Chapter 11: Representation and Description

- A segmented region can be represented by $\begin{cases} \text{boundary pixels} \\ \text{internal pixels} \end{cases}$
- When shape is important, a boundary (external) representation is used
- When colour or texture is important, an internal representation is used
- The description of a region is based on its representation, for example a boundary can be described by its length
- The features selected as descriptors are usually required to be as insensitive as possible to variations in (1) scale, (2) translation and (3) rotation, that is the features should be scale, translation and rotation invariant

11.1 Representation

Image data, for example a boundary, is usually represented in a more compact way so that it can be described more easily

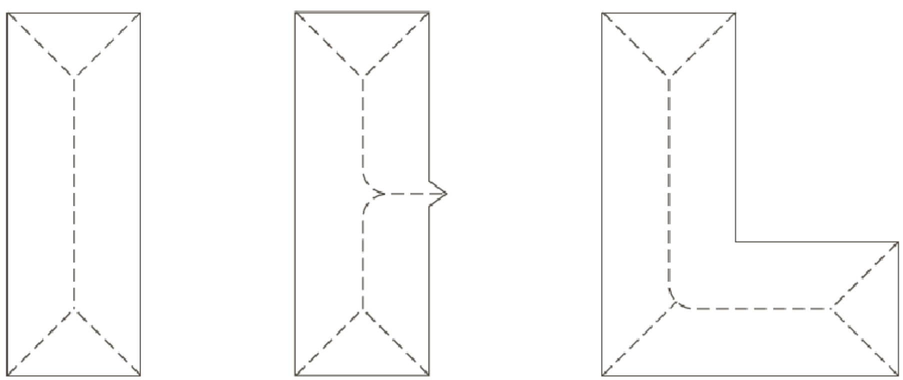
11.1.7 Skeletons (page 834) (We introduce this concept first...)

Skeletonization, that is the process that reduces the boundary of a region to a graph (which is a single pixel thick), often precedes other representation schemes and is therefore discussed first..

Brute force method: Medial axis transformation (MAT - 1967)

Consider a region R with boundary B...

- **For each point p in R find its closest neighbour in B**
- **If p has more than one closest neighbour, then p is part of the medial axis**



a b c

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

Problem: we have to calculate the distance between every internal point and every point on the edge of the boundary!

Thinning algorithm:

Edge points are deleted in an iterative way so that

(1) end points are not removed, (2) connectivity is not broken, and

(3) no excessive erosion is caused to the region

This algorithm thins a binary region, where an edge point $= 1$ and a background point $= 0$
Contour point: Edge point ($= 1$) with at least one neighbour with a value of 0

Step 1: A contour point is flagged for deletion if

(a) $2 \leq N(p_1) \leq 6$

(b) $T(p_1) = 1$

(c) $p_2 \cdot p_4 \cdot p_6 = 0$

(d) $p_4 \cdot p_6 \cdot p_8 = 0$

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

$N(p_1) \equiv$ **number of non-zero neighbours of** p_1
$T(p_1) \equiv$ **number of** $0{-}1$ **transitions in sequence:** $\{p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_2\}$

Now delete all the flagged contour points and consider the remaining contour points...

Step 2: A contour point is flagged for deletion if

(a) $2 \leq N(p_1) \leq 6$

(b) $T(p_1) = 1$

(c') $p_2 \cdot p_4 \cdot p_8 = 0$

(d') $p_2 \cdot p_6 \cdot p_8 = 0$

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

Delete all the flagged contour points; Repeat steps 1 and step 2 until no contour point is deleted during an iteration

Reasons for each of these conditions...

(a) $N(p_1) = 1 : \begin{array}{|c|c|c|}\hline 0 & 1 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}$ **: end point will be deleted!**

$N(p_1) = 7 : \begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 0 & 1 \\ \hline \end{array}$ **: erosion will occur!**

(b) $T(p_1) = 2 :$

0	**0**	**1**
0	**1**	**0**
1	**0**	**0**

: connectivity will be broken!

Note that $N(p_1) = 2$ **here**

Reasons for conditions (c), (d), (c') and (d'): see page 836

Example 11.5: The skeleton of a region

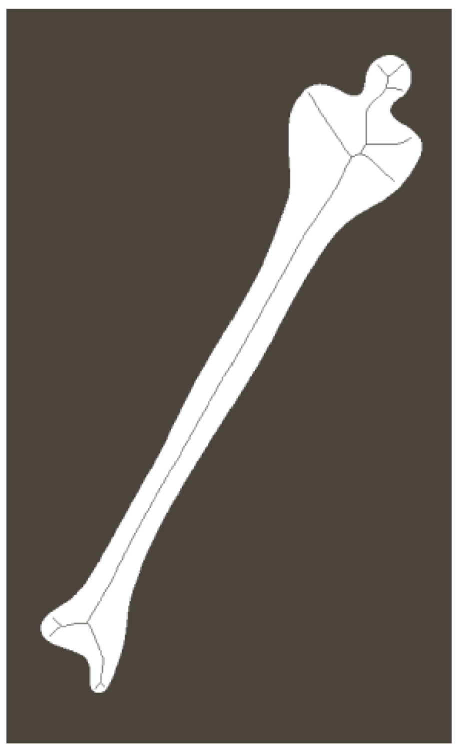


FIGURE 11.16 Human leg bone and skeleton of the region shown superimposed.

11.1.1 Boundary (border) following (page 818)

Moore boundary tracking algorithm (output is ordered sequence of points)

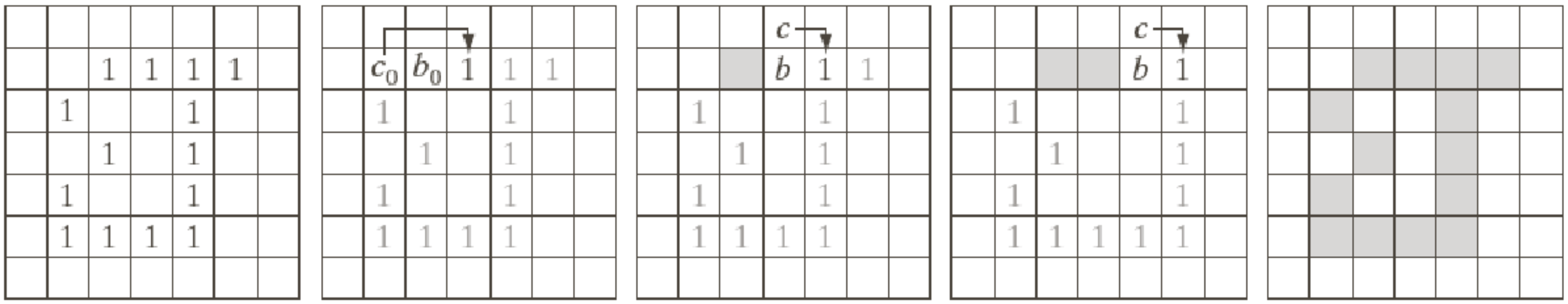
(1) Let the starting point b_0 be the uppermost, leftmost point in the image with label 1, and c_0 the west neighbour of b_0. Note that c_0 is always a background point. Examine the 8-neighbours of b_0, starting at c_0 and proceeding in a clockwise direction. Let b_1 denote the first neighbour encountered whose value is 1 and let c_1 be the (background) point immediately preceeding b_1 in the sequence. Store the locations of b_0 and b_1 for use in Step 5.

(2) Let $b = b_1$ and $c = c_1$.

(3) Let the 8-neighbours of b, starting at c and proceeding in a clockwise direction, be denoted by $n_1, n_2, \ldots, n_8$. Then find the first n_k labelled 1.

(4) Let $b = n_k$ and $c = n_{k-1}$.

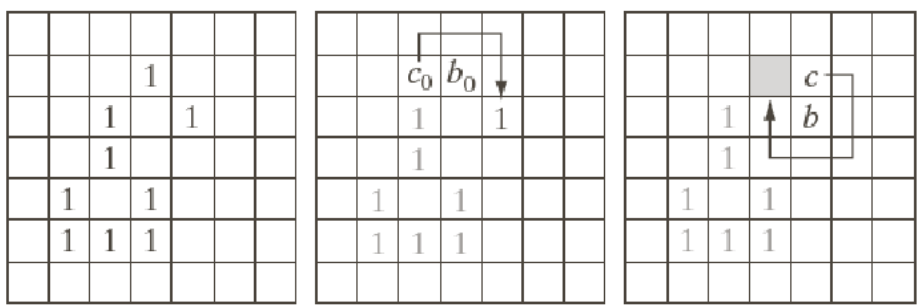
(5) Repeat Steps 3 and 4 until $b = b_0$ and the next boundary point found is b_1. The sequence of b points found when the algorithm stops constitutes the set of ordered boundary points.



a b c d e

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.

The algorithm should NOT stop the first time that b_0 is encountered again, as illustrated by the following example:



a b c

FIGURE 11.2 Illustration of an erroneous result when the stopping rule is such that boundary-following stops when the starting point, b_0, is encountered again.

Shape numbers
- **Representation: Chain codes** (page 820)
- **Description: Shape numbers** (page 838)
- **Matching shape numbers** (page 925)

11.1.2 Chain codes

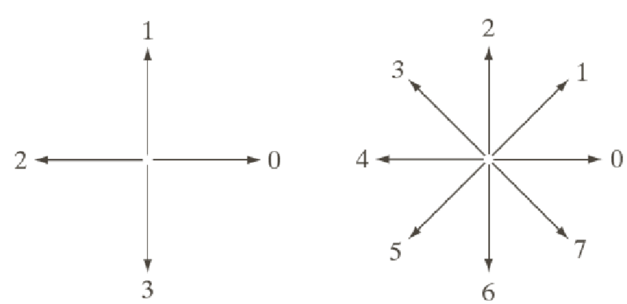
Assumption: thinning algorithm already applied to edge pixels

Generation of Freeman chain code: follow the boundary in an anti-clockwise direction and assign a direction to the segment between successive pixels

Difficulties:
- **Code generally very long**
- **Noise changes the code**

Solution: Resample the boundary using a larger grid spacing

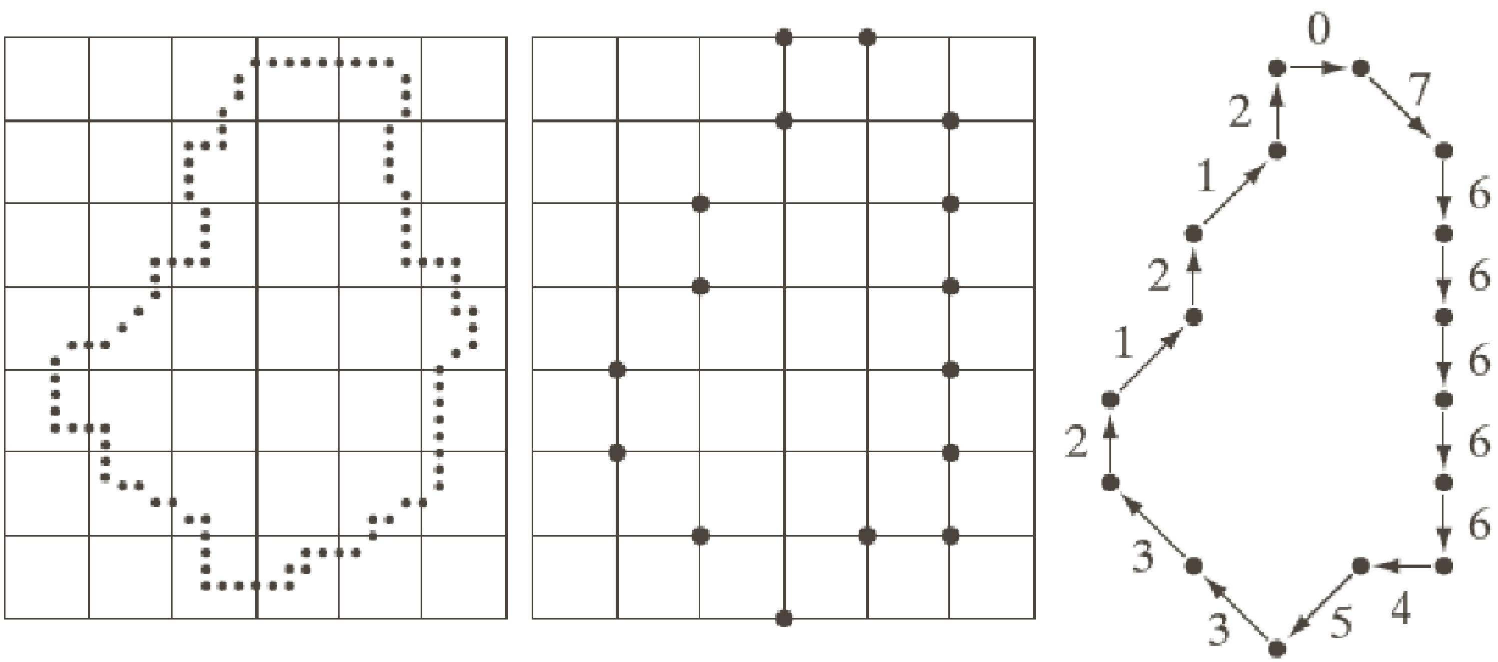
New boundary represented by 4- or 8-directional chain code



a b

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

Accuracy depends on grid size



a b c

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

Chain code depends on starting point

Normalization: consider the code to be circular and choose the starting point in such a way that the sequence represents the smallest integer

Example:

$$\begin{array}{ccccc} 1 & 3 & 0 & 2 & \\ 3 & 0 & 2 & 1 & \\ 0 & 2 & 1 & 3 & * \\ 2 & 1 & 3 & 0 & \end{array}$$

Rotation invariance: consider the first difference in the code

Example: (go counter-clockwise)

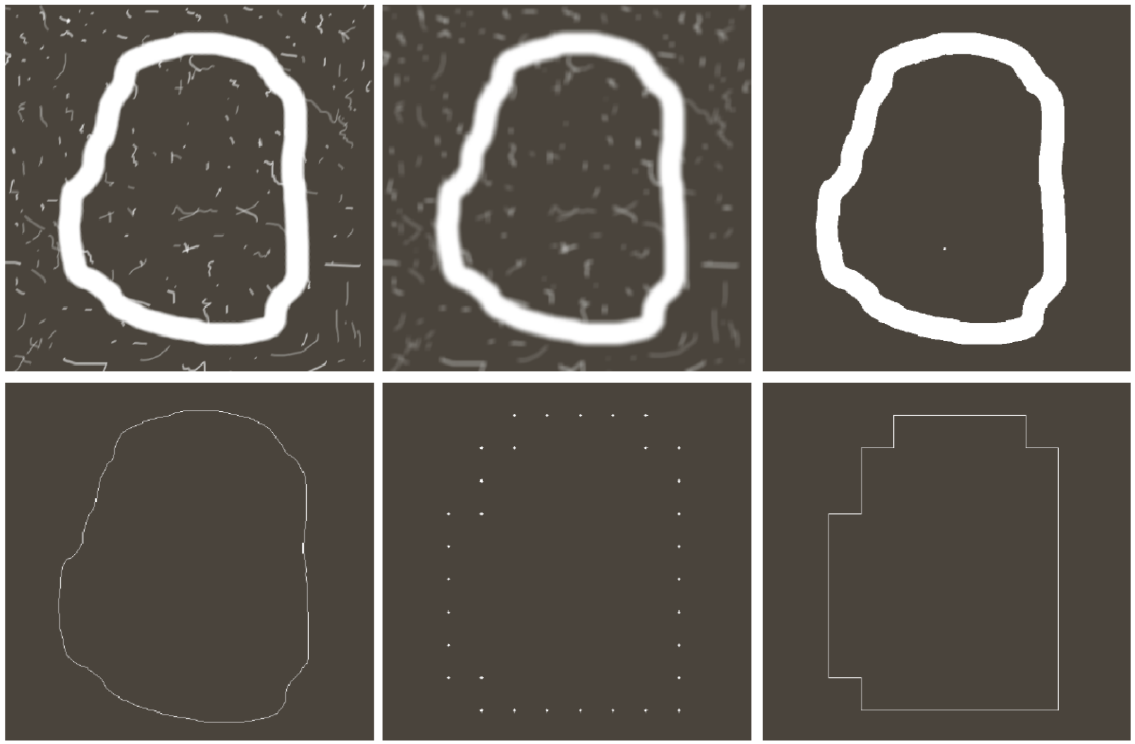
4-**directional code:**	1	0	1	0	3	3	2	2
first difference:	3	3	1	3	3	0	3	0

Scale invariance: chance grid size

Note: when objects differ in scale and orientation (rotation), they will be sampled differently!

$\longrightarrow$ **Shape numbers...** **(page 838)**

Example 11.1: Freeman chain code and some of its variations



a b c
d e f

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-directional Freeman chain code: 00006066666666444444242222202202

Int of min magnitude (start at $(2, 5)$): 00006066666666444444242222202202

First difference: 00062600000006000006260000620626

11.2.2 Shape numbers (page 838)

Shape number: first difference of chain code that represents smallest integer

Order n of shape number: number of digits in its representation

Suppose that we use a 4-directional chain code, then...

- **closed boundary $\Rightarrow$ n even**
- **n limits the number of possible shapes**

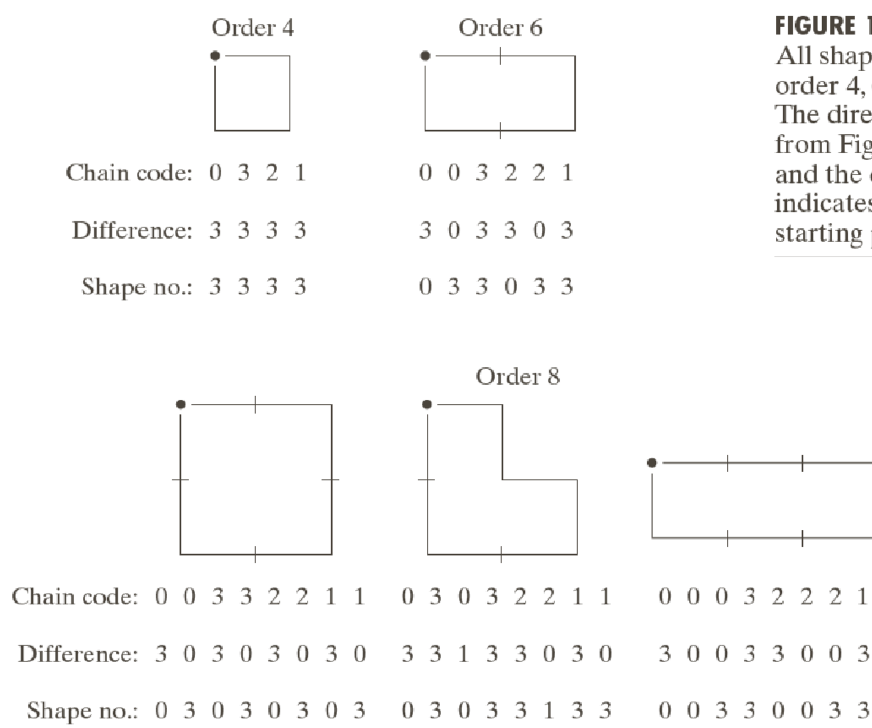


FIGURE 11.17 All shapes of order 4, 6, and 8. The directions are from Fig. 11.3(a), and the dot indicates the starting point.

Note: the first difference is rotation invariant, but the coded boundary depends on the orientation of the grid!

Normalization...

Major axis: the line through the centroid of the boundary pixels and parallel to the direction of maximum variance of the boundary pixels

Minor axis: the line through the centroid of the boundary pixels and perpendicular to the direction of maximum variance of the boundary pixels

When $C = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}$ **is the covariance matrix of the boundary pixel coordinates, the angle** θ **that the major axis makes with the horizontal is given by**

$$\theta = \arctan \frac{c_{22}}{c_{11}} = \arctan \frac{\sigma_y^2}{\sigma_x^2}$$

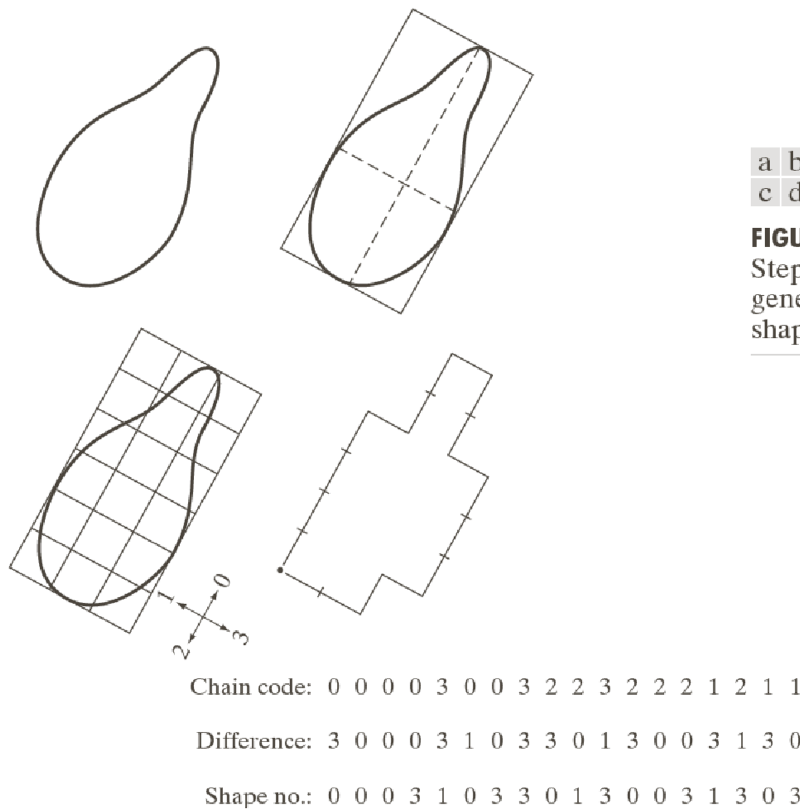
Construct a basic rectangle in such a way that the boundary fits within it

$$\textbf{Eccentricity of boundary} = \frac{\textbf{length of major axis}}{\textbf{length of minor axis}}$$

Therefore, for a specified order n, find a rectangle of order n of which the eccentricity is the closest to that of the basic rectangle. This determines the size of the grid cells

For example, options for $n = 12$: $\begin{cases} 2 \times 4\text{: } \textbf{eccentricity} = 2 \\ 3 \times 3\text{: } \textbf{eccentricity} = 1 \\ 1 \times 5\text{: } \textbf{eccentricity} = 5 \end{cases}$

Example 11.6: Computing shape numbers



a b
c d

FIGURE 11.18 Steps in the generation of a shape number.

12.3.1 Matching shape numbers (page 925)

The degree of similarity, k, between two region boundaries (shapes) is defined as the largest order for which their shape numbers still coincide

For example, when a and b denote shape numbers of closed boundaries represented by 4-directional chain codes, these two shapes have a degree of similarity k if

$$\begin{aligned} s_j(a) &= s_j(b) \ \textbf{ for } j = 4, 6, 8, \ldots, k \\ s_j(a) &\neq s_j(b) \ \textbf{ for } j = k+2, k+4, \ldots \end{aligned}$$

where s indicates shape number and the subscript indicates order

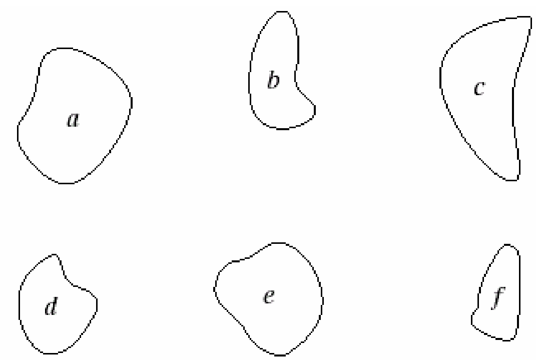
The distance between two shapes a and b is defined as

$$D(a, b) = \frac{1}{k}$$

where

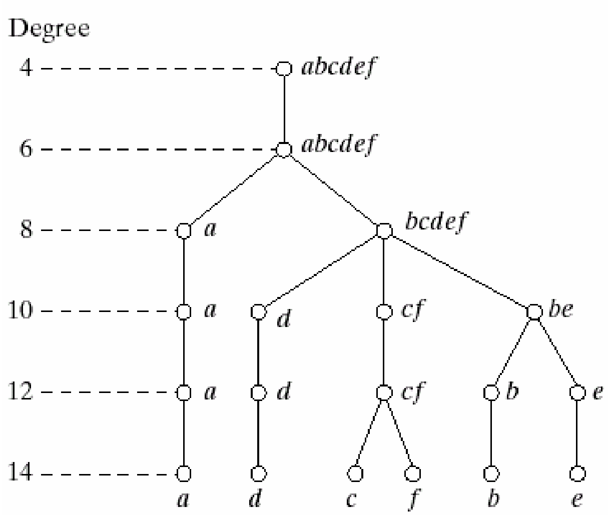
$$\begin{aligned} &D(a, b) \geq 0 \\ &D(a, b) = 0 \ \textbf{ iff } a = b \\ &D(a, c) \leq \max\left[D(a, b),\ D(b, c) \right] \end{aligned}$$

Example 12.7: Using shape numbers to compare shapes



a
b c

FIGURE 12.24
(a) Shapes.
(b) Hypothetical similarity tree.
(c) Similarity matrix. (Bribiesca and Guzman.)



	a	*b*	*c*	*d*	*e*	*f*
a	∞	6	6	6	6	6
b		∞	8	8	10	8
c			∞	8	8	12
d				∞	8	8
e					∞	8
f						∞

(a) Shapes (b) Hypothetical similarity tree (c) Similarity matrix