9 Morphological Image Processing

In form and feature, face and limb,
I grew so like my brother
That folks got taking me for him
And each for one another.

Henry Sambrooke Leigh, Carols of Cockayne, The Twins

Preview

The word *morphology* commonly denotes a branch of biology that deals with the form and structure of animals and plants. We use the same word here in the context of *mathematical morphology* as a tool for extracting image components that are useful in the representation and description of region shape, such as boundaries, skeletons, and the convex hull. We are interested also in morphological techniques for pre- or postprocessing, such as morphological filtering, thinning, and pruning.

In the following sections we develop and illustrate several important concepts in mathematical morphology. Many of the ideas introduced here can be formulated in terms of n-dimensional Euclidean space, E^n. However, our interest initially is on binary images whose components are elements of Z^2 (see Section 2.4.2). We discuss extensions to gray-scale images in Section 9.6.

The material in this chapter begins a transition from a focus on purely image processing methods, whose input and output are images, to processes in which the inputs are images, but the outputs are attributes extracted from those images, in the sense defined in Section 1.1. Tools such as morphology and related concepts are a cornerstone of the mathematical foundation that is utilized for extracting "meaning" from an image. Other approaches are developed and applied in the remaining chapters of the book.

9.1 Preliminaries

You will find it helpful to review Sections 2.4.2 and 2.6.4 before proceeding.

The language of mathematical morphology is set theory. As such, morphology offers a unified and powerful approach to numerous image processing problems. Sets in mathematical morphology represent objects in an image. For example, the set of all white pixels in a binary image is a complete morphological description of the image. In binary images, the sets in question are members of the 2-D integer space Z^2 (see Section 2.4.2), where each element of a set is a tuple (2-D vector) whose coordinates are the (x, y) coordinates of a white (or black, depending on convention) pixel in the image. Grayscale digital images of the form discussed in the previous chapters can be represented as sets whose components are in Z^3. In this case, two components of each element of the set refer to the coordinates of a pixel, and the third corresponds to its discrete intensity value. Sets in higher dimensional spaces can contain other image attributes, such as color and time varying components.

The set reflection operation is analogous to the flipping (rotating) operation performed in spatial convolution (Section 3.4.2).

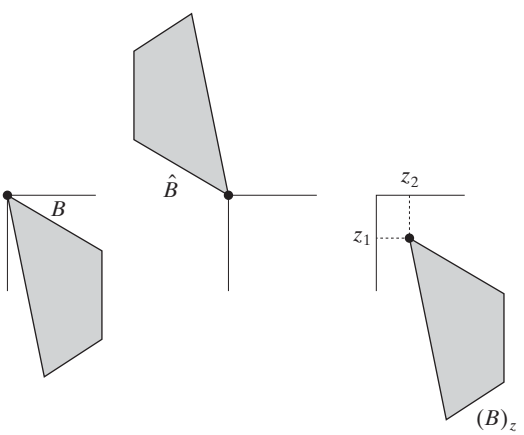
In addition to the basic set definitions in Section 2.6.4, the concepts of set reflection and translation are used extensively in morphology. The *reflection* of a set B, denoted $\hat{B}$, is defined as

$$\hat{B} = \{w | w = -b, \quad \text{for} \quad b \in B\} \tag{9.1-1}$$

If B is the set of pixels (2-D points) representing an object in an image, then $\hat{B}$ is simply the set of points in B whose (x, y) coordinates have been replaced by $(-x, -y)$. Figures 9.1(a) and (b) show a simple set and its reflection.[†]

a b c

FIGURE 9.1 (a) A set, (b) its reflection, and (c) its translation by z.



[†]When working with graphics, such as the sets in Fig. 9.1, we use shading to indicate points (pixels) that are members of the set under consideration. When working with binary images, the sets of interest are pixels corresponding to objects. We show these in white, and all other pixels in black. The terms *foreground* and *background* are used often to denote the sets of pixels in an image defined to be objects and non-objects, respectively.

The *translation* of a set B by point $z = (z_1, z_2)$, denoted $(B)_z$, is defined as

$$(B)_z = \{c | c = b + z, \quad \text{for} \quad b \in B\} \tag{9.1-2}$$

If B is the set of pixels representing an object in an image, then $(B)_z$ is the set of points in B whose (x, y) coordinates have been replaced by $(x + z_1, y + z_2)$. Figure 9.1(c) illustrates this concept using the set B from Fig. 9.1(a).

Set reflection and translation are employed extensively in morphology to formulate operations based on so-called *structuring elements* (SEs): small sets or subimages used to probe an image under study for properties of interest. The first row of Fig. 9.2 shows several examples of structuring elements where each shaded square denotes a member of the SE. When it does not matter whether a location in a given structuring element is or is not a member of the SE set, that location is marked with an "×" to denote a "don't care" condition, as defined later in Section 9.5.4. In addition to a definition of which elements are members of the SE, the origin of a structuring element also must be specified. The origins of the various SEs in Fig. 9.2 are indicated by a black dot (although placing the center of an SE at its center of gravity is common, the choice of origin is problem dependent in general). When the SE is symmetric and no dot is shown, the assumption is that the origin is at the center of symmetry.

When working with images, we require that structuring elements be rectangular arrays. This is accomplished by appending the smallest possible number of background elements (shown nonshaded in Fig. 9.2) necessary to form a rectangular array. The first and last SEs in the second row of Fig. 9.2 illustrate the procedure. The other SEs in that row already are in rectangular form.

As an introduction to how structuring elements are used in morphology, consider Fig. 9.3. Figures 9.3(a) and (b) show a simple set and a structuring element. As mentioned in the previous paragraph, a computer implementation requires that set A be converted also to a rectangular array by adding background elements. The background border is made large enough to accommodate the entire structuring element when its origin is on the border of the

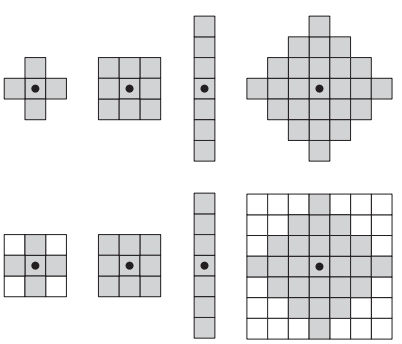
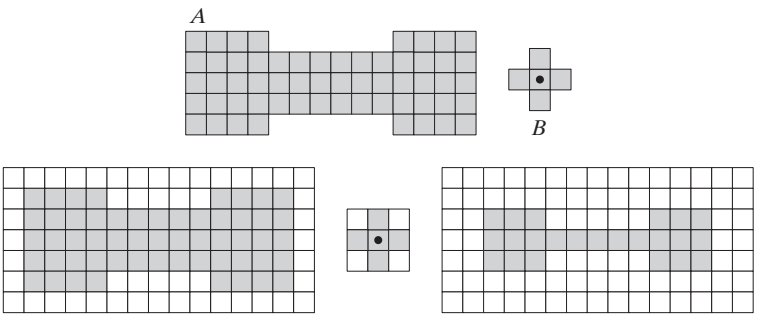


FIGURE 9.2 First row: Examples of structuring elements. Second row: Structuring elements converted to rectangular arrays. The dots denote the centers of the SEs.



a b
c d e

FIGURE 9.3 (a) A set (each shaded square is a member of the set). (b) A structuring element. (c) The set padded with background elements to form a rectangular array and provide a background border. (d) Structuring element as a rectangular array. (e) Set processed by the structuring element.

In future illustrations, we add enough background points to form rectangular arrays, but let the padding be implicit when the meaning is clear in order to simplify the figures.

original set (this is analogous to padding for spatial correlation and convolution, as discussed in Section 3.4.2). In this case, the structuring element is of size 3×3 with the origin in the center, so a one-element border that encompasses the entire set is sufficient, as Fig. 9.3(c) shows. As in Fig. 9.2, the structuring element is filled with the smallest possible number of background elements necessary to make it into a rectangular array [Fig. 9.3(d)].

Suppose that we define an operation on set A using structuring element B, as follows: Create a new set by running B over A so that the origin of B visits every element of A. At each location of the origin of B, if B is completely contained in A, mark that location as a member of the new set (shown shaded); else mark it as not being a member of the new set (shown not shaded). Figure 9.3(e) shows the result of this operation. We see that, when the origin of B is on a border element of A, part of B ceases to be contained in A, thus eliminating the location on which B is centered as a possible member for the new set. The net result is that the boundary of the set is *eroded*, as Fig. 9.3(e) shows. When we use terminology such as "the structuring element is contained in the set," we mean specifically that the elements of A and B fully overlap. In other words, although we showed A and B as arrays containing both shaded and nonshaded elements, only the shaded elements of both sets are considered in determining whether or not B is contained in A. These concepts form the basis of the material in the next section, so it is important that you understand the ideas in Fig. 9.3 fully before proceeding.

9.2 Erosion and Dilation

We begin the discussion of morphology by studying two operations: *erosion* and *dilation*. These operations are fundamental to morphological processing. In fact, many of the morphological algorithms discussed in this chapter are based on these two primitive operations.

9.2.1 Erosion

With A and B as sets in Z^2, the erosion of A by B, denoted $A \ominus B$, is defined as

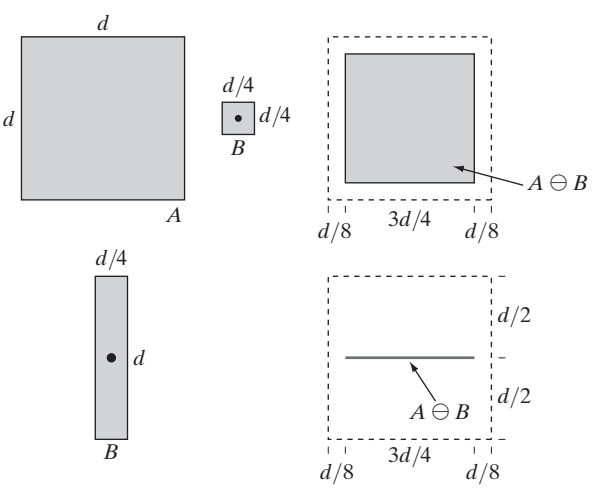
$$A \ominus B = \{z | (B)_z \subseteq A\} \tag{9.2-1}$$

In words, this equation indicates that the erosion of A by B is the set of all points z such that B, translated by z, is contained in A. In the following discussion, set B is assumed to be a structuring element. Equation (9.2-1) is the mathematical formulation of the example in Fig. 9.3(e), discussed at the end of the last section. Because the statement that B has to be contained in A is equivalent to B not sharing any common elements with the background, we can express erosion in the following equivalent form:

$$A \ominus B = \{z | (B)_z \cap A^c = \varnothing\} \tag{9.2-2}$$

where, as defined in Section 2.6.4, A^c is the complement of A and $\varnothing$ is the empty set.

Figure 9.4 shows an example of erosion. The elements of A and B are shown shaded and the background is white. The solid boundary in Fig. 9.4(c) is the limit beyond which further displacements of the origin of B would cause the structuring element to cease being completely contained in A. Thus, the locus of points (locations of the origin of B) within (and including) this boundary, constitutes the erosion of A by B. We show the erosion shaded in Fig. 9.4(c). Keep in mind that that erosion is simply the *set* of



a b c
d e

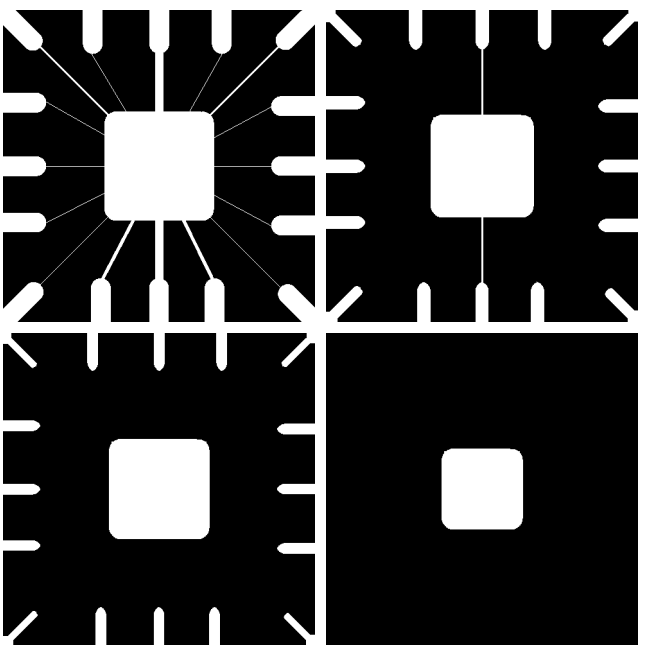
FIGURE 9.4 (a) Set A. (b) Square structuring element, B. (c) Erosion of A by B, shown shaded. (d) Elongated structuring element. (e) Erosion of A by B using this element. The dotted border in (c) and (e) is the boundary of set A, shown only for reference.

values of z that satisfy Eq. (9.2-1) or (9.2-2). The boundary of set A is shown dashed in Figs. 9.4(c) and (e) only as a reference; it is not part of the erosion operation. Figure 9.4(d) shows an elongated structuring element, and Fig. 9.4(e) shows the erosion of A by this element. Note that the original set was eroded to a line.

Equations (9.2-1) and (9.2-2) are not the only definitions of erosion (see Problems 9.9 and 9.10 for two additional, equivalent definitions.) However, these equations have the distinct advantage over other formulations in that they are more intuitive when the structuring element B is viewed as a spatial mask (see Section 3.4.1).

EXAMPLE 9.1: Using erosion to remove image components.

■ Suppose that we wish to remove the lines connecting the center region to the border pads in Fig. 9.5(a). Eroding the image with a square structuring element of size 11×11 whose components are all 1s removed most of the lines, as Fig. 9.5(b) shows. The reason the two vertical lines in the center were thinned but not removed completely is that their width is greater than 11 pixels. Changing the SE size to 15×15 and eroding the original image again did remove all the connecting lines, as Fig. 9.5(c) shows (an alternate approach would have been to erode the image in Fig. 9.5(b) again using the same 11×11 SE). Increasing the size of the structuring element even more would eliminate larger components. For example, the border pads can be removed with a structuring element of size 45×45, as Fig. 9.5(d) shows.



a b
c d

FIGURE 9.5 Using erosion to remove image components. (a) A 486×486 binary image of a wirebond mask. (b)–(d) Image eroded using square structuring elements of sizes 11×11, 15×15, and 45×45, respectively. The elements of the SEs were all 1s.

We see from this example that erosion shrinks or thins objects in a binary image. In fact, we can view erosion as a *morphological filtering* operation in which image details smaller than the structuring element are filtered (removed) from the image. In Fig. 9.5, erosion performed the function of a "line filter." We return to the concept of a morphological filter in Sections 9.3 and 9.6.3. ■

9.2.2 Dilation

With A and B as sets in Z^2, the *dilation* of A by B, denoted $A \oplus B$, is defined as

$$A \oplus B = \left\{ z | (\hat{B})_z \cap A \neq \varnothing \right\} \tag{9.2-3}$$

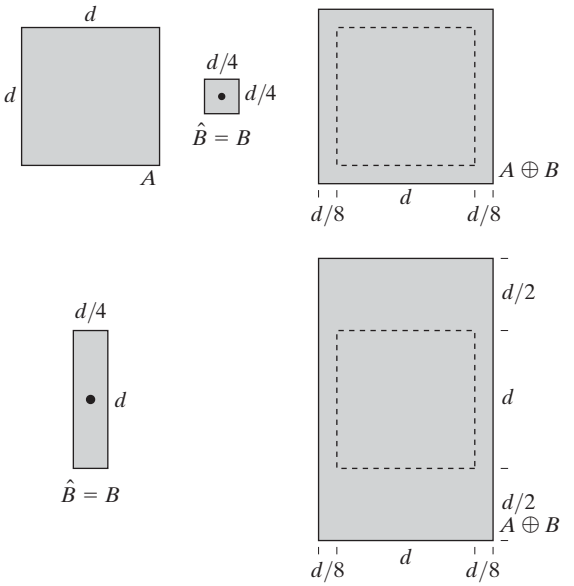
This equation is based on reflecting B about its origin, and shifting this reflection by z (see Fig. 9.1). The dilation of A by B then is the set of all displacements, z, such that $\hat{B}$ and A overlap by at least one element. Based on this interpretation, Eq. (9.2-3) can be written equivalently as

$$A \oplus B = \left\{ z | [(\hat{B})_z \cap A] \subseteq A \right\} \tag{9.2-4}$$

As before, we assume that B is a structuring element and A is the set (image objects) to be dilated.

Equations (9.2-3) and (9.2-4) are not the only definitions of dilation currently in use (see Problems 9.11 and 9.12 for two different, yet equivalent, definitions). However, the preceding definitions have a distinct advantage over other formulations in that they are more intuitive when the structuring element B is viewed as a convolution mask. The basic process of flipping (rotating) B about its origin and then successively displacing it so that it slides over set (image) A is analogous to spatial convolution, as introduced in Section 3.4.2. Keep in mind, however, that dilation is based on set operations and therefore is a nonlinear operation, whereas convolution is a linear operation.

Unlike erosion, which is a shrinking or thinning operation, dilation "grows" or "thickens" objects in a binary image. The specific manner and extent of this thickening is controlled by the shape of the structuring element used. Figure 9.6(a) shows the same set used in Fig. 9.4, and Fig. 9.6(b) shows a structuring element (in this case $\hat{B} = B$ because the SE is symmetric about its origin). The dashed line in Fig. 9.6(c) shows the original set for reference, and the solid line shows the limit beyond which any further displacements of the origin of $\hat{B}$ by z would cause the intersection of $\hat{B}$ and A to be empty. Therefore, all points on and inside this boundary constitute the dilation of A by B. Figure 9.6(d) shows a structuring element designed to achieve more dilation vertically than horizontally, and Fig. 9.6(e) shows the dilation achieved with this element.



a b c
d e

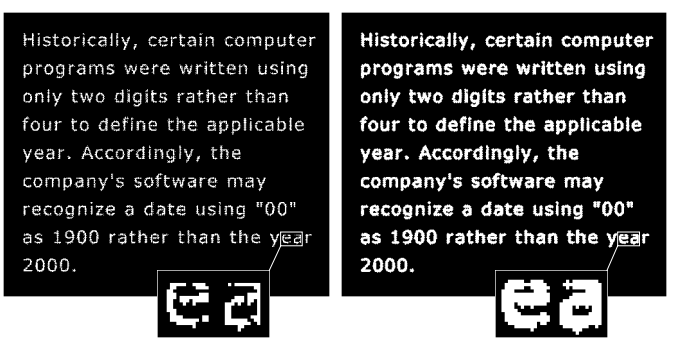
FIGURE 9.6
(a) Set A. (b) Square structuring element (the dot denotes the origin). (c) Dilation of A by B, shown shaded. (d) Elongated structuring element. (e) Dilation of A using this element. The dotted border in (c) and (e) is the boundary of set A, shown only for reference

EXAMPLE 9.2: An illustration of dilation.

■ One of the simplest applications of dilation is for bridging gaps. Figure 9.7(a) shows the same image with broken characters that we studied in Fig. 4.49 in connection with lowpass filtering. The maximum length of the breaks is known to be two pixels. Figure 9.7(b) shows a structuring element that can be used for repairing the gaps (note that instead of shading, we used 1s to denote the elements of the SE and 0s for the background; this is because the SE is now being treated as a subimage and not as a graphic). Figure 9.7(c) shows the result of dilating the original image with this structuring element. The gaps were bridged. One immediate advantage of the morphological approach over the lowpass filtering method we used to bridge the gaps in Fig. 4.49 is

a c
b

FIGURE 9.7
(a) Sample text of poor resolution with broken characters (see magnified view). (b) Structuring element. (c) Dilation of (a) by (b). Broken segments were joined.



0	1	0
1	1	1
0	1	0

that the morphological method resulted directly in a binary image. Lowpass filtering, on the other hand, started with a binary image and produced a gray-scale image, which would require a pass with a thresholding function to convert it back to binary form. ■

9.2.3 Duality

Erosion and dilation are duals of each other with respect to set complementation and reflection. That is,

$$(A \ominus B)^c = A^c \oplus \hat{B} \tag{9.2-5}$$

and

$$(A \oplus B)^c = A^c \ominus \hat{B} \tag{9.2-6}$$

Equation (9.2-5) indicates that erosion of A by B is the complement of the dilation of A^c by $\hat{B}$, and vice versa. The duality property is useful particularly when the structuring element is symmetric with respect to its origin (as often is the case), so that $\hat{B} = B$. Then, we can obtain the erosion of an image by B simply by dilating its background (i.e., dilating A^c) with the same structuring element and complementing the result. Similar comments apply to Eq. (9.2-6).

We proceed to prove formally the validity of Eq. (9.2-5) in order to illustrate a typical approach for establishing the validity of morphological expressions. Starting with the definition of erosion, it follows that

$$(A \ominus B)^c = \left\{z | (B)_z \subseteq A\right\}^c$$

If set $(B)_z$ is contained in A, then $(B)_z \cap A^c = \varnothing$, in which case the preceding expression becomes

$$(A \ominus B)^c = \left\{z | (B)_z \cap A^c = \varnothing\right\}^c$$

But the *complement* of the set of z's that satisfy $(B)_z \cap A^c = \varnothing$ is the set of z's such that $(B)_z \cap A^c \neq \varnothing$. Therefore,

$$\begin{aligned}(A \ominus B)^c &= \left\{z | (B)_z \cap A^c \neq \varnothing\right\} \\ &= A^c \oplus \hat{B}\end{aligned}$$

where the last step follows from Eq. (9.2-3). This concludes the proof. A similar line of reasoning can be used to prove Eq. (9.2-6) (see Problem 9.13).

9.3 Opening and Closing

As you have seen, dilation expands the components of an image and erosion shrinks them. In this section we discuss two other important morphological operations: opening and closing. *Opening* generally smoothes the contour of an object, breaks narrow isthmuses, and eliminates thin protrusions. *Closing* also tends to smooth sections of contours but, as opposed to opening, it generally fuses narrow breaks and long thin gulfs, eliminates small holes, and fills gaps in the contour.

The *opening* of set A by structuring element B, denoted $A \circ B$, is defined as

$$A \circ B = (A \ominus B) \oplus B \tag{9.3-1}$$

Thus, the opening A by B is the erosion of A by B, followed by a dilation of the result by B.

Similarly, the *closing* of set A by structuring element B, denoted $A \bullet B$, is defined as

$$A \bullet B = (A \oplus B) \ominus B \tag{9.3-2}$$

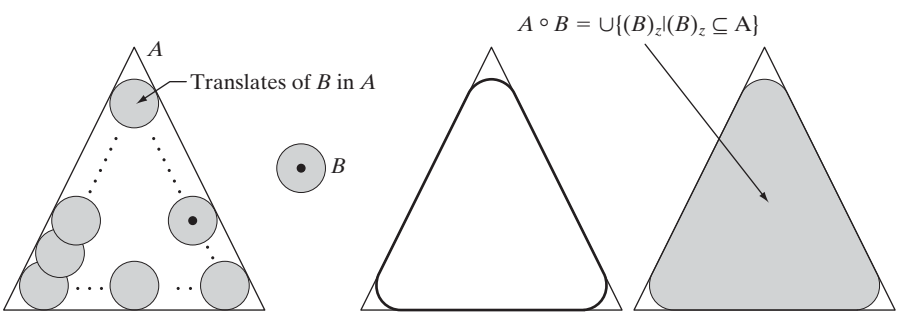
which says that the closing of A by B is simply the dilation of A by B, followed by the erosion of the result by B.

The opening operation has a simple geometric interpretation (Fig. 9.8). Suppose that we view the structuring element B as a (flat) "rolling ball." The *boundary* of $A \circ B$ is then established by the points in B that reach the *farthest* into the boundary of A as B is rolled around the *inside* of this boundary. This geometric *fitting* property of the opening operation leads to a set-theoretic formulation, which states that the opening of A by B is obtained by taking the union of all translates of B that fit into A. That is, opening can be expressed as a fitting process such that

$$A \circ B = \bigcup \{(B)_z | (B)_z \subseteq A\} \tag{9.3-3}$$

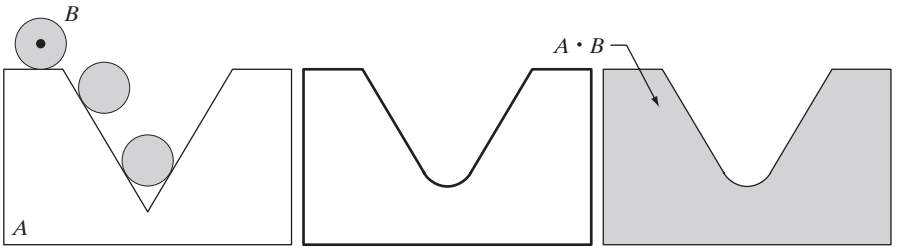
where $\cup\{\cdot\}$ denotes the union of all the sets inside the braces.

Closing has a similar geometric interpretation, except that now we roll B on the outside of the boundary (Fig. 9.9). As discussed below, opening and closing are duals of each other, so having to roll the ball on the outside is not unexpected. Geometrically, a point w is an element of $A \bullet B$ if and only if $(B)_z \cap A \neq \varnothing$ for any translate of $(B)_z$ that contains w. Figure 9.9 illustrates the basic geometrical properties of closing.



a b c d

FIGURE 9.8 (a) Structuring element B "rolling" along the inner boundary of A (the dot indicates the origin of B). (b) Structuring element. (c) The heavy line is the outer boundary of the opening. (d) Complete opening (shaded). We did not shade A in (a) for clarity.

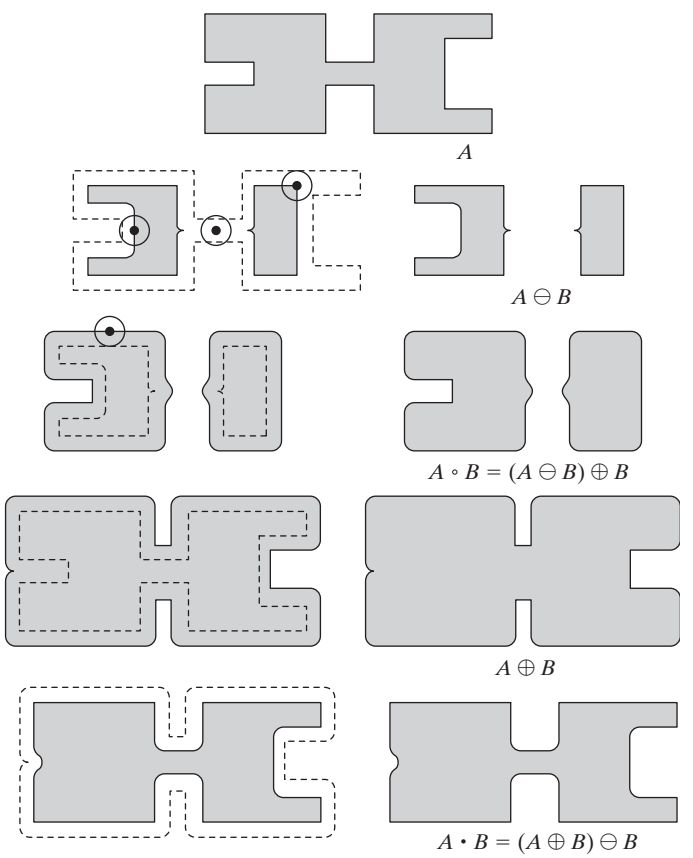


a b c

FIGURE 9.9 (a) Structuring element B "rolling" on the outer boundary of set A. (b) The heavy line is the outer boundary of the closing. (c) Complete closing (shaded). We did not shade A in (a) for clarity.

■ Figure 9.10 further illustrates the opening and closing operations. Figure 9.10(a) shows a set A, and Fig. 9.10(b) shows various positions of a disk structuring element during the erosion process. When completed, this process resulted in the disjoint figure in Fig. 9.10(c). Note the elimination of the bridge between the two main sections. Its width was thin in relation to the diameter of

EXAMPLE 9.3: A simple illustration of morphological opening and closing.



a
b c
d e
f g
h i

FIGURE 9.10 Morphological opening and closing. The structuring element is the small circle shown in various positions in (b). The SE was not shaded here for clarity. The dark dot is the center of the structuring element.

the structuring element; that is, the structuring element could not be completely contained in this part of the set, thus violating the conditions of Eq. (9.2-1). The same was true of the two rightmost members of the object. Protruding elements where the disk did not fit were eliminated. Figure 9.10(d) shows the process of dilating the eroded set, and Fig. 9.10(e) shows the final result of opening. Note that outward pointing corners were rounded, whereas inward pointing corners were not affected.

Similarly, Figs. 9.10(f) through (i) show the results of closing A with the same structuring element. We note that the inward pointing corners were rounded, whereas the outward pointing corners remained unchanged. The leftmost intrusion on the boundary of A was reduced in size significantly, because the disk did not fit there. Note also the smoothing that resulted in parts of the object from both opening and closing the set A with a circular structuring element. ■

As in the case with dilation and erosion, opening and closing are duals of each other with respect to set complementation and reflection. That is,

$$(A \bullet B)^c = (A^c \circ \hat{B}) \tag{9.3-4}$$

and

$$(A \circ B)^c = (A^c \bullet \hat{B}) \tag{9.3-5}$$

We leave the proof of this result as an exercise (Problem 9.14).

The opening operation satisfies the following properties:

(a) $A \circ B$ is a subset (subimage) of A.
(b) If C is a subset of D, then $C \circ B$ is a subset of $D \circ B$.
(c) $(A \circ B) \circ B = A \circ B$.

Similarly, the closing operation satisfies the following properties:

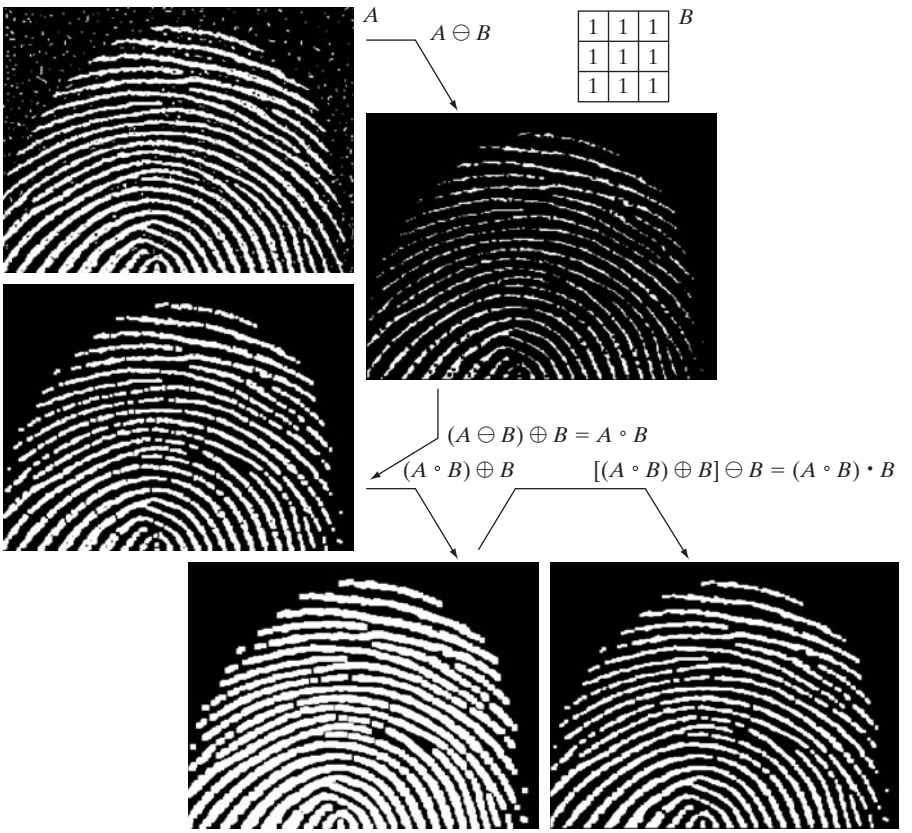
(a) A is a subset (subimage) of $A \bullet B$.
(b) If C is a subset of D, then $C \bullet B$ is a subset of $D \bullet B$.
(c) $(A \bullet B) \bullet B = A \bullet B$.

Note from condition (c) in both cases that multiple openings or closings of a set have no effect after the operator has been applied once.

EXAMPLE 9.4: Use of opening and closing for morphological filtering.

■ Morphological operations can be used to construct filters similar in concept to the spatial filters discussed in Chapter 3. The binary image in Fig. 9.11(a) shows a section of a fingerprint corrupted by noise. Here the noise manifests itself as random light elements on a dark background and as dark elements on the light components of the fingerprint. The objective is to eliminate the noise and its effects on the print while distorting it as little as possible. A morphological filter consisting of opening followed by closing can be used to accomplish this objective.

Figure 9.11(b) shows the structuring element used. The rest of Fig. 9.11 shows a step-by-step sequence of the filtering operation. Figure 9.11(c) is the



a b
d c
e f

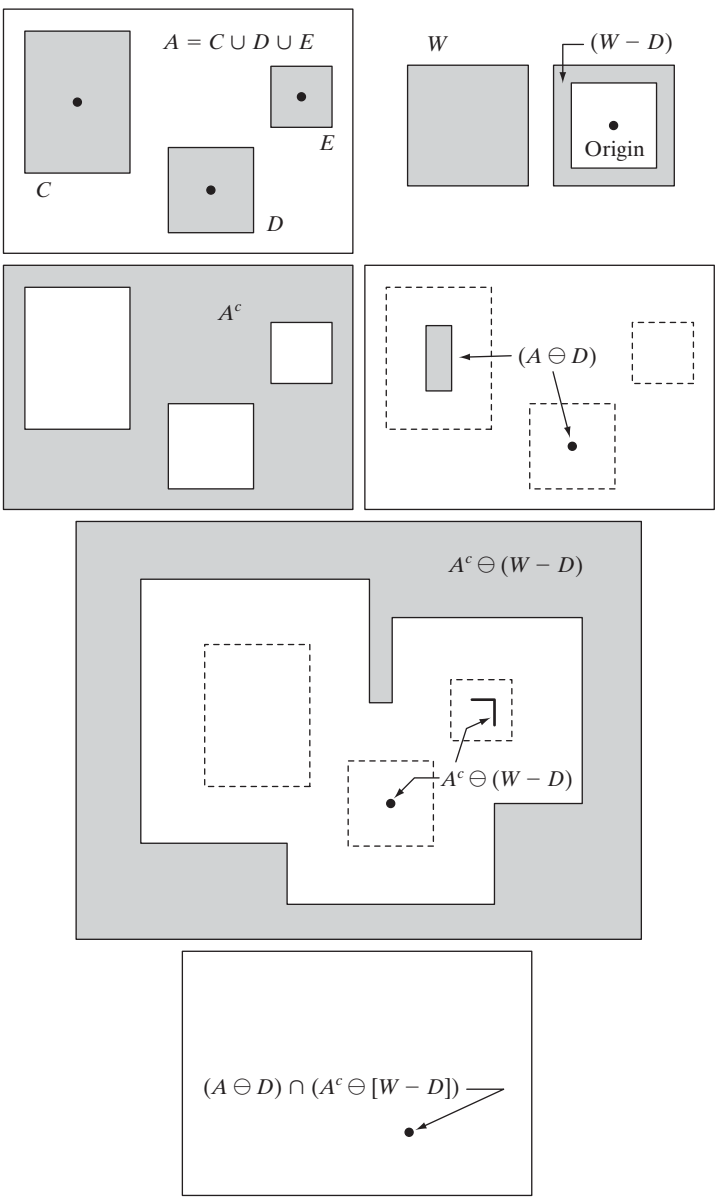
FIGURE 9.11
(a) Noisy image. (b) Structuring element. (c) Eroded image. (d) Opening of A. (e) Dilation of the opening. (f) Closing of the opening. (Original image courtesy of the National Institute of Standards and Technology.)

result of eroding A with the structuring element. The background noise was completely eliminated in the erosion stage of opening because in this case all noise components are smaller than the structuring element. The size of the noise elements (dark spots) contained within the fingerprint actually increased in size. The reason is that these elements are inner boundaries that increase in size as the object is eroded. This enlargement is countered by performing dilation on Fig. 9.11(c). Figure 9.11(d) shows the result. The noise components contained in the fingerprint were reduced in size or deleted completely.

The two operations just described constitute the opening of A by B. We note in Fig. 9.11(d) that the net effect of opening was to eliminate virtually all noise components in both the background and the fingerprint itself. However, new gaps between the fingerprint ridges were created. To counter this undesirable effect, we perform a dilation on the opening, as shown in Fig. 9.11(e). Most of the breaks were restored, but the ridges were thickened, a condition that can be remedied by erosion. The result, shown in Fig. 9.11(f), constitutes the closing of the opening of Fig. 9.11(d). This final result is remarkably clean of noise specks, but it has the disadvantage that some of the print ridges were not fully repaired, and thus contain breaks. This is not totally unexpected, because no conditions were built into the procedure for maintaining connectivity (we discuss this issue again in Example 9.8 and demonstrate ways to address it in Section 11.1.7). ■

9.4 The Hit-or-Miss Transformation

The morphological hit-or-miss transform is a basic tool for shape detection. We introduce this concept with the aid of Fig. 9.12, which shows a set A consisting of three shapes (subsets), denoted C, D, and E. The shading in Figs. 9.12(a) through (c) indicates the original sets, whereas the shading in Figs. 9.12(d) and (e) indicates the result of morphological operations. The objective is to find the location of one of the shapes, say, D.



a b
c d
e
f

FIGURE 9.12
(a) Set A. (b) A window, W, and the local background of D with respect to W, $(W - D)$. (c) Complement of A. (d) Erosion of A by D. (e) Erosion of A^c by $(W - D)$. (f) Intersection of (d) and (e), showing the location of the origin of D, as desired. The dots indicate the origins of C, D, and E.

Let the origin of each shape be located at its center of gravity. Let D be enclosed by a small window, W. The *local background* of D with respect to W is defined as the set difference $(W - D)$, as shown in Fig. 9.12(b). Figure 9.12(c) shows the complement of A, which is needed later. Figure 9.12(d) shows the erosion of A by D (the dashed lines are included for reference). Recall that the erosion of A by D is the set of locations of the *origin* of D, such that D is completely contained in A. Interpreted another way, $A \ominus D$ may be viewed geometrically as the set of all locations of the origin of D at which D found a match (hit) in A. Keep in mind that in Fig. 9.12 A consists only of the three *disjoint* sets C, D, and E.

Figure 9.12(e) shows the erosion of the complement of A by the local background set $(W - D)$. The outer shaded region in Fig. 9.12(e) is part of the erosion. We note from Figs. 9.12(d) and (e) that the set of locations for which D *exactly* fits inside A is the *intersection* of the erosion of A by D and the erosion of A^c by $(W - D)$ as shown in Fig. 9.12(f). This intersection is precisely the location sought. In other words, if B denotes the set composed of D and its background, the match (or set of matches) of B in A, denoted $A \circledast B$, is

$$A \circledast B = (A \ominus D) \cap \left[A^c \ominus (W - D)\right] \tag{9.4-1}$$

We can generalize the notation somewhat by letting $B = (B_1, B_2)$, where B_1 is the set formed from elements of B associated with an object and B_2 is the set of elements of B associated with the corresponding background. From the preceding discussion, $B_1 = D$ and $B_2 = (W - D)$. With this notation, Eq. (9.4-1) becomes

$$A \circledast B = (A \ominus B_1) \cap (A^c \ominus B_2) \tag{9.4-2}$$

Thus, set $A \circledast B$ contains all the (origin) points at which, simultaneously, B_1 found a match ("hit") in A and B_2 found a match in A^c. By using the definition of set differences given in Eq. (2.6-19) and the dual relationship between erosion and dilation given in Eq. (9.2-5), we can write Eq. (9.4-2) as

$$A \circledast B = (A \ominus B_1) - (A \oplus \hat{B}_2) \tag{9.4-3}$$

However, Eq. (9.4-2) is considerably more intuitive. We refer to any of the preceding three equations as the *morphological hit-or-miss transform*.

The reason for using a structuring element B_1 associated with objects and an element B_2 associated with the background is based on an assumed definition that two or more objects are distinct only if they form disjoint (disconnected) sets. This is guaranteed by requiring that each object have at least a one-pixel-thick background around it. In some applications, we may be interested in detecting certain patterns (combinations) of 1s and 0s within a set, in which case a background is not required. In such instances, the hit-or-miss transform reduces to simple erosion. As indicated previously, erosion is still a set of matches, but without the additional requirement of a background match for detecting individual objects. This simplified pattern detection scheme is used in some of the algorithms developed in the following section.

9.5 Some Basic Morphological Algorithms

With the preceding discussion as foundation, we are now ready to consider some practical uses of morphology. When dealing with binary images, one of the principal applications of morphology is in extracting image components that are useful in the representation and description of shape. In particular, we consider morphological algorithms for extracting boundaries, connected components, the convex hull, and the skeleton of a region. We also develop several methods (for region filling, thinning, thickening, and pruning) that are used frequently in conjunction with these algorithms as pre- or post-processing steps. We make extensive use in this section of "mini-images," designed to clarify the mechanics of each morphological process as we introduce it. These images are shown graphically with 1s shaded and 0s in white.

9.5.1 Boundary Extraction

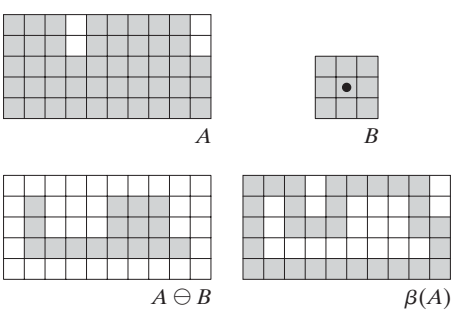
The boundary of a set A, denoted by $\beta(A)$, can be obtained by first eroding A by B and then performing the set difference between A and its erosion. That is,

$$\beta(A) = A - (A \ominus B) \tag{9.5-1}$$

where B is a suitable structuring element.

Figure 9.13 illustrates the mechanics of boundary extraction. It shows a simple binary object, a structuring element B, and the result of using Eq. (9.5-1). Although the structuring element in Fig. 9.13(b) is among the most frequently used, it is by no means unique. For example, using a 5×5 structuring element of 1s would result in a boundary between 2 and 3 pixels thick.

From this point on, we do not show border padding explicitly.



a b
c d

FIGURE 9.13 (a) Set A. (b) Structuring element B. (c) A eroded by B. (d) Boundary, given by the set difference between A and its erosion.

a b

FIGURE 9.14
(a) A simple binary image, with 1s represented in white. (b) Result of using Eq. (9.5-1) with the structuring element in Fig. 9.13(b).

EXAMPLE 9.5: Boundary extraction by morphological processing.

■ Figure 9.14 further illustrates the use of Eq. (9.5-1) with a 3×3 structuring element of 1s. As for all binary images in this chapter, binary 1s are shown in white and 0s in black, so the elements of the structuring element, which are 1s, also are treated as white. Because of the size of the structuring element used, the boundary in Fig. 9.14(b) is one pixel thick. ■

9.5.2 Hole Filling

A *hole* may be defined as a background region surrounded by a connected border of foreground pixels. In this section, we develop an algorithm based on set dilation, complementation, and intersection for filling holes in an image. Let A denote a set whose elements are 8-connected boundaries, each boundary enclosing a background region (i.e., a hole). Given a point in each hole, the objective is to fill all the holes with 1s.

We begin by forming an array, X_0, of 0s (the same size as the array containing A), except at the locations in X_0 corresponding to the given point in each hole, which we set to 1. Then, the following procedure fills all the holes with 1s:

$$X_k = (X_{k-1} \oplus B) \cap A^c \qquad k = 1, 2, 3, \ldots \tag{9.5-2}$$

where B is the symmetric structuring element in Fig. 9.15(c). The algorithm terminates at iteration step k if $X_k = X_{k-1}$. The set X_k then contains all the filled holes. The set union of X_k and A contains all the filled holes and their boundaries.

The dilation in Eq. (9.5-2) would fill the entire area if left unchecked. However, the intersection at each step with A^c limits the result to inside the region of interest. This is our first example of how a morphological process can be *conditioned* to meet a desired property. In the current application, it is appropriately called *conditional dilation*. The rest of Fig. 9.15 illustrates further the mechanics of Eq. (9.5-2). Although this example only has one hole, the concept clearly applies to any finite number of holes, assuming that a point inside each hole region is given.

a b c
d e f
g h i

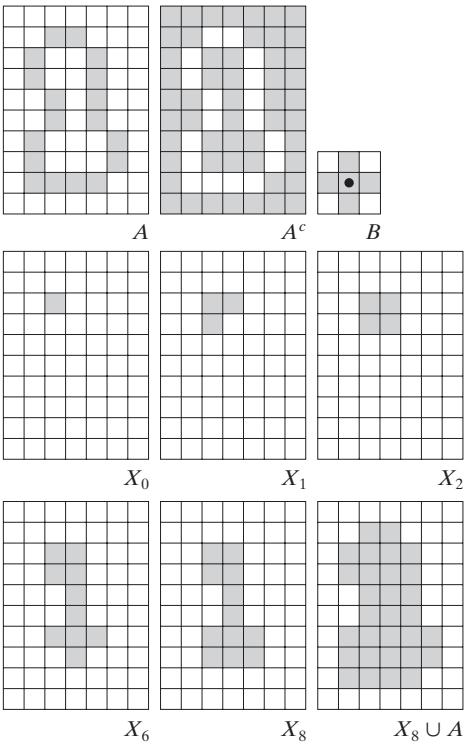
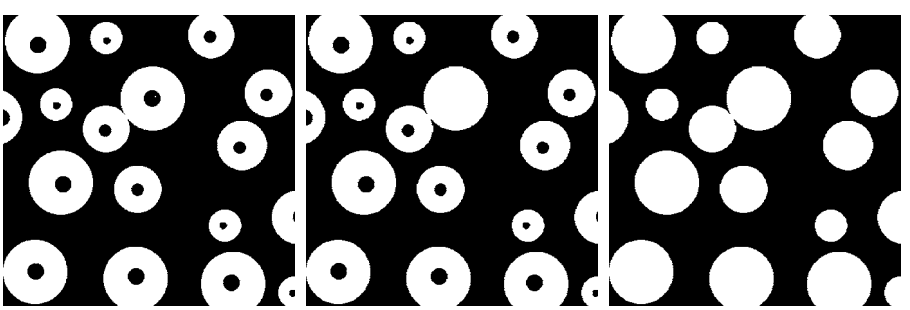


FIGURE 9.15 Hole filling. (a) Set A (shown shaded). (b) Complement of A. (c) Structuring element B. (d) Initial point inside the boundary. (e)–(h) Various steps of Eq. (9.5-2). (i) Final result [union of (a) and (h)].

EXAMPLE 9.6: Morphological hole filling.

■ Figure 9.16(a) shows an image composed of white circles with black inner spots. An image such as this might result from thresholding into two levels a scene containing polished spheres (e.g., ball bearings). The dark spots inside the spheres could be the result of reflections. The objective is to eliminate the reflections by hole filling. Figure 9.16(a) shows one point selected inside one of the spheres, and Fig. 9.16(b) shows the result of filling that component. Finally,



a b c

FIGURE 9.16 (a) Binary image (the white dot inside one of the regions is the starting point for the hole-filling algorithm). (b) Result of filling that region. (c) Result of filling all holes.

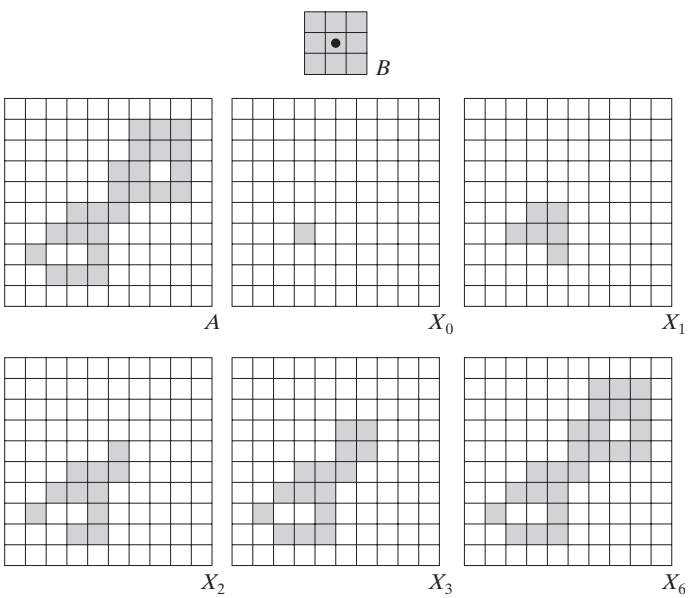
Fig. 9.16(c) shows the result of filling all the spheres. Because it must be known whether black points are background points or sphere inner points, fully automating this procedure requires that additional "intelligence" be built into the algorithm. We give a fully automatic approach in Section 9.5.9 based on morphological reconstruction. (See also Problem 9.23.) ■

9.5.3 Extraction of Connected Components

The concepts of connectivity and connected components were introduced in Section 2.5.2. Extraction of connected components from a binary image is central to many automated image analysis applications. Let A be a set containing one or more connected components, and form an array X_0 (of the same size as the array containing A) whose elements are 0s (background values), except at each location known to correspond to a point in each connected component in A, which we set to 1 (foreground value). The objective is to start with X_0 and find all the connected components. The following iterative procedure accomplishes this objective:

$$X_k = (X_{k-1} \oplus B) \cap A \qquad k = 1, 2, 3, \ldots \tag{9.5-3}$$

where B is a suitable structuring element (as in Fig. 9.17). The procedure terminates when $X_k = X_{k-1}$, with X_k containing all the connected components



a
b c d
e f g

FIGURE 9.17 Extracting connected components. (a) Structuring element. (b) Array containing a set with one connected component. (c) Initial array containing a 1 in the region of the connected component. (d)–(g) Various steps in the iteration of Eq. (9.5-3).

of the input image. Note the similarity in Eqs. (9.5-3) and (9.5-2), the only difference being the use of A as opposed to A^c. This is not surprising, because here we are looking for foreground points, while the objective in Section 9.5.2 was to find background points.

Figure 9.17 illustrates the mechanics of Eq. (9.5-3), with convergence being achieved for $k = 6$. Note that the shape of the structuring element used is based on 8-connectivity between pixels. If we had used the SE in Fig. 9.15, which is based on 4-connectivity, the leftmost element of the connected component toward the bottom of the image would not have been detected because it is 8-connected to the rest of the figure. As in the hole-filling algorithm, Eq. (9.5-3) is applicable to any finite number of connected components contained in A, assuming that a point is known in each.

See Problem 9.24 for an algorithm that does not require that a point in each connected component be known a priori.

EXAMPLE 9.7: Using connected components to detect foreign objects in packaged food.

■ Connected components are used frequently for automated inspection. Figure 9.18(a) shows an X-ray image of a chicken breast that contains bone fragments. It is of considerable interest to be able to detect such objects in processed food before packaging and/or shipping. In this particular case, the density of the bones is such that their nominal intensity values are different from the background. This makes extraction of the bones from the background

a
b
c d

FIGURE 9.18 (a) X-ray image of chicken filet with bone fragments. (b) Thresholded image. (c) Image eroded with a 5×5 structuring element of 1s. (d) Number of pixels in the connected components of (c). (Image courtesy of NTB Elektronische Geraete GmbH, Diepholz, Germany, www.ntbxray.com.)

Connected component	No. of pixels in connected comp
01	11
02	9
03	9
04	39
05	133
06	1
07	1
08	743
09	7
10	11
11	11
12	9
13	9
14	674
15	85

a simple matter by using a single threshold (thresholding was introduced in Section 3.1 and is discussed in more detail in Section 10.3). The result is the binary image in Fig. 9.18(b).

The most significant feature in this figure is the fact that the points that remain are clustered into objects (bones), rather than being isolated, irrelevant points. We can make sure that only objects of "significant" size remain by eroding the thresholded image. In this example, we define as significant any object that remains after erosion with a 5×5 structuring element of 1s. The result of erosion is shown in Fig. 9.18(c). The next step is to analyze the size of the objects that remain. We label (identify) these objects by extracting the connected components in the image. The table in Fig. 9.18(d) lists the results of the extraction. There are a total of 15 connected components, with four of them being dominant in size. This is enough to determine that significant undesirable objects are contained in the original image. If needed, further characterization (such as shape) is possible using the techniques discussed in Chapter 11. ■

9.5.4 Convex Hull

A set A is said to be *convex* if the straight line segment joining any two points in A lies entirely within A. The *convex hull* H of an arbitrary set S is the smallest convex set containing S. The set difference $H - S$ is called the *convex deficiency* of S. As discussed in more detail in Sections 11.1.6 and 11.3.2, the convex hull and convex deficiency are useful for object description. Here, we present a simple morphological algorithm for obtaining the convex hull, $C(A)$, of a set A.

Let B^i, $i = 1, 2, 3, 4$, represent the four structuring elements in Fig. 9.19(a). The procedure consists of implementing the equation:

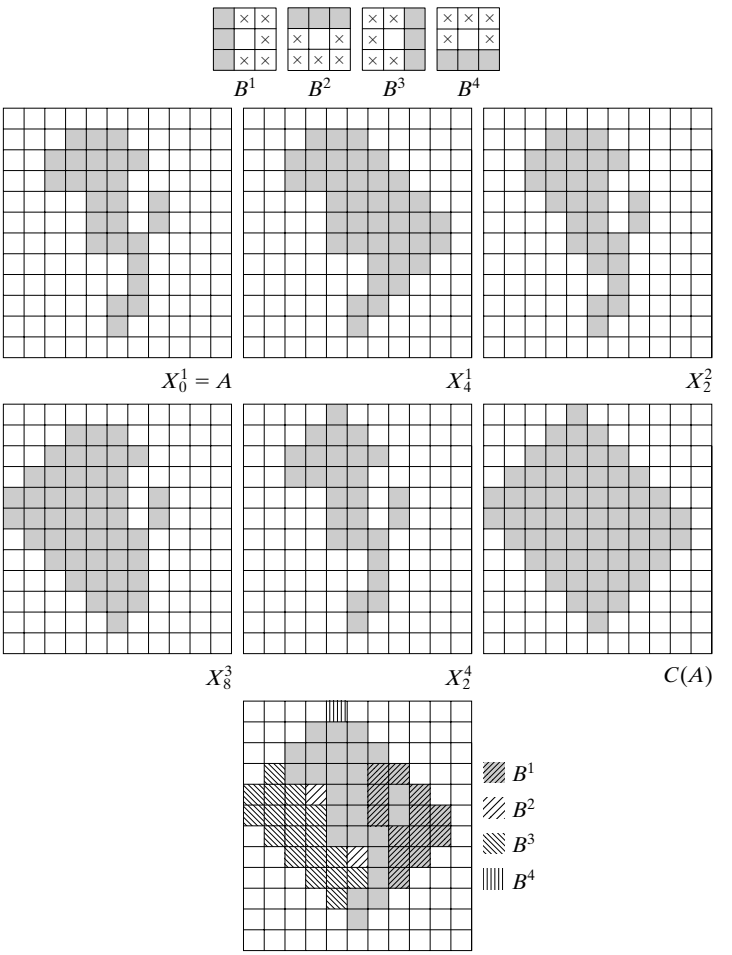
$$X_k^i = (X_{k-1} \circledast B^i) \cup A \quad i = 1, 2, 3, 4 \quad \text{and} \quad k = 1, 2, 3, \ldots \tag{9.5-4}$$

with $X_0^i = A$. When the procedure converges (i.e., when $X_k^i = X_{k-1}^i$), we let $D^i = X_k^i$. Then the convex hull of A is

$$C(A) = \bigcup_{i=1}^{4} D^i \tag{9.5-5}$$

In other words, the method consists of iteratively applying the hit-or-miss transform to A with B^1; when no further changes occur, we perform the union with A and call the result D^1. The procedure is repeated with B^2 (applied to A) until no further changes occur, and so on. The union of the four resulting Ds constitutes the convex hull of A. Note that we are using the simplified implementation of the hit-or-miss transform in which no background match is required, as discussed at the end of Section 9.4.

Figure 9.19 illustrates the procedure given in Eqs. (9.5-4) and (9.5-5). Figure 9.19(a) shows the structuring elements used to extract the convex hull. The origin of each element is at its center. The $\times$ entries indicate "don't care" conditions. This means that a structuring element is said to have found a match



a
b c d
e f g
h

FIGURE 9.19
(a) Structuring elements. (b) Set A. (c)–(f) Results of convergence with the structuring elements shown in (a). (g) Convex hull. (h) Convex hull showing the contribution of each structuring element.

in A if the 3×3 region of A under the structuring element mask at that location matches the pattern of the mask. For a particular mask, a pattern match occurs when the center of the 3×3 region in A is 0, and the three pixels under the shaded mask elements are 1. The values of the other pixels in the 3×3 region do not matter. Also, with respect to the notation in Fig. 9.19(a), B^i is a clockwise rotation of B^{i-1} by 90°.

Figure 9.19(b) shows a set A for which the convex hull is sought. Starting with $X_0^1 = A$ resulted in the set in Fig. 9.19(c) after four iterations of Eq. (9.5-4). Then, letting $X_0^2 = A$ and again using Eq. (9.5-4) resulted in the set in Fig. 9.19(d) (convergence was achieved in only two steps in this case). The next two results were obtained in the same way. Finally, forming the union of the sets in Figs. 9.19(c), (d), (e), and (f) resulted in the convex hull shown in Fig. 9.19(g). The contribution of each structuring element is highlighted in the composite set shown in Fig. 9.19(h).

One obvious shortcoming of the procedure just outlined is that the convex hull can grow beyond the minimum dimensions required to guarantee

convexity. One simple approach to reduce this effect is to limit growth so that it does not extend past the vertical and horizontal dimensions of the original set of points. Imposing this limitation on the example in Fig. 9.19 resulted in the image shown in Fig. 9.20. Boundaries of greater complexity can be used to limit growth even further in images with more detail. For example, we could use the maximum dimensions of the original set of points along the vertical, horizontal, and diagonal directions. The price paid for refinements such as this is additional complexity and increased computational requirements of the algorithm.

9.5.5 Thinning

The thinning of a set A by a structuring element B, denoted $A \otimes B$, can be defined in terms of the hit-or-miss transform:

$$\begin{aligned} A \otimes B &= A - (A \circledast B) \\ &= A \cap (A \circledast B)^c \end{aligned} \tag{9.5-6}$$

As in the previous section, we are interested only in pattern matching with the structuring elements, so no background operation is required in the hit-or-miss transform. A more useful expression for thinning A symmetrically is based on a *sequence* of structuring elements:

$$\{B\} = \{B^1, B^2, B^3, \ldots, B^n\} \tag{9.5-7}$$

where B^i is a rotated version of B^{i-1}. Using this concept, we now define thinning by a sequence of structuring elements as

$$A \otimes \{B\} = ((\ldots((A \otimes B^1) \otimes B^2)\ldots) \otimes B^n) \tag{9.5-8}$$

The process is to thin A by *one pass* with B^1, then thin the result with one pass of B^2, and so on, until A is thinned with one pass of B^n. The entire process is repeated until no further changes occur. Each individual thinning pass is performed using Eq. (9.5-6).

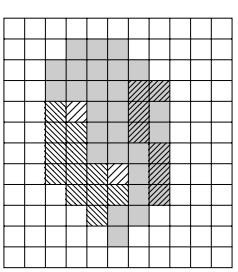


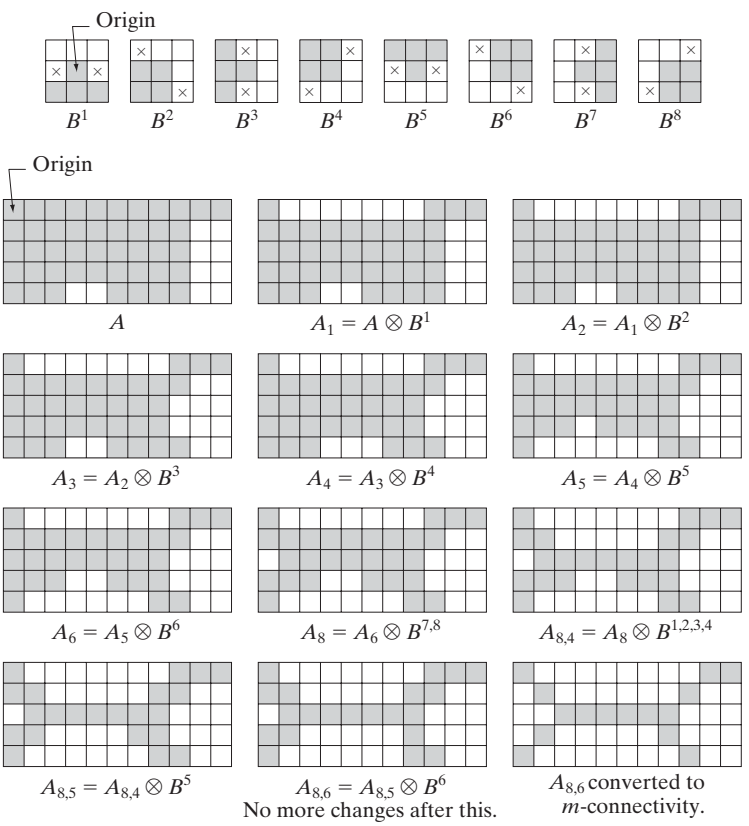
FIGURE 9.20 Result of limiting growth of the convex hull algorithm to the maximum dimensions of the original set of points along the vertical and horizontal directions.

Figure 9.21(a) shows a set of structuring elements commonly used for thinning, and Fig. 9.21(b) shows a set A to be thinned by using the procedure just discussed. Figure 9.21(c) shows the result of thinning after one pass of A with B^1, and Figs. 9.21(d) through (k) show the results of passes with the other structuring elements. Convergence was achieved after the second pass of B^6. Figure 9.21(l) shows the thinned result. Finally, Fig. 9.21(m) shows the thinned set converted to m-connectivity (see Section 2.5.2) to eliminate multiple paths.

9.5.6 Thickening

Thickening is the morphological dual of thinning and is defined by the expression

$$A \odot B = A \cup (A \circledast B) \tag{9.5-9}$$



a
b c d
e f g
h i j
k l m

FIGURE 9.21 (a) Sequence of rotated structuring elements used for thinning. (b) Set A. (c) Result of thinning with the first element. (d)–(i) Results of thinning with the next seven elements (there was no change between the seventh and eighth elements). (j) Result of using the first four elements again. (l) Result after convergence. (m) Conversion to m-connectivity.

where B is a structuring element suitable for thickening. As in thinning, thickening can be defined as a sequential operation:

$$A \odot \{B\} = ((\ldots((A \odot B^1) \odot B^2)\ldots) \odot B^n) \tag{9.5-10}$$

The structuring elements used for thickening have the same form as those shown in Fig. 9.21(a), but with all 1s and 0s interchanged. However, a separate algorithm for thickening is seldom used in practice. Instead, the usual procedure is to thin the background of the set in question and then complement the result. In other words, to thicken a set A, we form $C = A^c$, thin C, and then form C^c. Figure 9.22 illustrates this procedure.

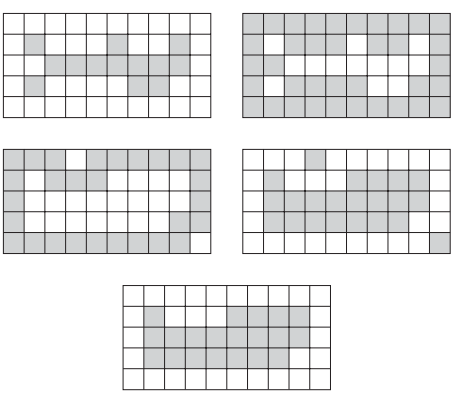
Depending on the nature of A, this procedure can result in disconnected points, as Fig. 9.22(d) shows. Hence thickening by this method usually is followed by postprocessing to remove disconnected points. Note from Fig. 9.22(c) that the thinned background forms a boundary for the thickening process. This useful feature is not present in the direct implementation of thickening using Eq. (9.5-10), and it is one of the principal reasons for using background thinning to accomplish thickening.

9.5.7 Skeletons

As Fig. 9.23 shows, the notion of a skeleton, $S(A)$, of a set A is intuitively simple. We deduce from this figure that

(a) If z is a point of $S(A)$ and $(D)_z$ is the largest disk centered at z and contained in A, one cannot find a larger disk (not necessarily centered at z) containing $(D)_z$ and included in A. The disk $(D)_z$ is called a *maximum disk*.

(b) The disk $(D)_z$ touches the boundary of A at two or more different places.



a b
c d
e

FIGURE 9.22 (a) Set A. (b) Complement of A. (c) Result of thinning the complement of A. (d) Thickened set obtained by complementing (c). (e) Final result, with no disconnected points.

a b
c d

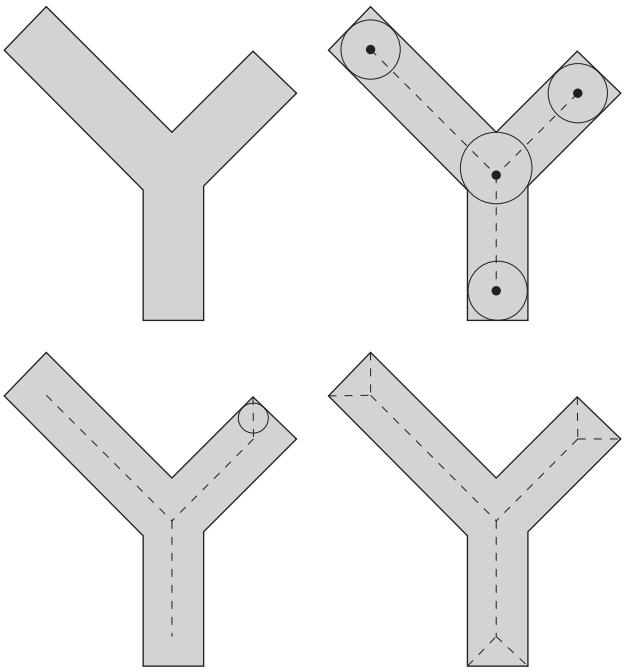


FIGURE 9.23
(a) Set A. (b) Various positions of maximum disks with centers on the skeleton of A. (c) Another maximum disk on a different segment of the skeleton of A. (d) Complete skeleton.

The skeleton of A can be expressed in terms of erosions and openings. That is, it can be shown (Serra [1982]) that

$$S(A) = \bigcup_{k=0}^{K} S_k(A) \tag{9.5-11}$$

with

$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B \tag{9.5-12}$$

where B is a structuring element, and $(A \ominus kB)$ indicates k successive erosions of A:

$$(A \ominus kB) = ((\ldots((A \ominus B) \ominus B) \ominus \ldots) \ominus B) \tag{9.5-13}$$

k times, and K is the last iterative step before A erodes to an empty set. In other words,

$$K = \max\{k | (A \ominus kB) \neq \varnothing\} \tag{9.5-14}$$

The formulation given in Eqs. (9.5-11) and (9.5-12) states that $S(A)$ can be obtained as the union of the *skeleton subsets* $S_k(A)$. Also, it can be shown that A can be *reconstructed* from these subsets by using the equation

$$A = \bigcup_{k=0}^{K} (S_k(A) \oplus kB) \tag{9.5-15}$$

where $(S_k(A) \oplus kB)$ denotes k successive dilations of $S_k(A)$; that is,

$$(S_k(A) \oplus kB) = ((\ldots((S_k(A) \oplus B) \oplus B) \oplus \ldots) \oplus B) \tag{9.5-16}$$

EXAMPLE 9.8: Computing the skeleton of a simple figure.

■ Figure 9.24 illustrates the concepts just discussed. The first column shows the original set (at the top) and two erosions by the structuring element B. Note that one more erosion of A would yield the empty set, so $K = 2$ in this case. The second column shows the opening of the sets in the first column by B. These results are easily explained by the fitting characterization of the opening operation discussed in connection with Fig. 9.8. The third column simply contains the set differences between the first and second columns.

The fourth column contains two partial skeletons and the final result (at the bottom of the column). The final skeleton not only is thicker than it needs to be but, more important, it is not connected. This result is not unexpected, as nothing in the preceding formulation of the morphological skeleton guarantees connectivity. Morphology produces an elegant formulation in terms of erosions and openings of the given set. However, heuristic formulations such as the algorithm developed in Section 11.1.7 are needed if, as is usually the case, the skeleton must be maximally thin, connected, and minimally eroded.

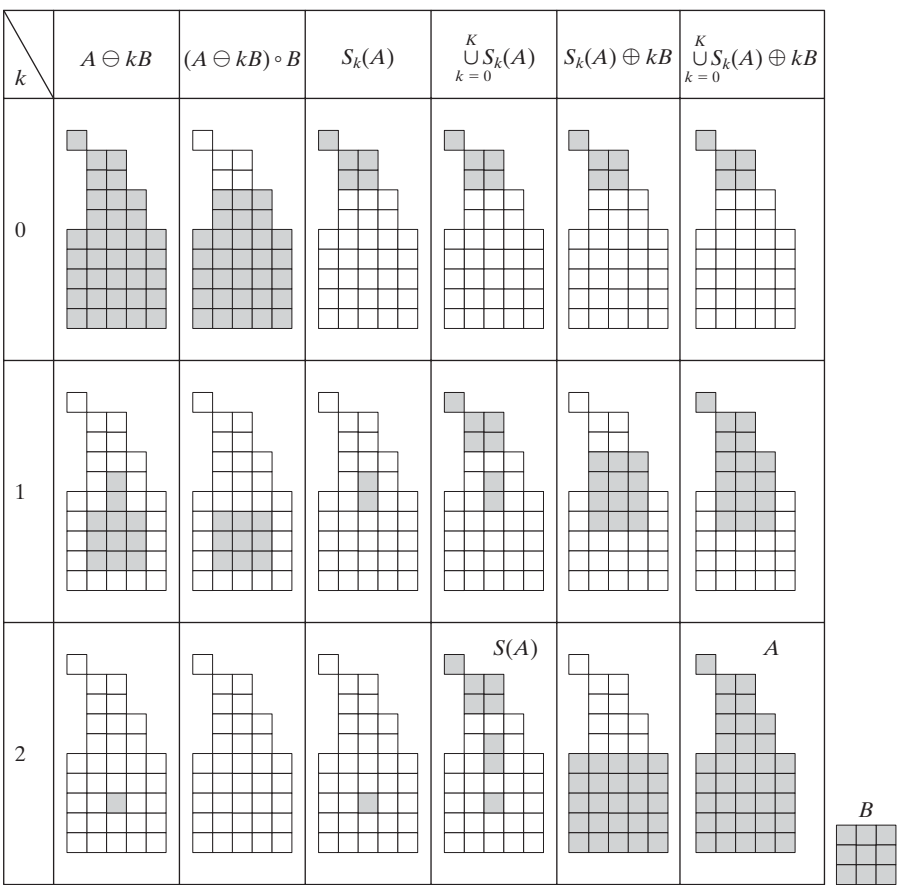


FIGURE 9.24 Implementation of Eqs. (9.5-11) through (9.5-15). The original set is at the top left, and its morphological skeleton is at the bottom of the fourth column. The reconstructed set is at the bottom of the sixth column.

The fifth column shows $S_0(A)$, $S_1(A) \oplus B$, and $(S_2(A) \oplus 2B) = (S_2(A) \oplus B) \oplus B$. Finally, the last column shows reconstruction of set A, which, according to Eq. (9.5-15), is the union of the dilated skeleton subsets shown in the fifth column. ■

9.5.8 Pruning

Pruning methods are an essential complement to thinning and skeletonizing algorithms because these procedures tend to leave parasitic components that need to be "cleaned up" by postprocessing. We begin the discussion with a pruning problem and then develop a morphological solution based on the material introduced in the preceding sections. Thus, we take this opportunity to illustrate how to go about solving a problem by combining several of the techniques discussed up to this point.

A common approach in the automated recognition of hand-printed characters is to analyze the shape of the skeleton of each character. These skeletons often are characterized by "spurs" (parasitic components). Spurs are caused during erosion by non uniformities in the strokes composing the characters. We develop a morphological technique for handling this problem, starting with the assumption that the length of a parasitic component does not exceed a specified number of pixels.

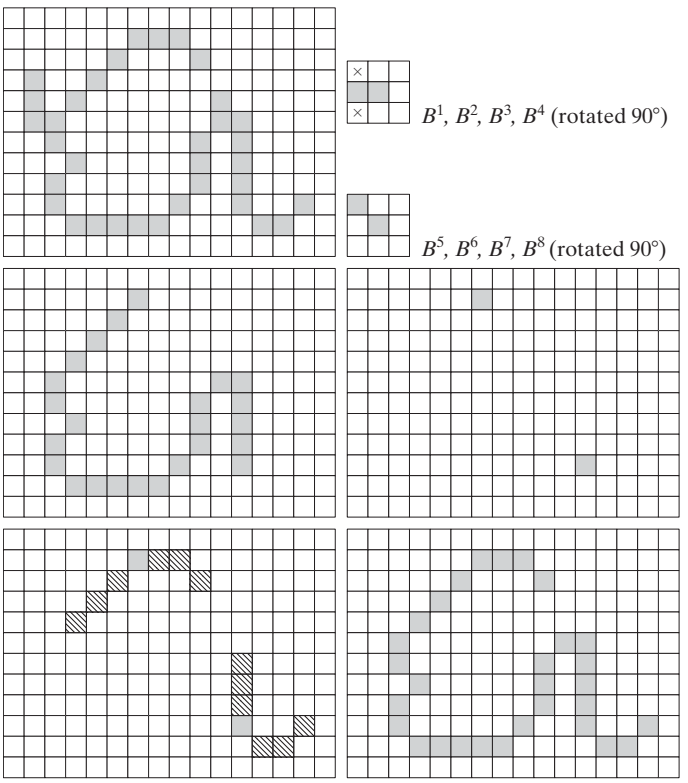
Figure 9.25(a) shows the skeleton of a hand-printed "a." The parasitic component on the leftmost part of the character is illustrative of what we are interested in removing. The solution is based on suppressing a parasitic branch by successively eliminating its end point. Of course, this also shortens (or eliminates) other branches in the character but, in the absence of other structural information, the assumption in this example is that any branch with three or less pixels is to be eliminated. Thinning of an input set A with a sequence of structuring elements designed to detect only end points achieves the desired result. That is, let

We may define an *end point* as the center point of a 3×3 region that satisfies any of the arrangements in Figs. 9.25(b) or (c).

$$X_1 = A \otimes \{B\} \tag{9.5-17}$$

where $\{B\}$ denotes the structuring element sequence shown in Figs. 9.25(b) and (c) [see Eq. (9.5-7) regarding structuring-element sequences]. The sequence of structuring elements consists of two different structures, each of which is rotated 90° for a total of eight elements. The $\times$ in Fig. 9.25(b) signifies a "don't care" condition, in the sense that it does not matter whether the pixel in that location has a value of 0 or 1. Numerous results reported in the literature on morphology are based on the use of a *single* structuring element, similar to the one in Fig. 9.25(b), but having "don't care" conditions along the entire first column. This is incorrect. For example, this element would identify the point located in the eighth row, fourth column of Fig. 9.25(a) as an end point, thus eliminating it and breaking connectivity in the stroke.

Applying Eq. (9.5-17) to A three times yields the set X_1 in Fig. 9.25(d). The next step is to "restore" the character to its original form, but with the parasitic



a b
c
d e
f g

FIGURE 9.25
(a) Original image. (b) and (c) Structuring elements used for deleting end points. (d) Result of three cycles of thinning. (e) End points of (d). (f) Dilation of end points conditioned on (a). (g) Pruned image.

branches removed. To do so first requires forming a set X_2 containing all end points in X_1 [Fig. 9.25(e)]:

$$X_2 = \bigcup_{k=1}^{8} (X_1 \circledast B^k) \tag{9.5-18}$$

where the B^k are the same end-point detectors shown in Figs. 9.25(b) and (c). The next step is dilation of the end points three times, using set A as a delimiter:

$$X_3 = (X_2 \oplus H) \cap A \tag{9.5-19}$$

Equation (9.5-19) is the basis for morphological reconstruction by dilation, as explained in the next section.

where H is a 3×3 structuring element of 1s and the intersection with A is applied after each step. As in the case of region filling and extraction of connected components, this type of conditional dilation prevents the creation of 1-valued elements outside the region of interest, as evidenced by the result shown in Fig. 9.25(f). Finally, the union of X_3 and X_1 yields the desired result,

$$X_4 = X_1 \cup X_3 \tag{9.5-20}$$

in Fig. 9.25(g).

In more complex scenarios, use of Eq. (9.5-19) sometimes picks up the "tips" of some parasitic branches. This condition can occur when the end

points of these branches are near the skeleton. Although Eq. (9.5-17) may eliminate them, they can be picked up again during dilation because they are valid points in A. Unless entire parasitic elements are picked up again (a rare case if these elements are short with respect to valid strokes), detecting and eliminating them is easy because they are disconnected regions.

A natural thought at this juncture is that there must be easier ways to solve this problem. For example, we could just keep track of all deleted points and simply reconnect the appropriate points to all end points left after application of Eq. (9.5-17). This option is valid, but the advantage of the formulation just presented is that the use of simple morphological constructs solved the entire problem. In practical situations when a set of such tools is available, the advantage is that no new algorithms have to be written. We simply combine the necessary morphological functions into a sequence of operations.

9.5.9 Morphological Reconstruction

The morphological concepts discussed thus far involve an image and a structuring element. In this section, we discuss a powerful morphological transformation called *morphological reconstruction* that involves two images and a structuring element. One image, the *marker*, contains the starting points for the transformation. The other image, the *mask*, constrains the transformation. The structuring element is used to define connectivity.[†]

Geodesic dilation and erosion

Central to morphological reconstruction are the concepts of geodesic dilation and geodesic erosion. Let F denote the marker image and G the mask image. It is assumed in this discussion that both are binary images and that $F \subseteq G$. The *geodesic dilation* of size 1 of the marker image with respect to the mask, denoted by $D_G^{(1)}(F)$, is defined as

$$D_G^{(1)}(F) = (F \oplus B) \cap G \tag{9.5-21}$$

where $\cap$ denotes the set intersection (here $\cap$ may be interpreted as a logical AND because the set intersection and logical AND operations are the same for binary sets). The geodesic dilation of size n of F with respect to G is defined as

$$D_G^{(n)}(F) = D_G^{(1)}\left[D_G^{(n-1)}(F)\right] \tag{9.5-22}$$

with $D_G^{(0)}(F) = F$. In this recursive expression, the set intersection in Eq. (9.5-21) is performed at each step.[‡] Note that the intersection operator guarantees that

[†]In much of the literature on morphological reconstruction, the structuring element is tacitly assumed to be isotropic and typically is called an *elementary isotropic structuring element*. In the context of this chapter, an example of such an SE is simply a 3×3 array of 1s with the origin at the center.

[‡]Although it is more intuitive to develop morphological-reconstruction methods using recursive formulations (as we do here), their practical implementation typically is based on more computationally efficient algorithms (see, for example, Vincent [1993] and Soille [2003]). All image-based examples in this section were generated using such algorithms.

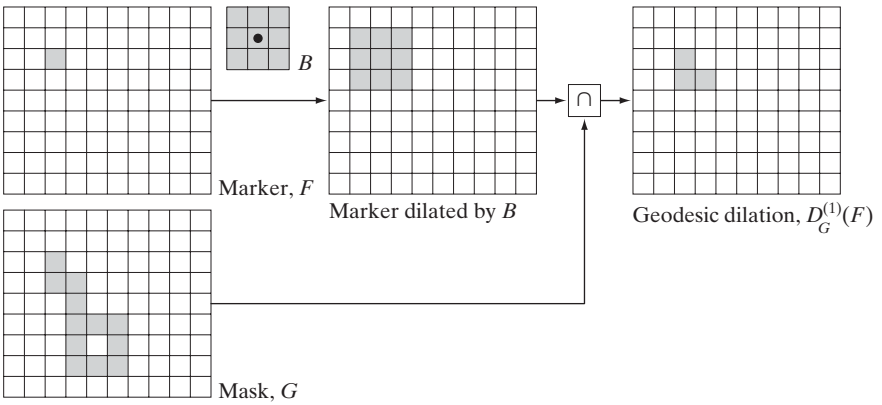


FIGURE 9.26 Illustration of geodesic dilation.

mask G will limit the growth (dilation) of marker F. Figure 9.26 shows a simple example of a geodesic dilation of size 1. The steps in the figure are a direct implementation of Eq. (9.5-21).

Similarly, the *geodesic erosion* of size 1 of marker F with respect to mask G is defined as

$$E_G^{(1)}(F) = (F \ominus B) \cup G \tag{9.5-23}$$

where $\cup$ denotes set union (or OR operation). The geodesic erosion of size n of F with respect to G is defined as

$$E_G^{(n)}(F) = E_G^{(1)}\left[E_G^{(n-1)}(F)\right] \tag{9.5-24}$$

with $E_G^{(0)}(F) = F$. The set union operation in Eq. (9.5-23) is performed at each iterative step, and guarantees that geodesic erosion of an image remains greater than or equal to its mask image. As expected from the forms in Eqs. (9.5-21) and (9.5-23), geodesic dilation and erosion are *duals* with respect to set complementation (see Problem 9.29). Figure 9.27 shows a simple example of geodesic erosion of size 1. The steps in the figure are a direct implementation of Eq. (9.5-23).

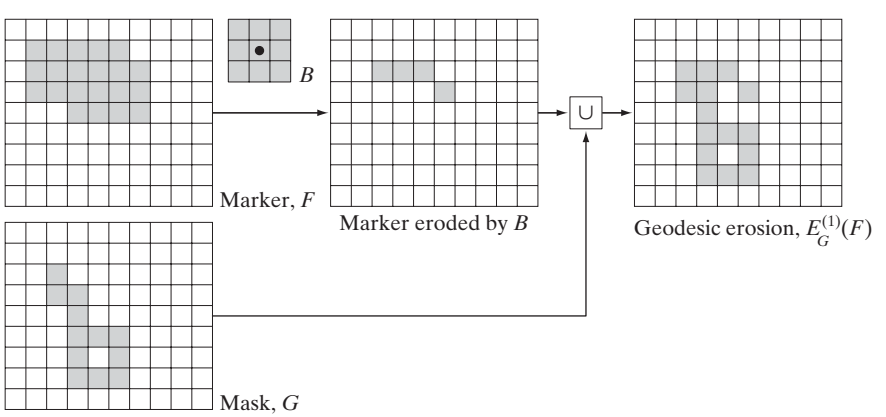


FIGURE 9.27 Illustration of geodesic erosion.

Geodesic dilation and erosion of finite images always converge after a finite number of iterative step because propagation or shrinking of the marker image is constrained by the mask.

Morphological reconstruction by dilation and by erosion

Based on the preceding concepts, *morphological reconstruction by dilation* of a mask image G from a marker image F, denoted $R_G^D(F)$, is defined as the geodesic dilation of F with respect to G, iterated until stability is achieved; that is,

$$R_G^D(F) = D_G^{(k)}(F) \tag{9.5-25}$$

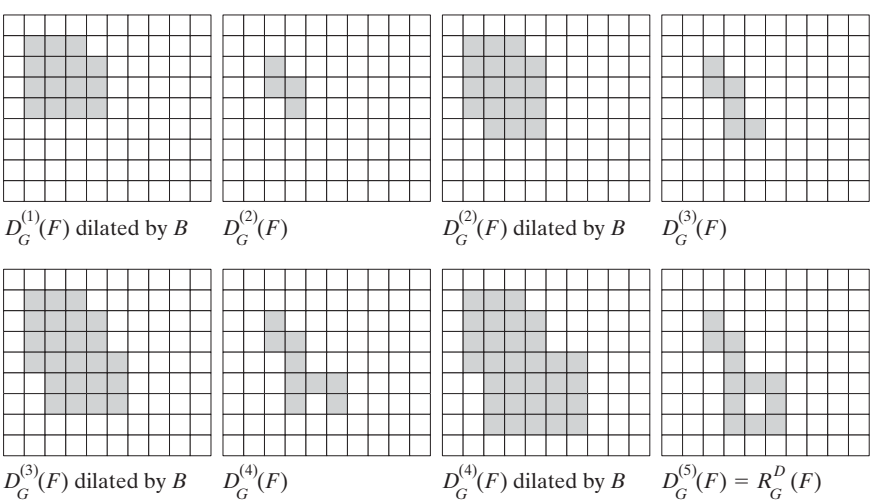
with k such that $D_G^{(k)}(F) = D_G^{(k+1)}(F)$.

Figure 9.28 illustrates reconstruction by dilation. Figure 9.28(a) continues the process begun in Fig. 9.26; that is, the next step in reconstruction after obtaining $D_G^{(1)}(F)$ is to dilate this result and then AND it with the mask G to yield $D_G^{(2)}(F)$, as Fig. 9.28(b) shows. Dilation of $D_G^{(2)}(F)$ and masking with G then yields $D_G^{(3)}(F)$, and so on. This procedure is repeated until stability is reached. If we carried this example one more step, we would find that $D_G^{(5)}(F) = D_G^{(6)}(F)$, so the morphologically reconstructed image by dilation is given by $R_G^D(F) = D_G^{(5)}(F)$, as indicated in Eq. (9.5-25). Note that the reconstructed image in this case is identical to the mask because F contained a single 1-valued pixel (this is analogous to convolution of an image with an impulse, which simply copies the image at the location of the impulse, as explained in Section 3.4.2).

In a similar manner, the *morphological reconstruction by erosion* of a mask image G from a marker image F, denoted $R_G^E(F)$, is defined as the geodesic erosion of F with respect to G, iterated until stability; that is,

$$R_G^E(F) = E_G^{(k)}(F) \tag{9.5-26}$$

with k such that $E_G^{(k)}(F) = E_G^{(k+1)}(F)$. As an exercise, you should generate a figure similar to Fig. 9.28 for morphological reconstruction by erosion.



a b c d
e f g h

FIGURE 9.28 Illustration of morphological reconstruction by dilation. F, G, B and $D_G^{(1)}(F)$ are from Fig. 9.26.

Reconstruction by dilation and erosion are duals with respect to set complementation (see Problem 9.30).

Sample applications

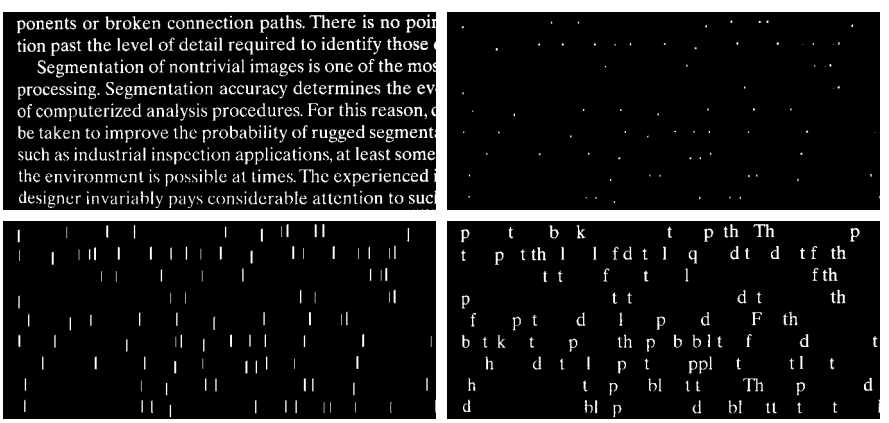
Morphological reconstruction has a broad spectrum of practical applications, each determined by the selection of the marker and mask images, by the structuring elements used, and by combinations of the primitive operations defined in the preceding discussion. The following examples illustrate the usefulness of these concepts.

Opening by reconstruction: In a morphological opening, erosion removes small objects and the subsequent dilation attempts to restore the shape of objects that remain. However, the accuracy of this restoration is highly dependent on the similarity of the shapes of the objects and the structuring element used. *Opening by reconstruction* restores *exactly* the shapes of the objects that remain after erosion. The opening by reconstruction of size n of an image F is defined as the reconstruction by dilation of F from the erosion of size n of F; that is,

$$O_R^{(n)}(F) = R_F^D\left[(F \ominus nB)\right] \tag{9.5-27}$$

where $(F \ominus nB)$ indicates n erosions of F by B, as explained in Section 9.5.7. Note that F is used as the mask in this application. A similar expression can be written for closing by reconstruction (see Table 9.1).

Figure 9.29 shows an example of opening by reconstruction. In this illustration, we are interested in extracting from Fig. 9.29(a) the characters that contain long, vertical strokes. Opening by reconstruction requires at least one erosion, so we perform that step first. Figure 9.29(b) shows the erosion



a b
c d

FIGURE 9.29 (a) Text image of size 918 × 2018 pixels. The approximate average height of the tall characters is 50 pixels. (b) Erosion of (a) with a structuring element of size 51 × 1 pixels. (c) Opening of (a) with the same structuring element, shown for reference. (d) Result of opening by reconstruction.

of Fig. 9.29(a) with a structuring element of length proportional to the average height of the tall characters (51 pixels) and width of one pixel. For the purpose of comparison, we computed the opening of the image using the same structuring element. Figure 9.29(c) shows the result. Finally, Fig. 9.29(d) is the opening by reconstruction (of size 1) of F [i.e., $O_R^{(1)}(F)$] given in Eq. (9.5-27). This result shows that characters containing long vertical strokes were restored accurately; all other characters were removed.

Filling holes: In Section 9.5.2, we developed an algorithm for filling holes based on knowing a starting point in each hole in the image. Here, we develop a fully automated procedure based on morphological reconstruction. Let $I(x, y)$ denote a binary image and suppose that we form a marker image F that is 0 everywhere, except at the image border, where it is set to $1 - I$; that is,

$$F(x, y) = \begin{cases} 1 - I(x, y) & \text{if } (x, y) \text{ is on the border of } I \\ 0 & \text{otherwise} \end{cases} \tag{9.5-28}$$

Then

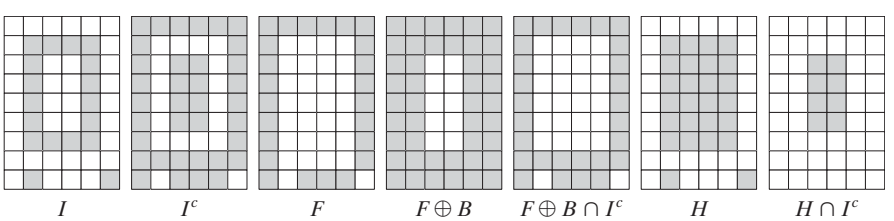
$$H = \left[R_{I^c}^D(F)\right]^c \tag{9.5-29}$$

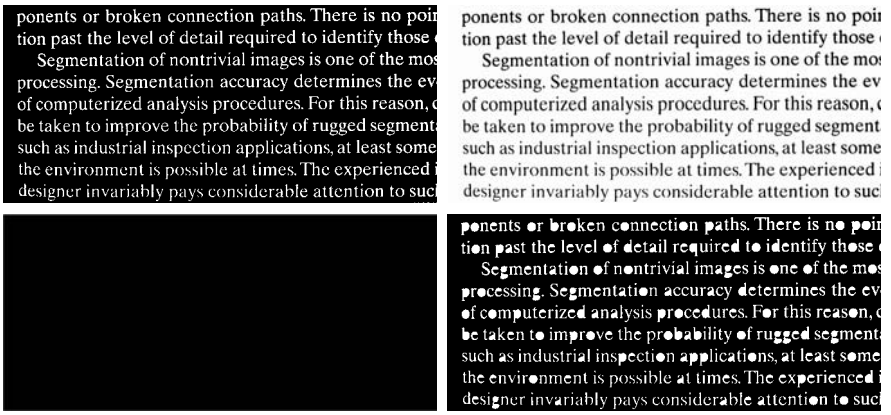
is a binary image equal to I with all holes filled.

Let us consider the individual components of Eq. (9.5-29) to see how this expression in fact leads to all holes in an image being filled. Figure 9.30(a) shows a simple image I containing one hole, and Fig. 9.30(b) shows its complement. Note that because the complement of I sets all foreground (1-valued) pixels to background (0-valued) pixels, and vice versa, this operation in effect builds a "wall" of 0s around the hole. Because I^c is used as an AND mask, all we are doing here is protecting all foreground pixels (including the wall around the hole) from changing during iteration of the procedure. Figure 9.30(c) is array F formed according to Eq. (9.5-28) and Fig. 9.30(d) is F dilated with a 3×3 SE whose elements are all 1s. Note that marker F has a border of 1s (except at locations where I is 1), so the dilation of F of the marker points starts at the border and proceeds inward. Figure 9.30(e) shows the geodesic dilation of F using I^c as the mask. As was just indicated, we see that all locations in this result corresponding to foreground pixels from I are 0, and that this is true now for the hole pixels as well. Another iteration will yield the same result which, when complemented as required by Eq. (9.5-29), gives the result in Fig. 9.30(f). As desired, the hole is now filled and the rest of image I was unchanged. The operation $H \cap I^c$ yields an image containing 1-valued pixels in the locations corresponding to the holes in I, as Fig. 9.30(g) shows.

a b c d e f g

FIGURE 9.30 Illustration of hole filling on a simple image.





a b
c d

FIGURE 9.31
(a) Text image of size 918 × 2018 pixels. (b) Complement of (a) for use as a mask image. (c) Marker image. (d) Result of hole-filling using Eq. (9.5-29).

Figure 9.31 shows a more practical example. Figure 9.31(b) shows the complement of the text image in Fig. 9.31(a), and Fig. 9.31(c) is the marker image, F, generated using Eq. (9.5-28). This image has a border of 1s, except at locations corresponding to 1s in the border of the original image. Finally, Fig. 9.31(d) shows the image with all the holes filled.

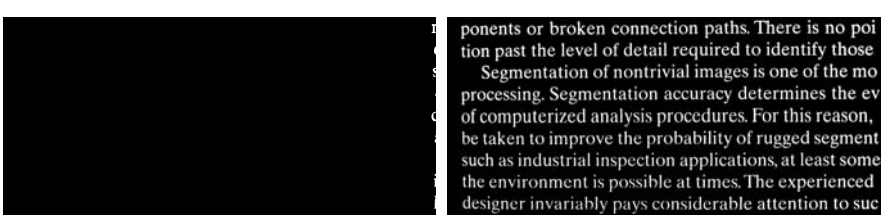
Border clearing: The extraction of objects from an image for subsequent shape analysis is a fundamental task in automated image processing. An algorithm for removing objects that touch (i.e., are connected to) the border is a useful tool because (1) it can be used to screen images so that only complete objects remain for further processing, or (2) it can be used as a signal that partial objects are present in the field of view. As a final illustration of the concepts introduced in this section, we develop a border-clearing procedure based on morphological reconstruction. In this application, we use the original image as the mask and the following marker image:

$$F(x, y) = \begin{cases} I(x, y) & \text{if } (x, y) \text{ is on the border of } I \\ 0 & \text{otherwise} \end{cases} \tag{9.5-30}$$

The border-Eclearing algorithm first computes the morphological reconstruction $R_I^D(F)$ (which simply extracts the objects touching the border) and then computes the difference

$$X = I - R_I^D(F) \tag{9.5-31}$$

to obtain an image, X, with no objects touching the border.



a b

FIGURE 9.32
Border clearing. (a) Marker image. (b) Image with no objects touching the border. The original image is Fig. 9.29(a).

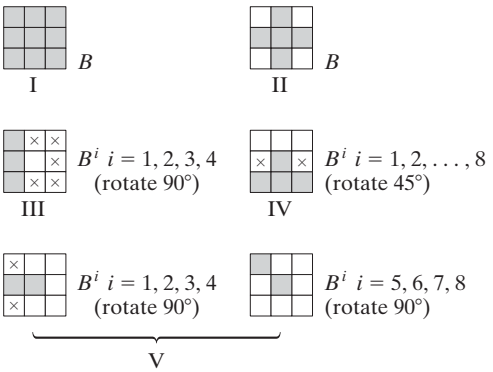


FIGURE 9.33 Five basic types of structuring elements used for binary morphology. The origin of each element is at its center and the ×'s indicate "don't care" values.

As an example, consider the text image again. Figure 9.32(a) in the previous page shows the reconstruction $R_I^D(F)$ obtained using a 3×3 structuring element of all 1s (note the objects touching the boundary on the right side), and Fig. 9.32(b) shows image X, computed using Eq. (9.5-31). If the task at hand were automated character recognition, having an image in which no characters touch the border is most useful because the problem of having to recognize partial characters (a difficult task at best) is avoided.

9.5.10 Summary of Morphological Operations on Binary Images

Table 9.1 summarizes the morphological results developed in the preceding sections, and Fig. 9.33 summarizes the basic types of structuring elements used in the various morphological processes discussed thus far. The Roman numerals in the third column of Table 9.1 refer to the structuring elements in Fig. 9.33.

TABLE 9.1 Summary of morphological operations and their properties.

Operation	Equation	Comments (The Roman numerals refer to the structuring elements in Fig. 9.33.)
Translation	$(B)_z = \{w \mid w = b + z, \text{ for } b \in B\}$	Translates the origin of B to point z.
Reflection	$\hat{B} = \{w \mid w = -b, \text{ for } b \in B\}$	Reflects all elements of B about the origin of this set.
Complement	$A^c = \{w \mid w \notin A\}$	Set of points not in A.
Difference	$A - B = \{w \mid w \in A, w \notin B\} = A \cap B^c$	Set of points that belong to A but not to B.
Dilation	$A \oplus B = \left\{z \mid (\hat{B}_z) \cap A \neq \varnothing\right\}$	"Expands" the boundary of A. (I)
Erosion	$A \ominus B = \left\{z \mid (B)_z \subseteq A\right\}$	"Contracts" the boundary of A. (I)
Opening	$A \circ B = (A \ominus B) \oplus B$	Smoothes contours, breaks narrow isthmuses, and eliminates small islands and sharp peaks. (I)

(*Continued*)

TABLE 9.1
(Continued)

Operation	Equation	Comments (The Roman numerals refer to the structuring elements in Fig. 9.33.)
Closing	$A \bullet B = (A \oplus B) \ominus B$	Smoothes contours, fuses narrow breaks and long thin gulfs, and eliminates small holes. (I)
Hit-or-miss transform	$A \circledast B = (A \ominus B_1) \cap (A^c \ominus B_2)$ $= (A \ominus B_1) - (A \oplus \hat{B}_2)$	The set of points (coordinates) at which, simultaneously, B_1 found a match ("hit") in A and B_2 found a match in A^c
Boundary extraction	$\beta(A) = A - (A \ominus B)$	Set of points on the boundary of set A. (I)
Hole filling	$X_k = (X_{k-1} \oplus B) \cap A^c$; $k = 1, 2, 3, \ldots$	Fills holes in A; X_0 = array of 0s with a 1 in each hole. (II)
Connected components	$X_k = (X_{k-1} \oplus B) \cap A$; $k = 1, 2, 3, \ldots$	Finds connected components in A; X_0 = array of 0s with a 1 in each connected component. (I)
Convex hull	$X_k^i = (X_{k-1}^i \circledast B^i) \cup A$; $i = 1, 2, 3, 4$; $k = 1, 2, 3, \ldots$; $X_0^i = A$; and $D^i = X_{\text{conv}}^i$	Finds the convex hull $C(A)$ of set A, where "conv" indicates convergence in the sense that $X_k^i = X_{k-1}^i$. (III)
Thinning	$A \otimes B = A - (A \circledast B)$ $= A \cap (A \circledast B)^c$ $A \otimes \{B\} =$ $((\ldots((A \otimes B^1) \otimes B^2)\ldots) \otimes B^n)$ $\{B\} = \{B^1, B^2, B^3, \ldots, B^n\}$	Thins set A. The first two equations give the basic definition of thinning. The last equations denote thinning by a sequence of structuring elements. This method is normally used in practice. (IV)
Thickening	$A \odot B = A \cup (A \circledast B)$ $A \odot \{B\} =$ $((\ldots(A \odot B^1) \odot B^2 \ldots) \odot B^n)$	Thickens set A. (See preceding comments on sequences of structuring elements.) Uses IV with 0s and 1s reversed.
Skeletons	$S(A) = \bigcup_{k=0}^{K} S_k(A)$ $S_k(A) = \bigcup_{k=0}^{K} \{(A \ominus kB)$ $- [(A \ominus kB) \circ B]\}$ Reconstruction of A: $A = \bigcup_{k=0}^{K} (S_k(A) \oplus kB)$	Finds the skeleton $S(A)$ of set A. The last equation indicates that A can be reconstructed from its skeleton subsets $S_k(A)$. In all three equations, K is the value of the iterative step after which the set A erodes to the empty set. The notation $(A \ominus kB)$ denotes the kth iteration of successive erosions of A by B. (I)

(*Continued*)

TABLE 9.1
(Continued)

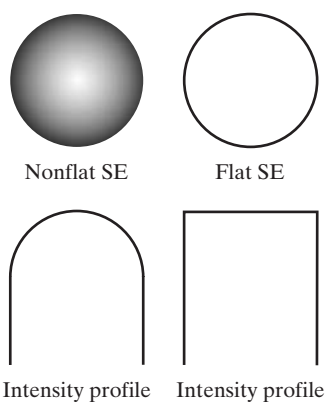
Operation	Equation	Comments (The Roman numerals refer to the structuring elements in Fig. 9.33.)
Pruning	$X_1 = A \otimes \{B\}$ $X_2 = \bigcup_{k=1}^{8} (X_1 \circledast B^k)$ $X_3 = (X_2 \oplus H) \cap A$ $X_4 = X_1 \cup X_3$	X_4 is the result of pruning set A. The number of times that the first equation is applied to obtain X_1 must be specified. Structuring elements V are used for the first two equations. In the third equation H denotes structuring element I.
Geodesic dilation of size 1	$D_G^{(1)}(F) = (F \oplus B) \cap G$	F and G are called the *marker* and *mask* images, respectively.
Geodesic dilation of size n	$D_G^{(n)}(F) = D_G^{(1)}\left[D_G^{(n-1)}(F)\right]$; $D_G^{(0)}(F) = F$	
Geodesic erosion of size 1	$E_G^{(1)}(F) = (F \ominus B) \cup G$	
Geodesic erosion of size n	$E_G^{(n)}(F) = E_G^{(1)}\left[E_G^{(n-1)}(F)\right]$; $E_G^{(0)}(F) = F$	
Morphological reconstruction by dilation	$R_G^D(F) = D_G^{(k)}(F)$	k is such that $D_G^{(k)}(F) = D_G^{(k+1)}(F)$
Morphological reconstruction by erosion	$R_G^E(F) = E_G^{(k)}(F)$	k is such that $E_G^{(k)}(F) = E_G^{(k+1)}(F)$
Opening by reconstruction	$O_R^{(n)}(F) = R_F^D[(F \ominus nB)]$	$(F \ominus nB)$ indicates n erosions of F by B.
Closing by reconstruction	$C_R^{(n)}(F) = R_F^E[(F \oplus nB)]$	$(F \oplus nB)$ indicates n dilations of F by B.
Hole filling	$H = \left[R_{I^c}^D(F)\right]^c$	H is equal to the input image I, but with all holes filled. See Eq. (9.5-28) for the definition of the marker image F.
Border clearing	$X = I - R_I^D(F)$	X is equal to the input image I, but with all objects that touch (are connected to) the boundary removed. See Eq. (9.5-30) for the definition of the marker image F.

9.6 Gray-Scale Morphology

In this section, we extend to gray-scale images the basic operations of dilation, erosion, opening, and closing. We then use these operations to develop several basic gray-scale morphological algorithms.

Throughout the discussion that follows, we deal with digital functions of the form $f(x, y)$ and $b(x, y)$, where $f(x, y)$ is a gray-scale image and $b(x, y)$ is a structuring element. The assumption is that these functions are discrete in the sense introduced in Section 2.4.2. That is, if Z denotes the set of real integers, then the coordinates (x, y) are integers from the Cartesian product Z^2 and f and b are functions that assign an intensity value (a real number from the set of real numbers, R) to each distinct pair of coordinates (x, y). If the intensity levels are integers also, then Z replaces R.

Structuring elements in gray-scale morphology perform the same basic functions as their binary counterparts: They are used as "probes" to examine a given image for specific properties. Structuring elements in gray-scale morphology belong to one of two categories: *nonflat* and *flat*. Figure 9.34 shows an example of each. Figure 9.34(a) is a hemispherical gray-scale SE shown as an image, and Fig. 9.34(c) is a horizontal intensity profile through its center. Figure 9.34(b) shows a flat structuring element in the shape of a disk and Fig. 9.34(d) is its corresponding intensity profile (the shape of this profile explains the origin of the word "flat"). The elements in Fig. 9.34 are shown as continuous quantities for clarity; their computer implementation is based on digital approximations (e.g., see the rightmost disk SE in Fig. 9.2). Due to a number of difficulties discussed later in this section, gray-scale SEs are used infrequently in practice. Finally, we mention that, as in the binary case, the origin of structuring elements must be clearly identified. Unless mentioned otherwise, all the examples in this section are based on symmetrical, flat structuring elements of unit height whose origins are at the center. The *reflection* of an SE in gray-scale morphology is as defined in Section 9.1, and we denote it in the following discussion by $\hat{b}(x, y) = b(-x - y)$.



a b
c d

FIGURE 9.34 Nonflat and flat structuring elements, and corresponding horizontal intensity profiles through their center. All examples in this section are based on flat SEs.

9.6.1 Erosion and Dilation

The *erosion* of f by a *flat* structuring element b at any location (x, y) is defined as the *minimum* value of the image in the region coincident with b when the origin of b is at (x, y). In equation form, the erosion at (x, y) of an image f by a structuring element b is given by

$$\big[f \ominus b\big](x, y) = \min_{(s,t)\in b}\big\{f(x + s, y + t)\big\} \tag{9.6-1}$$

where, in a manner similar to the correlation procedure discussed in Section 3.4.2, x and y are incremented through all values required so that the origin of b visits every pixel in f. That is, to find the erosion of f by b, we place the origin of the structuring element at every pixel location in the image. The erosion at any location is determined by selecting the minimum value of f from all the values of f contained in the region coincident with b. For example, if b is a square structuring element of size 3×3, obtaining the erosion at a point requires finding the minimum of the nine values of f contained in the 3×3 region defined by b when its origin is at that point.

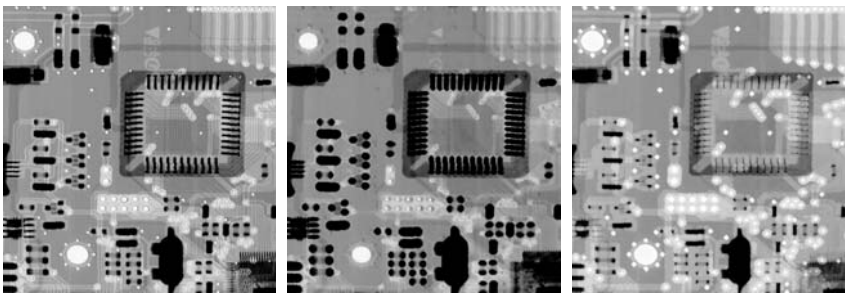
Similarly, the *dilation* of f by a *flat* structuring element b at any location (x, y) is defined as the *maximum* value of the image in the window outlined by $\hat{b}$ when the origin of $\hat{b}$ is at (x, y). That is,

$$\big[f \oplus b\big](x, y) = \max_{(s,t)\in b}\big\{f(x - s, y - t)\big\} \tag{9.6-2}$$

where we used the fact stated earlier that $\hat{b} = b(-x, -y)$. The explanation of this equation is identical to the explanation in the previous paragraph, but using the maximum, rather than the minimum, operation and keeping in mind that the structuring element is reflected about its origin, which we take into account by using $(-s, -t)$ in the argument of the function. This is analogous to spatial convolution, as explained in Section 3.4.2.

EXAMPLE 9.9: Illustration of gray-scale erosion and dilation.

■ Because gray-scale erosion with a flat SE computes the minimum intensity value of f in every neighborhood of (x, y) coincident with b, we expect in general that an eroded gray-scale image will be darker than the original, that the sizes (with respect to the size of the SE) of bright features will be reduced, and that the sizes of dark features will be increased. Figure 9.35(b) shows the erosion of Fig. 9.35(a) using a disk SE of unit height and a radius of two pixels. The effects just mentioned are clearly visible in the eroded image. For instance, note how the intensities of the small bright dots were reduced, making them barely visible in Fig. 9.35(b), while the dark features grew in thickness. The general background of the eroded image is slightly darker than the background of the original image. Similarly, Fig. 9.35(c) shows the result of dilation with the same SE. The effects are the opposite of those obtained with erosion. The bright features were thickened and the intensities of the dark features were reduced. Note in particular how the thin black connecting wires in the left, middle, and right, bottom of Fig. 9.35(a) are barely visible in Fig. 9.35(c). The sizes of the dark dots were reduced as a result of dilation but, unlike the eroded small white dots in Fig. 9.35(b), they still are easily visible in the dilated



a b c

FIGURE 9.35 (a) A gray-scale X-ray image of size 448 × 425 pixels. (b) Erosion using a flat disk SE with a radius of two pixels. (c) Dilation using the same SE. (Original image courtesy of Lixi, Inc.)

image. The reason is that the black dots were originally larger than the white dots with respect to the size of the SE. Finally, note that the background of the dilated image is slightly lighter than that of Fig. 9.35(a). ■

Nonflat SEs have gray-scale values that vary over their domain of definition. The erosion of image f by nonflat structuring element, b_N, is defined as

$$\left[f \ominus b_N\right](x, y) = \min_{(s,t)\in b_N}\left\{f(x + s, y + t) - b_N(s, t)\right\} \tag{9.6-3}$$

Here, we actually subtract values from f to determine the erosion at any point. This means that, unlike Eq. (9.6-1), erosion using a nonflat SE is not bounded in general by the values of f, which can present problems in interpreting results. Gray-scale SEs are seldom used in practice because of this, in addition to potential difficulties in selecting meaningful elements for b_N, and the added computational burden when compared with Eq. (9.6-1).

In a similar manner, dilation using a nonflat SE is defined as

$$\left[f \oplus b_N\right](x, y) = \max_{(s,t)\in b_N}\left\{f(x - s, y - t) + b_N(s, t)\right\} \tag{9.6-4}$$

The same comments made in the previous paragraph are applicable to dilation with nonflat SEs. When all the elements of b_N are constant (i.e., the SE is flat), Eqs. (9.6-3) and (9.6-4) reduce to Eqs. (9.6-1) and (9.6-2), respectively, within a scalar constant equal to the amplitude of the SE.

As in the binary case, erosion and dilation are duals with respect to function complementation and reflection; that is,

$$(f \ominus b)^c(x, y) = (f^c \oplus \hat{b})(x, y)$$

where $f^c = -f(x, y)$ and $\hat{b} = b(-x, -y)$. The same expression holds for nonflat structuring elements. Except as needed for clarity, we simplify the notation in the following discussion by omitting the arguments of all functions, in which case the preceding equation is written as

$$(f \ominus b)^c = (f^c \oplus \hat{b}) \tag{9.6-5}$$

Similarly,

$$(f \oplus b)^c = (f^c \ominus \hat{b}) \tag{9.6-6}$$

Erosion and dilation by themselves are not particularly useful in gray-scale image processing. As with their binary counterparts, these operations become powerful when used in combination to derive higher-level algorithms, as the material in the following sections demonstrates.

Although we deal with flat SEs in the examples in the remainder of this section, the concepts discussed are applicable also to nonflat structuring elements.

9.6.2 Opening and Closing

The expressions for opening and closing gray-scale images have the same form as their binary counterparts. The *opening* of image f by structuring element b, denoted $f \circ b$, is

$$f \circ b = (f \ominus b) \oplus b \tag{9.6-7}$$

As before, opening is simply the erosion of f by b, followed by a dilation of the result with b. Similarly, the *closing* of f by b, denoted $f \cdot b$, is

$$f \bullet b = (f \oplus b) \ominus b \tag{9.6-8}$$

The opening and closing for gray-scale images are duals with respect to complementation and SE reflection:

$$(f \bullet b)^c = f^c \circ \hat{b} \tag{9.6-9}$$

and

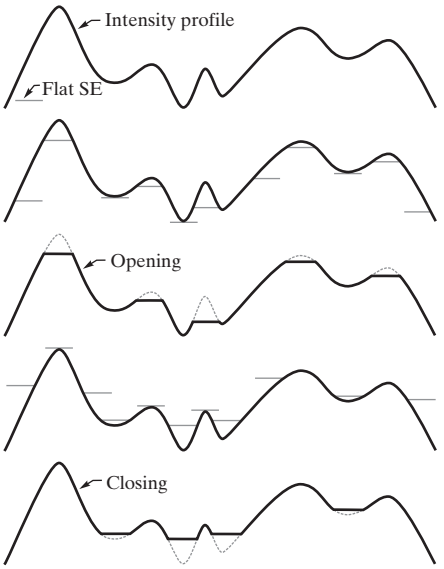
$$(f \circ b)^c = f^c \bullet \hat{b} \tag{9.6-10}$$

Because $f^c = -f(x, y)$, Eq. (9.6-9) can be written also as $-(f \bullet b) = (-f \circ \hat{b})$ and similarly for Eq. (9.6-10).

Opening and closing of images have a simple geometric interpretation. Suppose that an image function $f(x, y)$ is viewed as a 3-D surface; that is, its intensity values are interpreted as height values over the xy-plane, as in Fig. 2.18(a). Then the opening of f by b can be interpreted geometrically as pushing the structuring element up from below against the undersurface of f. At each location of the origin of b, the opening is the highest value reached by any part of b as it pushes up against the undersurface of f. The complete opening is then the set of all such values obtained by having the origin of b visit every (x, y) coordinate of f.

Sometimes opening and closing are illustrated by rolling a circle on the under and upper sides of a curve. In 3-D, the circle becomes a sphere and the resulting procedures are called *rolling-ball* algorithms.

Figure 9.36 illustrates the concept in one dimension. Suppose that the curve in Fig. 9.36(a) is the intensity profile along a single row of an image. Figure 9.36(b) shows a flat structuring element in several positions, pushed up against the bottom of the curve. The solid curve in Fig. 9.36(c) is the complete opening. Because the structuring element is too large to fit completely inside the upward peaks of the curve, the tops of the peaks are clipped by the opening, with the amount removed being proportional to how far the structuring element was able to reach into the peak. In general, openings are used to remove small, bright details, while leaving the overall intensity levels and larger bright features relatively undisturbed.



a
b
c
d
e

FIGURE 9.36 Opening and closing in one dimension. (a) Original 1-D signal. (b) Flat structuring element pushed up underneath the signal. (c) Opening. (d) Flat structuring element pushed down along the top of the signal. (e) Closing.

Figure 9.36(d) is a graphical illustration of closing. Observe that the structuring element is pushed down on top of the curve while being translated to all locations. The closing, shown in Fig. 9.36(e), is constructed by finding the lowest points reached by any part of the structuring element as it slides against the upper side of the curve.

The gray-scale opening operation satisfies the following properties:

(a) $f \circ b \leftarrow\!\lrcorner f$
(b) If $f_1 \leftarrow\!\lrcorner f_2$, then $(f_1 \circ b) \leftarrow\!\lrcorner (f_2 \circ b)$
(c) $(f \circ b) \circ b = f \circ b$

The notation $e \leftarrow\!\lrcorner r$ is used to indicate that the domain of e is a subset of the domain of r, and also that $e(x, y) \leq r(x, y)$ for any (x, y) in the domain of e.

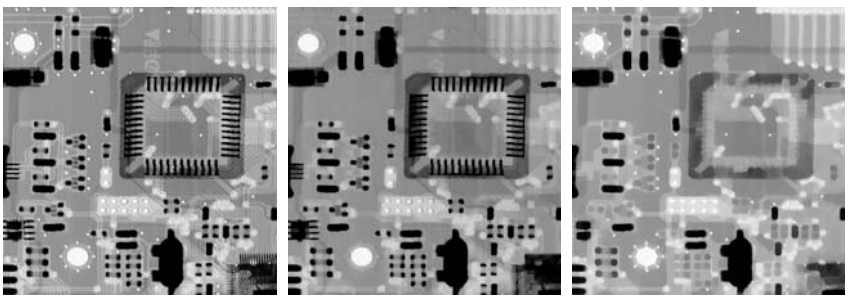
Similarly, the closing operation satisfies the following properties:

(a) $f \leftarrow\!\lrcorner f \bullet b$
(b) If $f_1 \leftarrow\!\lrcorner f_2$, then $(f_1 \bullet b) \leftarrow\!\lrcorner (f_2 \bullet b)$
(c) $(f \bullet b) \bullet b = f \bullet b$

The usefulness of these properties is similar to that of their binary counterparts.

EXAMPLE 9.10: Illustration of gray-scale opening and closing.

■ Figure 9.37 extends to 2-D the 1-D concepts illustrated in Fig. 9.36. Figure 9.37(a) is the same image we used in Example 9.9, and Fig. 9.37(b) is the opening obtained using a disk structuring element of unit height and radius of 3 pixels. As expected, the intensity of all bright features decreased, depending on the sizes of the features relative to the size of the SE. Comparing this figure with Fig. 9.35(b), we see that, unlike the result of erosion, opening had negligible effect on the dark features of the image, and the effect on the background was negligible. Similarly, Fig. 9.37(c) shows the closing of the image with a disk of radius 5 (the small round



a b c

FIGURE 9.37 (a) A gray-scale X-ray image of size 448 × 425 pixels. (b) Opening using a disk SE with a radius of 3 pixels. (c) Closing using an SE of radius 5.

black dots are larger than the small white dots, so a larger disk was needed to achieve results comparable to the opening). In this image, the bright details and background were relatively unaffected, but the dark features were attenuated, with the degree of attenuation being dependent on the relative sizes of the features with respect to the SE. ■

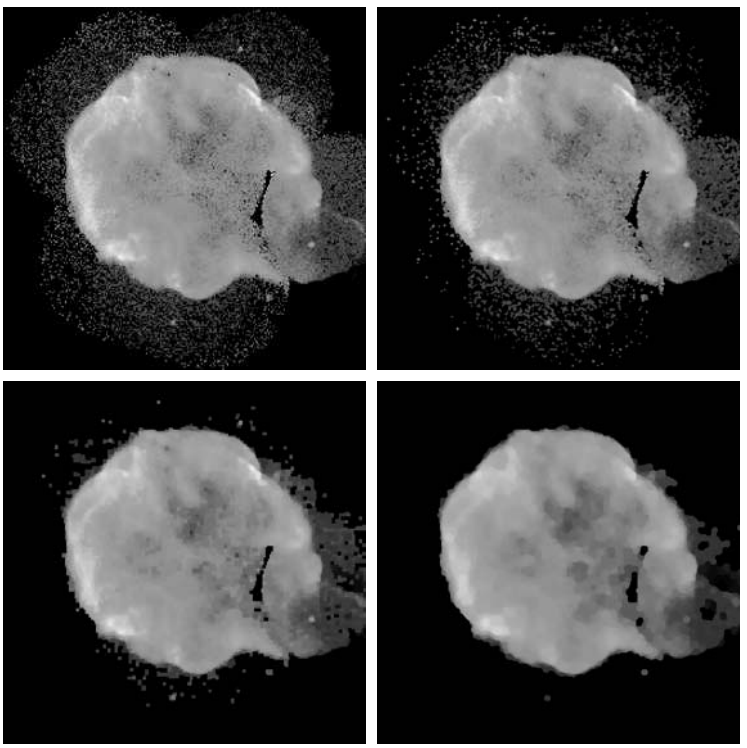
9.6.3 Some Basic Gray-Scale Morphological Algorithms

Numerous morphological techniques are based on the gray-scale morphological concepts introduced thus far. We illustrate some of these algorithms in the following discussion.

Morphological smoothing

Because opening suppresses bright details smaller than the specified SE, and closing suppresses dark details, they are used often in combination as *morphological filters* for image smoothing and noise removal. Consider Fig. 9.38(a), which shows an image of the Cygnus Loop supernova taken in the X-ray band (see Fig. 1.7 for details about this image). For purposes of the present discussion, suppose that the central light region is the object of interest and that the smaller components are noise. The objective is to remove the noise. Figure 9.38(b) shows the result of opening the original image with a flat disk of radius 2 and then closing the opening with an SE of the same size. Figures 9.38(c) and (d) show the results of the same operation using SEs of radii 3 and 5, respectively. As expected, this sequence shows progressive removal of small components as a function of SE size. In the last result, we see that the object of interest has been extracted. The noise components on the lower side of the image could not be removed completely because of their density.

The results in Fig. 9.38 are based on opening the original image and then closing the opening. A procedure used sometimes is to perform *alternating sequential filtering*, in which the opening–closing sequence starts with the original image, but subsequent steps perform the opening and closing on the results



a b
c d

FIGURE 9.38
(a) 566 × 566 image of the Cygnus Loop supernova, taken in the X-ray band by NASA's Hubble Telescope. (b)–(d) Results of performing opening and closing sequences on the original image with disk structuring elements of radii, 1, 3, and 5, respectively. (Original image courtesy of NASA.)

of the previous step. This type of filtering is useful in automated image analysis, in which results at each step are compared against a specified metric. Generally, this approach produces more blurring for the same size SE than the method illustrated in Fig. 9.38.

Morphological gradient

Dilation and erosion can be used in combination with image subtraction to obtain the morphological gradient of an image, denoted by g, where

See Section 3.6.4 for a definition of the image gradient.

$$g = (f \oplus b) - (f \ominus b) \tag{9.6-11}$$

The dilation thickens regions in an image and the erosion shrinks them. Their difference emphasizes the boundaries between regions. Homogenous areas are not affected (as long as the SE is relatively small) so the subtraction operation tends to eliminate them. The net result is an image in which the edges are enhanced and the contribution of the homogeneous areas are suppressed, thus producing a "derivative-like" (gradient) effect.

Figure 9.39 shows an example. Figure 9.39(a) is a head CT scan, and the next two figures are the opening and closing with a 3 × 3 SE of all 1s. Note the thickening and shrinking just mentioned. Figure 9.39(d) is the morphological gradient obtained using Eq. (9.6-11), in which the boundaries between regions are clearly delineated, as expected of a 2-D derivative image.

a b
c d

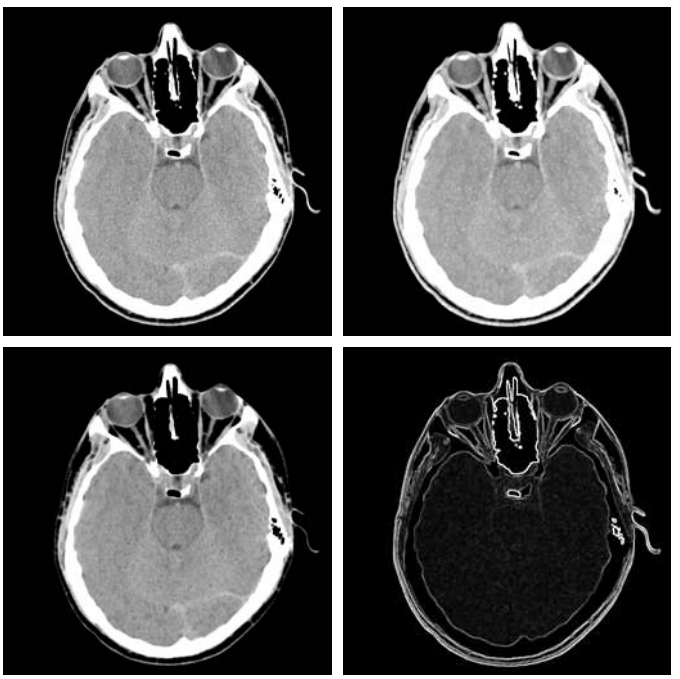


FIGURE 9.39
(a) 512 × 512 image of a head CT scan. (b) Dilation. (c) Erosion. (d) Morphological gradient, computed as the difference between (b) and (c). (Original image courtesy of Dr. David R. Pickens, Vanderbilt University.)

Top-hat and bottom-hat transformations

Combining image subtraction with openings and closings results in so-called *top-hat and bottom-hat* transformations. The *top-hat transformation* of a grayscale image f is defined as f minus its opening:

$$T_{\text{hat}}(f) = f - (f \circ b) \tag{9.6-12}$$

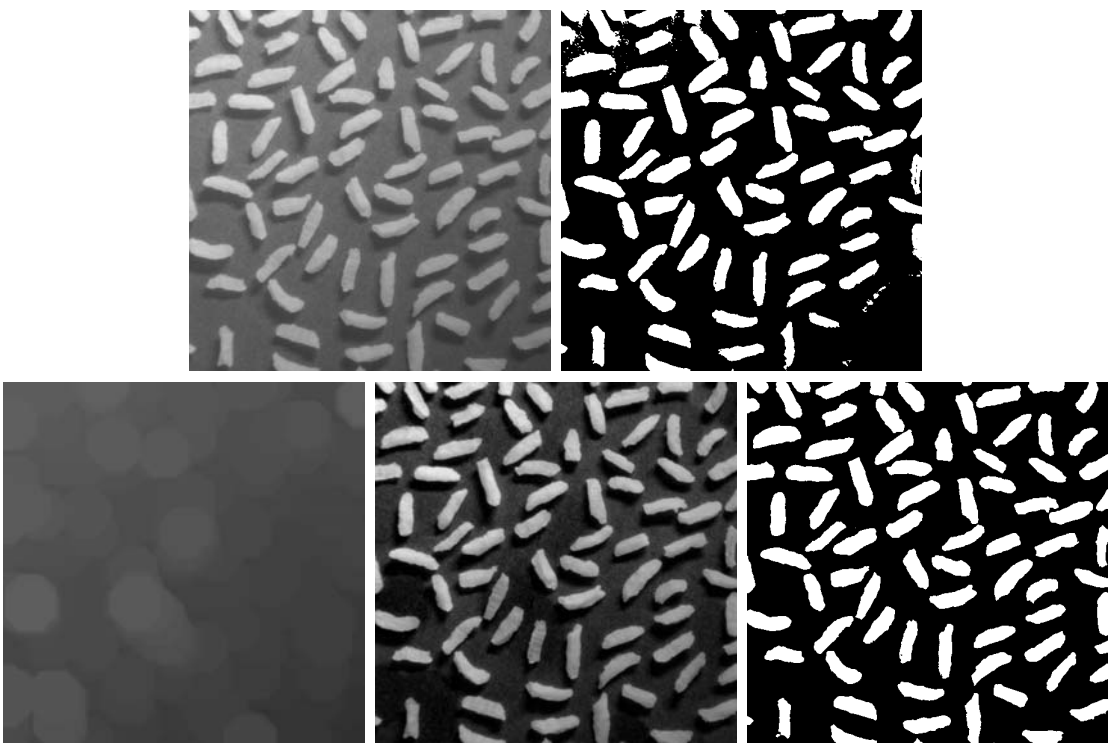
Similarly, the *bottom-hat transformation* of f is defined as the closing of f minus f:

$$B_{\text{hat}}(f) = (f \bullet b) - f \tag{9.6-13}$$

One of the principal applications of these transformations is in removing objects from an image by using a structuring element in the opening or closing operation that does not fit the objects to be removed. The difference operation then yields an image in which only the removed components remain. The top-hat transform is used for light objects on a dark background, and the bottom-hat transform is used for the converse. For this reason, the names *white top-hat* and *black top-hat*, respectively, are used frequently when referring to these two transformations.

An important use of top-hat transformations is in correcting the effects of nonuniform illumination. As we will see in the next chapter, proper (uniform) illumination plays a central role in the process of extracting objects from the background. This process, called *segmentation*, is one of the first steps performed in automated image analysis. A commonly used segmentation approach is to threshold the input image.

To illustrate, consider Fig. 9.40(a), which shows a 600 × 600 image of grains of rice. This image was obtained under nonuniform lighting, as evidenced by the darker area in the bottom, rightmost part of the image. Figure 9.40(b) shows the result of thresholding using Otsu's method, an optimal thresholding method discussed in Section 10.3.3. The net result of nonuniform illumination was to cause segmentation errors in the dark area (several grains of rice were not extracted from the background), as well as in the top left part of the image, where parts of the background were misclassified. Figure 9.40(c) shows the opening of the image with a disk of radius 40. This SE was large enough so that it would not fit in any of the objects. As a result, the objects were eliminated, leaving only an approximation of the background. The shading pattern is clear in this image. By subtracting this image from the original (i.e., performing a top-hat transformation), the background should become more uniform. This is indeed the case, as Fig. 9.40(d) shows. The background is not perfectly uniform, but the differences between light and dark extremes are less, and this was enough to yield a correct



a b
c d e

FIGURE 9.40 Using the top-hat transformation for *shading correction*. (a) Original image of size 600 × 600 pixels. (b) Thresholded image. (c) Image opened using a disk SE of radius 40. (d) Top-hat transformation (the image minus its opening). (e) Thresholded top-hat image.

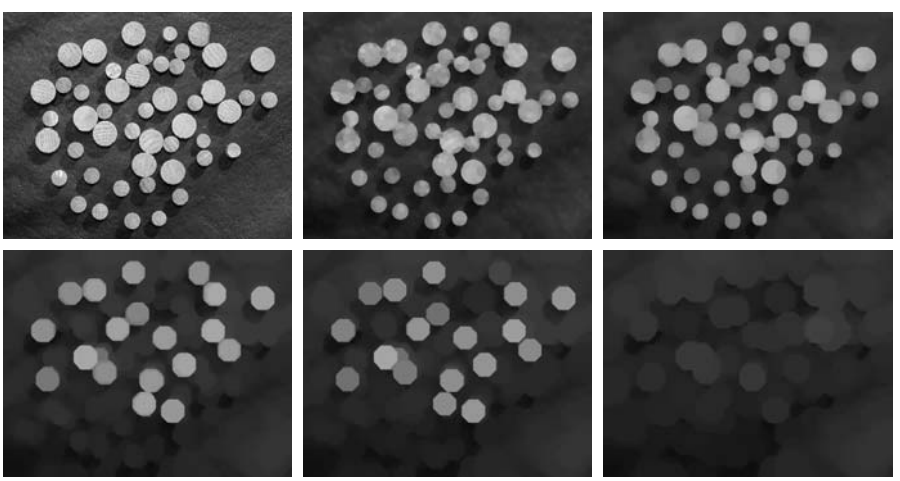
segmentation result in which all rice grains were detected, as Fig. 9.40(e) shows. This image was obtained using Otsu's method, as before.

Granulometry

In terms of image processing, *granulometry* is a field that deals with determining the size distribution of particles in an image. In practice, particles seldom are neatly separated, which makes particle counting by identifying individual particles a difficult task. Morphology can be used to estimate particle size distribution indirectly, without having to identify and measure every particle in the image.

The approach is simple in principle. With particles having regular shapes that are lighter than the background, the method consists of applying openings with SEs of increasing size. The basic idea is that opening operations of a particular size should have the most effect on regions of the input image that contain particles of similar size. For each opening, the *sum* of the pixel values in the opening is computed. This sum, sometimes called the *surface area*, decreases as a function of increasing SE size because, as we noted earlier, openings decrease the intensity of light features. This procedure yields a 1-D array of such numbers, with each element in the array being equal to the sum of the pixels in the opening for the size SE corresponding to that location in the array. To emphasize changes between successive openings, we compute the difference between adjacent elements of the 1-D array. To visualize the results, the differences are plotted. The peaks in the plot are an indication of the predominant size distributions of the particles in the image.

As an example, consider Fig. 9.41(a) which is an image of wood dowel plugs of two dominant sizes. The wood grain in the dowels are likely to introduce variations in the openings, so smoothing is a sensible pre-processing step. Figure 9.41(b) shows the image smoothed using the morphological smoothing



a b c
d e f

FIGURE 9.41 (a) 531 × 675 image of wood dowels. (b) Smoothed image. (c)–(f) Openings of (b) with disks of radii equal to 10, 20, 25, and 30 pixels, respectively. (Original image courtesy of Dr. Steve Eddins, The MathWorks, Inc.)

filter discussed earlier, with a disk of radius 5. Figures 9.41(c) through (f) show examples of image openings with disks of radii 10, 20, 25, and 30. Note in Fig. 9.41(d) that the intensity contribution due to the small dowels has been almost eliminated. In Fig. 9.41(e) the contribution of the large dowels has been significantly reduced, and in Fig. 9.41(f) even more so. (Observe in Fig. 9.41(e) that the large dowel near the top right of the image is much darker than the others because of its smaller size. This would be useful information if we had been attempting to detect defective dowels.)

Figure 9.42 shows a plot of the difference array. As mentioned previously, we expect significant differences (peaks in the plot) around radii at which the SE is large enough to encompass a set of particles of approximately the same diameter. The result in Fig. 9.42 has two distinct peaks, clearly indicating the presence of two dominant object sizes in the image.

Textural segmentation

Figure 9.43(a) shows a noisy image of dark blobs superimposed on a light background. The image has two textural regions: a region composed on large blobs on the right and a region on the left composed of smaller blobs. The objective is to find a boundary between the two regions based on their textural content (we discuss texture in Section 11.3.3). As noted earlier, the process of subdividing an image into regions is called *segmentation*, which is the topic of Chapter 10.

The objects of interest are darker than the background, and we know that if we close the image with a structuring element larger than the small blobs, these blobs will be removed. The result in Fig. 9.43(b), obtained by closing the input image using a disk with a radius of 30 pixels, shows that indeed this is the case (the radius of the blobs is approximately 25 pixels). So, at this point, we have an image with large, dark blobs on a light background. If we *open* this image with a structuring element that is large relative to the separation between these blobs, the net result should be an image in which the light patches between the blobs are removed, leaving the dark blobs and now equally dark patches between these blobs. Figure 9.43(c) shows the result, obtained using a disk of radius 60.

Performing a morphological gradient on this image with, say, a 3×3 SE of 1s, will give us the boundary between the two regions. Figure 9.43(d) shows the boundary obtained from the morphological gradient operation superimposed

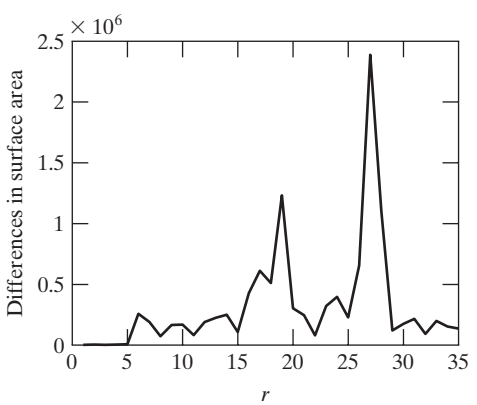


FIGURE 9.42 Differences in surface area as a function of SE disk radius, r. The two peaks are indicative of two dominant particle sizes in the image.

a b
c d

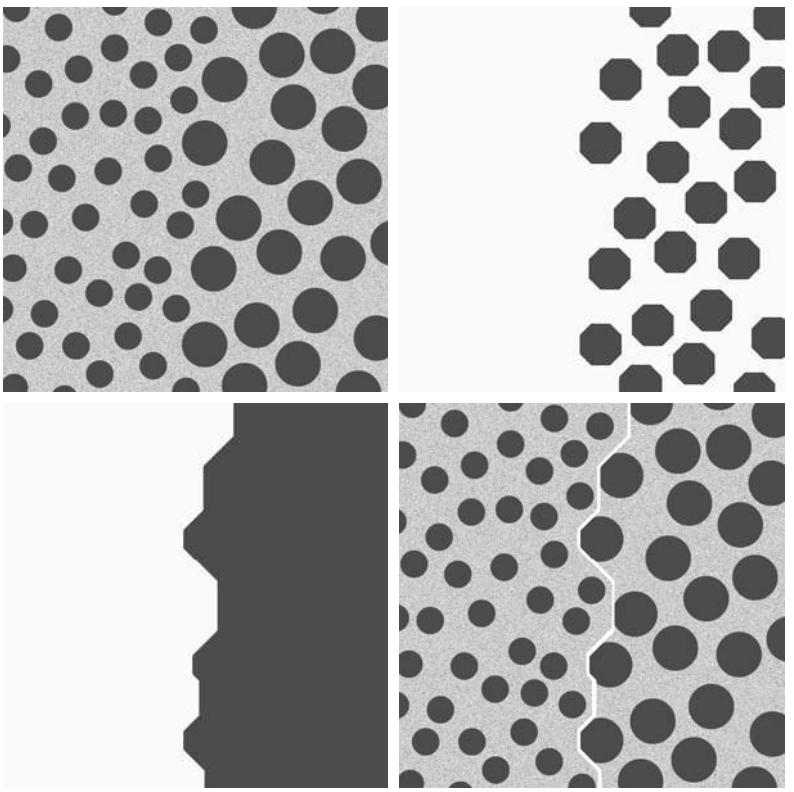


FIGURE 9.43 Textural segmentation. (a) A 600 × 600 image consisting of two types of blobs. (b) Image with small blobs removed by closing (a). (c) Image with light patches between large blobs removed by opening (b). (d) Original image with boundary between the two regions in (c) superimposed. The boundary was obtained using a morphological gradient operation.

on the original image. All pixels to the right of this boundary are said to belong to the texture region characterized by large blobs, and conversely for the pixels on the left of the boundary. You will find it instructive to work through this example in more detail using the graphical analogy for opening and closing illustrated in Fig. 9.36.

9.6.4 Gray-Scale Morphological Reconstruction

Gray-scale morphological reconstruction is defined basically in the same manner introduced in Section 9.5.9 for binary images. Let f and g denote the *marker* and *mask* images, respectively. We assume that both are gray-scale images of the same size and that $f \leq g$. The *geodesic dilation* of size 1 of f with respect to g is defined as

It is understood that these expressions are functions of (x, y). We omit the coordinates to simplify the notation.

$$D_g^{(1)}(f) = (f \oplus b) \wedge g \tag{9.6-14}$$

where $\wedge$ denotes the point-wise minimum operator. This equation indicates that the geodesic dilation of size 1 is obtained by first computing the dilation of f by b and then selecting the minimum between the result and g at every point (x,y). The dilation is given by Eq. (9.6-2) if b is a flat SE or by Eq. (9.6-4) if it is not. The geodesic dilation of size n of f with respect to g is defined as

$$D_g^{(n)}(f) = D_g^{(1)}[D_g^{(n-1)}(f)] \tag{9.6-15}$$

with $D_g^{(0)}(f) = f$.

Similarly, the *geodesic erosion* of size 1 of f with respect to g is defined as

$$E_g^{(1)}(f) = (f \ominus b) \vee g \tag{9.6-16}$$

where $\vee$ denotes the point-wise maximum operator. The geodesic erosion of size n is defined as

$$E_g^{(n)}(f) = E_g^{(1)}\left[E_g^{(n-1)}(f)\right] \tag{9.6-17}$$

with $E_g^{(0)}(f) = f$.

See Problem 9.33 for a list of dual relationships between expressions in this section.

The *morphological reconstruction by dilation* of a gray-scale mask image, g, by a gray-scale marker image, f, is defined as the geodesic dilation of f with respect to g, iterated until stability is reached; that is,

$$R_g^D(f) = D_g^{(k)}(f) \tag{9.6-18}$$

with k *such* that $D_g^{(k)}(f) = D_g^{(k+1)}(f)$. The *morphological reconstruction by erosion* of g by f is similarly defined as

$$R_g^E(f) = E_g^{(k)}(f) \tag{9.6-19}$$

with k *such* that $E_g^{(k)}(f) = E_g^{(k+1)}(f)$.

As in the binary case, opening by reconstruction of gray-scale images first erodes the input image and uses it as a marker. The *opening by reconstruction* of size n of an image f is defined as the reconstruction by dilation of f from the erosion of size n of f; that is,

$$O_R^{(n)}(f) = R_f^D\left[(f \ominus nb)\right] \tag{9.6-20}$$

where $(f \ominus nb)$ denotes n erosions of f by b, as explained in Section 9.5.7. Recall from the discussion of Eq. (9.5-27) for binary images that the objective of opening by reconstruction is to preserve the shape of the image components that remain after erosion.

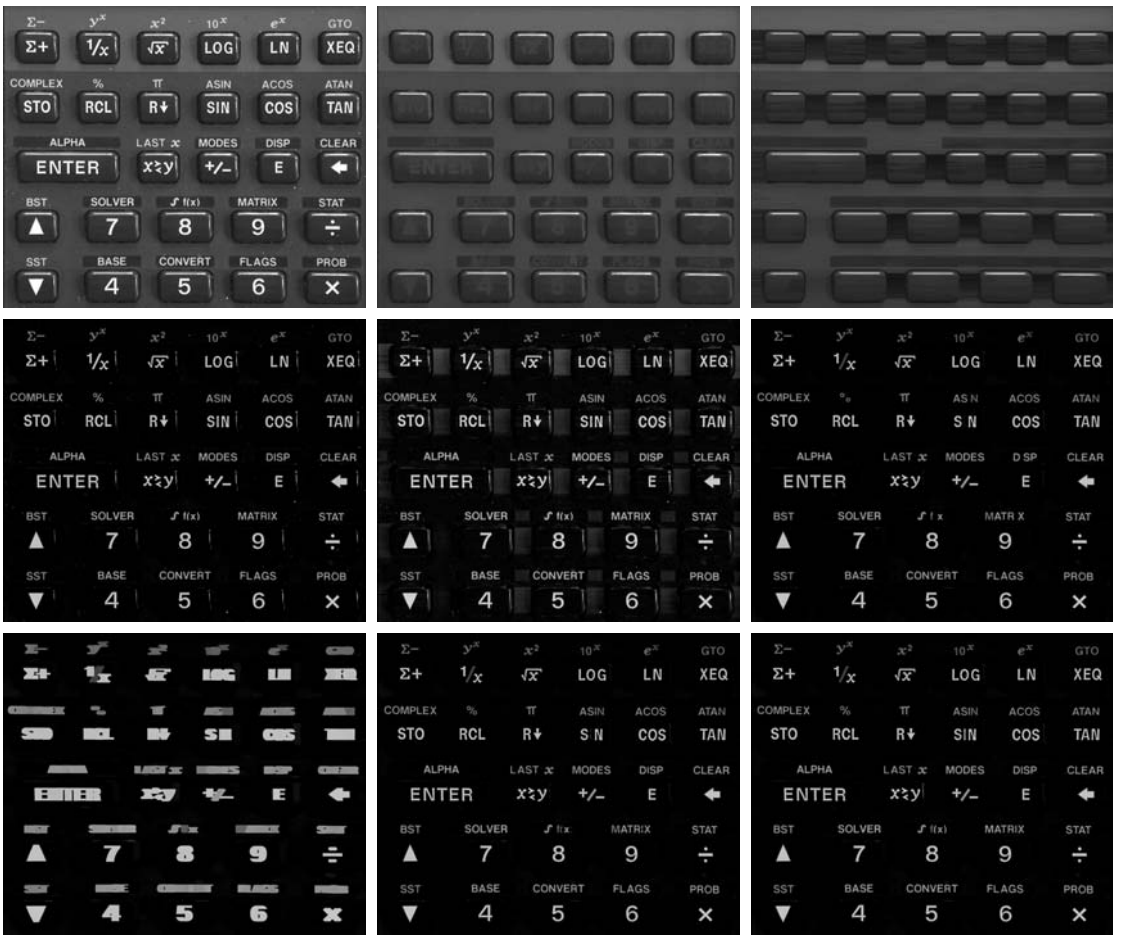
Similarly, the *closing by reconstruction* of size n of an image f is defined as the reconstruction by erosion of f from the dilation of size n of f; that is,

$$C_R^{(n)}(f) = R_f^E\left[(f \oplus nb)\right] \tag{9.6-21}$$

where $(f \oplus nb)$ denotes n dilations of f by b. Because of duality, the closing by reconstruction of an image can be obtained by complementing the image, obtaining the opening by reconstruction, and complementing the result. Finally, as the following example shows, a useful technique called *top-hat by reconstruction* consists of subtracting from an image its opening by reconstruction.

EXAMPLE 9.11: Using morphological reconstruction to flatten a complex background.

■ In this example, we illustrate the use of gray-scale reconstruction in several steps to normalize the irregular background of the image in Fig. 9.44(a), leaving only the text on a background of constant intensity. The solution of this problem is a good illustration of the power of morphological concepts. We begin by suppressing the horizontal reflection on the top of the keys. The reflections are wider than any single character in the image, so we should be able to suppress them by performing an opening by reconstruction using a long horizontal line in the erosion operation. This operation will yield the background containing the keys and their reflections. Subtracting this from



a b c
d e f
g h i

FIGURE 9.44 (a) Original image of size 1134 × 1360 pixels. (b) Opening by reconstruction of (a) using a horizontal line 71 pixels long in the erosion. (c) Opening of (a) using the same line. (d) Top-hat by reconstruction. (e) Top-hat. (f) Opening by reconstruction of (d) using a horizontal line 11 pixels long. (g) Dilation of (f) using a horizontal line 21 pixels long. (h) Minimum of (d) and (g). (i) Final reconstruction result. (Images courtesy of Dr. Steve Eddins, The MathWorks, Inc.)

the original image (i.e., performing a top-hat by reconstruction) will eliminate the horizontal reflections and variations in background from the original image.

Figure 9.44(b) shows the result of opening by reconstruction of the original image using a horizontal line of size 1 × 71 pixels in the erosion operation. We could have used just an opening to remove the characters, but the resulting background would not have been as uniform, as Fig. 9.44(c) shows (for example, compare the regions between the keys in the two images). Figure 9.44(d)

shows the result of subtracting Fig. 9.44(b) from Fig. 9.44(a). As expected, the horizontal reflections and variations in background were suppressed. For comparison, Fig. 9.44(e) shows the result of performing just a top-hat transformation (i.e., subtracting the "standard" opening from the image, as discussed earlier in this section). As expected from the characteristics of the background in Fig. 9.44(c), the background in Fig. 9.44(e) is not nearly as uniform as in Fig. 9.44(d).

The next step is to remove the vertical reflections from the edges of keys, which are quite visible in Fig. 9.44(d). We can do this by performing an opening by reconstruction with a line SE whose width is approximately equal to the reflections (about 11 pixels in this case). Figure 9.44(f) shows the result of performing this operation on Fig. 9.44(d). The vertical reflections were suppressed, but so were thin, vertical strokes that are valid characters (for example, the I in SIN), so we have to find a way to restore the latter. The suppressed characters are very close to the other characters so, if we dilate the remaining characters horizontally, the dilated characters will overlap the area previously occupied by the suppressed characters. Figure 9.44(g), obtained by dilating Fig. 9.44(f) with a line SE of size 1×21, shows that indeed this is case.

All that remains at this point is to restore the suppressed characters. Consider an image formed as the point-wise minimum between the dilated image in Fig. 9.44(g) and the top-hat by reconstruction in Fig. 9.44(d). Figure 9.44(h) shows the minimum image (although this result appears to be close to our objective, note that the I in SIN is still missing). By using this image as a marker and the dilated image as the mask in gray-scale reconstruction [Eq. (9.6-18)] we obtain the final result in Fig. 9.44(i). This image shows that all characters were properly extracted from the original, irregular background, including the background of the keys. The background in Fig. 9.44(i) is uniform throughout. ■

Summary

The morphological concepts and techniques introduced in this chapter constitute a powerful set of tools for extracting features of interest in an image. One of the most appealing aspects of morphological image processing is the extensive set-theoretic foundation from which morphological techniques have evolved. A significant advantage in terms of implementation is the fact that dilation and erosion are primitive operations that are the basis for a broad class of morphological algorithms. As shown in the following chapter, morphology can be used as the basis for developing image segmentation procedures with numerous applications. As discussed in Chapter 11, morphological techniques also play a major role in procedures for image description.

References and Further Reading

The book by Serra [1982] is a fundamental reference on morphological image processing. See also Serra [1988], Giardina and Dougherty [1988], and Haralick and Shapiro [1992]. Additional early references relevant to our discussion include Blum [1967], Lantuéjoul [1980], Maragos [1987], and Haralick et al. [1987]. For an overview of both binary and gray-scale morphology, see Basart and Gonzalez [1992] and Basart et al.

[1992]. This set of references provides ample basic background for the material covered in Sections 9.1 through 9.4. For a good overview of the material in Sections 9.5 and 9.6, see the book by Soille [2003].

Important issues of implementing morphological algorithms such as the ones given in Section 9.5 and 9.6 are exemplified in the papers by Jones and Svalbe [1994], Park and Chin [1995], Sussner and Ritter [1997], Anelli et al. [1998], and Shaked and Bruckstein [1998]. A paper by Vincent [1993] is especially important in terms of practical details for implementing gray-scale morphological algorithms. See also the book by Gonzalez, Woods, and Eddins [2004].

For additional reading on the theory and applications of morphological image processing, see the book by Goutsias and Bloomberg [2000] and a special issue of *Pattern Recognition* [2000]. See also a compilation of references by Rosenfeld [2000]. The books by Marchand-Maillet and Sharaiha [2000] on binary image processing and by Ritter and Wilson [2001] on image algebra also are of interest. Current work in the application of morphological techniques for image processing is exemplified in the papers by Kim [2005] and Evans and Liu [2006].

Problems

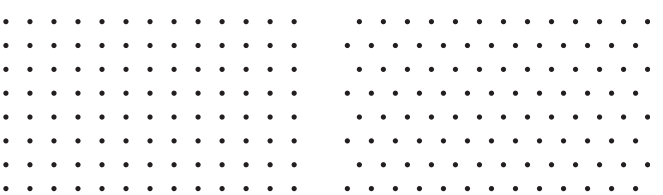
Detailed solutions to the problems marked with a star can be found in the book Web site. The site also contains suggested projects based on the material in this chapter.

9.1 Digital images in this book are embedded in square grid arrangements and pixels are allowed to be 4-, 8-, or *m*-connected. However, other grid arrangements are possible. Specifically, a hexagonal grid arrangement that leads to 6-connectivity, is used sometimes (see the following figure).

(a) How would you convert an image from a square grid to a hexagonal grid?

(b) Discuss the shape invariance to rotation of objects represented in a square grid as opposed to a hexagonal grid.

(c) Is it possible to have ambiguous diagonal configurations in a hexagonal grid, as is the case with 8-connectivity? (See Section 2.5.2.)



9.2 ★ **(a)** Give a morphological algorithm for converting an 8-connected binary boundary to an *m*-connected boundary (see Section 2.5.2). You may assume that the boundary is fully connected and that it is one pixel thick.

(b) Does the operation of your algorithm require more than one iteration with each structuring element? Explain your reasoning.

(c) Is the performance of your algorithm independent of the order in which the structuring elements are applied? If your answer is yes, prove it; otherwise give an example that illustrates the dependence of your procedure on the order of application of the structuring elements.

9.3 Erosion of a set A by structuring element B is a subset of A as long as the origin of B is contained by B. Give an example in which the erosion $A \ominus B$ lies outside, or partially outside, A.

9.4 The following four statements are true. Advance an argument that establishes the reason(s) for their validity. Part (a) is true in general. Parts (b) through (d) are true only for *digital* sets. To show the validity of (b) through (d), draw a discrete, square grid (as shown in Problem 9.1) and give an example for each case using sets composed of points on this grid. (*Hint:* Keep the number of points in each case as small as possible while still establishing the validity of the statements.)

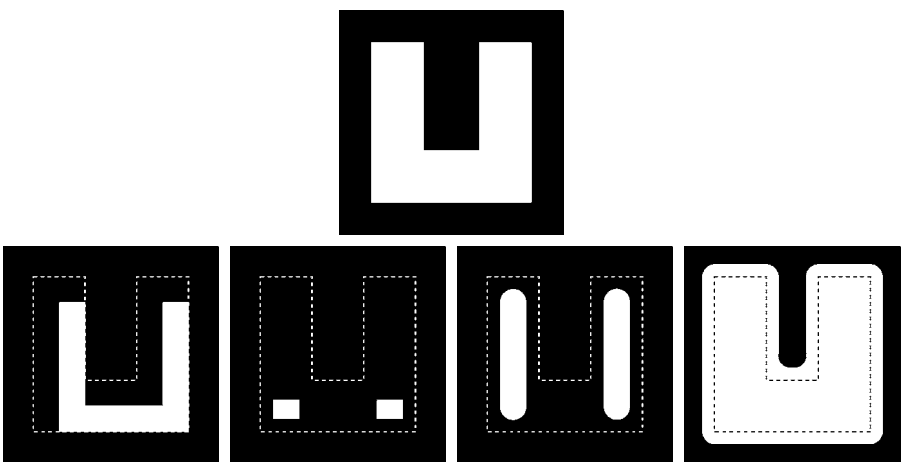
★ **(a)** The erosion of a convex set by a convex structuring element is a convex set.

★ **(b)** The dilation of a convex set by a convex structuring element is not necessarily always convex.

(c) The points in a convex digital set are not always connected.

(d) It is possible to have a set of points in which a line joining every pair of points in the set lies within the set but the set is not convex.

★**9.5** With reference to the image shown, give the structuring element and morphological operation(s) that produced each of the results shown in images (a) through (d). Show the origin of each structuring element clearly. The dashed lines show the boundary of the original set and are included only for reference. Note that in (d) all corners are rounded.



(a) (b) (c) (d)

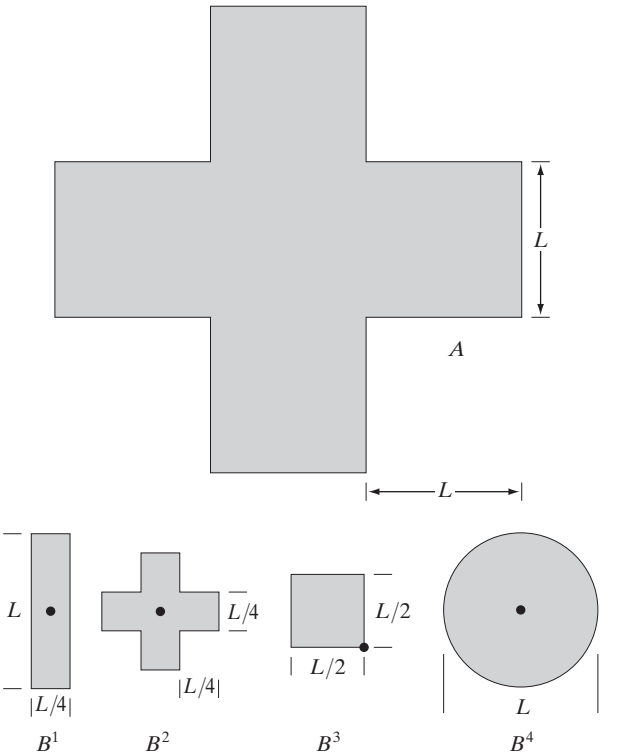
9.6 Let A denote the set shown shaded in the following figure. Refer to the structuring elements shown (the black dots denote the origin). Sketch the result of the following morphological operations:

(a) $(A \ominus B^4) \oplus B^2$

(b) $(A \ominus B^1) \oplus B^3$

(c) $(A \oplus B^1) \oplus B^3$

(d) $(A \oplus B^3) \ominus B^2$



★9.7 **(a)** What is the limiting effect of repeatedly dilating an image? Assume that a trivial (one point) structuring element is not used.

(b) What is the smallest image from which you can start in order for your answer in part (a) to hold?

9.8 **(a)** What is the limiting effect of repeatedly eroding an image? Assume that a trivial (one point) structuring element is not used.

(b) What is the smallest image from which you can start in order for your answer in part (a) to hold?

★9.9 An alternative definition of erosion is

$$A \ominus B = \{w \in Z^2 | w + b \in A, \text{ for every } b \in B\}$$

Show that this definition is equivalent to the definition in Eq. (9.2-1).

9.10 **(a)** Show that the definition of erosion given in Problem 9.9 is equivalent to yet another definition of erosion:

$$A \ominus B = \bigcap_{b \in B} (A)_{-b}$$

(If $-b$ is replaced with b, this expression is called the *Minkowsky subtraction* of two sets.)

(b) Show that the expression in (a) also is equivalent to the definition in Eq. (9.2-1).

★**9.11** An alternative definition of dilation is

$$A \oplus B = \{w \in Z^2 | w = a + b, \text{ for some } a \in A \text{ and } b \in B\}$$

Show that this definition and the definition in Eq. (9.2-3) are equivalent.

9.12 **(a)** Show that the definition of dilation given in Problem 9.11 is equivalent to yet another definition of dilation:

$$A \oplus B = \bigcup_{b \in B} (A)_b$$

(This expression also is called the *Minkowsky addition* of two sets.)

(b) Show that the expression in (a) also is equivalent to the definition in Eq. (9.2-3).

9.13 Prove the validity of the duality expression in Eq. (9.2-6).

★**9.14** Prove the validity of the duality expressions $(A \bullet B)^c = (A^c \circ \hat{B})$ and $(A \circ B)^c = (A^c \bullet \hat{B})$.

9.15 Prove the validity of the following expressions:

★**(a)** $A \circ B$ is a subset (subimage) of A.

(b) If C is a subset of D, then $C \circ B$ is a subset of $D \circ B$.

(c) $(A \circ B) \circ B = A \circ B$.

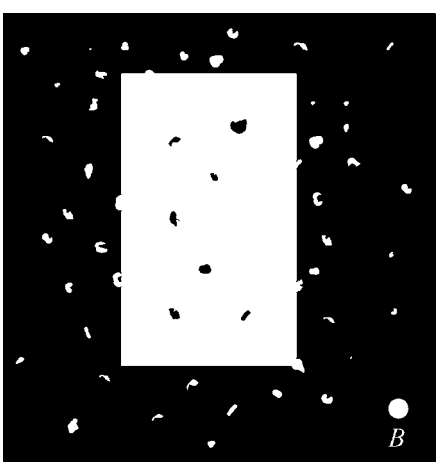
9.16 Prove the validity of the following expressions (assume that the origin of B is contained in B and that Problems 9.14 and 9.15 are true):

(a) A is a subset (subimage) of $A \bullet B$.

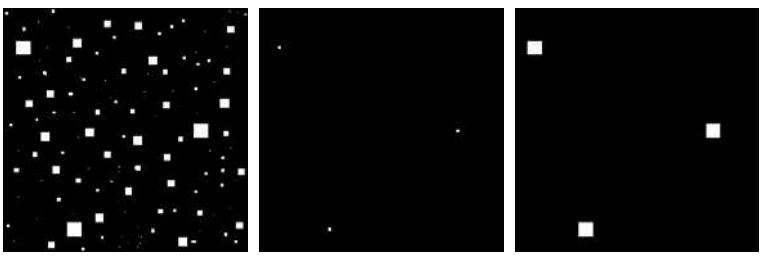
(b) If C is a subset of D, then $C \bullet B$ is a subset of $D \bullet B$.

(c) $(A \bullet B) \bullet B = A \bullet B$.

9.17 Refer to the image and structuring element shown. Sketch what the sets C, D, E, and F would look like in the following sequence of operations: $C = A \ominus B$; $D = C \oplus B$; $E = D \oplus B$; and $F = E \ominus B$. The initial set A consists of all the image components shown in white, with the exception of the structuring element B. Note that this sequence of operations is simply the opening of A by B, followed by the closing of that opening by B. You may assume that B is just large enough to enclose each of the noise components.



★9.18 Consider the three binary images shown in the following figure. The image on the left is composed of squares of sizes 1, 3, 5, 7, 9, and 15 pixels on the side. The image in the middle was generated by eroding the image on the left with a square structuring element of 1s, of size 13×13 pixels, with the objective of eliminating all the squares, except the largest ones. Finally, the image on the right is the result of dilating the image in the center with the same structuring element, with the objective of restoring the largest squares. You know that erosion followed by dilation is the opening of an image, and you know also that opening generally does not restore objects to their original form. Explain why full reconstruction of the large squares was possible in this case.

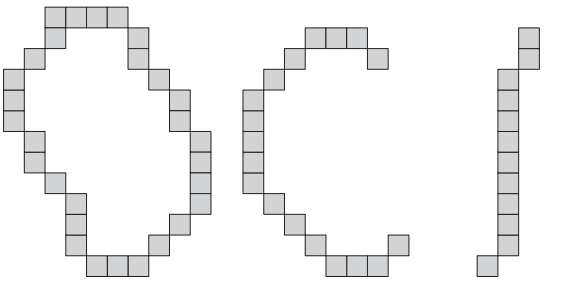


9.19 Sketch the result of applying the hit-or-miss transform to the image and structuring element shown. Indicate clearly the origin and border you selected for the structuring element.

Image

Structuring element

★9.20 Three features (lake, bay, and line segment) useful for differentiating thinned objects in an image are shown in the following page. Develop a morphological/logical algorithm for differentiating among these shapes. The input to your algorithm would be one of these three shapes. The output must be the identity of the input. You may assume that the features are 1 pixel thick and that each is fully connected. However, they can appear in any orientation.



Lake Bay Line segment

9.21 Discuss what you would expect the result to be in each of the following cases:

(a) The starting point of the hole filling algorithm of Section 9.5.2 is a point on the boundary of the object.

(b) The starting point in the hole filling algorithm is outside of the boundary.

(c) Sketch what the convex hull of the figure in Problem 9.6 would look like as computed with the algorithm given in Section 9.5.4. Assume that $L = 3$ pixels.

9.22 ★**(a)** Discuss the effect of using the structuring element in Fig. 9.15(c) for boundary extraction, instead of the one shown in Fig. 9.13(b).

(b) What would be the effect of using a 3×3 structuring element composed of all 1s in the hole filling algorithm of Eq. (9.5-2), instead of the structuring element shown in Fig. 9.15(c)?

9.23 ★**(a)** Propose a method (using any of the techniques from Sections 9.1 through 9.5) for automating the example in Fig. 9.16. You may assume that the spheres do not touch each other and that none touch the border of the image.

(b) Repeat (a), but allowing the spheres to touch in arbitrary ways, including touching the border.

★**9.24** The algorithm given in Section 9.5.3 for extracting connected components requires that a point be known in each connected component in order to extract them all. Suppose that you are given a binary image containing an arbitrary (unknown) number of connected components. Propose a completely automated procedure for extracting all connected components. Assume that points belonging to connected components are labeled 1 and background points are labeled 0.

9.25 Give an expression based on reconstruction by dilation capable of extracting all the holes in a binary image.

9.26 With reference to the hole-filling algorithm in Section 9.5.9:

(a) Explain what would happen if all border points of f are 1.

(b) If the result in (a) gives the result that you would expect, explain why. If it does not, explain how you would modify the algorithm so that it works as expected.

★**9.27** Explain what would happen in binary erosion and dilation if the structuring element is a single point, valued 1. Give the reason(s) for your answer.

9.28 As explained in Eq. (9.5-27) and Section 9.6.4, opening by reconstruction preserves the shape of the image components that remain after erosion. What does closing by reconstruction do?

★**9.29** Show that geodesic erosion and dilation (Section 9.5.9) are duals with respect to set complementation. That is, show that $E_G^{(n)}(F) = \left[D_{G^c}^{(1)}\left[D_{G^c}^{(n-1)}(F^c)\right]\right]^c$ and, conversely, that $D_G^{(n)}(F) = \left[E_{G^c}^{(1)}\left[E_{G^c}^{(n-1)}(F^c)\right]\right]^c$. Assume that the structuring element is symmetric about its origin.

9.30 Show that reconstruction by dilation and reconstruction by erosion (Section 9.5.9) are duals with respect to set complementation. That is, show that $R_G^D(F) = \left[R_{G^c}^E(F^c)\right]^c$ and, vice versa, that $R_G^E(F) = \left[R_{G^c}^D(F^c)\right]^c$. Assume that the structuring element is symmetric about its origin.

★**9.31** Advance an argument showing that:

(a) $[(F \ominus nB)]^c = (F^c \oplus n\hat{B})$, where $(F \ominus nB)$ indicates n erosions of F by B.

(b) $[(F \oplus nB)]^c = (F^c \ominus n\hat{B})$.

9.32 Show that binary closing by reconstruction is the dual of opening by reconstruction with respect to set complementation: $O_R^{(n)}(F) = \left[C_R^{(n)}(F^c)\right]^c$, and similarly that $C_R^{(n)}(F) = \left[O_R^{(n)}(F^c)\right]^c$. Assume that the structuring element is symmetric with respect to its origin.

9.33 Prove the validity of the following gray-scale morphology expressions. You may assume that b is a flat structuring element. Recall that $f^c(x, y) = -f(x, y)$ and that $\hat{b}(x, y) = b(-x - y)$.

★ **(a)** Duality of erosion and dilation: $(f \ominus b)^c = f^c \oplus \hat{b}$ and $(f \oplus b)^c = f^c \ominus \hat{b}$.

(b) $(f \bullet b)^c = f^c \circ \hat{b}$ and $(f \circ b)^c = f^c \bullet \hat{b}$.

★ **(c)** $D_g^{(n)}(f) = \left[E_{g^c}^{(1)}\left[E_{g^c}^{(n-1)}(f^c)\right]\right]^c$ and $E_g^{(n)}(f) = \left[D_{g^c}^{(1)}\left[D_{g^c}^{(n-1)}(f^c)\right]\right]^c$. Assume a symmetric structuring element.

(d) $R_g^D(f) = \left[R_{g^c}^E(f^c)\right]^c$ and $R_g^E(f) = \left[R_{g^c}^D(f^c)\right]^c$.

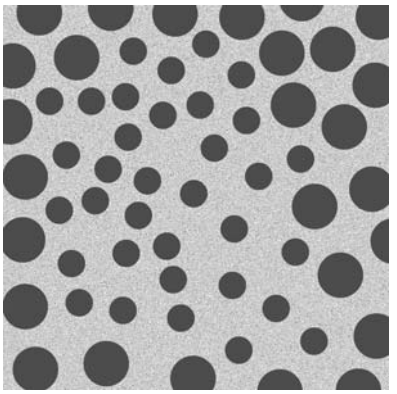
(e) $[(f \ominus nb)]^c = (f^c \oplus n\hat{b})$, where $(f \ominus nb)$ indicates n erosions of f by b. Also, $[(f \oplus nb)]^c = (f^c \ominus n\hat{b})$.

(f) $O_R^{(n)}(f) = \left[C_R^{(n)}(f^c)\right]^c$ and $C_R^{(n)}(f) = \left[O_R^{(n)}(f^c)\right]^c$. Assume that the structuring element is symmetric with respect to its origin.

9.34 In Fig. 9.43, a boundary between distinct texture regions was established without difficulty. Consider the image at the top of the facing page, which shows a region of small circles enclosed by a region of larger circles.

(a) Would the method used to generate Fig. 9.43(d) work with this image as well? Explain your reasoning, including any assumptions that you need to make for the method to work.

(b) If your answer was yes, sketch what the boundary will look like.



9.35 A gray-scale image, $f(x, y)$, is corrupted by nonoverlapping noise spikes that can be modeled as small, cylindrical artifacts of radii $R_{\min} \leq r \leq R_{\max}$ and amplitude $A_{\min} \leq a \leq A_{\max}$.

★ **(a)** Develop a morphological filtering approach for cleaning up the image.

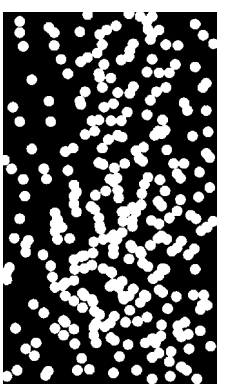
(b) Repeat (a), but now assume that there is overlapping of, at most, four noise spikes.

9.36 A preprocessing step in an application of microscopy is concerned with the issue of isolating individual round particles from similar particles that overlap in groups of two or more particles (see following image). Assuming that all particles are of the same size, propose a morphological algorithm that produces three images consisting respectively of

★ **(a)** Only of particles that have merged with the boundary of the image.

(b) Only overlapping particles.

(c) Only nonoverlapping particles.



9.37 A high-technology manufacturing plant wins a government contract to manufacture high-precision washers of the form shown in the following figure. The terms of the contract require that the shape of all washers be inspected by an imaging system. In this context, shape inspection refers to deviations from round on the inner and outer edges of the washers. You may assume the following: (1) A "golden" (perfect with respect to the problem) image of an acceptable washer is available; and (2) the imaging and positioning systems ultimately used in the system will have an accuracy high enough to allow you to ignore errors due to digitalization and positioning. You are hired as a consultant to help specify the visual inspection part of the system. Propose a solution based on morphological/logic operations. Your answer should be in the form of a block diagram.