DIGITAL IMAGE PROCESSING

IMAGE SEGMENTATION

Introduction

- Image segmentation divides an image into regions that are connected and have some similarity within the region and some difference between adjacent regions.

- The goal is usually to find individual objects in an image.

- For the most part there are fundamentally two kinds of approaches to segmentation: discontinuity and similarity.
  - Similarity may be due to pixel intensity, color or texture.
  - Differences are sudden changes (discontinuities) in any of these, but especially sudden changes in intensity along a boundary line, which is called an edge.

Detection of Discontinuities

- There are three kinds of discontinuities of intensity: points, lines and edges.
- The most common way to look for discontinuities is to scan a small mask over the image. The mask determines which kind of discontinuity to look for.

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

FIGURE 10.1 A general 3 × 3 mask.

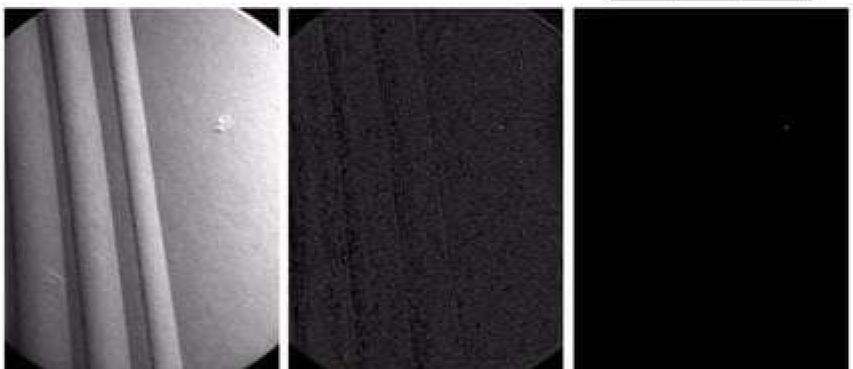
w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Detection of Discontinuities
Point Detection

$$|R| \geq T$$

where T : a nonnegative threshold

−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.)

Detection of Discontinuities
Line Detection

- Only slightly more common than point detection is to find a one pixel wide line in an image.
- For digital images the only three point straight lines are only horizontal, vertical, or diagonal (+ or –45°).

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

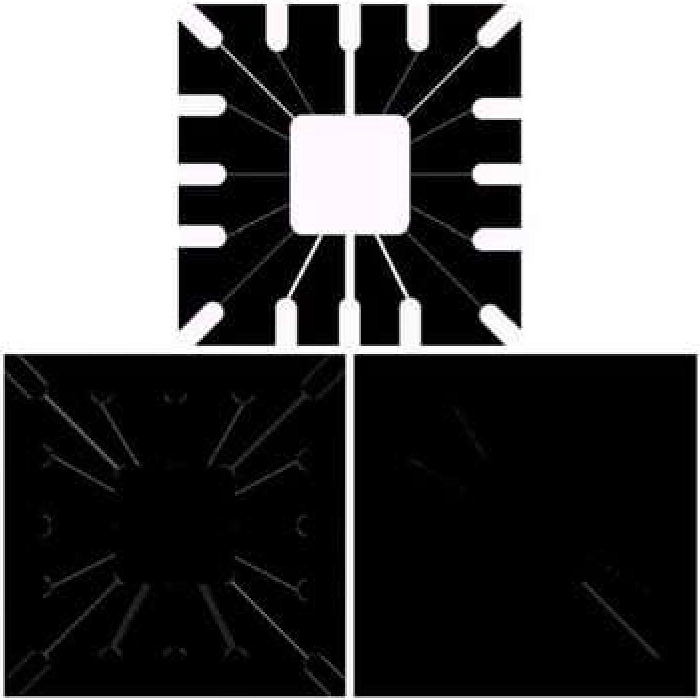
Vertical

−1	2	−1
−1	2	−1
−1	2	−1

−45°

2	−1	−1
−1	2	−1
−1	−1	2

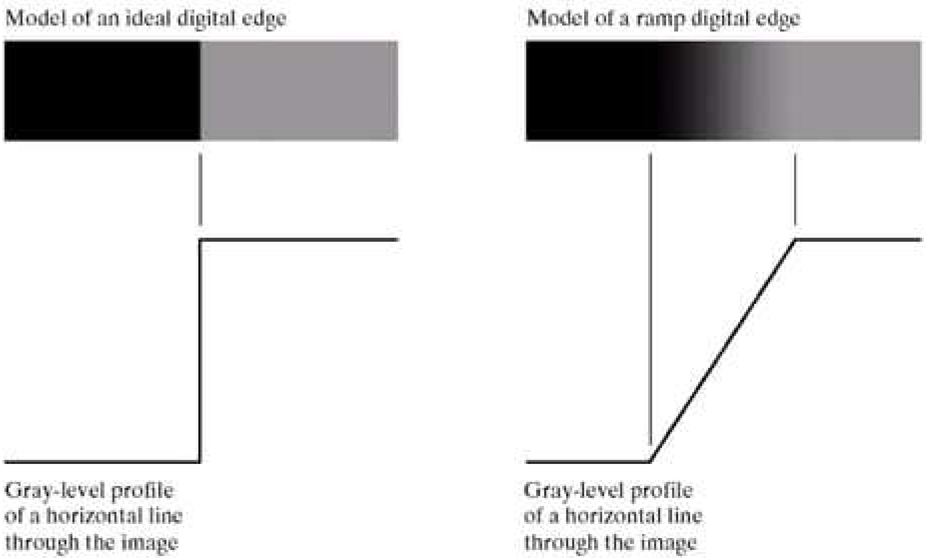
Detection of Discontinuities
Line Detection



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).

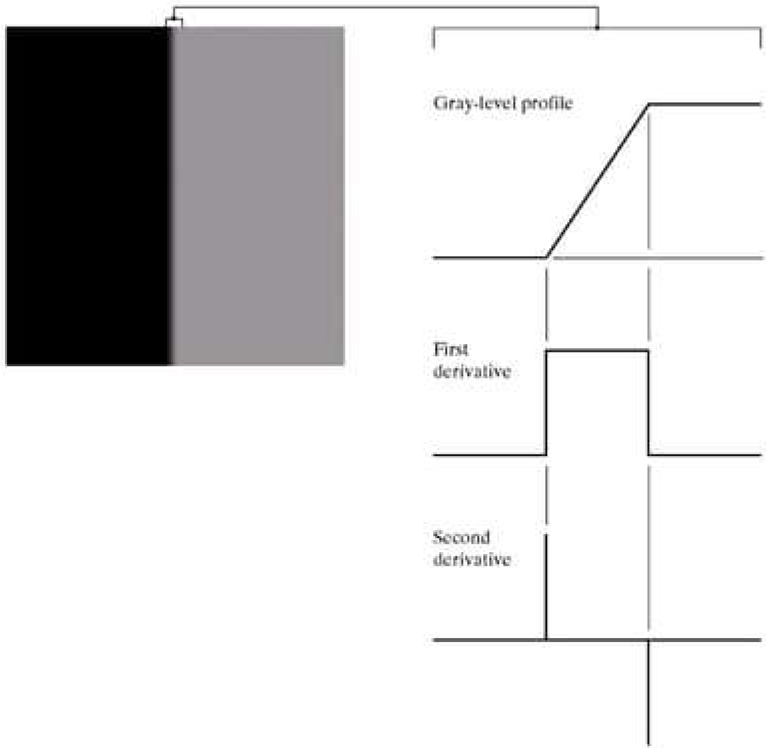
Detection of Discontinuities
Edge Detection



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.

Detection of Discontinuities
Edge Detection



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.

Detection of Discontinuities
Edge Detection

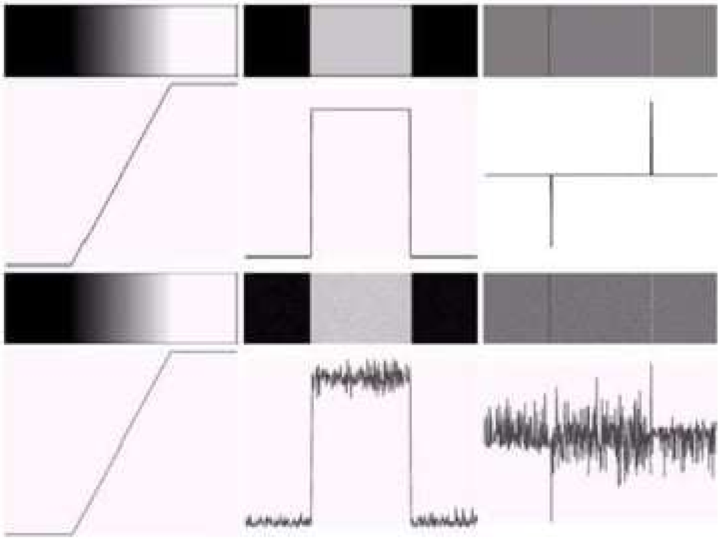


FIGURE 10.7 First column: images and gray-level profiles of a ramp edge corrupted by random Gaussian noise of mean 0 and σ = 0.0, 0.1, 1.0, and 10.0, respectively. Second column: first-derivative images and gray-level profiles. Third column: second-derivative images and gray-level profiles.

a
b
c
d

Detection of Discontinuities
Edge Detection

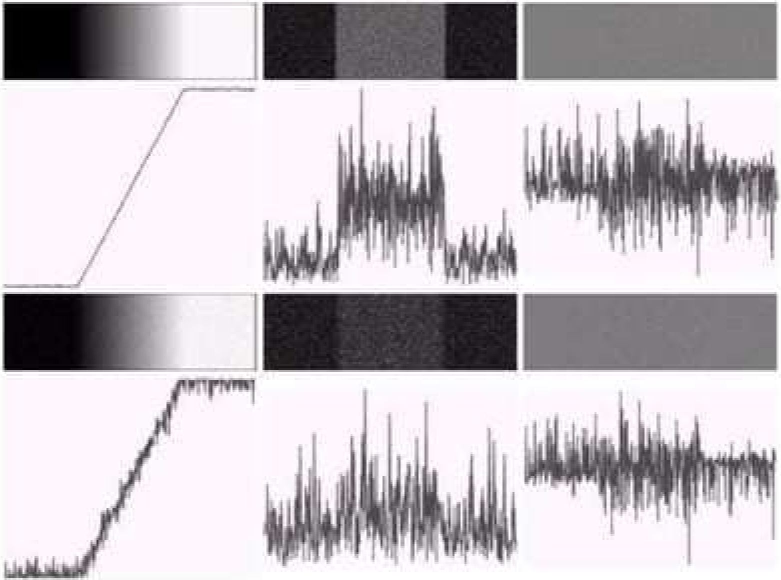


FIGURE 10.7 First column: images and gray-level profiles of a ramp edge corrupted by random Gaussian noise of mean 0 and σ = 0.0, 0.1, 1.0, and 10.0, respectively. Second column: first-derivative images and gray-level profiles. Third column: second-derivative images and gray-level profiles.

a
b
c
d

Detection of Discontinuities
Gradient Operators

- First-order derivatives:
  - The gradient of an image $f(x,y)$ at location (x,y) is defined as the vector:

$$\nabla \mathbf{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

  - The magnitude of this vector: $\nabla f = \mathrm{mag}(\nabla \mathbf{f}) = \left[G_x^2 + G_y^2\right]^{1/2}$
  - The direction of this vector: $\alpha(x,y) = \tan^{-1}\left(\frac{G_y}{G_x}\right)$

Detection of Discontinuities
Gradient Operators

Roberts cross-gradient operators

−1	0	0	−1
0	1	1	0

Roberts

Prewitt operators

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

Prewitt

Sobel operators

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

Sobel

Detection of Discontinuities
Gradient Operators

Prewitt masks for detecting diagonal edges

0	1	1
−1	0	1
−1	−1	0

−1	−1	0
−1	0	1
0	1	1

Prewitt

Sobel masks for detecting diagonal edges

0	1	2
−1	0	1
−2	−1	0

−2	−1	0
−1	0	1
0	1	2

Sobel

a b
c d

FIGURE 10.9 Prewitt and Sobel masks for detecting diagonal edges.

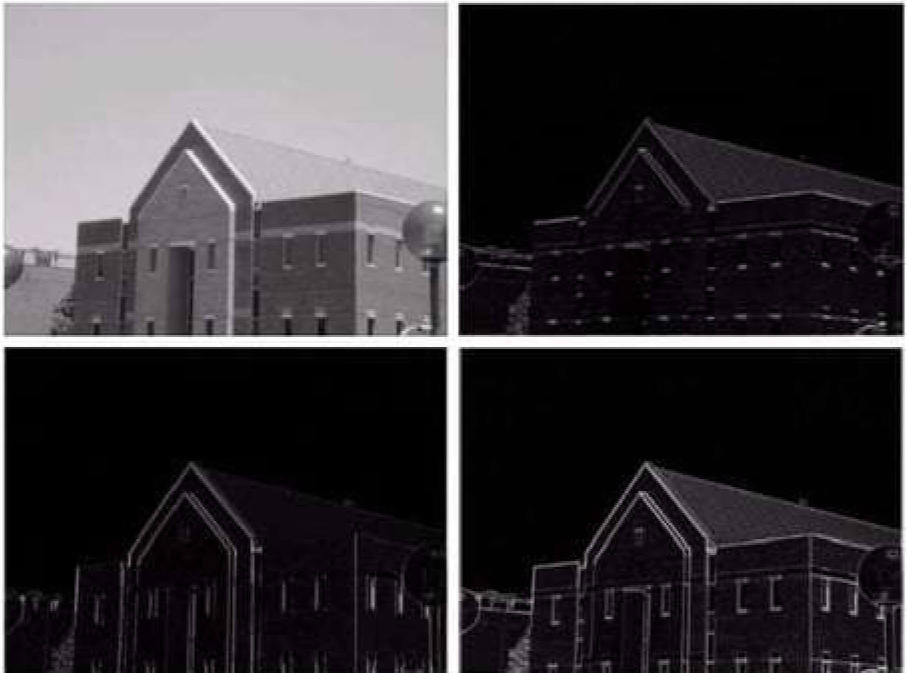
Detection of Discontinuities
Gradient Operators: Example

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, $|G_x| + |G_y|$.

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

Detection of Discontinuities
Gradient Operators: Example



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.

Detection of Discontinuities
Gradient Operators: Example

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).

0	1	2
−1	0	1
−2	−1	0

−2	−1	0
−1	0	1
0	1	2

Detection of Discontinuities
Gradient Operators

- Second-order derivatives: (The Laplacian)
  - The Laplacian of an 2D function $f(x,y)$ is defined as

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

  - Two forms in practice:

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

Detection of Discontinuities
Gradient Operators

- Consider the function:

A Gaussian function

$$h(r) = -e^{-\frac{r^2}{2\sigma^2}} \quad \text{where } r^2 = x^2 + y^2$$

$$\text{and } \sigma : \text{the standard deviation}$$

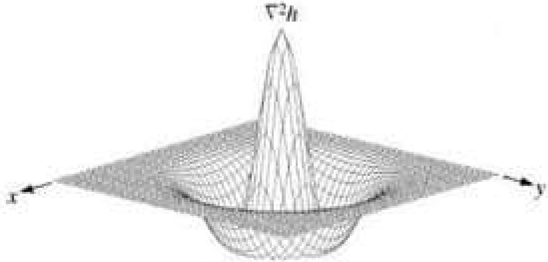
- The Laplacian of h is

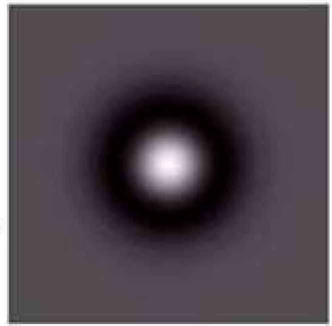
$$\nabla^2 h(r) = -\left[\frac{r^2 - \sigma^2}{\sigma^4}\right] e^{-\frac{r^2}{2\sigma^2}}$$

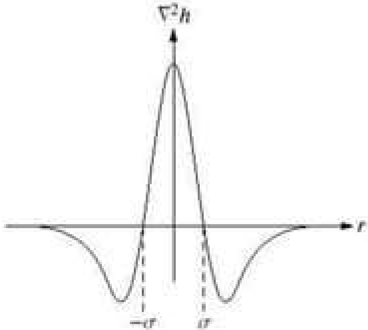
The Laplacian of a Gaussian (LoG)

- The Laplacian of a Gaussian sometimes is called the Mexican hat function. It also can be computed by smoothing the image with the Gaussian smoothing mask, followed by application of the Laplacian mask.

Detection of Discontinuities
Gradient Operators





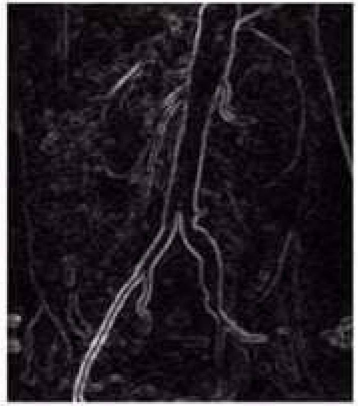


0	0	−1	0	0
0	−1	−2	−1	0
−1	−2	16	−2	−1
0	−1	−2	−1	0
0	0	−1	0	0

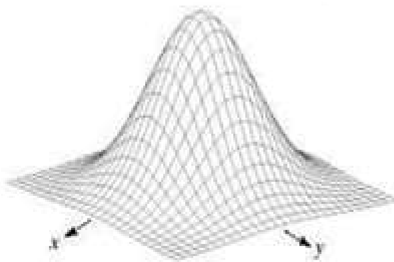
a b
c d

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).

Detection of Discontinuities Gradient Operators: Example



Sobel gradient



Gaussian smooth function

−1	−1	−1
−1	8	−1
−1	−1	−1

Laplacian mask

Detection of Discontinuities
Gradient Operators: Example

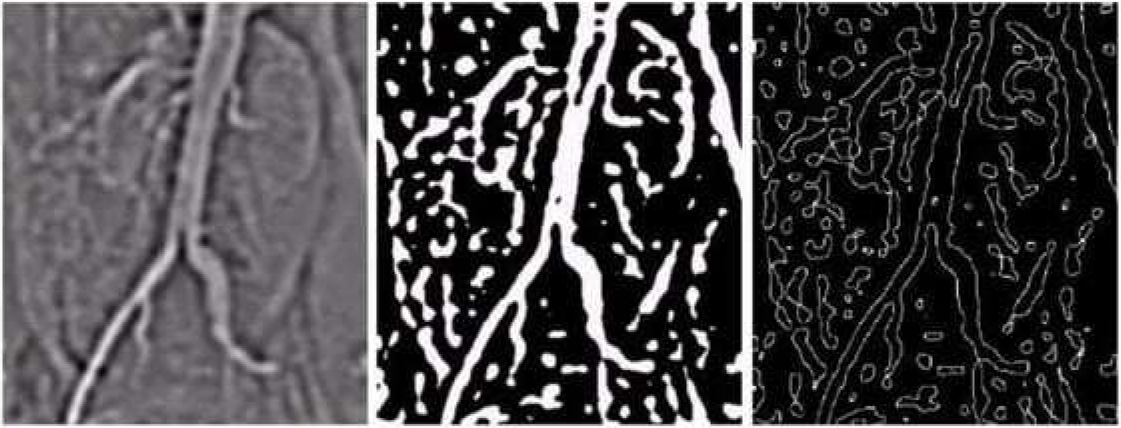


FIGURE 10.15 (a) Original image. (b) Sobel gradient (shown for comparison). (c) Spatial Gaussian smoothing function. (d) Laplacian mask. (e) LoG. (f) Thresholded LoG. (g) Zero crossings. (Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University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

Edge Linking and Boundary Detection
Local Processing

- Two properties of edge points are useful for edge linking:
  - the strength (or magnitude) of the detected edge points
  - their directions (determined from gradient directions)
- This is usually done in local neighborhoods.
- Adjacent edge points with similar magnitude and direction are linked.
- For example, an edge pixel with coordinates (x_0,y_0) in a predefined neighborhood of (x,y) is similar to the pixel at (x,y) if

$$|\nabla f(x,y)-\nabla(x_0,y_0)|\le E,\quad E:\text{a nonnegative threshold}$$

$$|\alpha(x,y)-\alpha(x_0,y_0)|< A,\quad A:\text{a nonegative angle threshold}$$

Edge Linking and Boundary Detection Local Processing: Algorithm

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

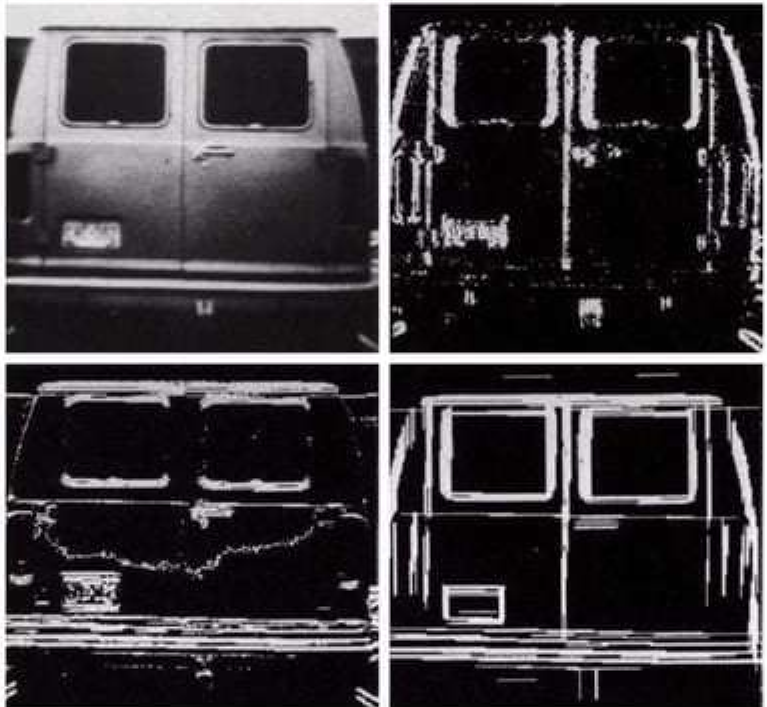
$$g(x,y)=\begin{cases}1, & M(x,y)>T_M \text{ AND } \alpha(x,y)=A\pm T_A \\ 0, & \text{Otherwise}\end{cases}$$

   where T_M is a threshold, A is a specific angle direction, and $\pm T_A$ defines a "band" of acceptable direction about A.
3. Scan the rows of g and fill all gaps in each row that do not exceed a specified length, K.
4. To detect gaps in other direction, θ, rotate g by this angle and apply the horizontal scanning procedure in step 3. Rotate the result back by −θ.

Edge Linking and Boundary Detection
Local Processing: Example

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.)



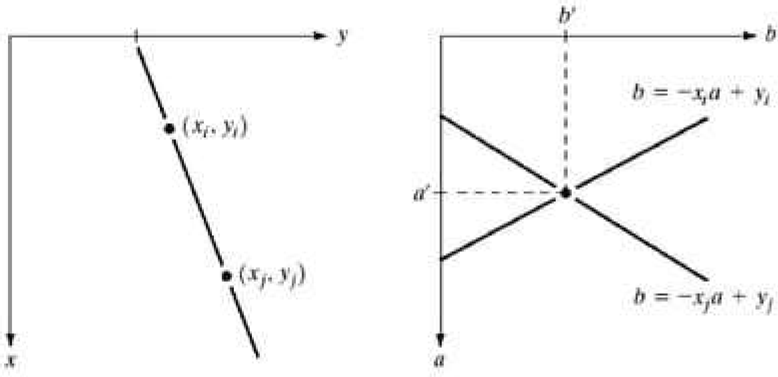
In this example, we can find the license plate candidate after edge linking process.

Edge Linking and Boundary Detection
Global Processing via the Hough Transform

Edge Linking and Boundary Detection
Global Processing via the Hough Transform

- Hough transform: a way of finding edge points in an image that lie along a straight line.
- Example: *xy*-plane v.s. *ab*-plane (parameter space)

$$y_i = ax_i + b$$

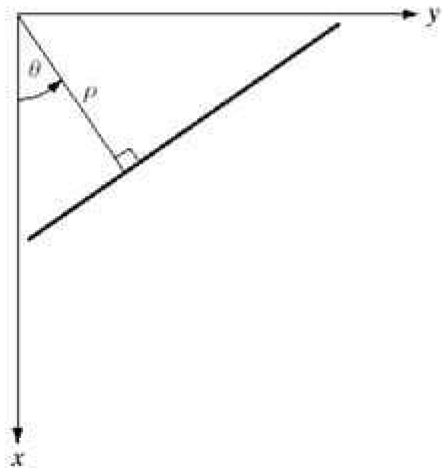


a b

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

Edge Linking and Boundary Detection
Global Processing via the Hough Transform

- The Hough transform consists of finding all pairs of values of θ and ρ which satisfy the equations that pass through (x,y).
- These are accumulated in what is basically a 2-dimensional histogram.
- When plotted these pairs of θ and ρ will look like a sine wave. The process is repeated for all appropriate (x,y) locations.



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

Edge Linking and Boundary Detection
Hough Transform Example

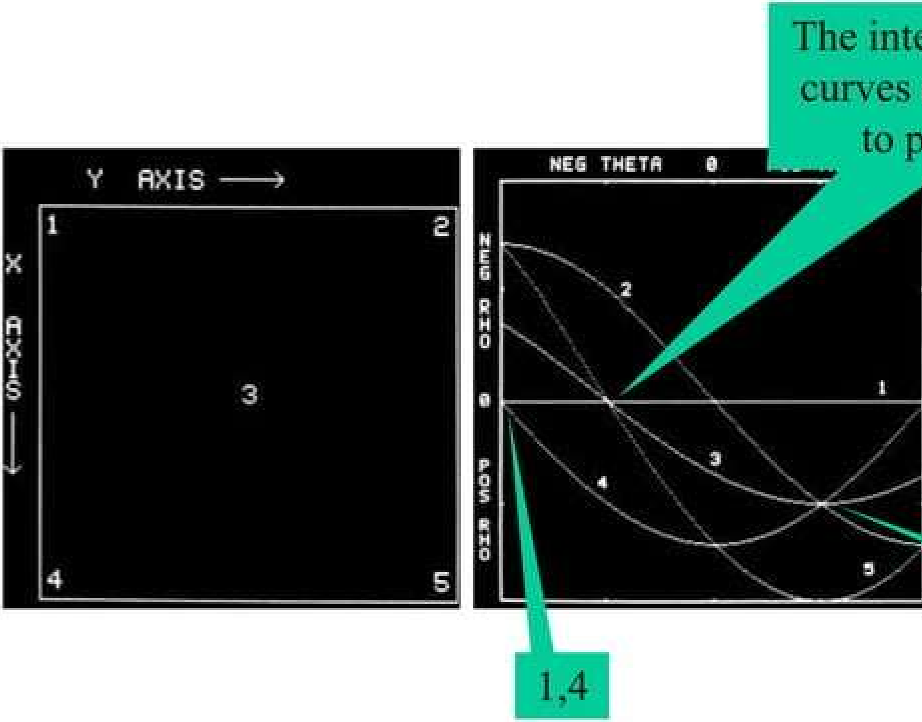
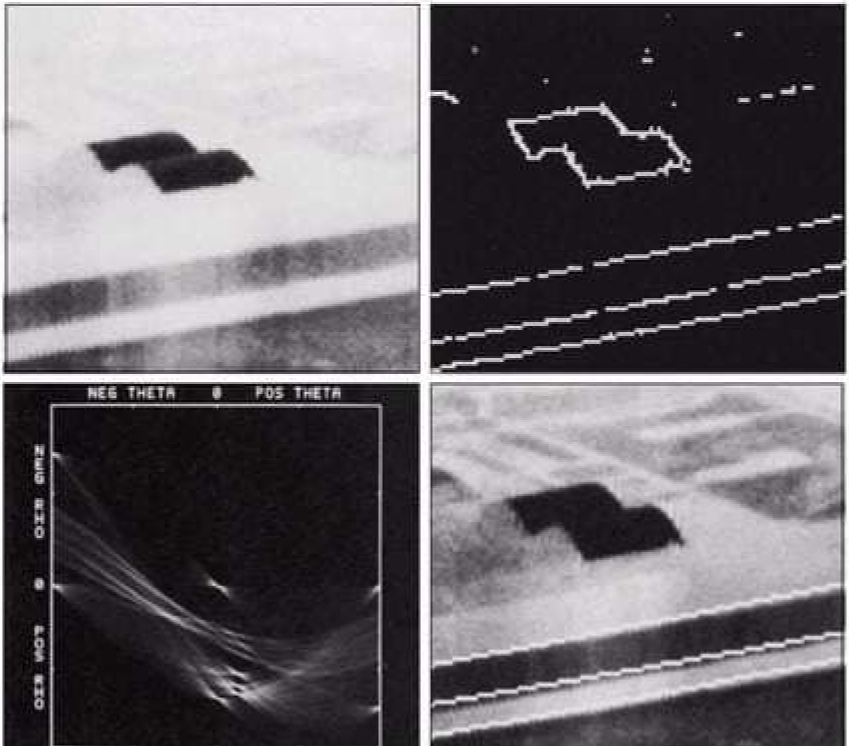


FIGURE 10.20 Illustration of the Hough transform. (Courtesy of Mr. D. R. Cate, Texas Instruments, Inc.)

Edge Linking and Boundary Detection
Hough Transform Example



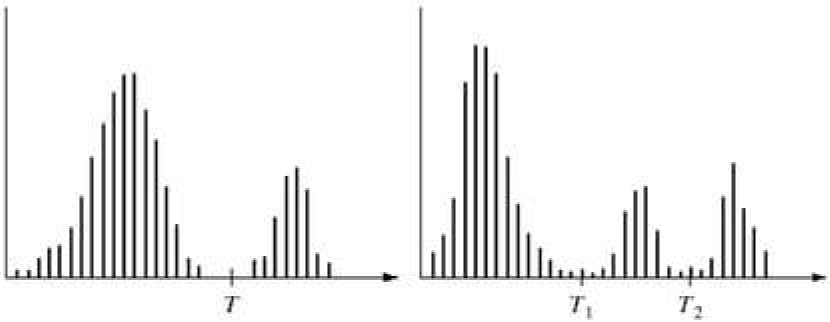
a b
c d

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.)

Thresholding

- Assumption: the range of intensity levels covered by objects of interest is different from the background.

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



a b Single threshold Multiple threshold

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

Thresholding
The Role of Illumination

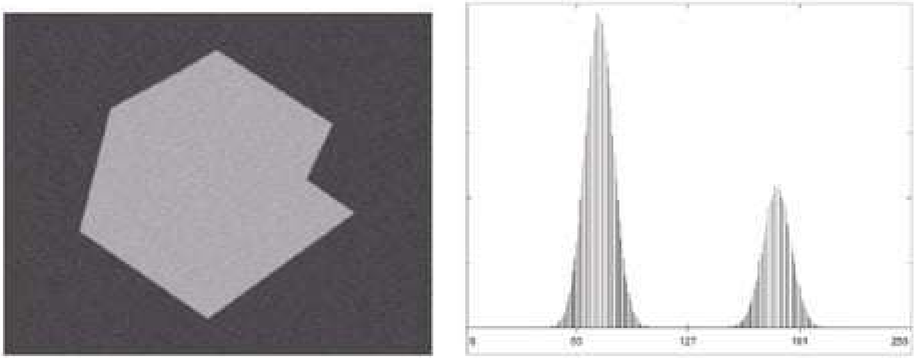
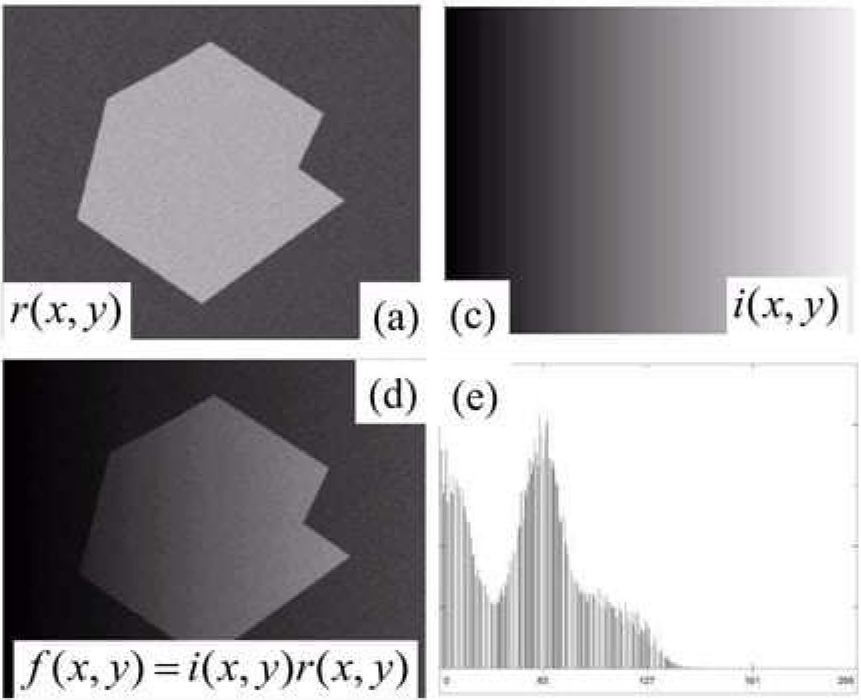


FIGURE 10.27
(a) Computer generated reflectance function.
(b) Histogram of reflectance function.

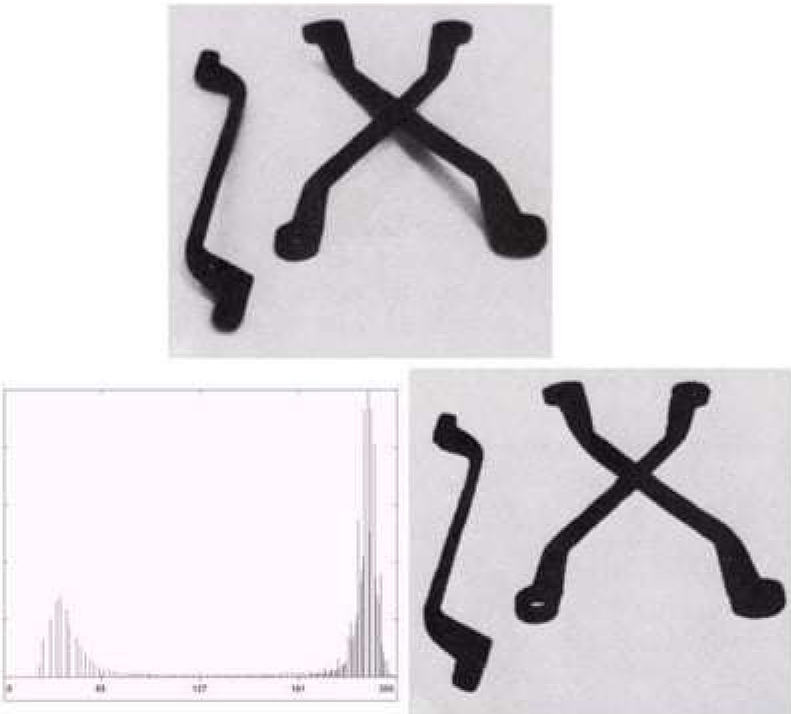
Thresholding
The Role of Illumination



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.

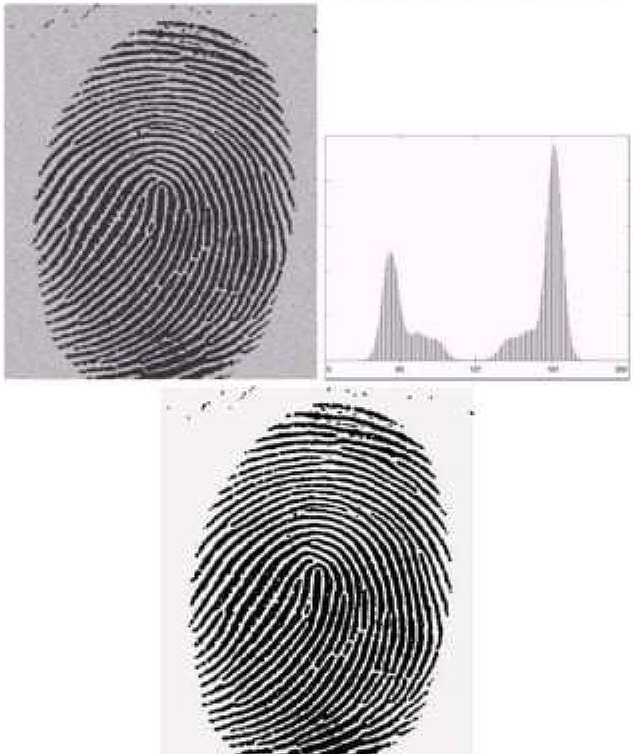
Thresholding
Basic Global Thresholding



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.

Thresholding
Basic Global 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.)

Thresholding
Basic Adaptive Thresholding

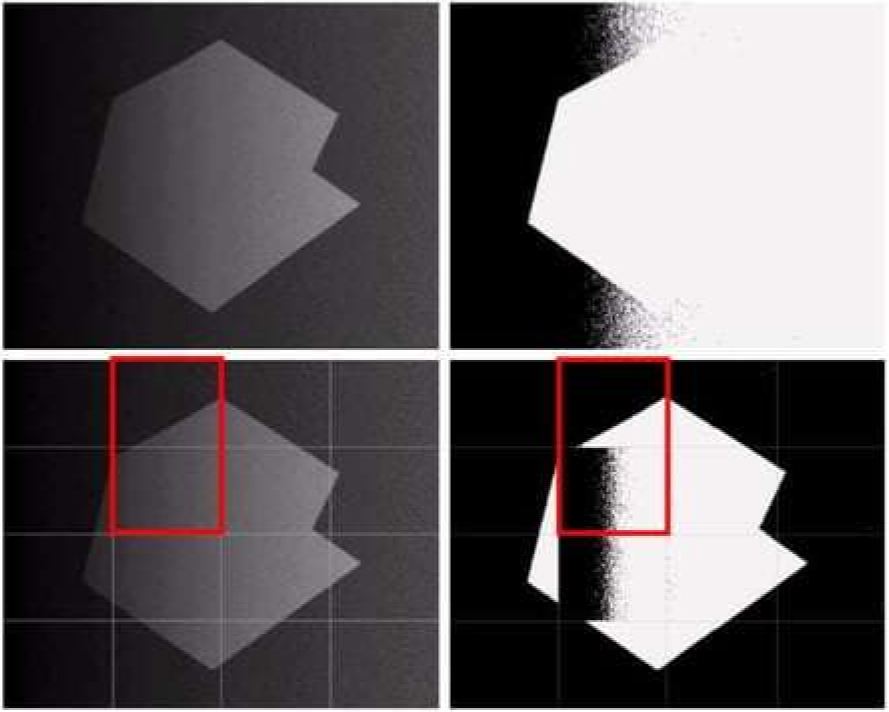
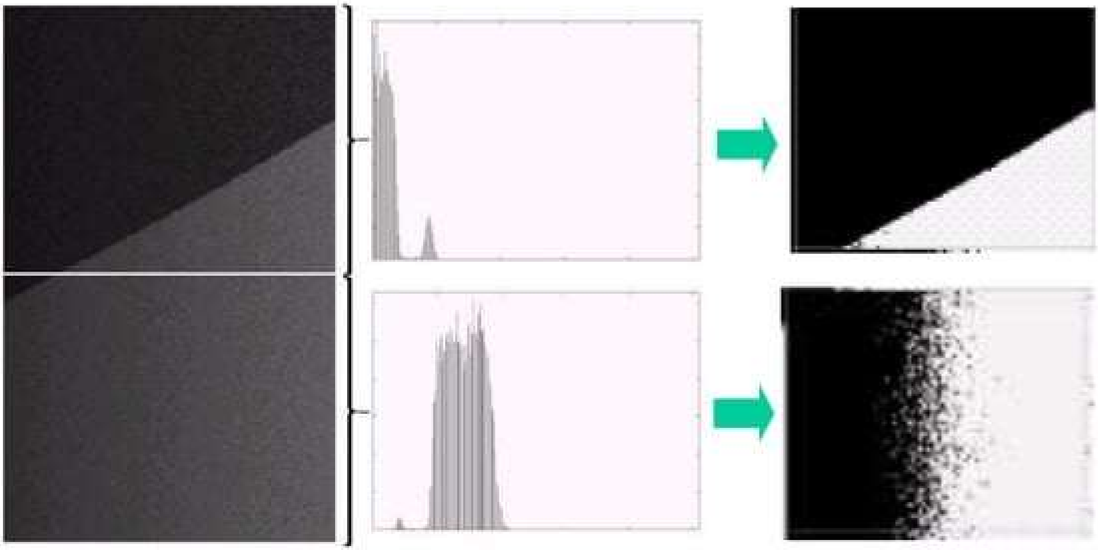


FIGURE 10.30 (a) Original image. (b) Result of global thresholding. (c) Image subdivided into individual subimages. (d) Result of adaptive thresholding.

Thresholding
Basic Adaptive Thresholding



How to solve this problem?

Thresholding
Basic Adaptive Thresholding

Answer: subdivision

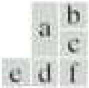
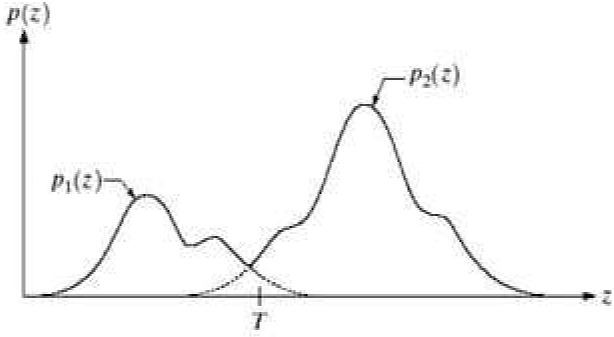


FIGURE 10.31 (a) Properly and improperly segmented subimages from Fig. 10.30. (b)–(c) Corresponding histograms. (d) Further subdivision of the improperly segmented subimage. (e) Histogram of small subimage at top, left. (f) Result of adaptively segmenting (d).

Thresholding
Optimal Global and Adaptive Thresholding

- This method treats pixel values as probability density functions.
- The goal of this method is to minimize the probability of misclassifying pixels as either object or background.
- There are two kinds of error:
  - mislabeling an object pixel as background, and
  - mislabeling a background pixel as object.

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



Thresholding
Use of Boundary Characteristics

a
b

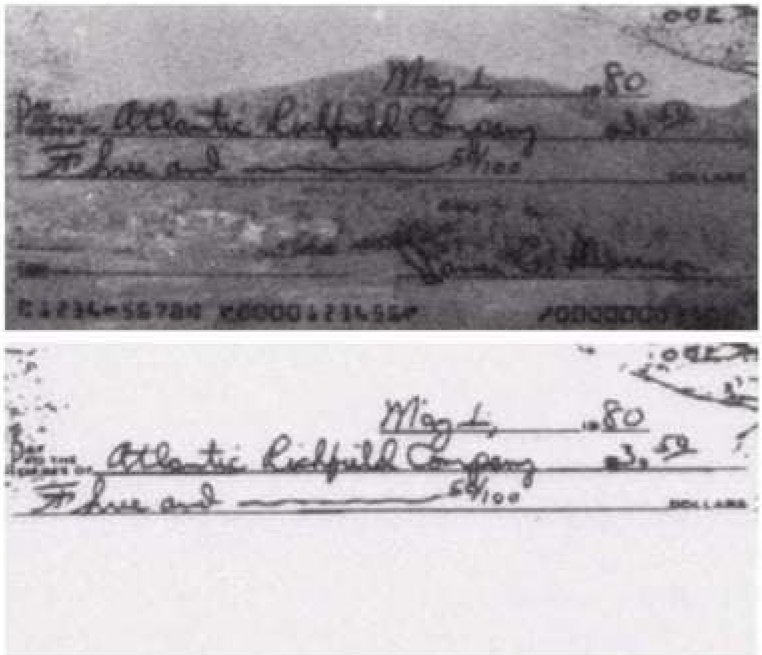


FIGURE 10.37 (a) Original image. (b) Image segmented by local thresholding. (Courtesy of IBM Corporation.)

Thresholding
Thresholds Based on Several Variables

Color image

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.

Region-Based Segmentation

- Edges and thresholds sometimes do not give good results for segmentation.
- Region-based segmentation is based on the connectivity of similar pixels in a region.
  - Each region must be uniform.
  - Connectivity of the pixels within the region is very important.
- There are two main approaches to region-based segmentation: region growing and region splitting.

Region-Based Segmentation
Basic Formulation

- Let R represent the entire image region.
- Segmentation is a process that partitions R into subregions, $R_1, R_2, \ldots, R_n$, such that

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

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

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

(d) $P(R_i) = \text{TRUE}$ for $i = 1, 2, \ldots, n$

(e) $P(R_i \cup R_j) = \text{FALSE}$ for any adjacent regions R_i and R_j

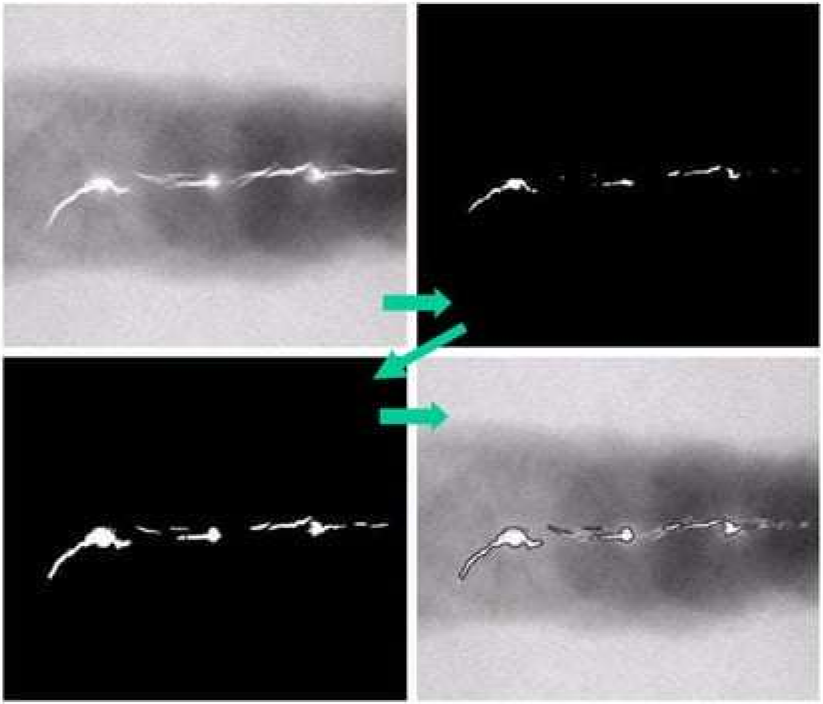
where $P(R_k)$: a logical predicate defined over the points in set R_k

For example: $P(R_k)$=TRUE if all pixels in R_k have the same gray level.

Region-Based Segmentation
Region Growing

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.).



Region-Based Segmentation
Region Growing

- Fig. 10.41 shows the histogram of Fig. 10.40 (a). It is difficult to segment the defects by thresholding methods. (Applying region growing methods are better in this case.)

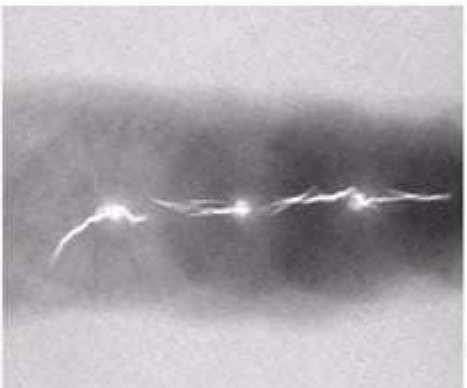


Figure 10.40(a)

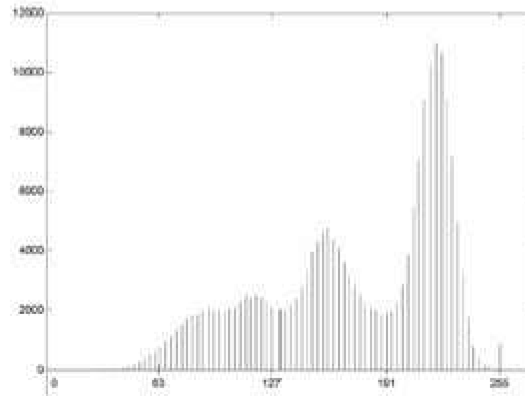


Figure 10.41

Region-Based Segmentation
Region Splitting and Merging

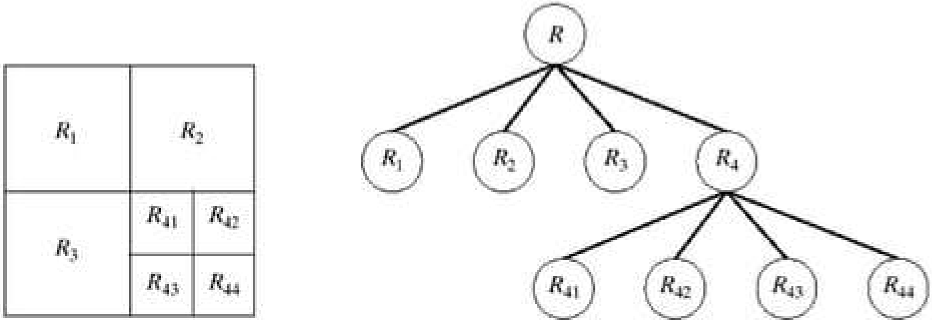
- Region splitting is the opposite of region growing.
  - First there is a large region (possible the entire image).
  - Then a predicate (measurement) is used to determine if the region is uniform.
  - If not, then the method requires that the region be split into two regions.
  - Then each of these two regions is independently tested by the predicate (measurement).
  - This procedure continues until all resulting regions are uniform.

Region-Based Segmentation
Region Splitting

- The main problem with region splitting is determining where to split a region.
- One method to divide a region is to use a quadtree structure.
- Quadtree: a tree in which nodes have exactly four descendants.

a b

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



Region-Based Segmentation
Region Splitting and Merging

- The split and merge procedure:
  - Split into four disjoint quadrants any region R_i for which $P(R_i)$ = FALSE.
  - Merge any adjacent regions R_j and R_k for which $P(R_j \cup R_k)$ = TRUE. (the quadtree structure may not be preserved)
  - Stop when no further merging or splitting is possible.

a b c

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

Segmentation by Morphological Watersheds

- The concept of watersheds is based on visualizing an image in three dimensions: two spatial coordinates versus gray levels.
- In such a topographic interpretation, we consider three types of points:
  - (a) points belonging to a regional minimum
  - (b) points at which a drop of water would fall with certainty to a single minimum
  - (c) points at which water would be equally likely to fall to more than one such minimum
- The principal objective of segmentation algorithms based on these concepts is to find the watershed lines.

Segmentation by Morphological Watersheds Example

a b
c d

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

Segmentation by Morphological Watersheds Example

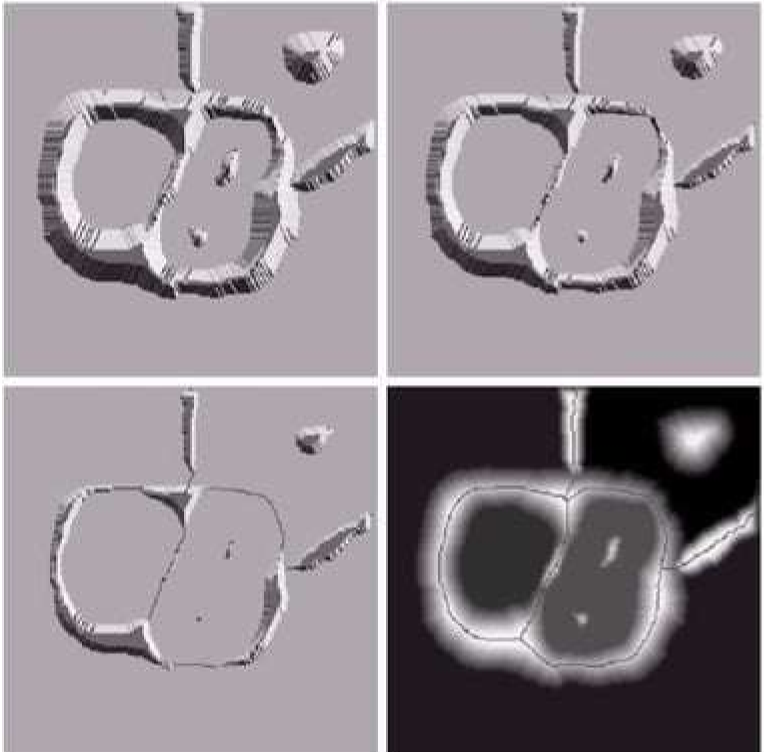
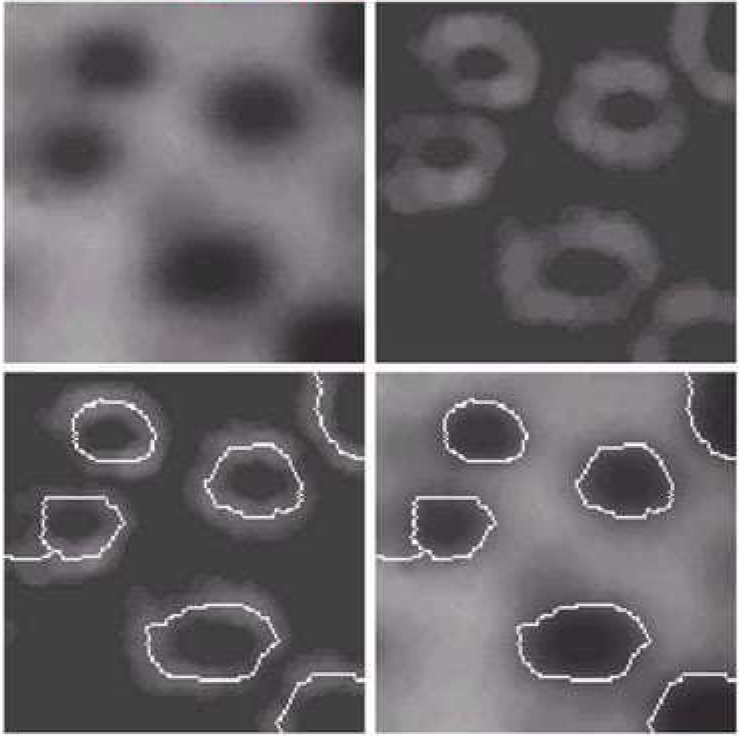


FIGURE 10.44 *(Continued)* (e) Result of further flooding. (f) Beginning of merging of water from two catchment basins (a short dam was built between them). (g) Longer dams. (h) Final watershed (segmentation) lines. (Courtesy of Dr. S. Beucher, CMM/Ecole des Mines de Paris.)

Segmentation by Morphological Watersheds Example

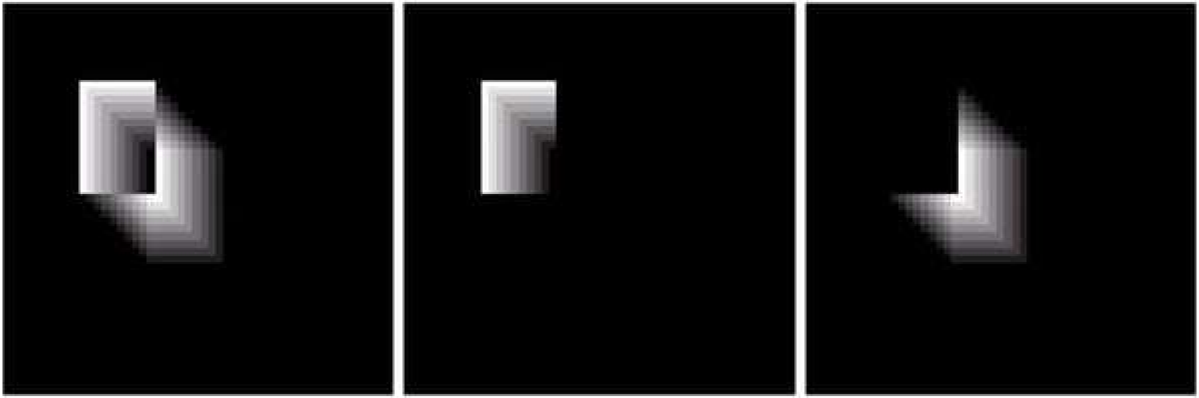


a b
c d

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.)

The Use of Motion in Segmentation

- ADI: accumulative difference 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.

The Use of Motion in Segmentation

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.)