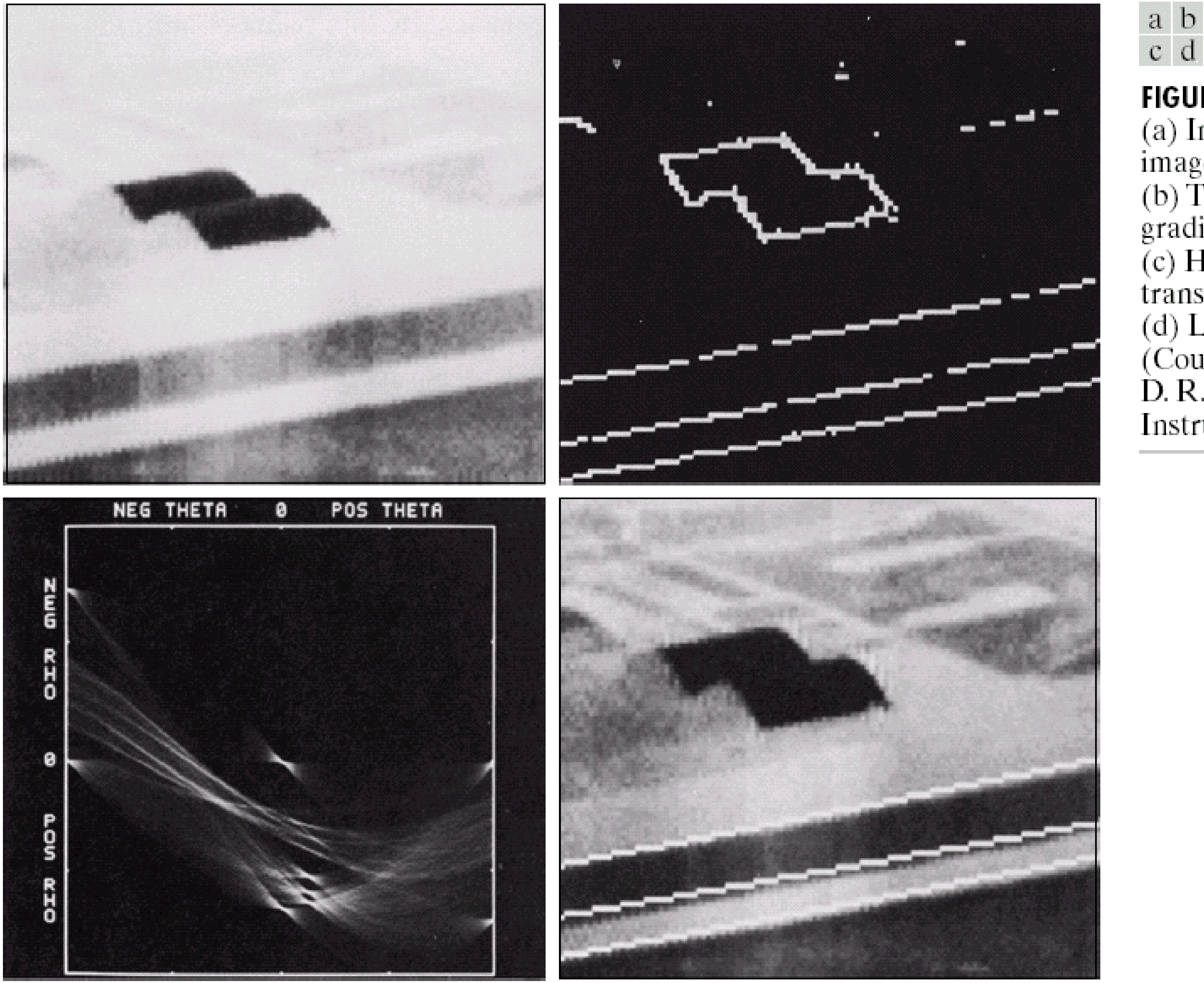


FIGURE 10.21 (a) Infrared image. (b) Thresholded gradient image. (c) Hough transform. (d) Linked pixels. (Courtesy of Mr. D. R. Cate, Texas Instruments, Inc.)

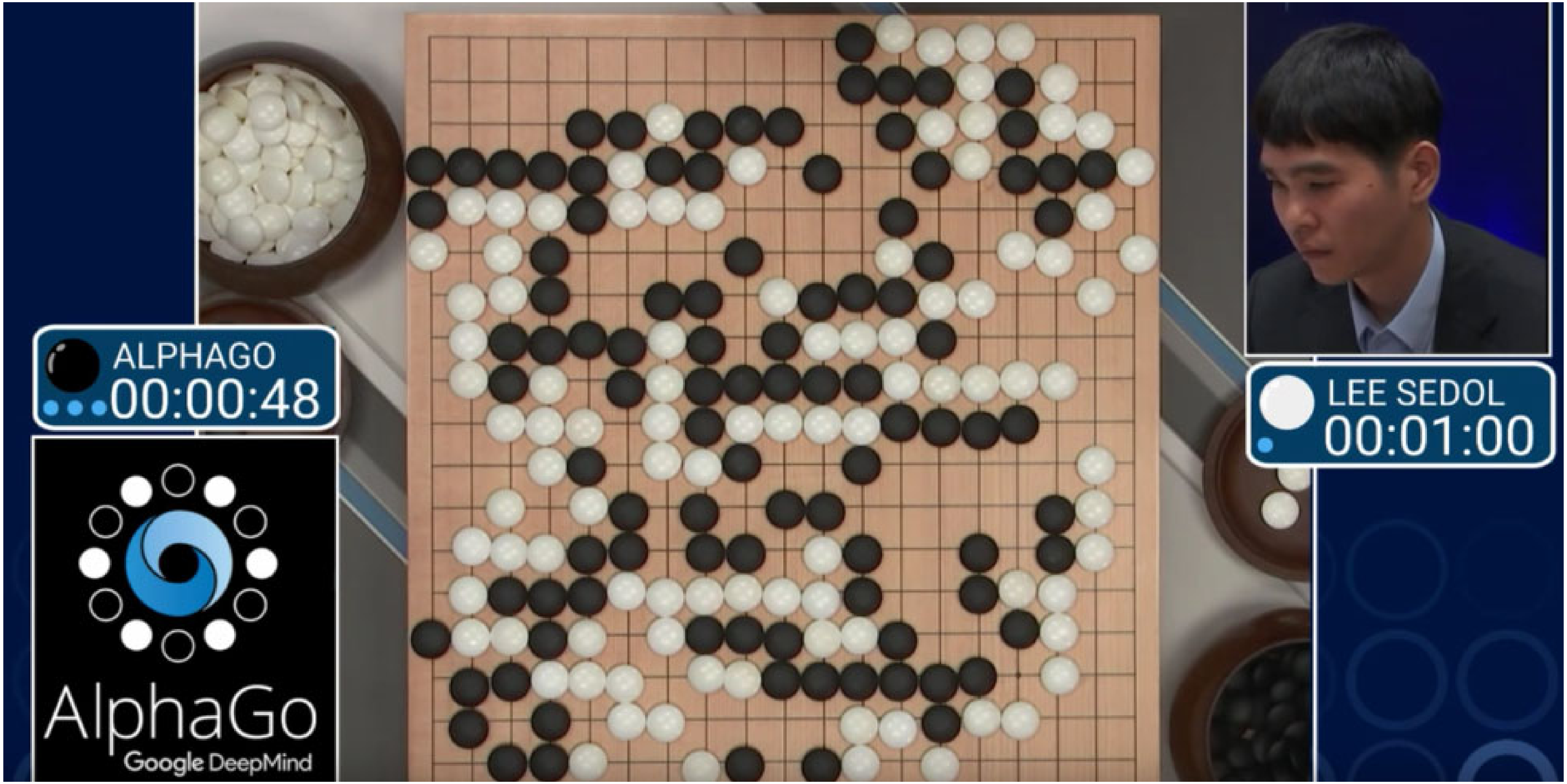
Global processing via Hough transform

6. This method can also use circle instead of straight line. How?

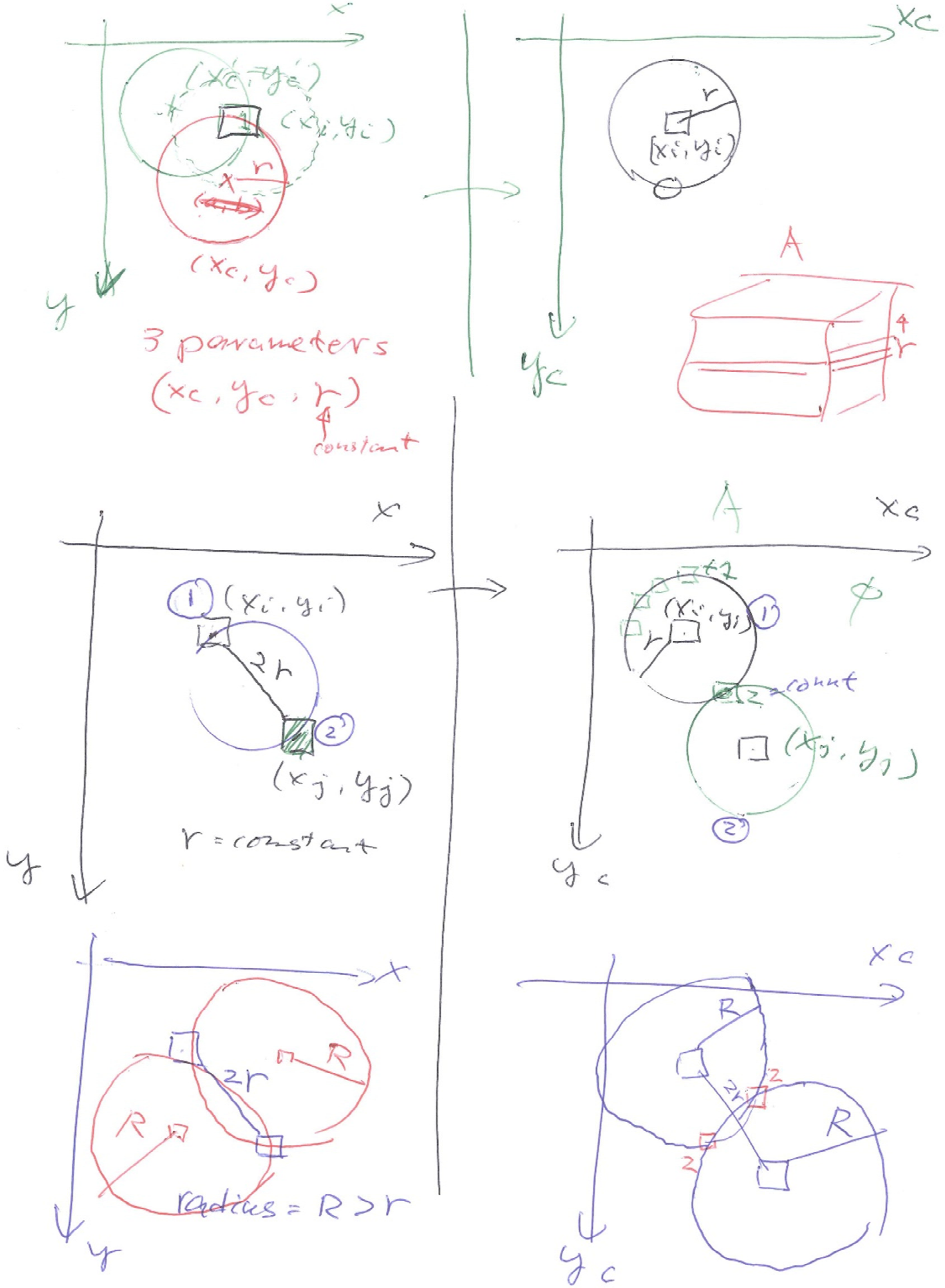
$$\left(x-c_1\right)^2+\left(y-c_2\right)^2=c_3$$

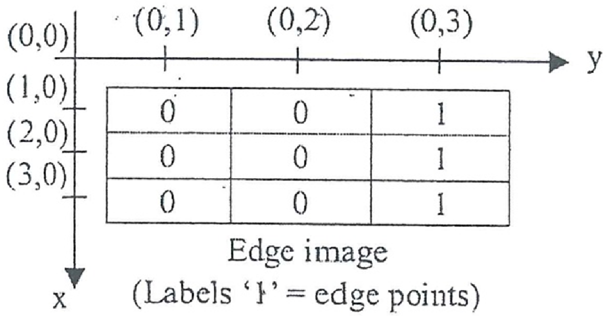
7. Solution to the edge linking problem:
   a. Compute the gradient of an image and threshold it to obtain a binary image.
   b. Partition the parameter space, e.g., *a-b* plane or theta-rho plane.
   c. Examine the bin counts, pick the highest count.
   d. Examine the relationship (based on criteria) between pixels.

Finding stones in a board of a 19x19 grid……

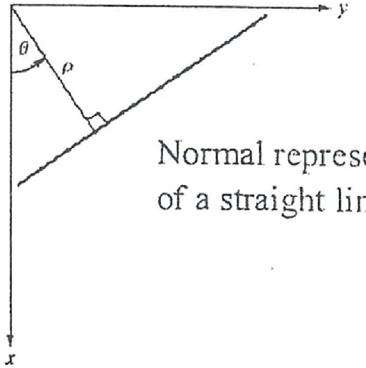


www.popularmechanics.com


x
y
(xc, yc)
(xi, yi)
r
(xc, yc)
3 parameters
(xc, yc, r)
constant
xc
yc
r
(xi, yi)
A
r
A
x
y
1
(xi, yi)
2r
2
(xj, yj)
r = constant
xc
yc
A
φ
(xi, yi)
r
1
2 = const
(xj, yj)
2
x
y
2r
R
R
radius = R > r
xc
yc
R
R
2r
2
2



Edge image
(Labels '1' = edge points)



Normal representation of a straight line

Assuming that the normal representation of a line $x\cos\theta + y\sin\theta = \rho$ is used, fill in the accumulator cells $A(\rho,\theta)$ below. (Hints: for every edge point, find the corresponding value of ρ when θ is fixed. In the table below, round ρ to the nearest integer)

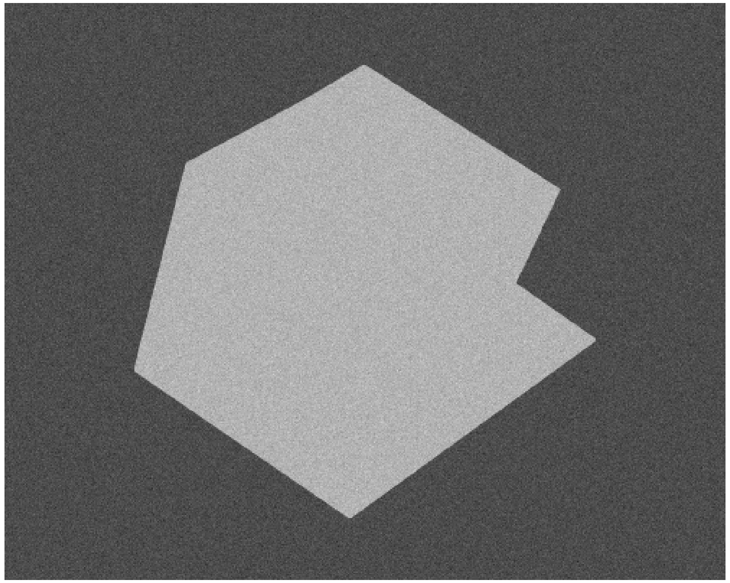
Accumulator Cells $A(\rho,\theta)$

	$\theta = 0^\circ$	$\theta = 45^\circ$	$\theta = 90^\circ$	$\theta = 135^\circ$
$\rho = 0$				
$\rho = 1$				
$\rho = 2$				
$\rho = 3$				
$\rho = 4$				
$\rho = 5$				

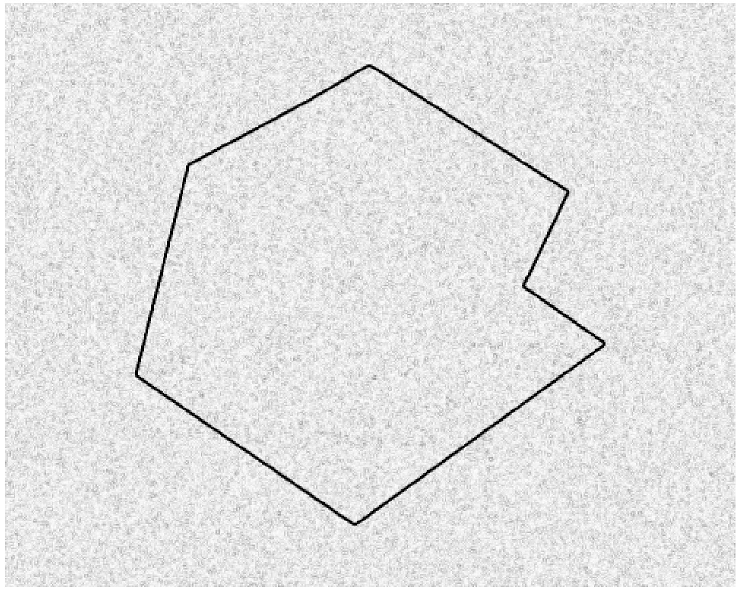
Global intensity based segmentation (Global thresholding)

Global thresholding

1. Previously, we focused on image segmentation based on the gradient values and boundaries.



Original image

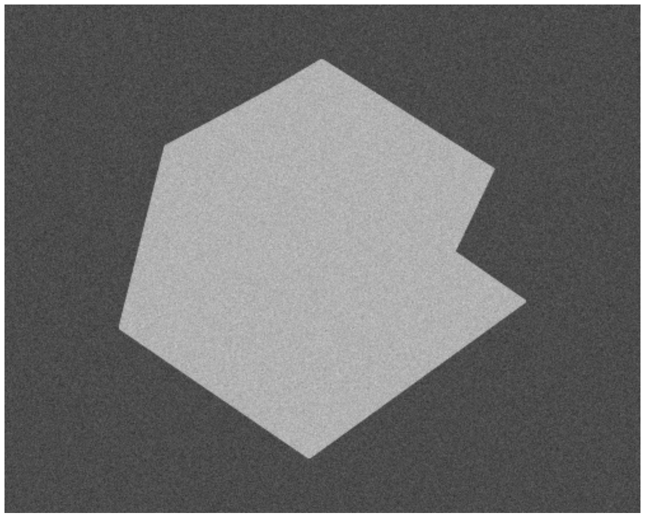


Edge map

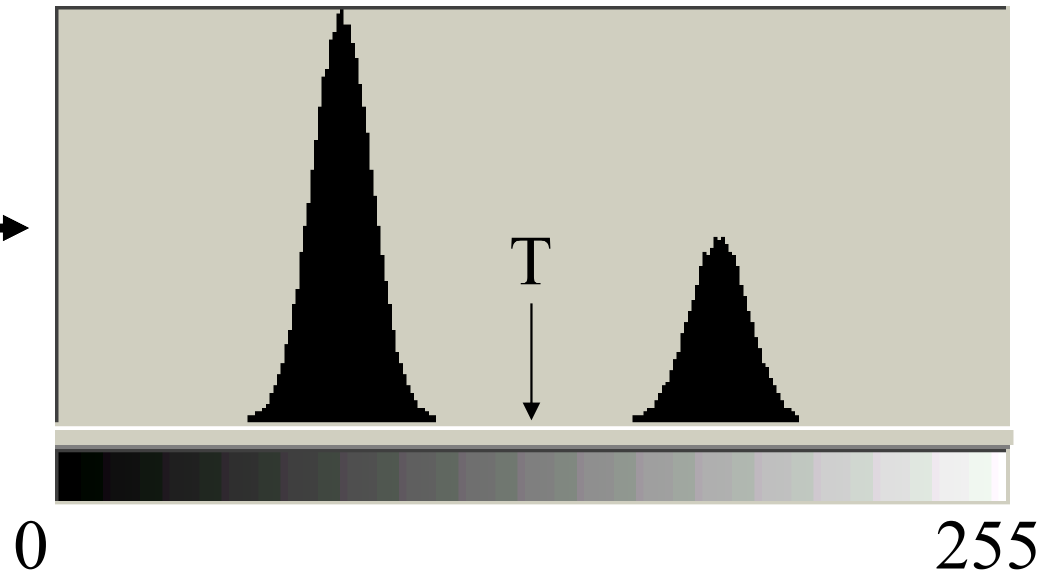
Object can be detected by using edge detection and linking methods

Global thresholding

2. Now, this handout will cover image segmentation by using threshold estimated based on pixel statistics.



Original image



Histogram of the image (bimodal)

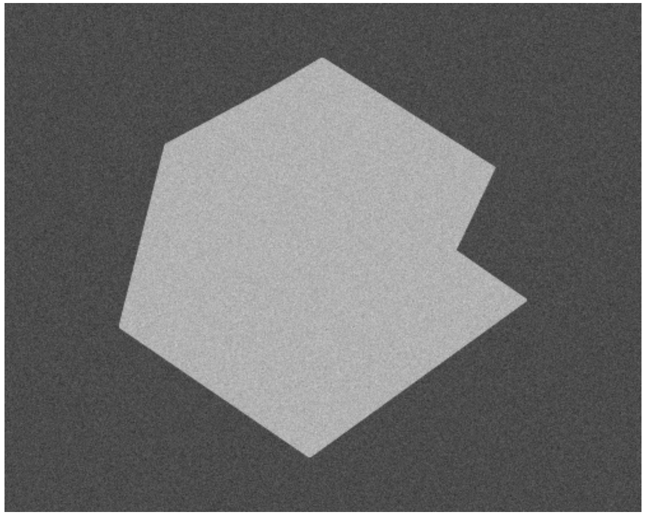
Global thresholding

1. Intensity histogram $h(i)$ = occurrences of pixel intensity value i. For example, $h(i=3) = 100$ means that there are 100 pixels having intensity value $i=3$.

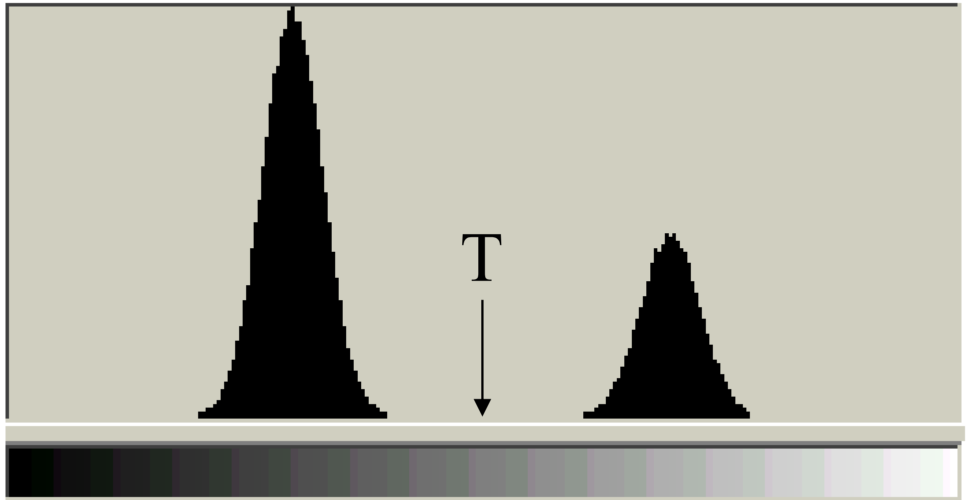
2. For an image $f(x,y)$ having light object and dark background, the histogram consists of two dominant modes (two groups on intensity values).

3. The object is extracted by selecting a threshold T that clearly separates these modes.

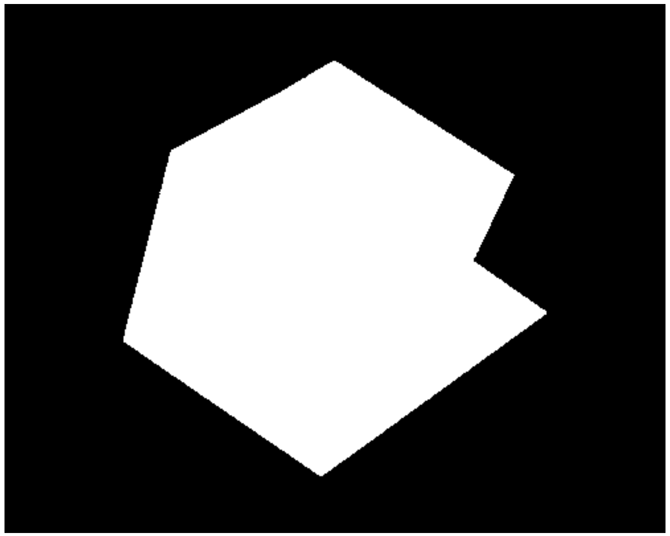
Global thresholding



Original image



Histogram of the image (bimodal)



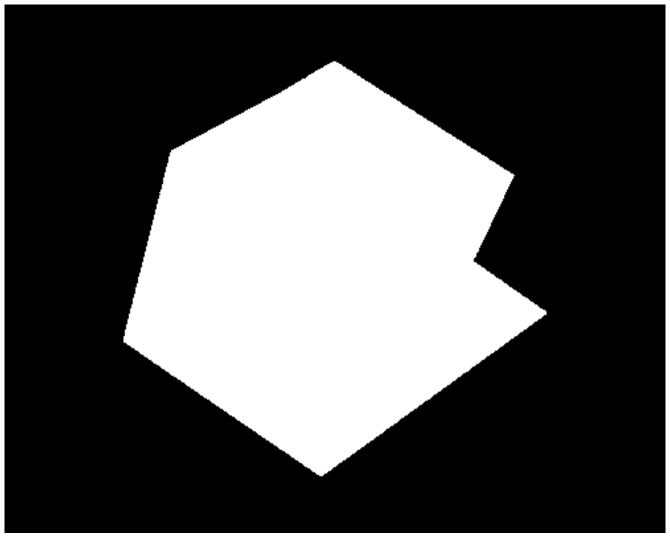
The thresholded (binary) image

Global thresholding

4. Each pixel is then labelled according to a pre-defined pixel labelling rule (2-class labelling), which is given by

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

5. If $T = 127$,

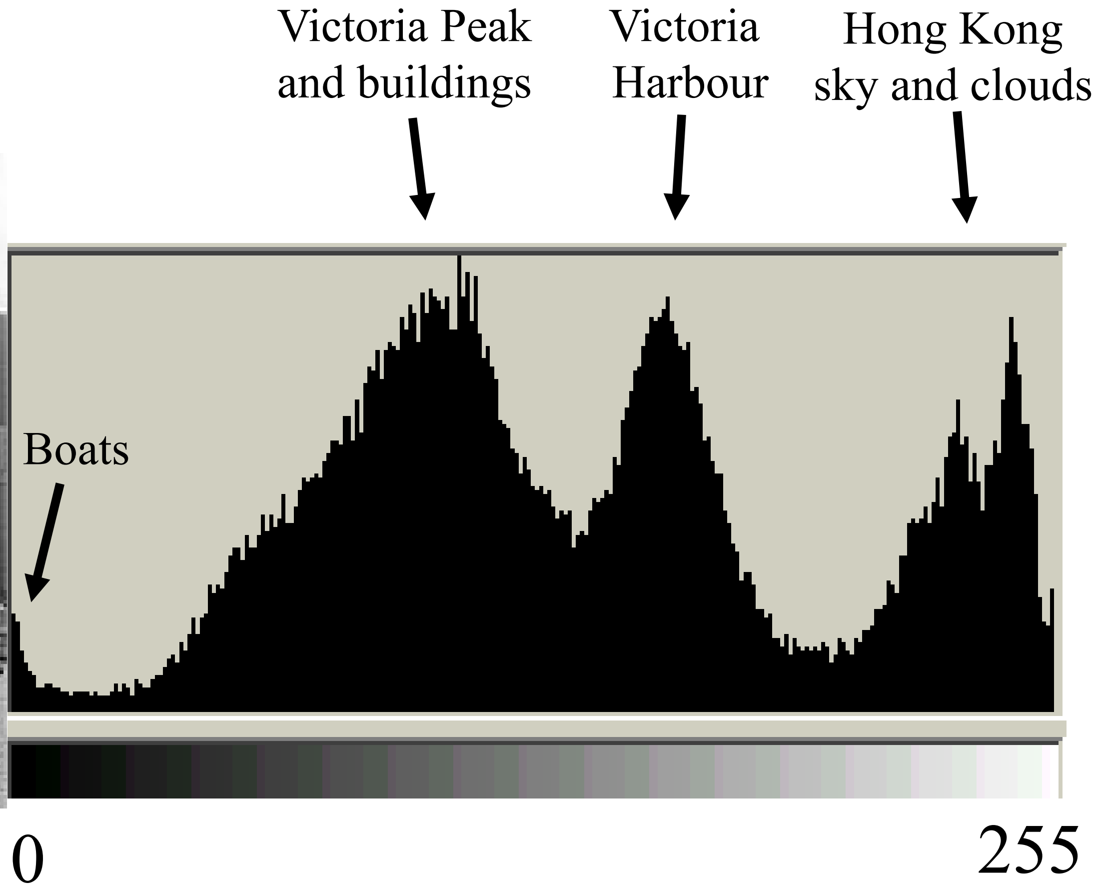


The thresholded (binary) image

Global thresholding

6. Multi-level thresholding (multi-class labelling)

Original image



Histogram of the image
(multi-modal)

Global thresholding

7. Global vs local/adaptive thresholding
    a. Global thresholding – threshold does not depend on image locations or local features/neighbouring pixel properties.

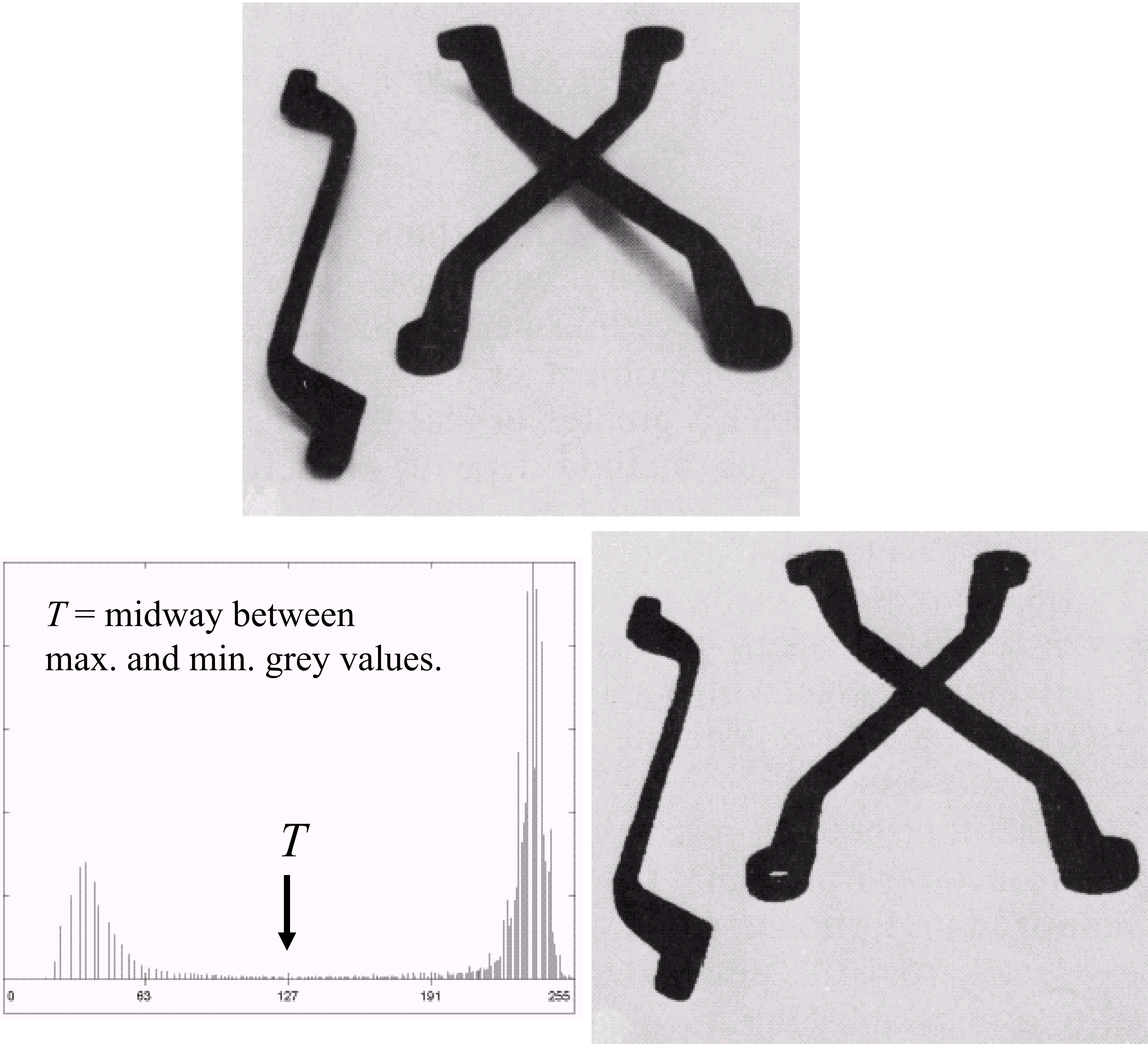
    b. Local thresholding – threshold depends on image locations or local features/neighbouring pixel properties.

Global Thresholding

1. Idea: partitioning the histogram into distinct regions by using a single, global threshold.

2. Segmentation is accomplished by scanning an image pixel by pixel and labelling each pixel as object or background according to a pixel labelling rule.

3. The pixel labelling rule assigns labels to each pixel. The label assignment depends on whether pixel grey level is greater or less than the value of T.



a
b c

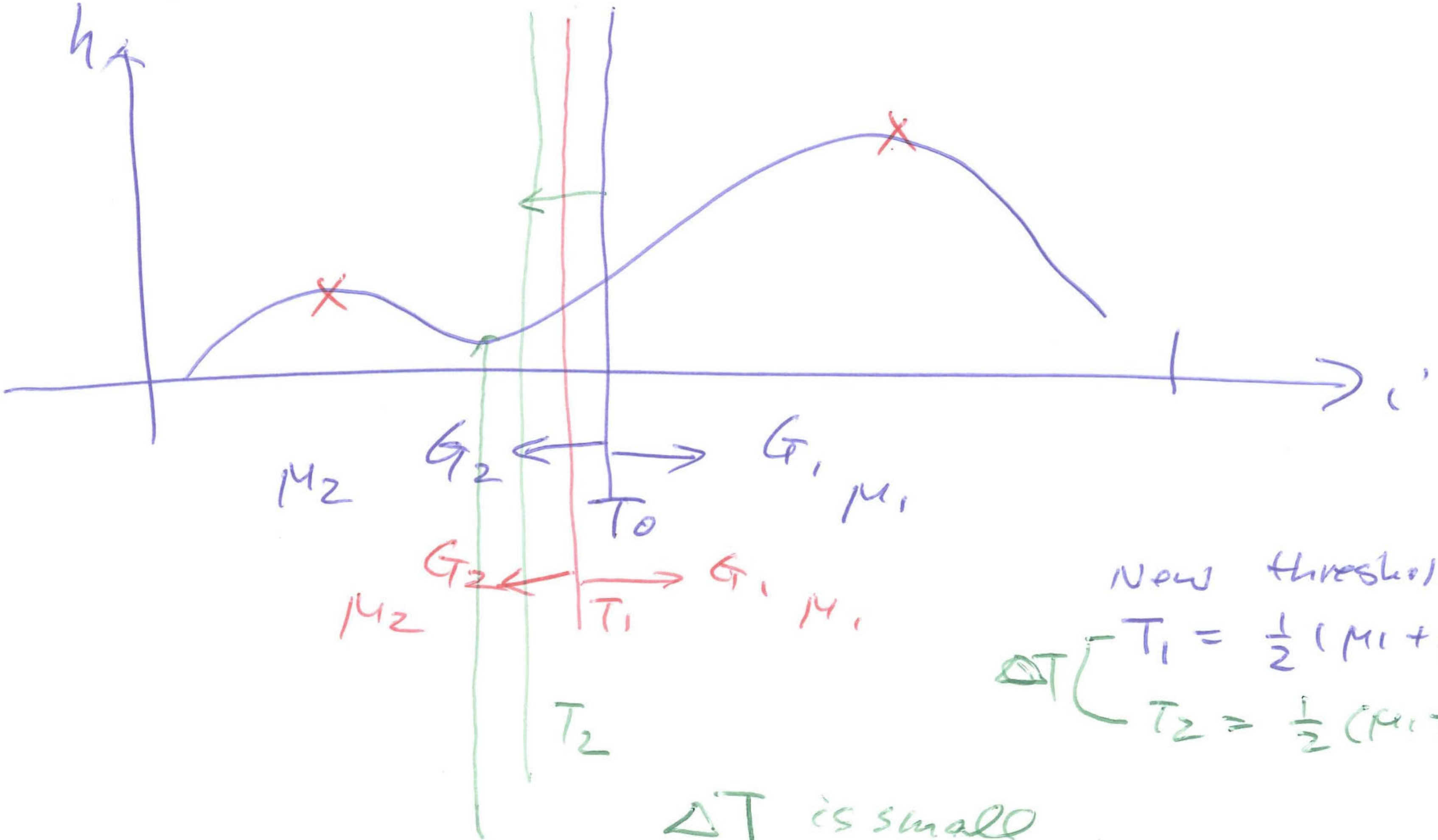
FIGURE 10.28 (a) Original image. (b) Image histogram. (c) Result of global thresholding with T midway between the maximum and minimum gray levels.

Global Thresholding

4. How to estimate the global threshold T?
5. T = midway between max. and min. grey levels.
6. T = midway between two peaks of distributions.
7. Method 1 (automatic, iterative approach).
   a. Select an initial estimate for T.
   b. Segment the image using T. It creates two regions G_1 (all pixel intensity values > T) and G_2 (otherwise).
   c. Compute the average grey level values μ_1 and μ_2 for the pixels in regions G_1 and G_2 respectively.
   d. Compute a new threshold value

$$T = \frac{1}{2}\left(\mu_1 + \mu_2\right)$$

   e. Repeat steps b through d until the change in T is small.



$T_{optimal}$

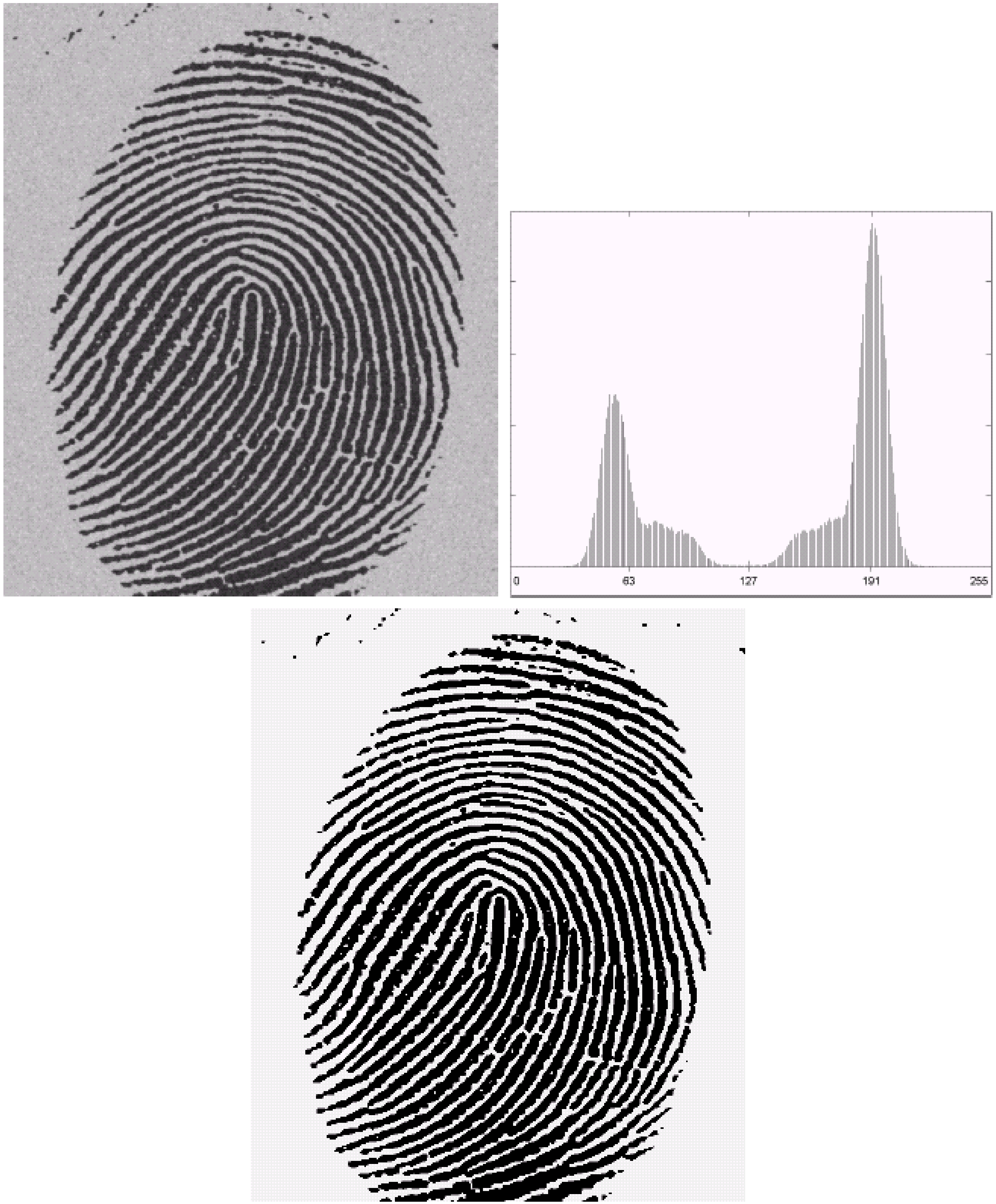
New threshold

$T_1 = \frac{1}{2}(\mu_1 + \mu_2)$

ΔT

$T_2 = \frac{1}{2}(\mu_1 + \mu_2)$

ΔT is small $\rightarrow$ terminate



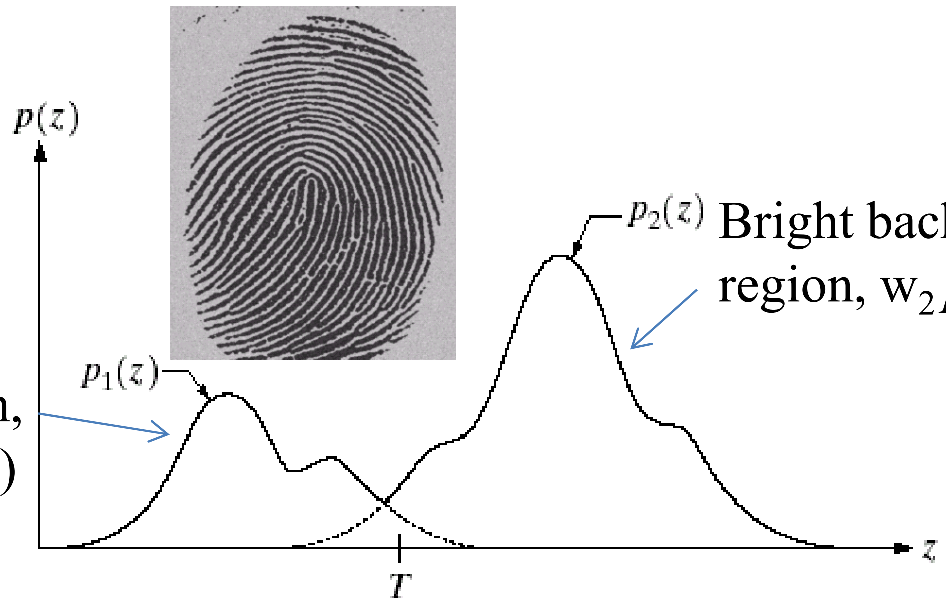
a b
c

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.)

Global Thresholding

8. Method 2 (automatic, statistical, minimum error approach).
   a. We assume that the image histogram is a mixture of two distributions.

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



   b. $p(z|1)$ and $p(z|2)$ can be the Gaussian distributions, Rayleigh or others.
   c. The mixture model is defined as

$$p(z) = w_1 p(z|1) + w_2 p(z|2)$$

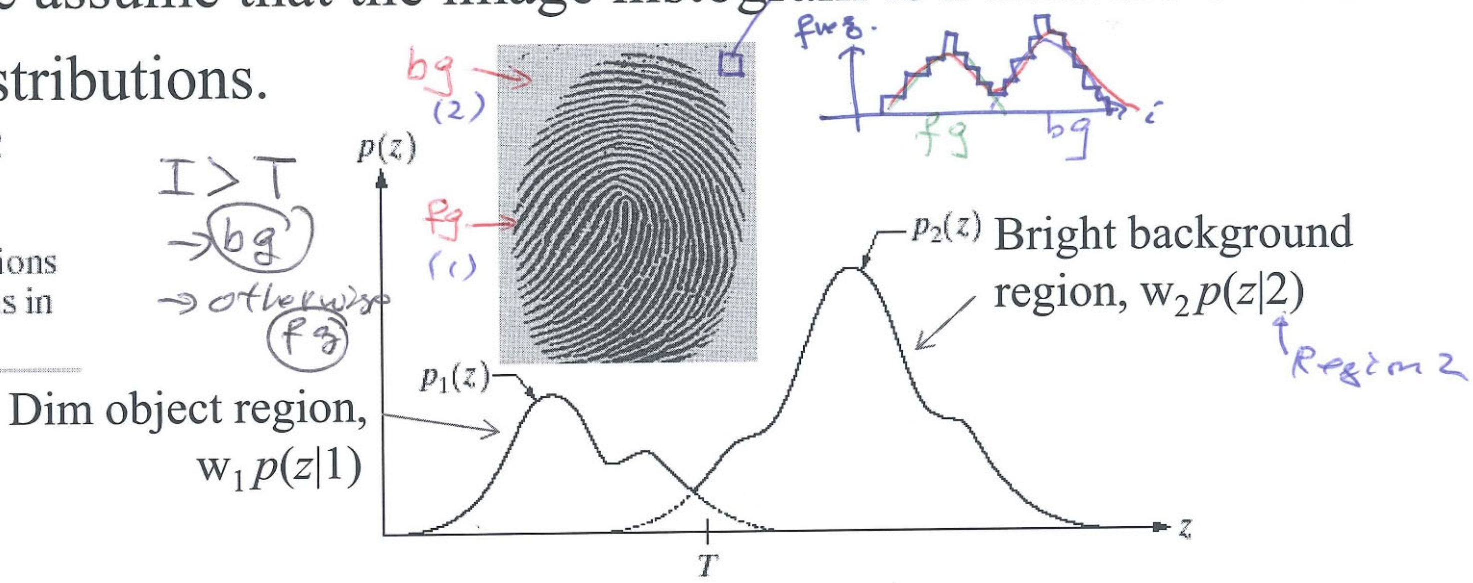
where $w_1 + w_2 = 1$, and $0 \le w_1, w_2 \le 1$

Global Thresholding

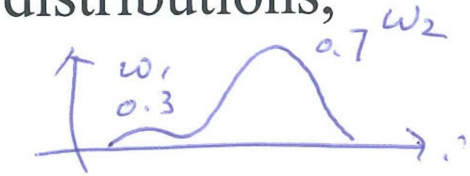
8. Method 2 (automatic, statistical, minimum error approach).

   a. We assume that the image histogram is a mixture of two distributions.

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



   b. $p(z|1)$ and $p(z|2)$ can be the Gaussian distributions, Rayleigh or others.



   c. The mixture model is defined as

$$p(z) = \overset{0.5}{w_1} p(z|1) + \overset{0.5}{w_2} p(z|2)$$

where $w_1 + w_2 = 1$, and $0 \le w_1, w_2 \le 1$

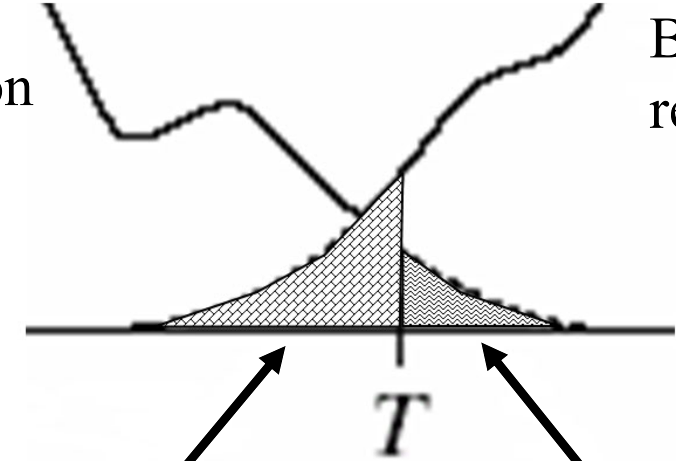
Global Thresholding

8. Method 2 (automatic, statistical, minimum error approach).

   d. Error is defined as

Object region $p(z|1)$

Background region, $p(z|2)$



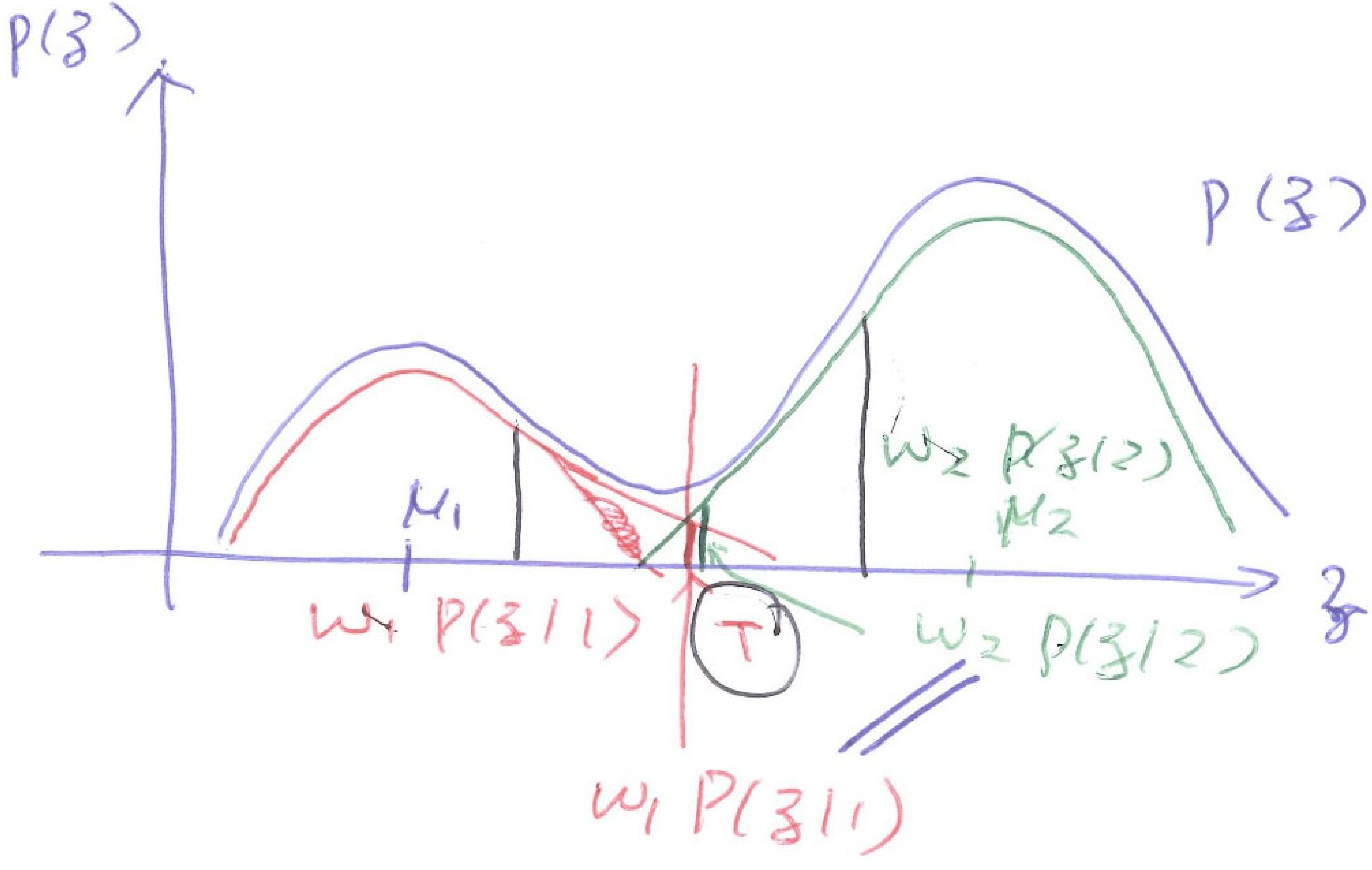
$$E(T) = \int_{-\infty}^{T} w_2 p(z|2)\,dz + \int_{T}^{\infty} w_1 p(z|1)\,dz$$

Misclassifying background pixels into object pixels.

Misclassifying object pixels into background pixels.

   e. By minimising $E(T)$ with respect to T, we get

> when $w_1 p(T|1) = w_2 p(T|2)$
> $E(T)$ is minimum.



p(z)
p(z)
w2 p(z|2)
μ1
μ2
z
w1 p(z|1)
T
w2 p(z|2)
w1 p(z|1)

Basic Global Thresholding

8. Method 2 (automatic, statistical, minimum error approach).

   f. If $p(z|1)$ and $p(z|2)$ are Gaussian distributed,

$$p(z) = \frac{w_1}{\sqrt{2\pi\sigma}} e^{\frac{-(z-\mu_1)^2}{2\sigma^2}} + \frac{w_2}{\sqrt{2\pi\sigma}} e^{\frac{-(z-\mu_2)^2}{2\sigma^2}}$$

   then the minimum error threshold is given by

$$T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2} \ln\left(\frac{w_2}{w_1}\right)$$

Given the observed histogram and statistical mixture model, how to estimate the parameters (w_1, w_2, means and SDs)?

Answer: Expectation-Maximization (EM) method can be used.

https://en.wikipedia.org/wiki/Expectation%E2%80%93maximization_algorithm

Statistical mixture models

1. The probability density function (PDF) is formed from a linear combination of M basis functions (e.g. Gaussian distribution).
2. Mixture distribution (or PDF) is given by

$$p(z) = \sum_{j=1}^{M} p(z \mid j)\, p(j)$$

where M represents the number of basis functions

$p(z \mid j)$ represents basis function

$p(j) = w_j$ represents mixing parameter

Interesting Links:
https://www.ll.mit.edu/mission/cybersec/publications/publication-files/full_papers/0802_Reynolds_Biometrics-GMM.pdf
http://www.stat.cmu.edu/~cshalizi/uADA/12/lectures/ch20.pdf
Or, search "Gaussian Mixture Model" in the Internet for more information.

Statistical mixture models

3. Constraints

(1) $$\sum_{j=1}^{M} p(j) = 1$$

(2) $$0 \le p(j) \le 1$$

(3) $$\int p(z \mid j)\, dx = 1$$

e.g., $$p(z \mid j) = \frac{1}{\sqrt{2\pi\sigma_j^2}} \cdot \exp\left[\frac{-(z-\mu_j)^2}{2\sigma_j^2}\right]$$

Gaussian distribution

Statistical mixture models

4. Posterior probability

$$p(j \mid z) = \frac{p(z \mid j)\, p(j)}{p(z)}$$

$$\sum_{j=1}^{M} p(j \mid z) = 1$$

Expectation-Maximisation (EM) method

1. The EM method is an iterative scheme for finding the values of the parameters in a mixture model. The concept is to maximize the expected negative log-likelihood function $E = -\ln L$ of an observed image.

Initial values of μ_j, σ_j and $p(j)$

Estimate the expected $p^{old}(j \mid z)$

Estimate the values of μ_j^{new}, σ_j^{new} and $p^{new}(j)$ based on the principle of maximum likelihood

Expectation-Maximisation (EM) method

2. Update equations are shown below. Let N be the number of pixels in an image. As such, pixel index $n = 1, \ldots, N$. z^n represents pixel intensity at the n^{th} pixel.
3. Iteration continues until convergence is reached.

$$\mu_j^{new} = \frac{\sum_n p^{old}\left(j \mid z^n\right) z^n}{\sum_n p^{old}\left(j \mid z^n\right)}$$

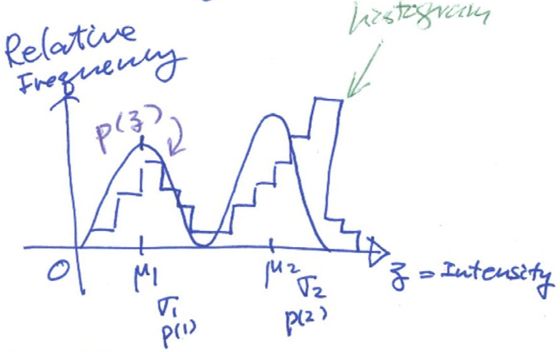
$$(\sigma_j^{new})^2 = \frac{\sum_n p^{old}\left(j \mid z^n\right)\left(z^n - \mu_j^{new}\right)^2}{\sum_n p^{old}\left(j \mid z^n\right)}$$

$$p\left(j\right)^{new} = \frac{1}{N} \sum_n p^{old}\left(j \mid z^n\right)$$

E.g.

z^1	z^2	z^3
z^4	z^5	z^6
z^7	z^8	z^9

$\boxed{\text{Example}}$



Iteration 0 (2 classes)

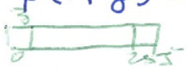
$$\boxed{\mu_1,\ \sigma_1,\ p(1),\ \mu_2,\ \sigma_2,\ p(2)}$$

Iteration 1

E step:

posterior, likelihood, prior

$$p(1|z) = \frac{p(z|1)\,p(1)}{p(z)}$$



where GMM

$$p(z) = p(z|1)p(1) + p(z|2)p(2)$$

$$p(2|z) = \frac{p(z|2)\,p(2)}{p(z)}$$

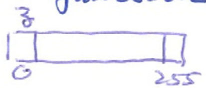
mixing parameter, basis function

where

$$p(z) = p(1)\,p(z|1) + p(2)\,p(z|2)$$

likelihood or basis fct

$$p(z|1) = \frac{1}{\sqrt{2\pi}\,\sigma_1^2}\, e^{-\frac{(z-\mu_1)^2}{2\sigma_1^2}}$$



$$p(z|2) = \frac{1}{\sqrt{2\pi}\,\sigma_2^2}\, e^{-\frac{(z-\mu_2)^2}{2\sigma_2^2}}$$

M step:

posterior probability

$$\mu_1^{new} = \frac{\sum_{n=1}^{9} p(1|z^n)\cdot z^n}{\sum_{m=1}^{9} p(1|z^m)},$$

z^n = intensity value at nth pixel

$$= \frac{p(1|z^1)z^1 + p(1|z^2)z^2 + \cdots + p(1|z^9)z^9}{p(1|z^1) + p(1|z^2) + \cdots + p(1|z^9)}$$

$$(\sigma_1^{new})^2 = \frac{\sum_{n=1}^{9} p(1|z^n)\,(z^n - \mu_1^{new})^2}{\sum_{m=1}^{9} p(1|z^m)}$$

$$p(1)^{new} = \frac{1}{9} \sum_{n=1}^{9} p(1 \mid z^n)$$

Find μ_1^{new}, $\cancel{p}(\sigma_1^{new})^2$, $p(1)^{new}$

μ_2^{new}, $(\sigma_2^{new})^2$, $p(2)^{new}$

---

update all parameters

$\mu_1 = \mu_1^{new}$

$\sigma_1 = \sigma_1^{new}$

$p(1) = p(1)^{new}$

$\mu_2 = \mu_2^{new}$

$\sigma_2 = \sigma_2^{new}$

$p(2) = p(2)^{new}$

<u>iteration 2</u>

E step : find $p(1 \mid z)$ & $p(2 \mid z)$

M step : find μ_1^{new}, $(\sigma_1^{new})^2$, $p(1)^{new}$

μ_2^{new}, $(\sigma_2^{new})^2$, $p(2)^{new}$

⋮

Stops when

$\Delta\mu_1$, $\Delta\sigma_1$, $\Delta p(1)$

$\Delta\mu_2$, $\Delta\sigma_2$, $\Delta p(2)$

are significantly small.