Lecture 18
Representation and description I

1. Boundary representation
2. Boundary descriptors

What is representation

- What is representation
  - After segmentation, we obtain binary image with interested regions (objects) high-lighted.
  - Presentation is to represent the interested regions in a format which is convenient to process by human or computer program.
- How to represent regions: 2 ways
  - In terms of its external characteristics (its boundary), focusing on shape characteristics
  - In terms of its internal characteristics (its region) focusing on regional properties, e.g., color, texture
  - Or use both

Description

- Description: describes the region using certain features based on the chosen representation
- Feature selected should be **as insensitive as** possible to variations in
  - size
  - translation
  - rotation
- Example
  - Representation ⇨ boundary
  - Description ⇨ length of the boundary, orientation of the straight line joining its extreme points, and the number of concavities in the boundary.

Moors boundary tracking algorithm

- Given 0/1 binary image, in which objects are labeled 1 and background is labeled 0. Given a region R, i.e., a set of connected pixels with label 1.

Step 1. Let the start point b_0 be the uppermost and leftmost point of R

Let c_0 be the west neighbor of b_0.

Step 2. Examine the 8-neighbor of b_i

starting at c_i in clockwise direction. Let b_{i+1} be the first point with lable 1, and c_{i+1} be the proceding b_{i+1} in the sequence.

Store b_{i+1}

Step 3. Repeat Step 2 while not ($b_{i+1} = b_0$ and $b_{i+2} = b_1$)

Step 5. Output $b_0, \ldots, b_{i+1}$

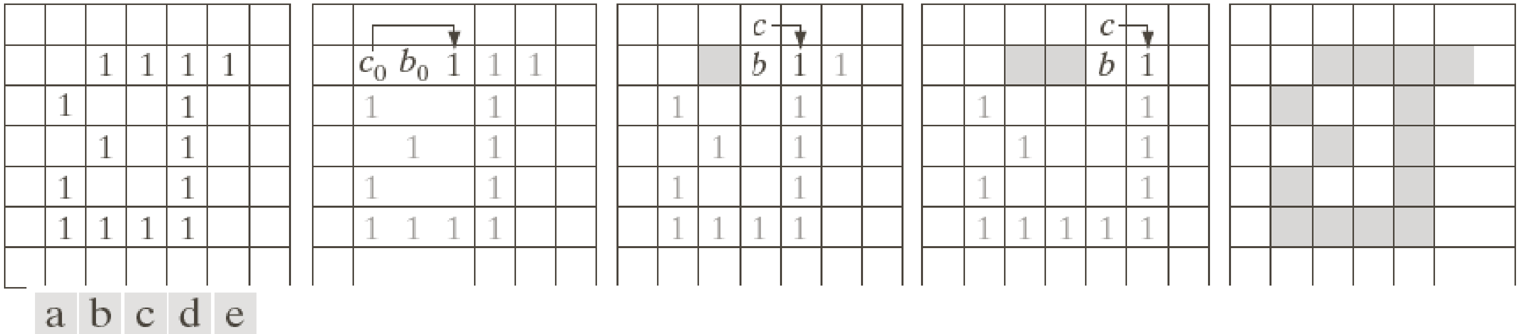
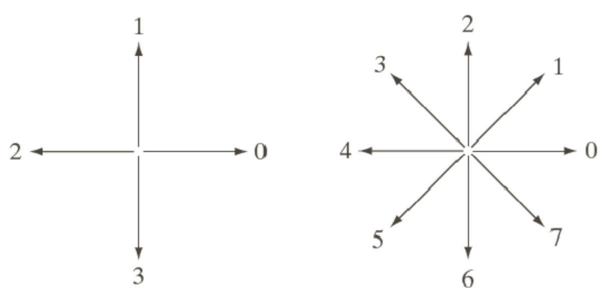


FIGURE 11.1 Illustration of the first few steps in the boundary-following algorithm. The point to be processed next is labeled in black, the points yet to be processed are gray, and the points found by the algorithm are labeled as gray squares.

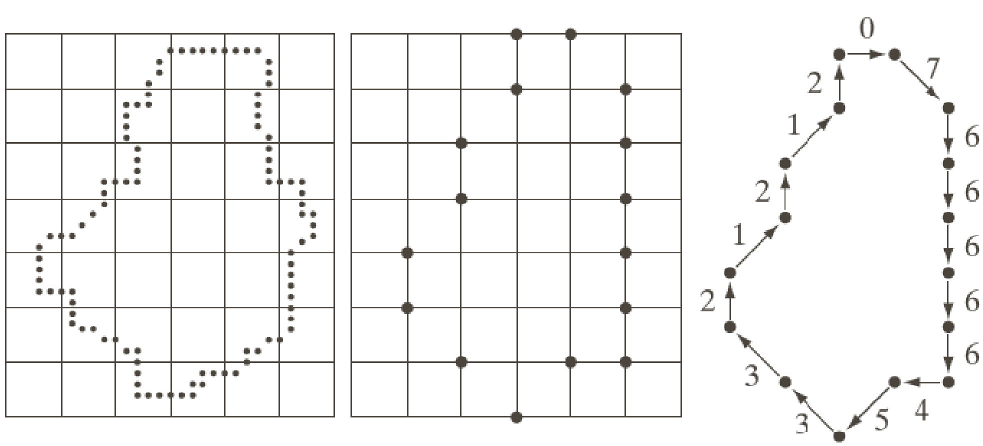
Chain codes

- Chain codes are used to represent a boundary by a connected sequence of straight line segments of special length and direction.
- Freeman chain code: based on 4- or 8-connectivity of segments. The direction is coded using a number scheme.
- Drawbacks: long and sensitive to disturbances
  - Solution: Resample by large grid spacing



a b

FIGURE 11.3 Direction numbers for (a) 4-directional chain code, and (b) 8-directional chain code.

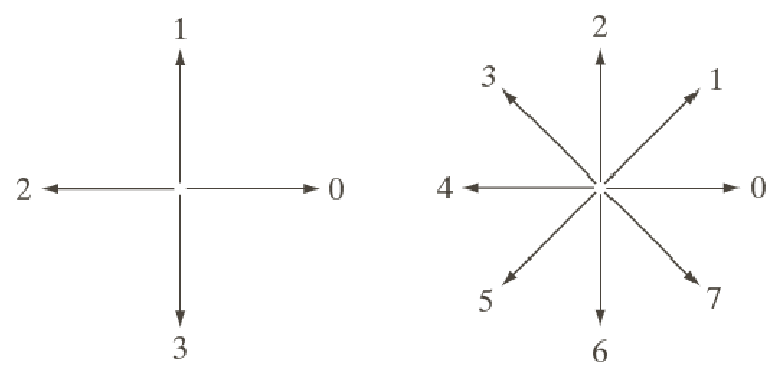


a b c

FIGURE 11.4 (a) Digital boundary with resampling grid superimposed. (b) Result of resampling. (c) 8-directional chain-coded boundary.

Normalized chain codes

- Normalize for starting point: treat the chain code as a circular sequence of direction numbers and redefine the starting point so that the resulting sequence of numbers forms an integer of minimum magnitude
- Normalize for rotation: use the first different chain code instead, difference = the number of direction changes in a counterclockwise direction



- Example
  - code 10103322
  - different is 3133030
  - circular chain code: 33133030

Example

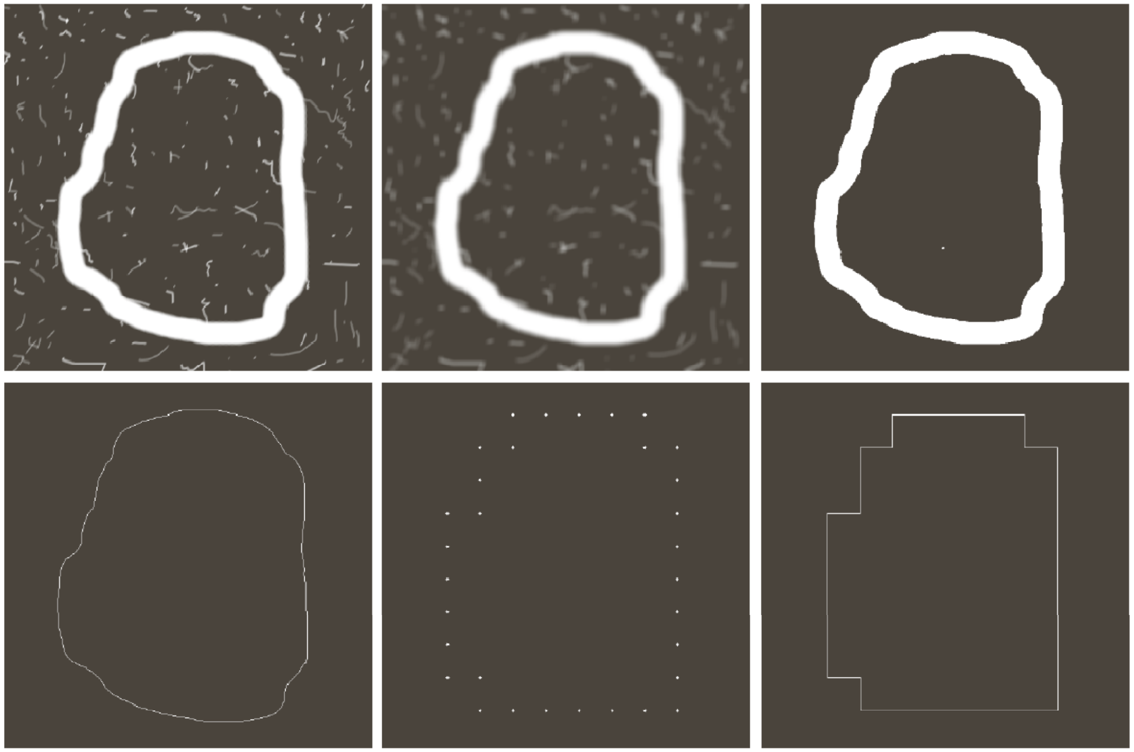
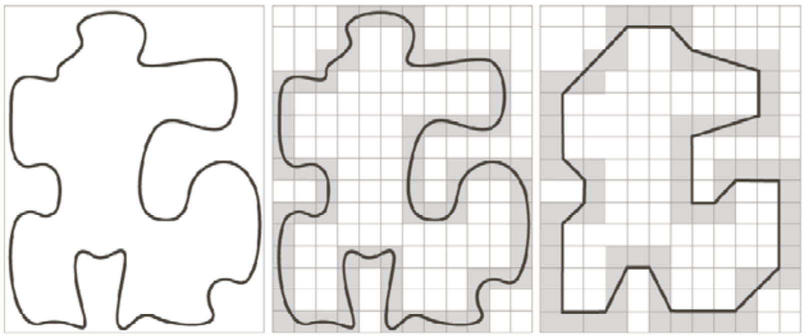


FIGURE 11.5 (a) Noisy image. (b) Image smoothed with a 9 × 9 averaging mask. (c) Smoothed image, thresholded using Otsu's method. (d) Longest outer boundary of (c). (e) Subsampled boundary (the points are shown enlarged for clarity). (f) Connected points from (e).

- 8-direction Freeman chain code of the simplified boundary: 00006066666666444444242222202202
- The first difference: 00062600000006000006200006200626

Polygonal Approximations

- Boundary can be approximated with arbitrary accuracy by a polygon, try to capture the essence of the boundary shape with the fewest possible polygonal segments.
- Minimum-perimeter polygon (MPP)
  - Cellular complex to enclose a boundary
  - Idea: rubber band constrained by the inner and outer walls of bounding region
  - if each cell encompass only one point on the boundary error is at most be $\sqrt{2}d$, d is the minimum possible distance between different pixels

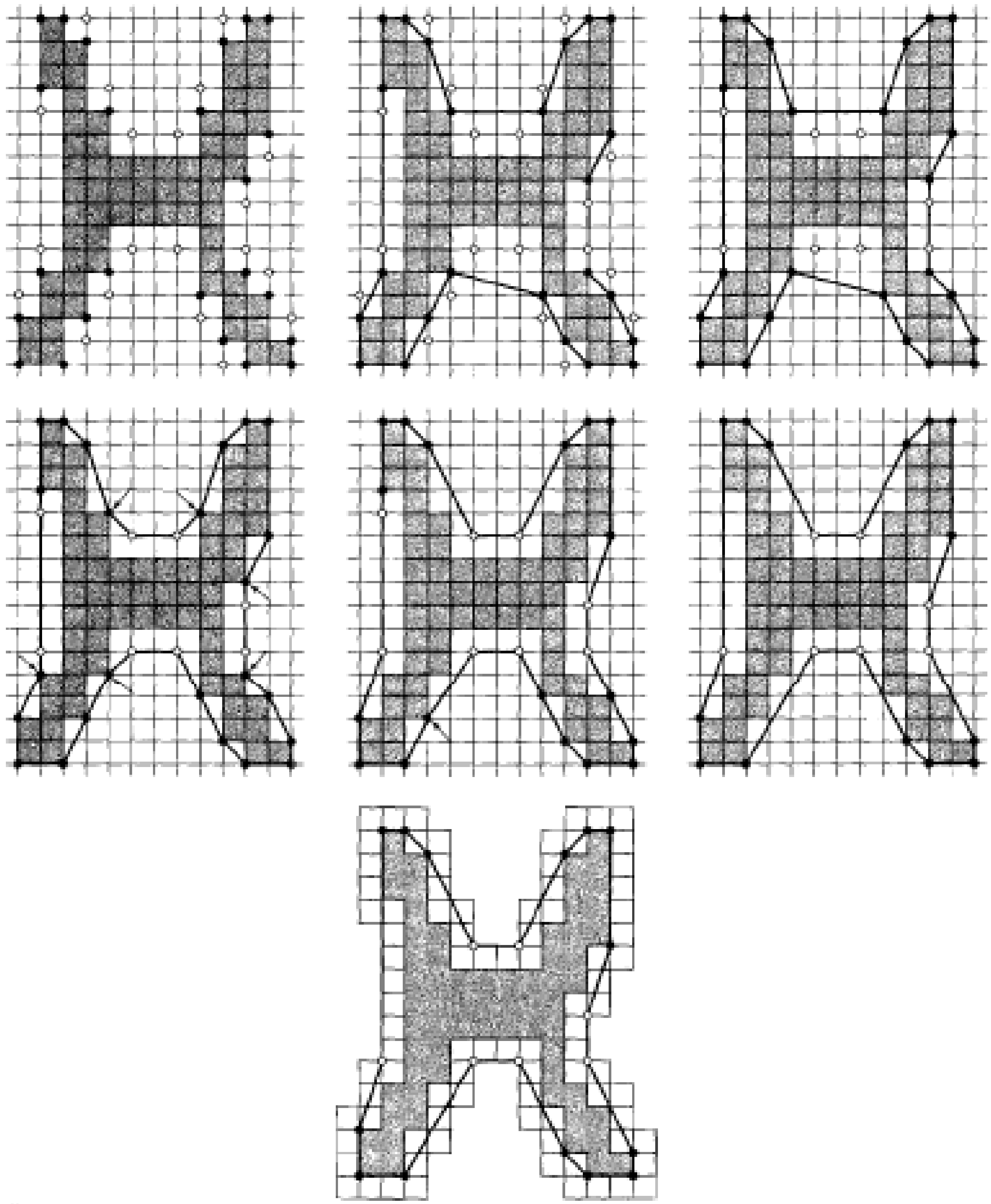


a b c

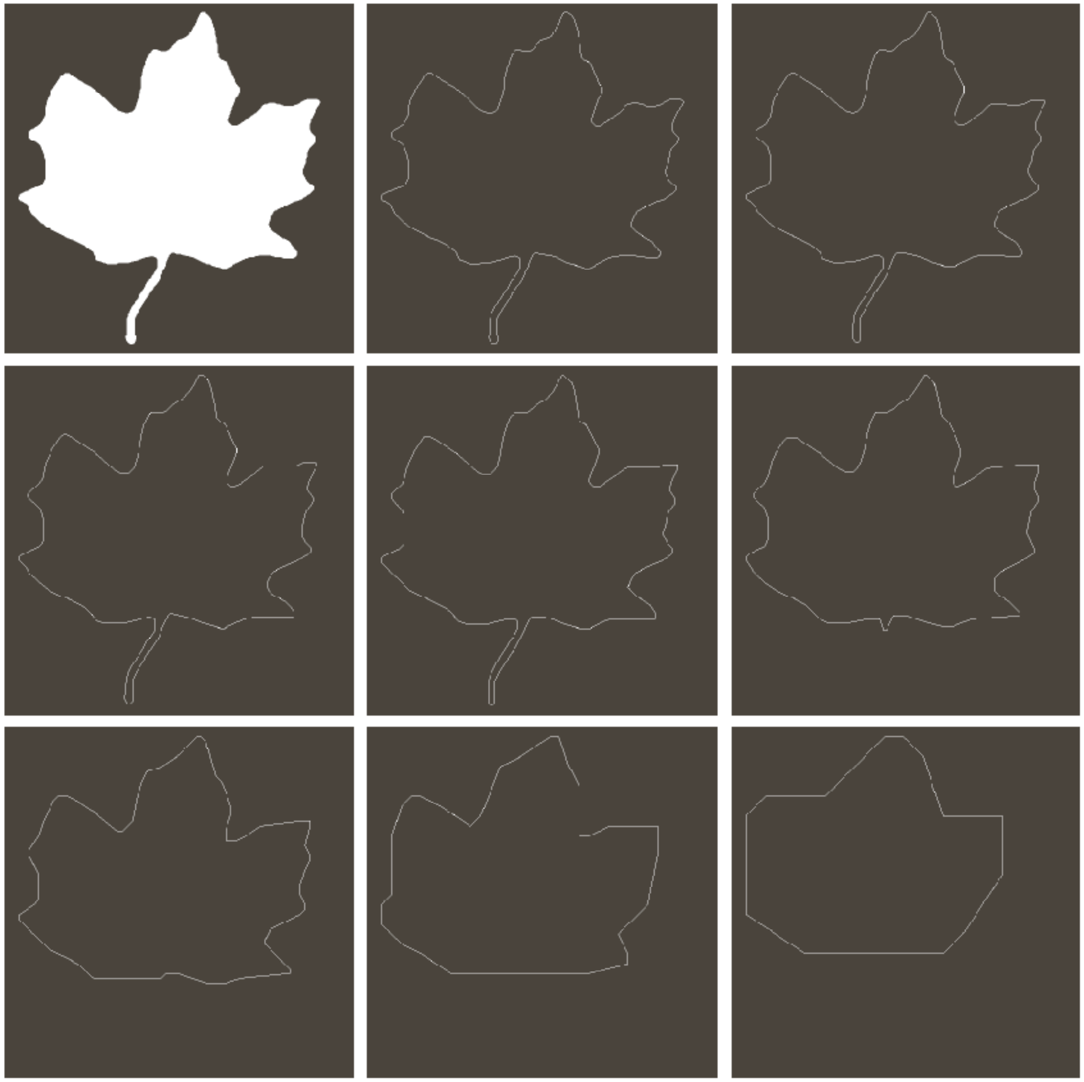
FIGURE 11.6 (a) An object boundary (black curve). (b) Boundary enclosed by cells (in gray). (c) Minimum-perimeter polygon obtained by allowing the boundary to shrink. The vertices of the polygon are created by the corners of the inner and outer walls of the gray region.

MPP algorithm

1. Obtain the cellular complex
2. Obtain the region internal to the cellular complex
3. Use function boundaries to obtain the boundary of the region in step 2 as a 4-connected, clockwise sequence of coordinates
4. Obtain the Freeman chain code of the sequence
5. Obtain the convex (white) concave (black) vertices from the chain code
6. From an initial polygon using the convex vertices and delete any concave vertices that outside the polygon
7. Form a polygon with the remaining convex and concave vertices.
8. Delete all black dots that are concave
9. Repeat 7-8 until all changes cease. The remaining vertices are the vertices of MPP



Example

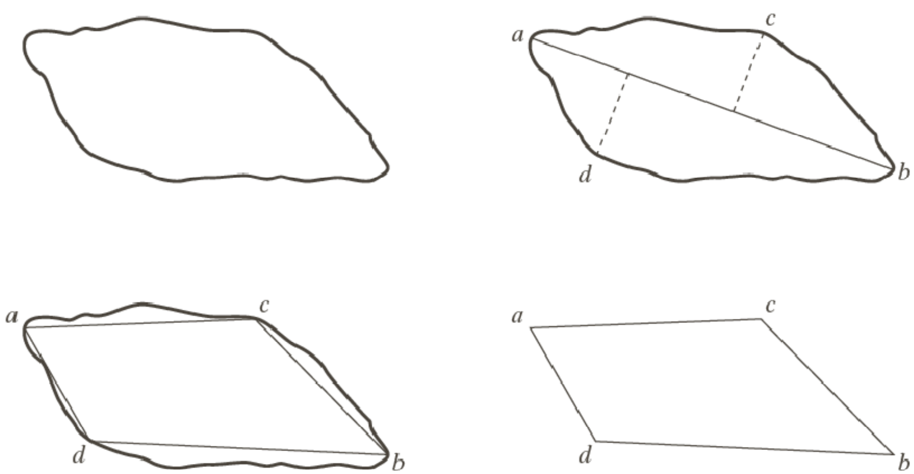


a	b	c
d	e	f
g	h	i

FIGURE 11.8 (a) 566 × 566 binary image. (b) 8-connected boundary. (c) through (i), MMPs obtained using square cells of sizes 2, 3, 4, 6, 8, 16, and 32, respectively (the vertices were joined by straight lines for display). The number of boundary points in (b) is 1900. The numbers of vertices in (c) through (i) are 206, 160, 127, 92, 66, 32, and 13, respectively.

Other polygonal approximation approaches

- Merging techniques
  - based on average error or other criteria, merge points along the boundary until the least square error line fit of the points merged so far exceeds a preset threshold
- Splitting techniques
  - find the major axis
  - find minor axes which perpendicular to major axis and has distance greater than a threshold
  - repeat until we can’t split anymore

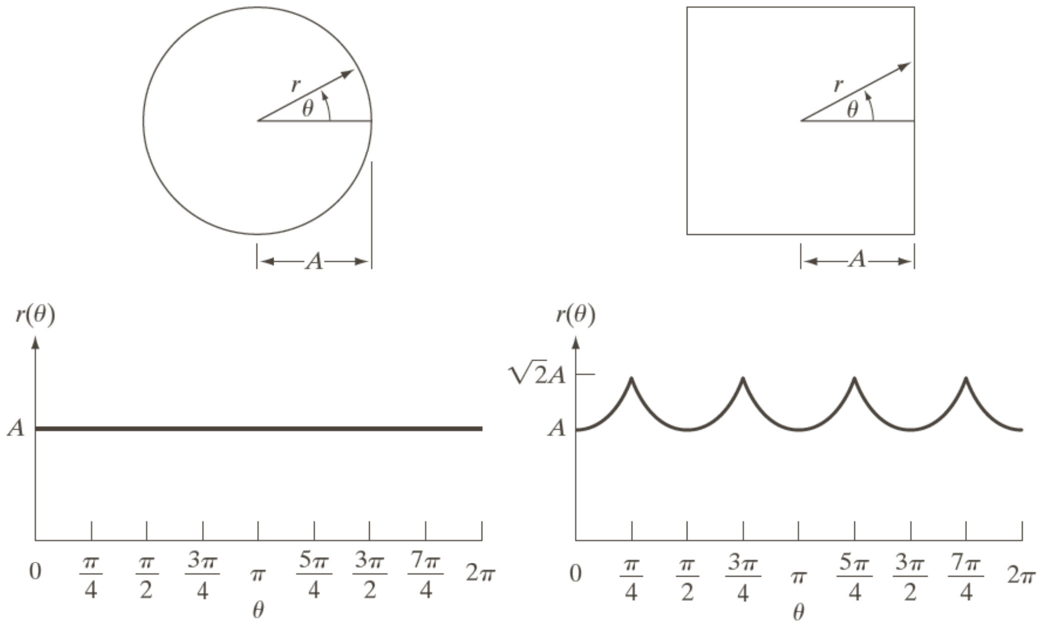


a b
c d

FIGURE 11.9
(a) Original boundary.
(b) Boundary divided into segments based on extreme points. (c) Joining of vertices.
(d) Resulting polygon.

Signature

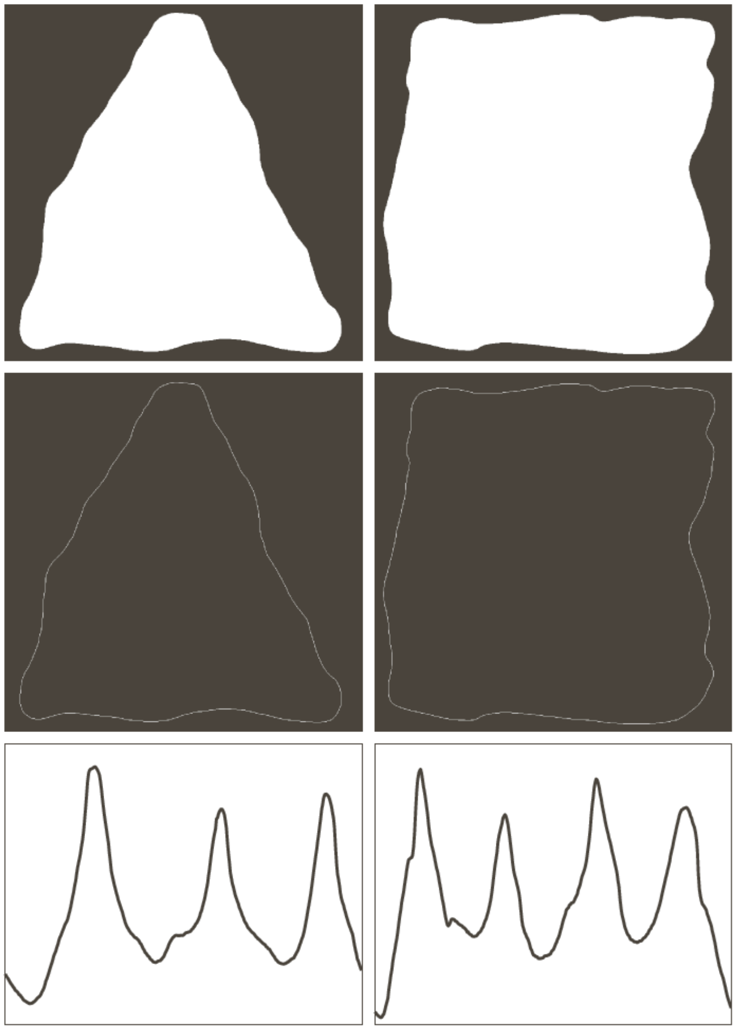
- A 1-D function representing a boundary.



a b

FIGURE 11.10 Distance-versus-angle signatures. In (a) $r(\theta)$ is constant. In (b), the signature consists of repetitions of the pattern $r(\theta) = A \sec \theta$ for $0 \leq \theta \leq \pi/4$ and $r(\theta) = A \csc \theta$ for $\pi/4 < \theta \leq \pi/2$.

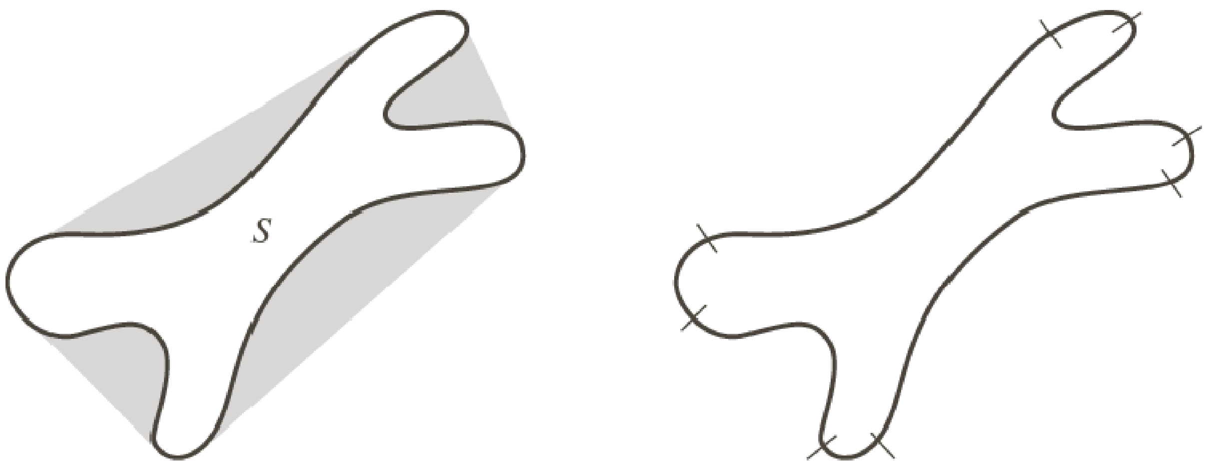
Example



a b
c d
e f

FIGURE 11.11 Two binary regions, their external boundaries, and their corresponding $r(\theta)$ signatures. The horizontal axes in (e) and (f) correspond to angles from 0° to 360°, in increments of 1°.

Boundary Segments



a b

FIGURE 11.12
(a) A region, *S*, and its convex deficiency (shaded).
(b) Partitioned boundary.

- Decompose a boundary into segment.
- Example: Find the convex hull H of an arbitrary set S. The set different H-S is called convex deficiency D of the set S, which can be computed by the boundary segmentation.

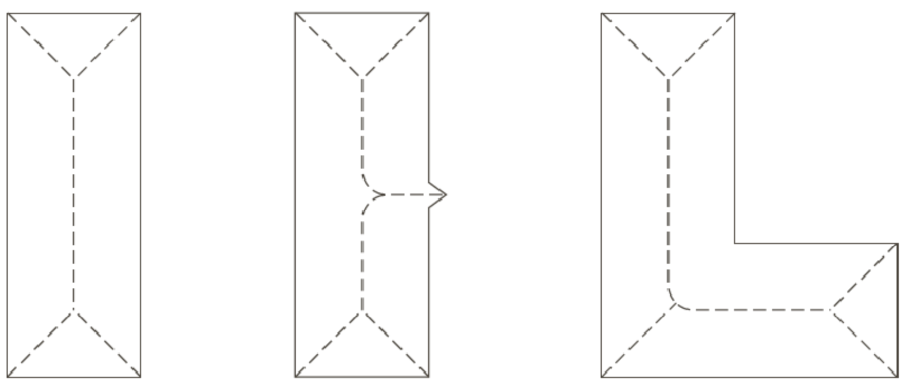
Skeletons

- Reduce a plane region shape to a graph.
- As the result of thinning: obtained by thinning operation, but may not keep the skeleton connected
- Skeleton defined by medial axis: points in the region has more than one closest neighbors
- Medial Axis Transformation (MAT):
  - MAT of region R with border B is as follows.

  for each point p in R, we find its closet neighbor in B.

  if p has more than one such neighbor, it is said to belong to the medial axis of R

  closest depends on the definition of a distance



a b c

FIGURE 11.13 Medial axes (dashed) of three simple regions.

Using thinning

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

FIGURE 11.14 Neighborhood arrangement used by the thinning algorithm.

iteratively deleting edge points of a region with constraints

1. does not remove end points
2. does not break connectivity
3. does not cause excessive erosion of the region

Skeleton Algorithm by thinning

Assume region points have value 1 and background points have value 0

Step 1: flag a contour point p_1 for deletion if the following conditions are satisfied:

$$(a)\quad 2 \le N(p_i) \le 6, (b)\quad T(p_1) = 1, (c)\quad p_2 \cdot p_4 \cdot p_6 = 0, (d)\quad p_4 \cdot p_6 \cdot p_8 = 0$$

$$where\ N(p_1) = p_2 + p_3 + \ldots + p_8 + p_9$$

$N(p_i)$ is the number of nonzero neighbors of p_i

contour point is any pixel with value 1 and having at least one 8-neighbor valued 0.

After step 1 has marked every boundary points satisfy all 4 conditions, delete those pixels.

step 2: remain condition (a) and (b) but change conditions (c) and (d) to follows

$$(c)'\quad p_2 \cdot p_4 \cdot p_8 = 0, (d)'\quad p_2 \cdot p_6 \cdot p_8 = 0$$

flagged the remain border points for deletion. then delete the marked points

repeat step 1) and 2) until no more points to delete

Example

0	0	1
1	p_1	0
1	0	1

FIGURE 11.15 Illustration of conditions (a) and (b) in Eq. (11.1-4). In this case $N(p_1) = 4$ and $T(p_1) = 3$.

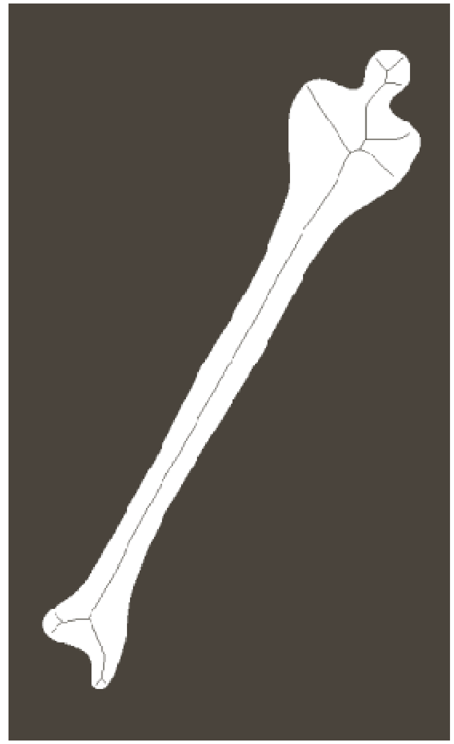


FIGURE 11.16 Human leg bone and skeleton of the region shown superimposed.

2. Boundary Descriptors

1. length of a boundary
2. diameters
3. eccentricity
4. curvature
5. shape numbers
6. Fourier descriptors
7. Statistical moments

Length of a boundary and Diameters

- Length of a boundary : the number of pixels along a boundary, give a rough approximation of its length

- Diameters: $$Diam(B) = \max_{i,j}[D(p_i, p_j)]$$

  D is a distance measure, p_i and p_j are points on the boundary B

Eccentricity, Curvature

- Eccentricity
  - Ratio of the major to the minor axis,
  - major axis = the line connecting the two extreme points that comprise the diameter
  - minor axis = the line perpendicular to the major axis

- Curvature
  - The rate of change of slope
    - difficult to do as digital boundaries tend to be locally ragged
    - using the difference between the slopes of adjacent boundary segments (which represented as straight lines)
    - use Merging and Splitting to create adjacent boundary segments
    - concave, convex and coner

Shape numbers

The number of digits in its representation

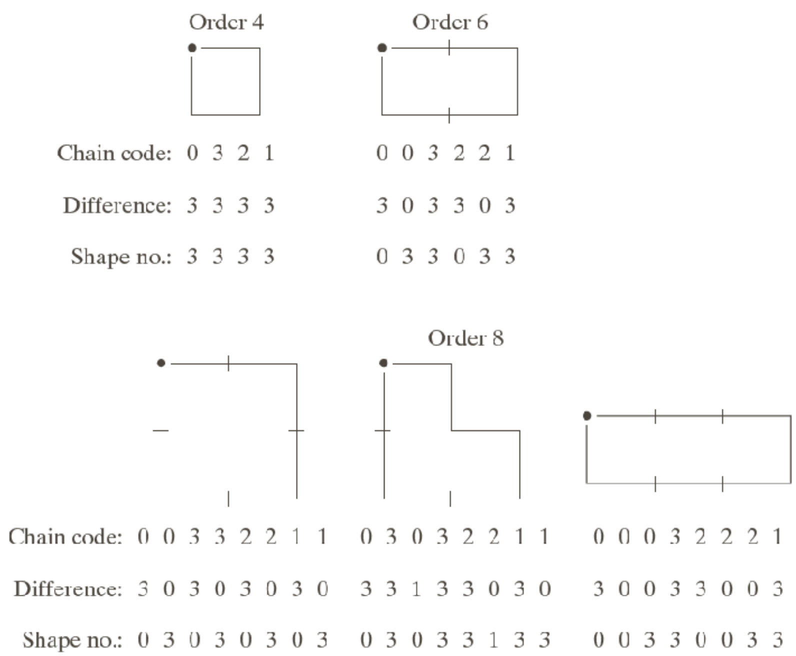
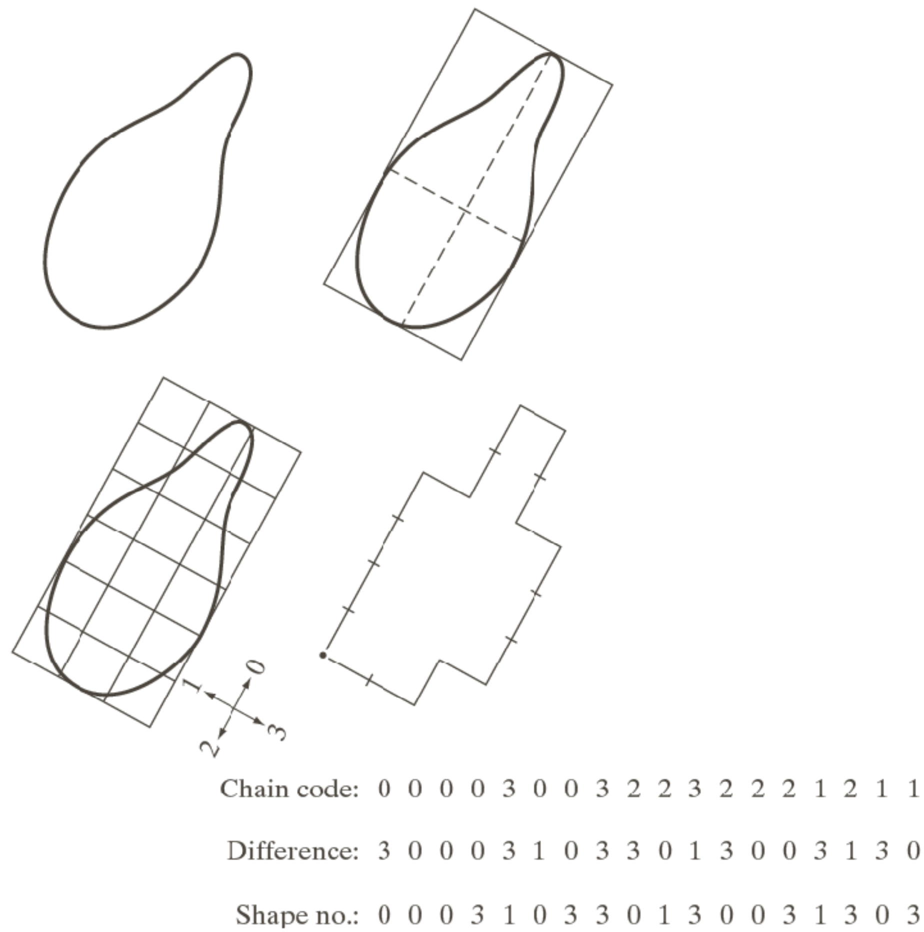


FIGURE 11.17 All shapes of order 4, 6, and 8. The directions are from Fig. 11.3(a), and the dot indicates the starting point.



a b
c d

FIGURE 11.18 Steps in the generation of a shape number.

Fourier Descriptors

boundary = $(x_0,y_0), \ldots , (x_{K-1},y_{k-1})$

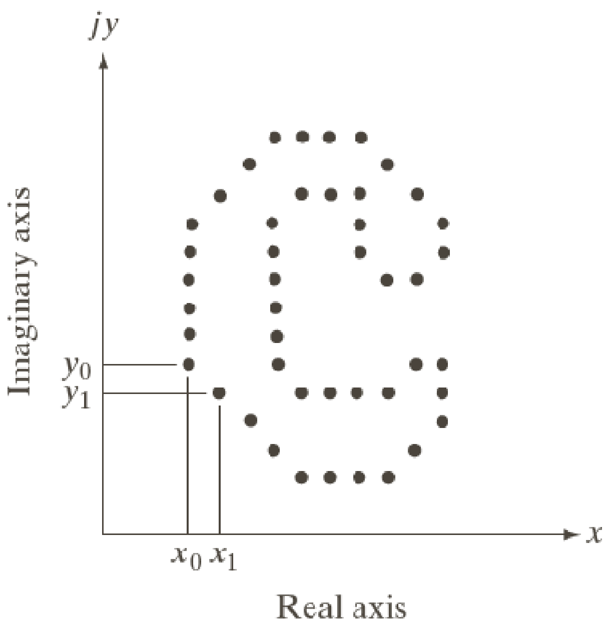


FIGURE 11.19 A digital boundary and its representation as a complex sequence. The points (x_0, y_0) and (x_1, y_1) shown are (arbitrarily) the first two points in the sequence.

$$s(k) = x(k) + jy(k) \quad \text{for } k = 0,1,\ldots,K-1$$

Fourier Descriptors

Fourier transformation (DFT)

$$a_r(u) = \frac{1}{K}\sum_{k=0}^{K-1} s(k)e^{-j2\pi uk/K} \quad \text{for } u = 0,1,\ldots,K\text{-}1$$

a(u) : Fourier coefficients (Fourier Descriptors)

Inverse Fourier transformation

$$s_t(k) = \sum_{k=0}^{K-1} a(u)e^{j2\pi uk/K} \quad \text{for } k = 0,1,\ldots,K\text{-}1$$

Invariant

Transformation	Boundary	Fourier Descriptor
Identity	$s(k)$	$a(u)$
Rotation	$s_r(k) = s(k)e^{j\theta}$	$a_r(u) = a(u)e^{j\theta}$
Translation	$s_t(k) = s(k) + \Delta_{xy}$	$a_t(u) = a(u) + \Delta_{xy}\delta(u)$
Scaling	$s_s(k) - \alpha s(k)$	$a_s(u) - \alpha a(u)$
Starting point	$s_p(k) = s(k - k_0)$	$a_p(u) = a(u)e^{-j2\pi k_0 u/K}$

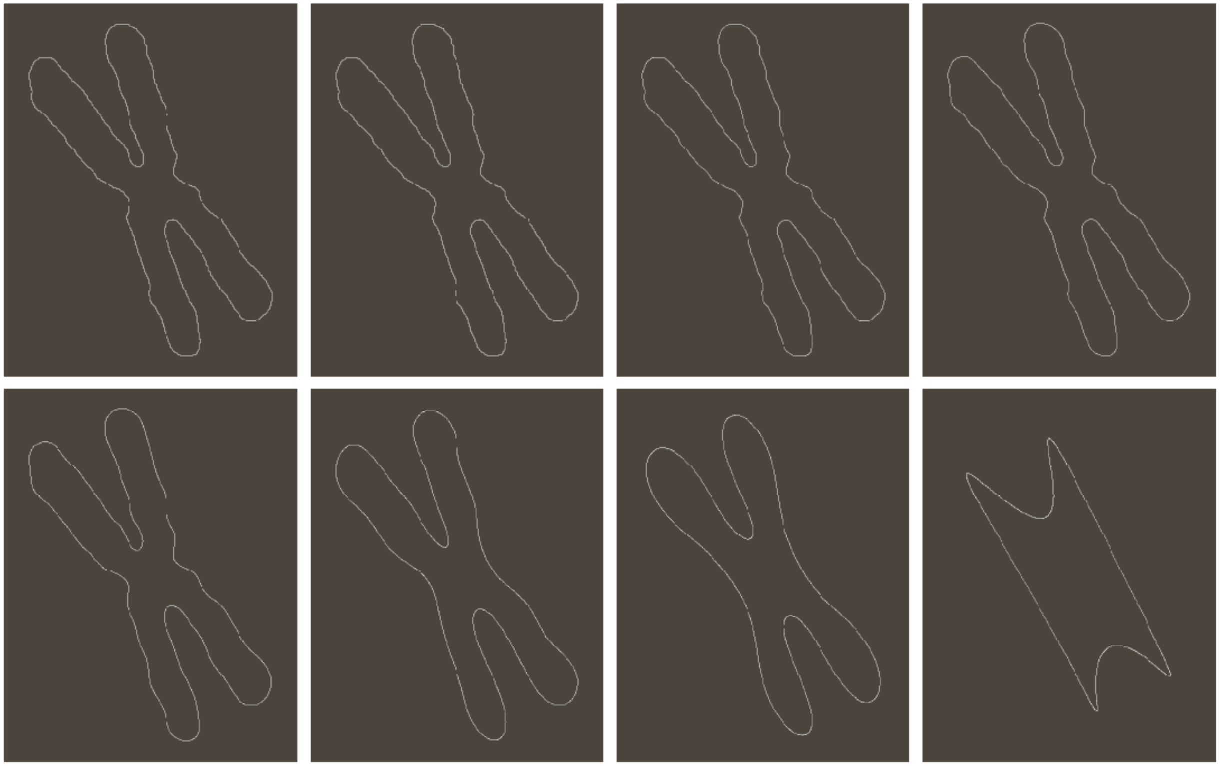
TABLE 11.1
Some basic properties of Fourier descriptors.

P Coefficient of Fourier Descriptors

$$\hat{s}(k) = \sum_{k=0}^{P-1} a(u)e^{j2\pi uk/K} \quad \text{for } k = 0,1,\ldots,K-1$$

approximation to *s(k)*

descriptors => P number of coefficients



a b c d
e f g h

FIGURE 11.20 (a) Boundary of human chromosome (2868 points). (b)–(h) Boundaries reconstructed using 1434, 286, 144, 72, 36, 18, and 8 Fourier descriptors, respectively. These numbers are approximately 50%, 10%, 5%, 2.5%, 1.25%, 0.63%, and 0.28% of 2868, respectively.

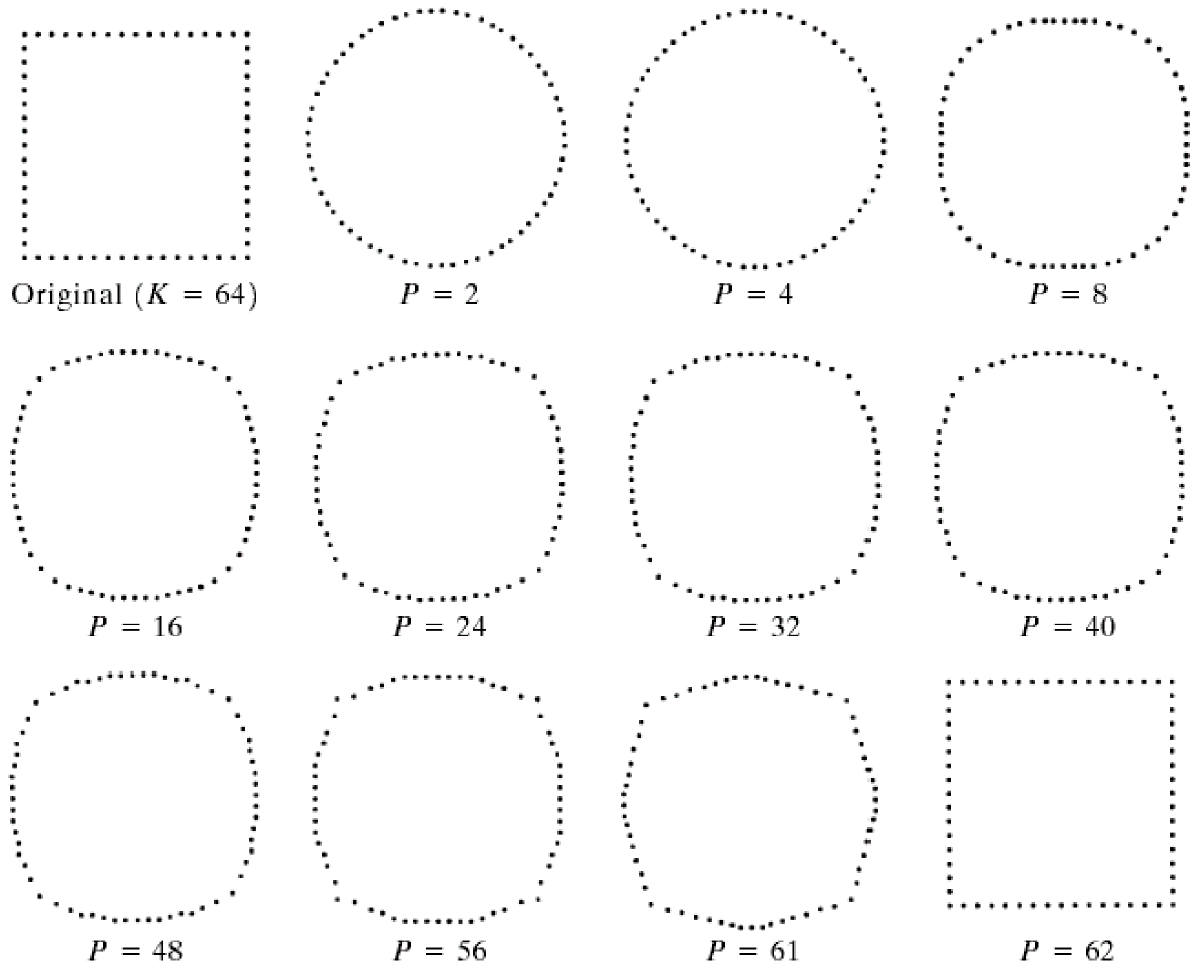


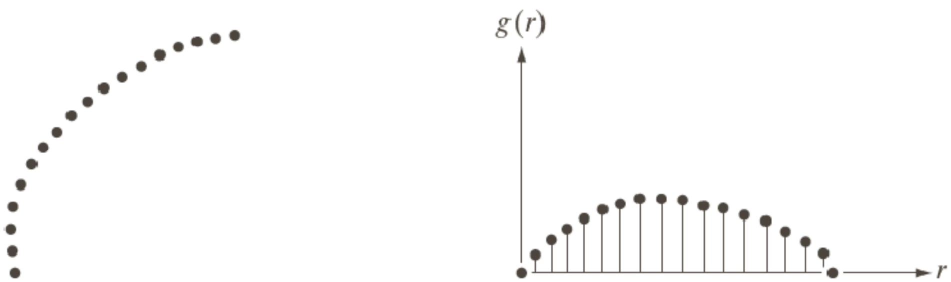
FIGURE 11.14 Examples of reconstruction from Fourier descriptors. P is the number of Fourier coefficients used in the reconstruction of the boundary.

Statistical moments

- The shape of boundary segments can be described by statistical moments such as mean, variance, higher-order moments.
- Assume the segmentation is represented by 1-D funciton g(r).
- p(vi), I = 0, 1, 2, …, A-1, A is the number of discrete amplitude increments

$$\mu_n(v) = \sum_{i=0}^{A-1} (v_i - m)^n p(v_i)$$

$$\text{where } m = \sum_{i=0}^{A-1} v_i p(v_i)$$



a b

FIGURE 11.21 (a) Boundary segment. (b) Representation as a 1-D function.