
Computer Vision & Digital Image Processing

Morphological Image Processing II

Opening and Closing

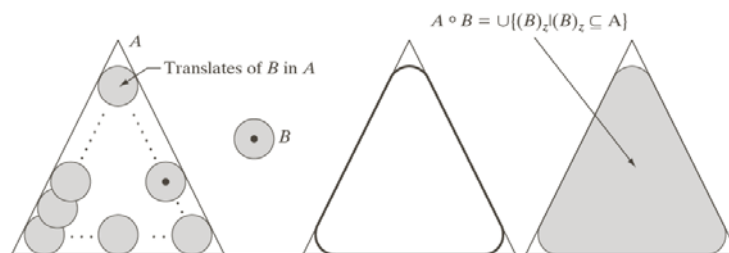
- Morphological *opening* generally
 - Smooths the contour of an object
 - Breaks narrow isthmuses
 - Eliminates thin protrusions
- Morphological *closing* generally
 - Smooths contour sections
 - Fuses narrow breaks and long thin gulfs
 - Eliminates small holes
 - Fills gaps in a contour

Opening and Closing

- The opening of set A by structuring element B , denoted $A \circ B$, is defined as
 - $A \circ B = (A \ominus B) \oplus B$
- Thus, opening is defined as the erosion of A by B followed by a dilation of the result by B
- The closing of set A by structuring element B , denoted $A \bullet B$, is defined as
 - $A \bullet B = (A \oplus B) \ominus B$
- Thus, closing is defined as the dilation of A by B followed by an erosion of the result by B

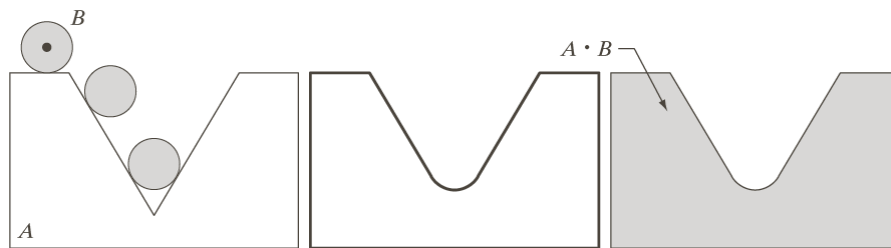
Opening: Geometric Interpretation

- Suppose the structuring element, B , is viewed as a rolling ball
- The boundary of $A \circ B$ is established by all points in B that reach the *farthest* into the boundary of A as B is rolled about the *inside* of this boundary
- The opening of A by B is obtained by taking the union of all translates of B that fit into A
 - $A \circ B = \cup \{(B)_z \mid (B)_z \subseteq A\}$

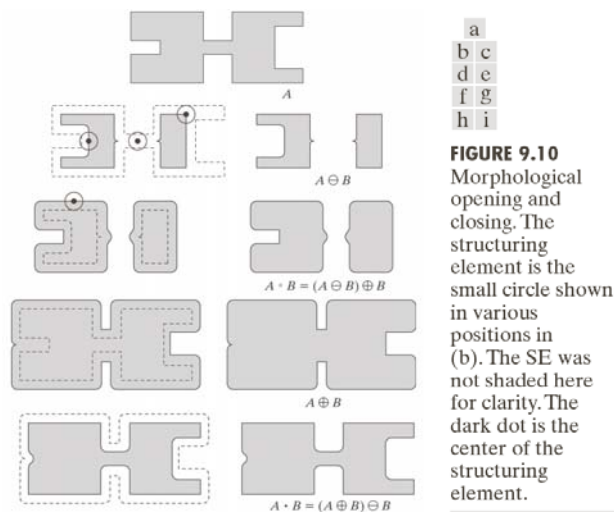


Closing: Geometric Interpretation

- Suppose the structuring element, B , is viewed as a rolling ball
- The boundary of $A \bullet B$ is established by all points in B that reach the *closest* to the boundary of A as B is rolled about the *outside* of this boundary
 - $A \bullet B = \{w \mid (B)_z \cap A \neq \emptyset \text{ for any translate of } (B)_z \text{ containing } w\}$



Opening and Closing Examples



Morphological Filtering

- Morphological operators can be used to construct filters similar in concept to spatial filters
- If the filtering objective in question is to eliminate noise and distort data of interest as little as possible, then a morphological filter consisting of an opening followed by a closing can be used
- Recall from the definitions of opening and closing that the more primitive operations of erosion and dilation are used

– Opening

$$A \circ B = (A \ominus B) \oplus B$$

erode
dilate

Closing

$$A \bullet B = (A \oplus B) \ominus B$$

dilate
erode

Morphological Filtering Example

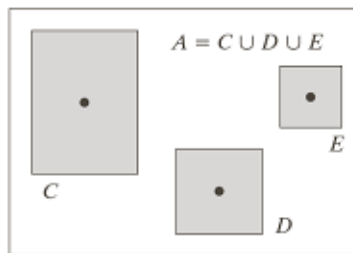


FIGURE 9.11

(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.)

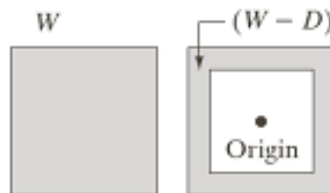
Hit-or-Miss Transform

- The morphological hit-or-miss transform is a basic tool for shape detection
- The basic intent is to find the location of a known shape within a set of shapes
- Assume a set A consists of a set of shapes (subsets) C , D , and E
- It is desired to find the location of one of the shapes, D
- Let the origin of each shape be its center of gravity



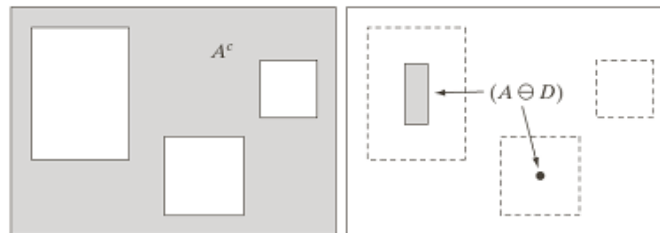
Hit-or-Miss Transform

- Let D be enclosed by a small window, W
- The *local background* of D with respect to W is defined as the set difference $(W - D)$



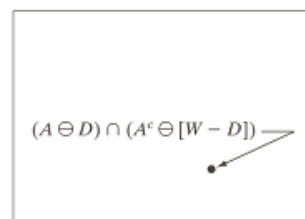
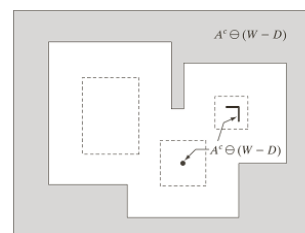
Hit-or-Miss Transform

- The complement of A , A^c , is needed in the transform operation
- Let A be eroded by D
- The erosion of A by D is the set of locations of the *origin* of D , such that D is completely contained in A
- Viewed geometrically, this is the set of all locations of the origin of D at which D found a match (hit) in A



Hit-or-Miss Transform

- Erode the complement of A , A^c , by the local background set ($W-D$)
- If we now compute the intersection of the two computed values, this give use the location of D
- If B denotes the set composed of D and its background, the match (or set of matches) of B in A , denoted $A^{\oplus} B$, is
- $A^{\oplus} B = (A \ominus D) \cap [A^c \ominus (W-D)]$



Boundary Extraction

- The boundary of a set A , denoted $\beta(A)$, is obtained by first eroding A by B and then performing the set difference between A and its erosion
 - $\beta(A) = A - (A \ominus B)$
- B , as always, is a suitable structuring element

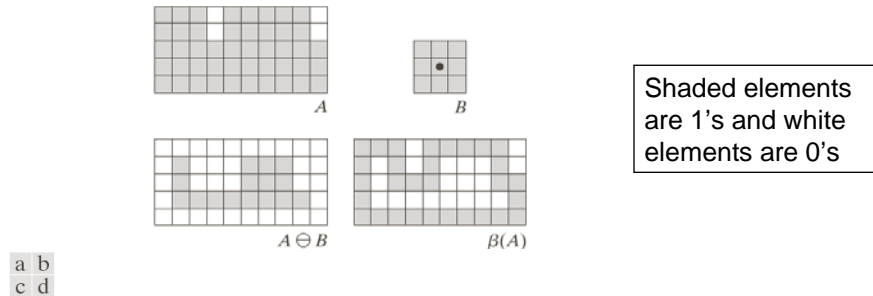


FIGURE 9.13 (a) Set A . (b) Structuring element B . (c) A eroded by B . (d) Boundary, given by the set difference between A and its erosion.

Boundary Extraction

- Although a 3x3 structuring element is commonly used, it is not unique
- A 5x5 structuring element of 1's would result in a boundary between 2 and 3 pixels thick
- When the origin of the structuring element, B , is on the edges of the set, part of B may be outside the image
 - The normal treatment of this condition is to assume that values outside the borders of the image are 0

Boundary Extraction



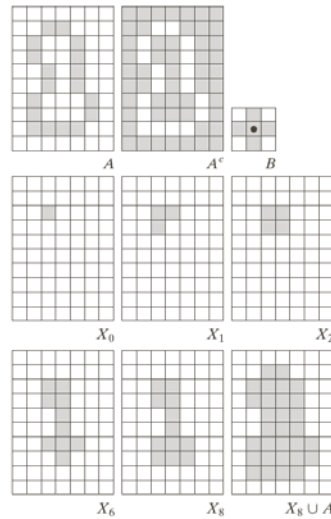
FIGURE 9.14
(a) A simple binary image, with 1s represented in white. (b) Result of using Eq. (9.5-1) with the structuring element in Fig. 9.13(b).

Region Filling

- Assume A denotes a set containing a subset whose elements are 8-connected boundary points of a region.
- Beginning with a point p inside the boundary, the object is to fill the entire region (with 1's)
- Adopt the convention that all nonboundary (background) points are labeled 0 and assign a value of 1 to p to begin
- The following procedure fills the region with 1's
 - $X_k = (X_{k-1} \oplus B) \cap A^c \quad k = 1, 2, 3, \dots$
- Where $X_0 = p$ and B is the symmetric structuring element:

$$B = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$
- The algorithm terminates at step k if $X_k = X_{k-1}$
- The set $X_k \cup A$ contains the filled set and its boundary

Region Filling

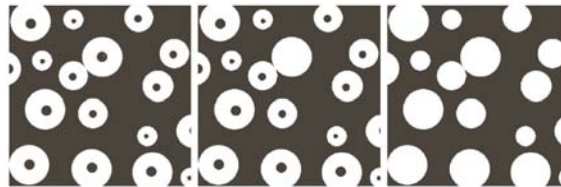


a	b	c
d	e	f
g	h	i

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)].

Region Filling

- The dilation process in the algorithm would fill the entire area if left unchecked
- The intersection with A^c at each iteration limits the result to inside the region of interest
 - This is the first example of how a morphological process can be conditioned to meet a desired property
- This process is called *conditional dilation*



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.

Extraction of Connected Components

- Concepts of connectivity and connected components are used
- Let Y represent a connected component in a set A and assume that a point p of Y is known
- The following expression yields all the elements of Y
 - $X_k = (X_{k-1} \oplus B) \cap A \quad k = 1, 2, 3, \dots$
- Where $X_0 = p$ and B is a suitable structuring element
- If $X_k = X_{k-1}$, the algorithm has converged and $Y = X_k$

Extraction of Connected Components

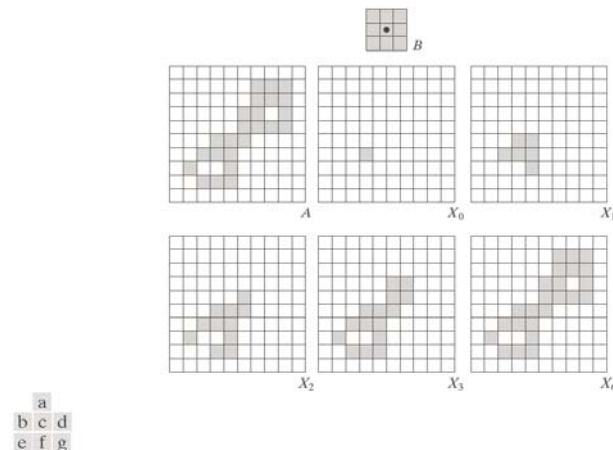
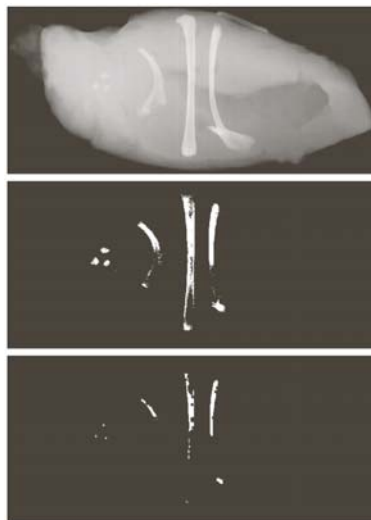


FIGURE 9.17 Extracting connected components. (a) Structuring element. (b) Array containing a set with one connected component. (c) Initial array containing a 1 in the region of the connected component. (d)–(g) Various steps in the iteration of Eq. (9.5-3).

Extraction of Connected Components

- This process is similar to region filling except that we use A instead of A^c in the process
- The difference arises because all of the elements sought (the elements of the connected component) are labeled 1
- The intersection with A at each step eliminates dilations that are centered on elements labeled 0
- The shape of the structuring element assumes 8-connectivity between pixels

Extraction of Connected Components



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

FIGURE 9.18
 (a) X-ray image of chicken file 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 Gerate GmbH, Diepholz, Germany, www.ntbxray.com.)

Convex Hull

- A set A is *convex* if and only if the straight line segment joining any two points of A lie 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*
- The convex hull and convex deficiency will be useful for object description
- We present an algorithm for obtaining the convex hull, $C(A)$, of a set A

Convex Hull

- Let B^i , $i=1,2,3,4$ represent the four structuring elements shown below
- The procedure consists of implementing the following:

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

$$\text{with } X_0^i = A$$

	X	X
		X
	X	X

B_1

X		X
X	X	X

B_2

X	X	
X		
X	X	

B_3

X	X	X
X		X

B_4

shaded=1, white=0, X=don't care

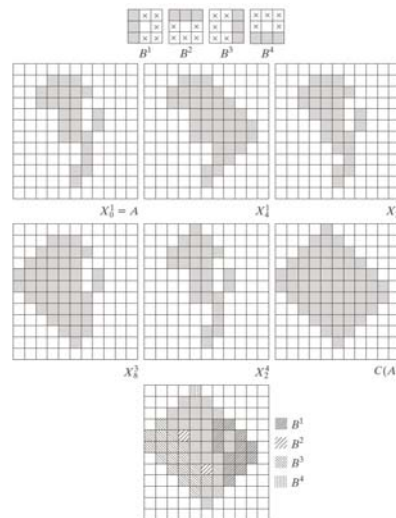
Convex Hull

- Now let $D^i = X_k^i$, where there is convergence in the sense that $X_k^i = X_{k-1}^i$
- The convex hull of A is

$$C(A) = \bigcup_{i=1}^4 D^i$$

- In other words, the procedure 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 set D^1
- Sets D^2 , D^3 and D^4 are generated in a similar manner
- The union of the four sets is the convex hull of A

Convex Hull



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.

Convex Hull

- One obvious shortcoming of the procedure is that the convex hull can grow beyond the minimum dimensions required to guarantee convexity
- One approach to reduce this effect is to limit the growth of the convex hull such that it does not extend beyond the horizontal and vertical dimensions of the original set

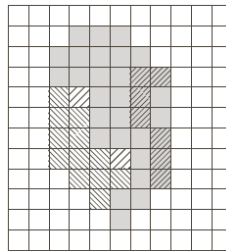


FIGURE 9.20
Result of limiting growth of the convex hull algorithm to the maximum dimensions of the original set of points along the vertical and horizontal directions.

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

$$\begin{aligned} A \otimes B &= A - (A * B) \\ &= A \cap (A * B)^c \end{aligned}$$

- A more useful expression for thinning A symmetrically is based on a *sequence* of structuring elements:
 - $\{B\} = \{B^1, B^2, B^3, \dots, B^n\}$

$$A \otimes \{B\} = ((\dots((A \otimes B^1) \otimes B^2) \dots) \otimes B^n)$$

Thinning

- The process is to thin A by *one pass* with B^1
- 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 $A \cap (A \circledast B)^c$
- As a post-processing step, the thinned set may be converted to m -connectivity to eliminate multiple paths

Thinning

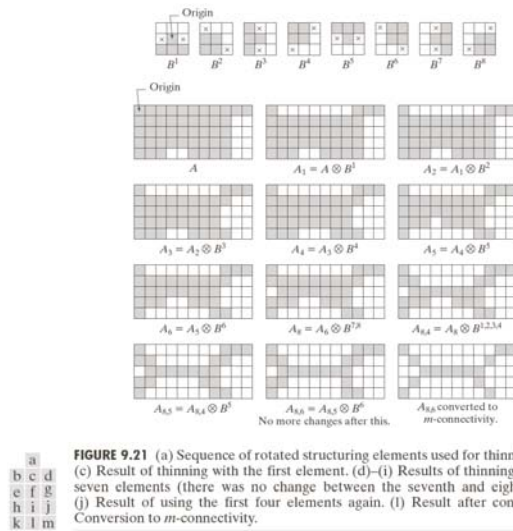


FIGURE 9.21 (a) Sequence of rotated structuring elements used for thinning. (b) Set A . (c) Result of thinning with the first element. (d)–(i) Results of thinning with the next seven elements (there was no change between the seventh and eighth elements). (j) Result of using the first four elements again.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

Thickening

- *Thickening* is the morphological dual of thinning and is defined by the expression
 - $A \odot B = A \cup (A^* \circ B)$
- Where B is a structuring element suitable for thickening
- As with thinning, thickening can be defined as a sequential operation

$$A \odot \{B\} = ((\dots((A \odot B^1) \odot B^2) \dots) \odot B^n)$$

- The structuring elements for thickening have the same form as those for thinning, but with all 1's and 0's interchanged

Thickening

- A separate algorithm for thickening is not absolutely required
- In practice, we can thin the background of the set in question and complement the result
- In other words, to thicken set A , we form $C = A^c$, thin C , and then form C^c
- Depending on the nature of A , this procedure may result in some disconnected points
- Therefore, thickening is commonly followed with a post-processing step to remove disconnected points

Thickening

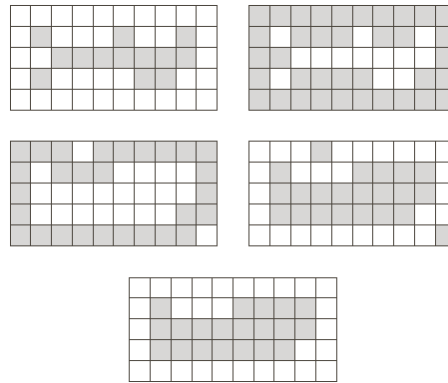


FIGURE 9.22 (a) Set A . (b) Complement of A . (c) Result of thinning the complement of A . (d) Thickened set obtained by complementing (c). (e) Final result, with no disconnected points.