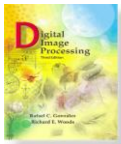


Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

Chapter 9

Morphological Image Processing

Gonzalez & Woods, Digital Image Processing (2009)

In this lecture, we will consider

- What is morphology?
- Simple morphological operations
- Compound operations
- Some Morphological algorithms

Note: Our interest initially is on binary images. Whether 0 and 1 refer to white or black is a little interchangeable at times, in these slides.

2

What Is Morphology?

•Morphological image processing (or *morphology*) describes a range of image processing techniques that deal with the shape (or morphology) of features in an image

•Morphological operations are typically applied to remove imperfections introduced during segmentation, and so typically operate on bi-level images

•Imprecisely, Morphology - A mathematical tool for investigating geometric structure in binary and grayscale images. Helps in shaper processing and analysis

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

3

Quick Example

Images taken from Gonzalez & Woods, Digital Image Processing (2009)

Image after segmentation

Image after segmentation and morphological processing

4

Morphology

- **Morphology operates like the other neighborhood processing methods by applying a kernel to each pixel in the input.**
- **In morphology, the kernel is denoted a *structuring element and contains '0's and '1's.***
- ***You can design the structuring element* as you please, but normally the pattern of '1's form a box or a disk.**

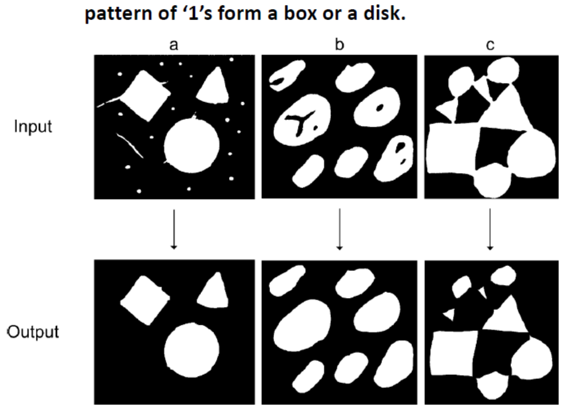


Fig. Three examples of the uses of morphology.

(a) Removing small objects.
(b) Filling holes.
(c) Isolating objects

Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

5

Structuring Elements

Structuring elements can be any size and make any shape

However, for simplicity we will use rectangular structuring elements with their origin at the middle pixel

1	1	1
1	**1**	1
1	1	1

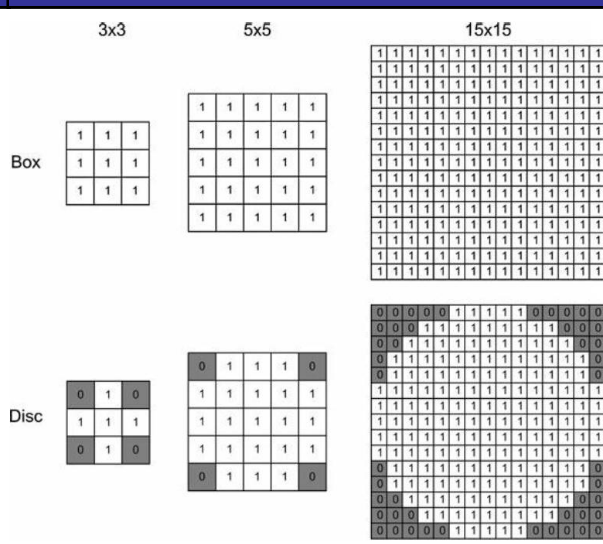
0	1	0
1	**1**	1
0	1	0

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

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

6

Types of Structuring Elements



Here white colour corresponds to pixels of importance.

Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

7

Hit and Fit

Hit

- For each ‘1’ in the structuring element we investigate whether the pixel at the same position in the image is also a ‘1’.
- If this is the case for just one of the ‘1’s in the structuring element we say that the structuring element *hits the image at the pixel* position in question (the one on which the structuring element is centered). This pixel is therefore set to ‘1’ in the output image. Otherwise it is set to ‘0’.

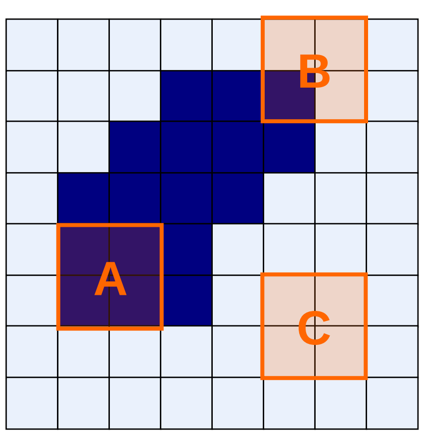
Fit

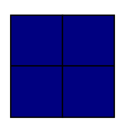
- For each ‘1’ in the structuring element we investigate whether the pixel at the same position in the image is also a ‘1’.
- If this is the case for *all the ‘1’s in the structuring* element we say that the structuring element *fits the image at the pixel position in* question (the one on which the structuring element is centered). This pixel is therefore set to ‘1’ in the output image. Otherwise it is set to ‘0’.

Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

8

Structuring Elements, Hits & Fits





Structuring Element

Fit: All *on pixels* in the structuring element cover *on pixels* in the image

Hit: Any *on pixel* in the structuring element covers an *on pixel* in the image

All morphological processing operations are based on these simple ideas

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

9

Fitting & Hitting

0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0	0	0
0	0	1	1 B	1	1	1	0	0 C	0	0	0
0	1	1	1	1	1	1	1	0	0	0	0
0	1	1	1	1	1	1	1	0	0	0	0
0	0	1	1	1	1	1	1	0	0	0	0
0	0	1	1	1	1	1	1	1	0	0	0
0	0	1	1	1	1	1	1 A	1	1	1	0
0	0	0	0	0	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0

1	1	1
1	1	1
1	1	1

Structuring Element 1

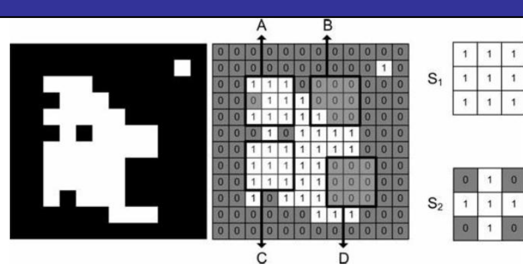
0	1	0
1	1	1
0	1	0

Structuring Element 2

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

10

Hit and Fit



Position	*SE*	Fit	Hit
A	S_1	No	Yes
A	S_2	No	Yes
B	S_1	No	Yes
B	S_2	No	No
C	S_1	Yes	Yes
C	S_2	Yes	Yes
D	S_1	No	No
D	S_2	No	No

Here white colour corresponds to pixels of importance.

Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

11

Fundamental Operations

Fundamentally morphological image processing is very like spatial filtering

The structuring element is moved across every pixel in the original image to give a pixel in a new processed image

The value of this new pixel depends on the operation performed

There are two basic morphological operations: **erosion** and **dilation**

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

12

Dilation

Dilation of image f by structuring element s is given by $f \oplus s$

The structuring element s is positioned with its origin at (x, y) and the new pixel value is determined using the rule:

$$g(x, y) = \begin{cases} 1 \text{ if } s \text{ hits } f \\ 0 \text{ otherwise} \end{cases}$$

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

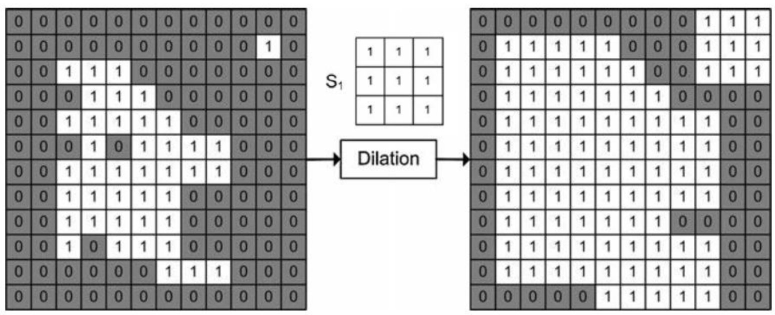
13

Dilation

Dilation

Applying Hit to an entire image is denoted *Dilation and is written as*

$$g(x, y) = f(x, y) \oplus SE$$



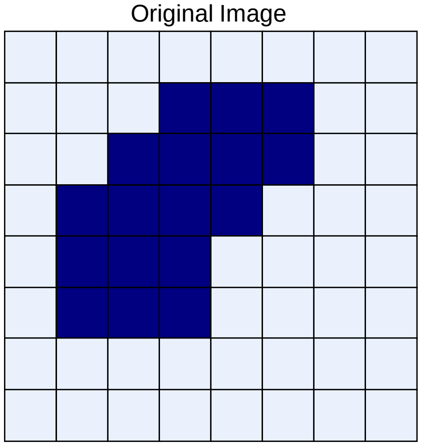
Here white colour corresponds to pixels of importance.

Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

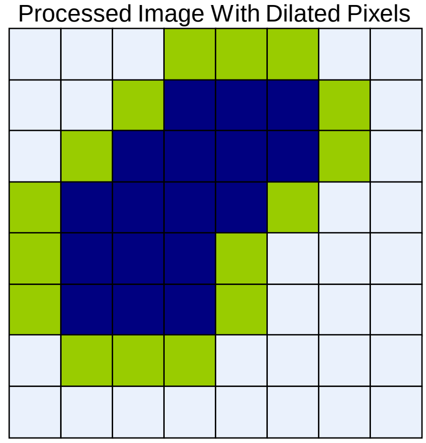
14

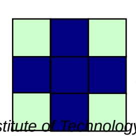
Dilation Example

Original Image



Processed Image With Dilated Pixels





Structuring Element

Credits: *Dr. Brian Mac, Dublin Institute of Technology, Ireland*

15

Dilation Example

Original image

Dilation by 3*3 square structuring element

Dilation by 5*5 square structuring element

Watch out: In these examples a 1 refers to a black pixel!

That implies here black colour corresponds to pixels of importance.

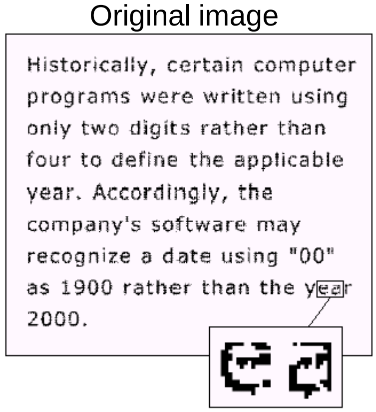
Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

16

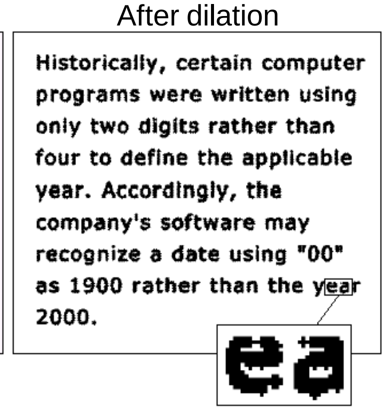
Dilation Example

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

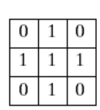
Original image


Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

After dilation


Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

One immediate advantage over lowpass-filtering method that dilation method resulted directly in a binary image.



0	1	0
1	1	1
0	1	0

Structuring element

Here white colour corresponds to pixels of importance.

17

What Is Dilation For?

Dilation can repair breaks

Dilation can repair intrusions

Here BLACK colour corresponds to pixels of importance.

Watch out: Dilation enlarges objects

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

18

Dilation

- The term *dilation refers to the fact that the object in the binary image is increased* in size.
- In general, dilating an image results in objects becoming bigger, small holes being filled, and objects being merged.
- How big the effect is depends on the size of the structuring element.
- It should be noticed that a large structuring element can be implemented by iteratively applying a smaller structuring element.

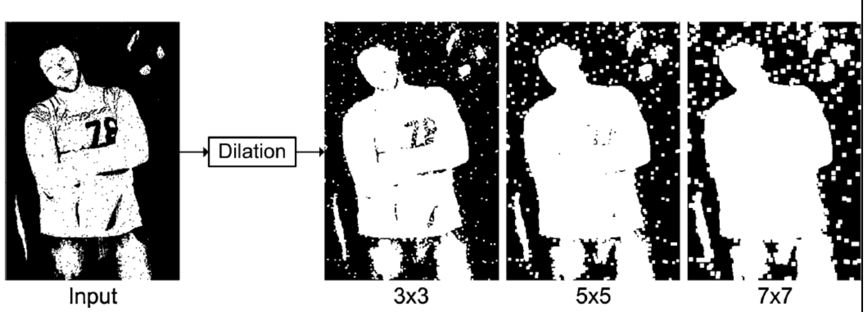


Fig. Dilation with different sized structuring elements

Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

Here white colour corresponds to pixels of importance.

19

Erosion

Erosion of image f by structuring element s is given by $f \ominus s$

The structuring element s is positioned with its origin at (x, y) and the new pixel value is determined using the rule:

$$g(x,y) = \begin{cases} 1 \text{ if } s \text{ fits } f \\ 0 \text{ otherwise} \end{cases}$$

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

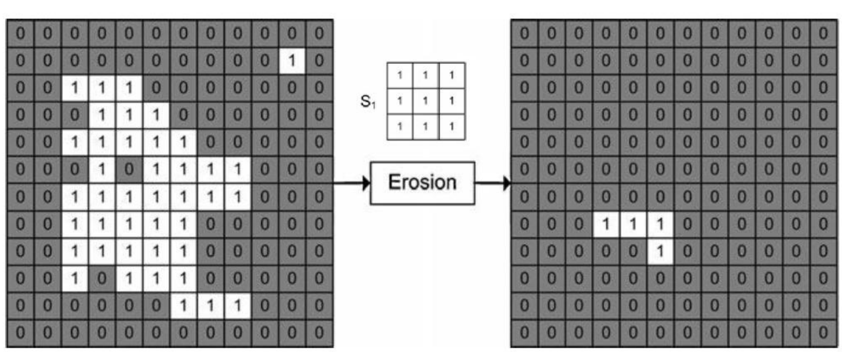
20

Erosion

Erosion

Applying Fit to an entire image is denoted Erosion and is written as

$$g(x, y) = f(x, y) \ominus SE$$



Here white colour corresponds to pixels of importance.

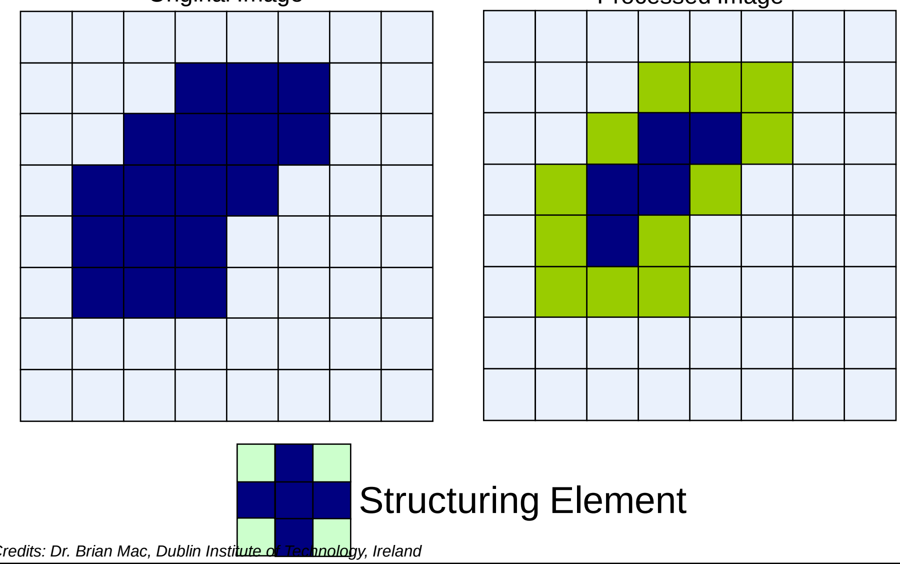
Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

21

Erosion Example

Original Image

Processed Image

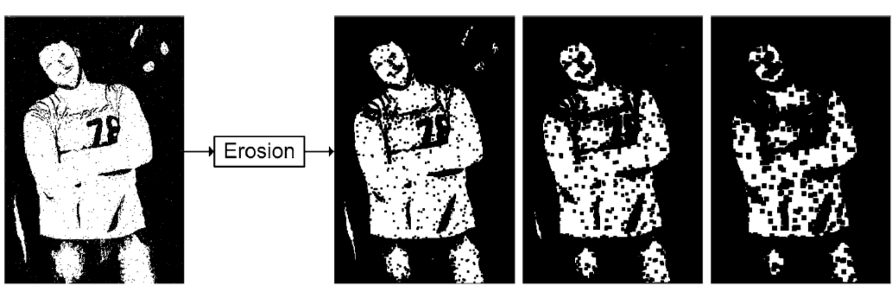


Structuring Element

Credits: *Dr. Brian Mac, Dublin Institute of Technology, Ireland*

22

Erosion Example



Input 3x3 5x5 7x7

Fig. Erosion with different sized structuring elements

Here white colour corresponds to pixels of importance.

Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

23

Erosion Example

Original image

Erosion by 3*3 square structuring element

Erosion by 5*5 square structuring element

Watch out: In these examples a 1 refers to a black pixel!

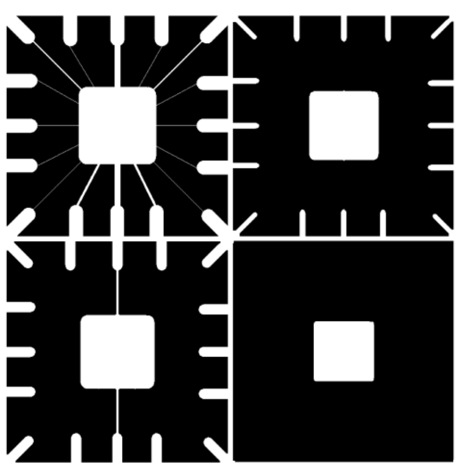
Here BLACK colour corresponds to pixels of importance.

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

24

Erosion Example

Images taken from Gonzalez & Woods, Digital Image Processing (2002)



Original image

After erosion with a disc of radius 10

After erosion with a disc of radius 5

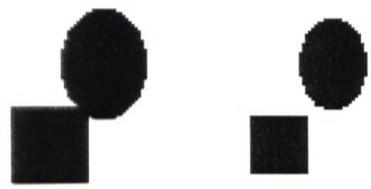
After erosion with a disc of radius 20

Here white colour corresponds to pixels of importance.

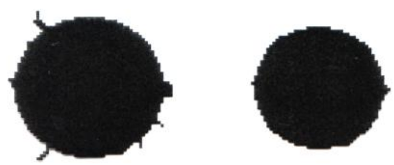
25

What Is Erosion For?

Erosion can split apart joined objects



Erosion can strip away extrusions



Here BLACK colour corresponds to pixels of importance.

Watch out: Erosion shrinks objects

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

- **Language of mathematical morphology: set theory**
- **Sets** $\equiv$ **objects in an image**
- **Binary images: sets** $\in Z^2$
- **Gray-scale images: sets** $\in Z^3$

9.1 Preliminaries

- **Let** A **be a set in** Z^2**. If** $a = (a_1, a_2)$ **is an element of** A**, then we write** $a \in A$
- **Subset, union, intersection:**

$$A \subseteq B,\ C = A \bigcup B,\ D = A \bigcap B$$

- **Disjoint or mutually exclusive:** $A \bigcap B = \emptyset$
- **Complement:** $A^c = \{w | w \notin A\}$
- **Difference:** $A - B = \{w | w \in A, w \notin B\} = A \bigcap B^c$

Credits: Dr Hanno Coetzer, Department of Applied Mathematics, Stellenbosch University, S. Africa

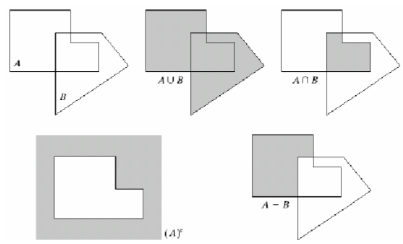
Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

- $\boxed{\text{Reflection:}}$ $\hat{B} = \{w | w = -b, \text{ for } b \in B\}$
- $\boxed{\text{Translation}}$ of set A by point $z = (z_1, z_2)$: $(A)_z = \{c | c = a + z, \text{ for } a \in A\}$

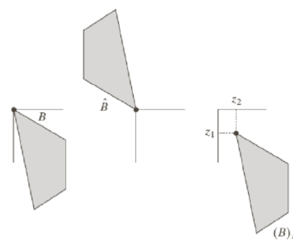
(Ed 2)



a b c
d e

FIGURE 9.1 (a) Two sets A and B. (b) The union of A and B. (c) The intersection of A and B. (d) The complement of A. (e) The difference between A and B.

(Ed 3)



a b c

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

Credits: Dr Hanno Coetzer, Department of Applied Mathematics, Stellenbosch University, S. Africa

Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

- Reflection and translation are employed to formulate operations based on structuring elements (SEs): small sets (subimages) used to probe an image for properties of interest

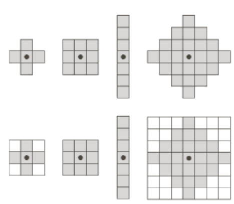
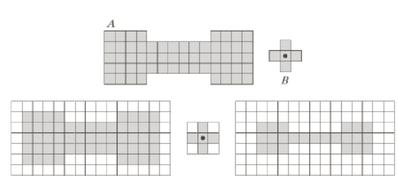


FIGURE 9.2 First row: Examples of structuring elements. Second row: Structuring elements converted to rectangular arrays. The dots denote the centers of the SEs.



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.

Credits: Dr Hanno Coetzer, Department of Applied Mathematics, Stellenbosch University, S. Africa

Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

9.2 Erosion and dilation

These operations are fundamental to morphological processing

9.2.1 Erosion

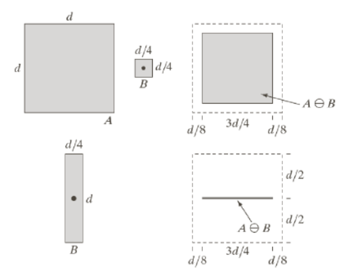
With A and B sets in Z^2, the erosion of A by B, is defined as

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

or alternatively

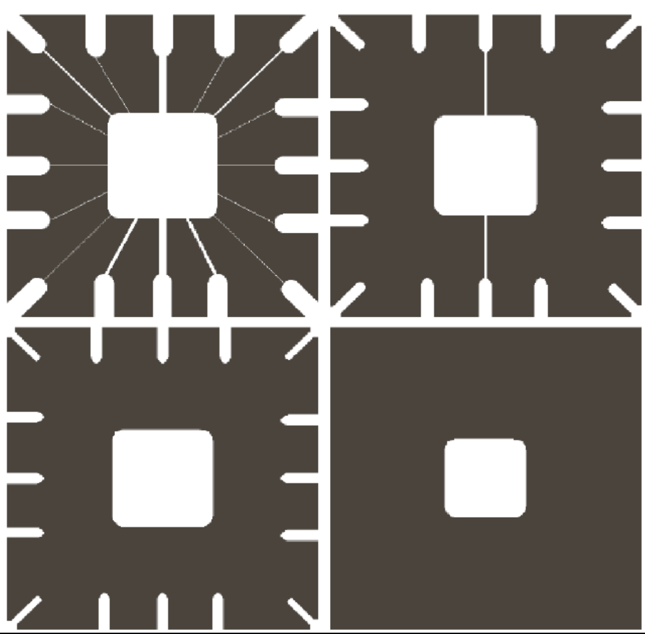
$$A \ominus B = \left\{ z | (B)_z \bigcap A^c = \emptyset \right\}$$

- B is the SE
- Convolution process



Credits: Dr Hanno Coetzer, Department of Applied Mathematics, Stellenbosch University, S. Africa

Example 9.1



a b
c d

FIGURE 9.5 Using erosion to remove image components. (a) A 486×486 binary image of a wire-bond 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.

Here white colour corresponds to pixels of importance.

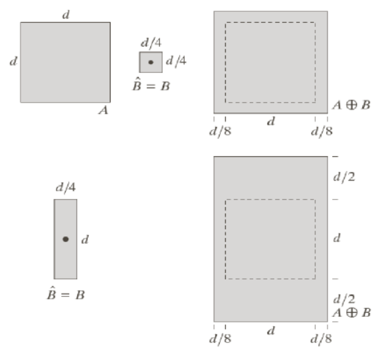
9.2.2 Dilation

With A and B sets in Z^2, the dilation of A by B, is defined as

$$A \oplus B = \left\{ z | (\hat{B})_z \bigcap A \neq \emptyset \right\}$$

or alternatively

$$A \oplus B = \left\{ z | \left[(\hat{B})_z \bigcap A \right] \subseteq A \right\}$$



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

Credits: Dr Hanno Coetzer, Department of Applied Mathematics, Stellenbosch University, S. Africa

Digital Image Processing, 3rd ed.
Gonzalez & Woods
www.ImageProcessingPlace.com

9.2.3 Duality

Dilation and erosion are duals of each other with respect to set complementation and reflection, that is

$$\boxed{(A \ominus B)^c = A^c \oplus \hat{B}} \qquad (*)$$

and

$$\boxed{(A \oplus B)^c = A^c \ominus \hat{B}}$$

Proof of $(*)$:

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

Credits: Dr Hanno Coetzer, Department of Applied Mathematics, Stellenbosch University, S. Africa

33

Compound Operations

More interesting morphological operations can be performed by performing combinations of erosions and dilations

The most widely used of these *compound operations* are:

- Opening
- Closing

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

9.3 Opening and closing

USES

Opening: Smoothes the contour of an object
Breaks narrow isthmuses ("bridges")
Eliminates thin protrusions

Closing: Smoothes sections of contours
Fuses narrow breaks and long thin gulfs
Eliminates small holes in contours
Fills gaps in contours

Definitions

The opening of set A by structuring element B:

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

The closing of set A by structuring element B:

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

Credits: Dr Hanno Coetzer, Department of Applied Mathematics, Stellenbosch University, S. Africa

35

Compound Operations

Compound Operations

Closing

$$g(x,y) = f(x,y) \bullet SE = \big(f(x,y) \oplus SE\big) \ominus SE$$

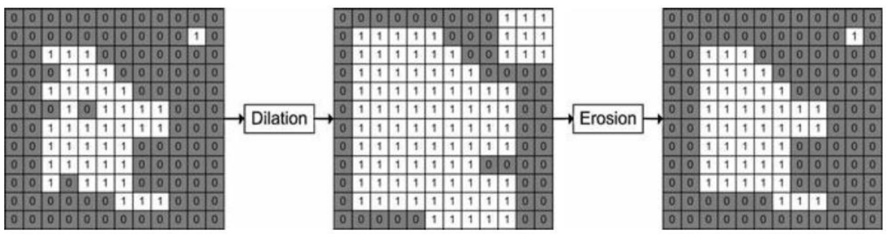


Fig. Closing of the binary image using *S1*

Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

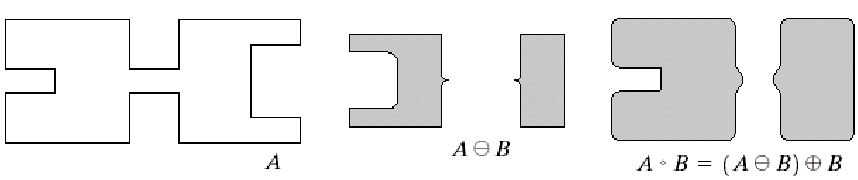
36

Opening

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

The opening of image f by structuring element s, denoted $f \circ s$ is simply an erosion followed by a dilation

$$f \circ s = (f \ominus s) \oplus s$$



Original shape

After erosion

After dilation (opening)

Note: a disc shaped structuring element is used

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

37

Opening Example

Original Image

Image After Opening

Here white colour corresponds to pixels of importance.

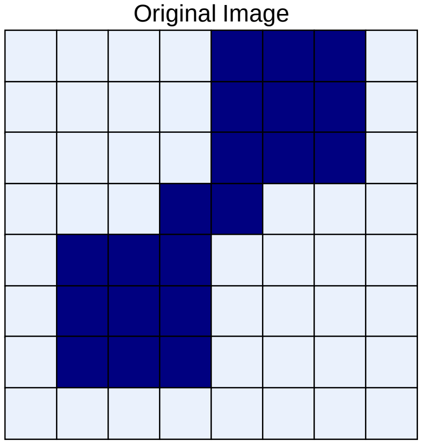
Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

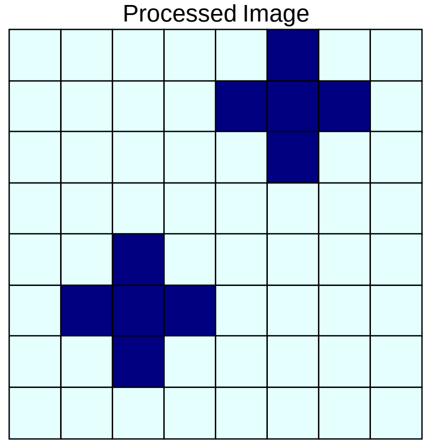
38

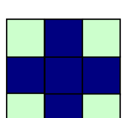
Opening Example

Original Image



Processed Image





Structuring Element

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

39

Opening

Opening

$$g(x,y) = f(x,y) \circ SE = \left(f(x,y) \ominus SE\right) \oplus SE$$

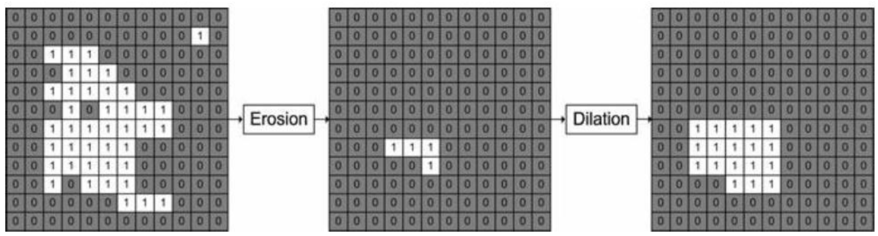


Fig. Opening of the binary image using *S1*

Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

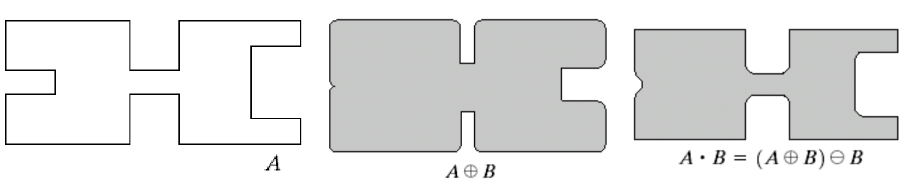
40

Closing

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

The closing of image f by structuring element s, denoted $f \bullet s$ is simply a dilation followed by an erosion

$$f \bullet s = (f \oplus s) \ominus s$$



Original shape

After dilation

After erosion (closing)

Note a disc shaped structuring element is used

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

41

Closing

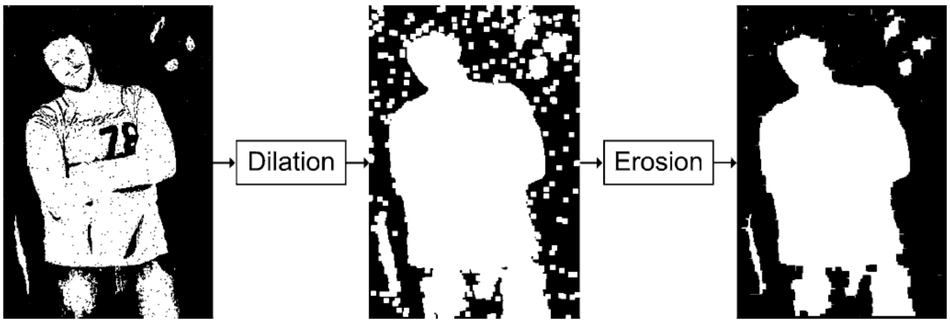


Fig. Closing performed using 7 ×7 box-shaped structuring elements

Here white colour corresponds to pixels of importance.

Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

42

Closing Example

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Original Image

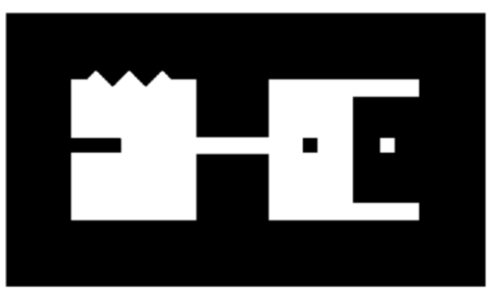


Image After Closing

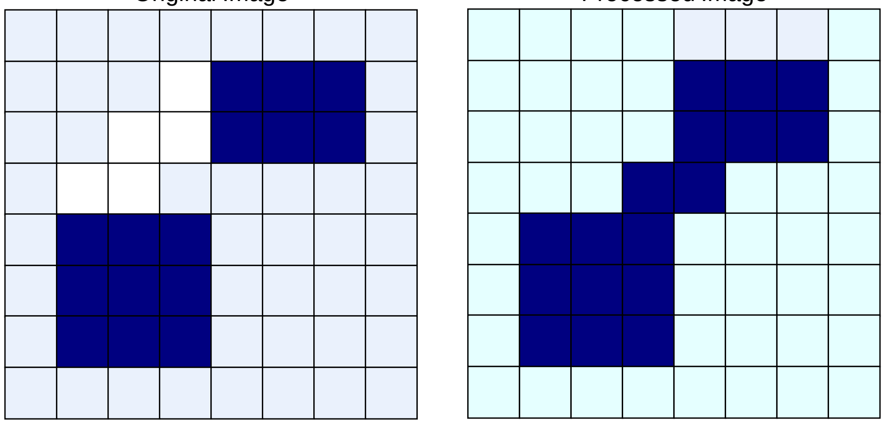
Here white colour corresponds to pixels of importance.

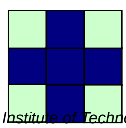
Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

43

Closing Example

Original Image Processed Image





Structuring Element

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

44

Opening / Closing

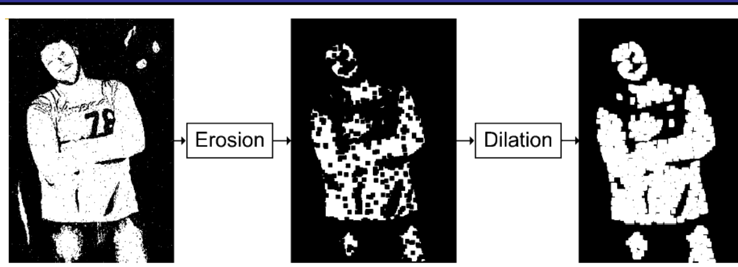


Fig. Opening performed using a 7×7 box-shaped structuring element

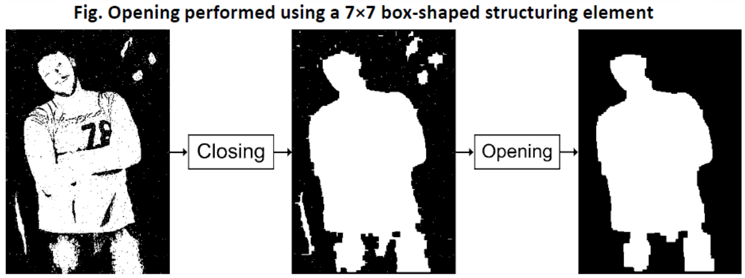


Fig. Filtering a binary image where both holes and small noisy objects are present

Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

45

Applications

Application: Boundary Detection

$$g(x, y) = f(x, y) - \big(f(x, y) \ominus SE\big)$$

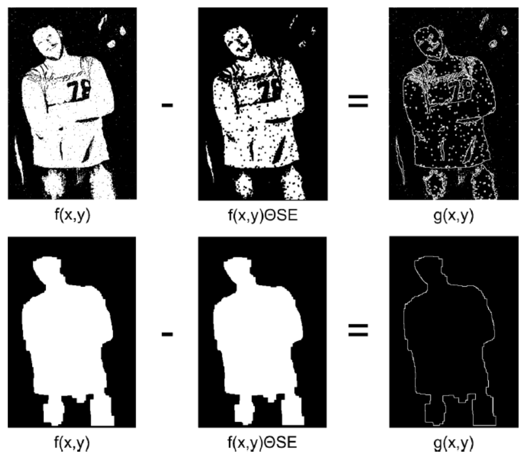


Fig. Boundary detection

Credits: Dr. S. Murala, EEL484 Lecture notes/slides, IIT Ropar

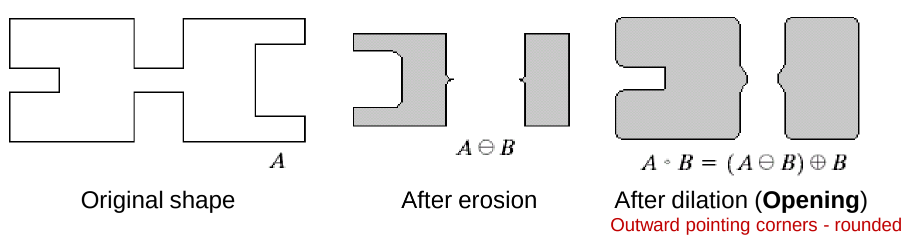
46

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

Smoothes the contour of an object
Breaks narrow isthmuses ("bridges")
Eliminates thin protrusions

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

Smoothes sections of contours
Fuses narrow breaks and long thin gulfs
Eliminates small holes in contours
Fills gaps in contours



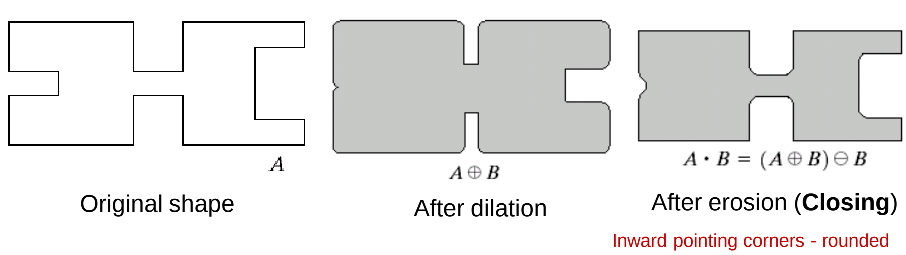
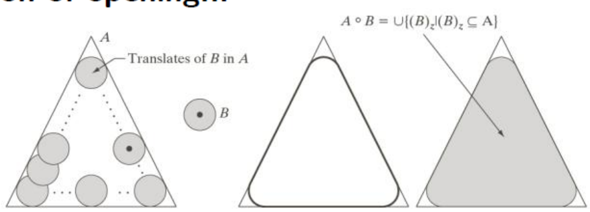


Illustration of opening…



Points in B that reach farthest into the boundary of A as B is rolled around inside of this boundary

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.

Alternative definition for opening:

$$A \circ B = \bigcup \{(B)_z | (B)_z \subseteq A\}$$

Digital Image Processing, 3rd ed.

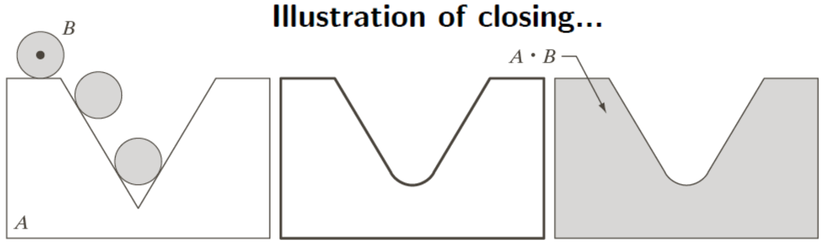
Gonzalez & Woods

www.ImageProcessingPlace.com

Chapter 9

Morphological Image Processing

Illustration of closing…

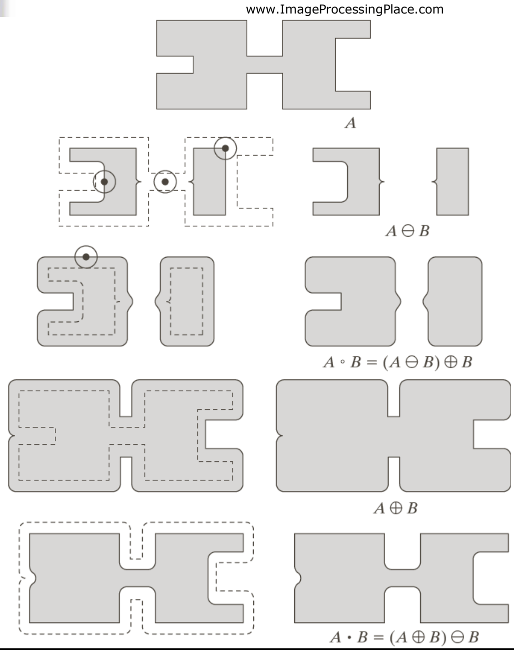


a b c

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.

Alternative definition for closing:

A point w is an element of $A \bullet B$ if and only if $(B)_z \cap A \neq \emptyset$ for any translate of $(B)_z$ that contains w



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.

Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

Chapter 9

Morphological Image Processing

Opening and closing are also duals of each other with respect to set complementation and reflection, that is

$$(A \bullet B)^c = A^c \circ \hat{B}$$

The opening operation satisfies the following properties:

(i) $A \circ B \subseteq A$ (ii) **If** $C \subseteq D$, **then** $C \circ B \subseteq D \circ B$ (iii) $(A \circ B) \circ B = A \circ B$

The closing operation satisfies the following properties:

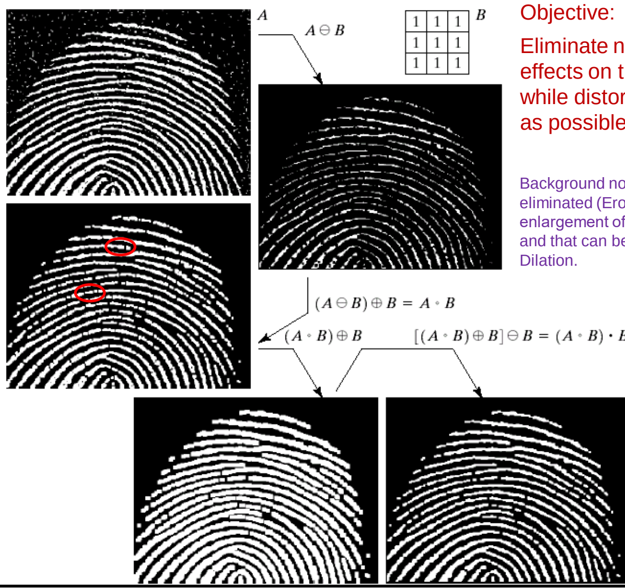
(i) $A \subseteq A \bullet B$ (ii) **If** $C \subseteq D$, **then** $C \bullet B \subseteq D \bullet B$ (iii) $(A \bullet B) \bullet B = A \bullet B$

Credits: Dr Hanno Coetzer, Department of Applied Mathematics, Stellenbosch University, S. Africa

51

Morphological Processing Example

Images taken from Gonzalez & Woods, Digital Image Processing (2002)



Objective:

Eliminate noise and its effects on the fingerprint while distorting it as little as possible

Background noise completely eliminated (Erosion), but there is enlargement of the "dark spots" and that can be countered by Dilation.

Virtually all noise components in foreground and background eliminated by "Opening", but new gaps between fingerprint ridges were created.

52

Hit or Miss Transformation

- **Objective is to find a disjoint region (set) in an image**
- **If B denotes the set composed of D and its background, the match/hit (or set of matches/hits) of B in A, is**

$$A \circledast B = (A \ominus D) \cap [A^c \ominus (W - D)]$$

This transform is considered as a basic tool for shape detection

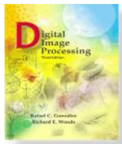
- **Generalized notation:** $B = (B_1, B_2)$
  - B_1: **Set formed from elements of B associated with an object**
  - B_2: **Set formed from elements of B associated with the corresponding background**

[Preceeding discussion: $B_1 = D$ and $B_2 = (W - D)$]

- **More general definition:** $A \circledast B = (A \ominus B_1) - (A \oplus \hat{B}_2)$

$$A \circledast B = (A \ominus B_1) \cap [A^c \ominus B_2]$$

- $A \circledast B$ **contains all the origin points at which, simultaneously, B_1 found a hit in A and B_2 found a hit in A^c**



a b
c d
e
f

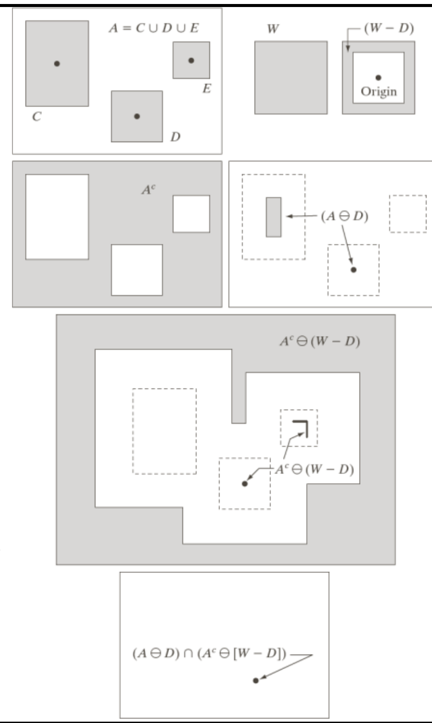


FIGURE 9.12 (a) Set A. (b) A window, W, and the local background of D with respect to W, $(W - D)$. (c) Complement of A. (d) Erosion of A by D. (e) Erosion of A^c by $(W - D)$. (f) Intersection of (d) and (e), showing the location of the origin of D, as desired. The dots indicate the origins of C, D,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

54

Morphological Algorithms

Principal applications of morphology is in extracting image components that are useful in the representation and description of shape.

We will consider morphological algorithms for extracting

- Boundaries, Connected Components, Convex hull , Skeleton of a region

We will also discuss some other methods like:

- Region filling, Thinning , Thickening, Pruning

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

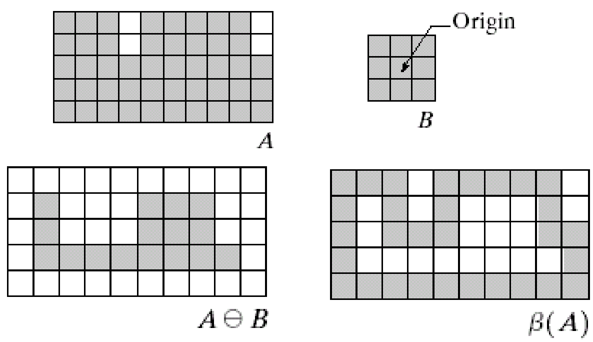
55

Boundary Extraction

Extracting the boundary (or outline) of an object is often extremely useful

The boundary can be given simply as

$$\beta(A) = A - (A \ominus B)$$



Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

56

Boundary Extraction Example

A simple image and the result of performing boundary extraction using a square 3x3 structuring element


Original Image


Extracted Boundary

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

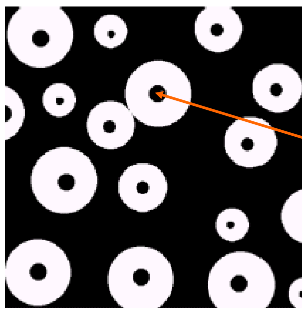
Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

57

Region / Hole Filling

Images taken from Gonzalez & Woods, Digital Image Processing (2002)

Given a pixel inside a boundary, *region filling* attempts to fill that boundary with object pixels (1s)



Given a point inside here, can we fill the whole circle?

Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

58

Region/Hole Filling

- $A \equiv$ set whose elements are 8-connected boundaries that enclose a background region (hole)
- Given a point p in each hole, the objective is to fill all the holes with 1's
- All non-boundary (background) points are labeled 0
- Begin by forming an array X_0 of 0's, except at the locations in X_0 that correspond to the points p in each hole, which is set to 1...
- The following procedure fills all the holes with 1's,

$$X_k = (X_{k-1} \oplus B) \cap A^c, \quad k = 1, 2, 3, \dots,$$

where B is the symmetric structuring element in figure 9.15 (c)

- The algorithm terminates at iteration step k if $X_k = X_{k-1}$
- The set union of X_k and A contains the filled set and its boundary

Note that the intersection at each step with A^c limits the dilation result to inside the region of interest

59

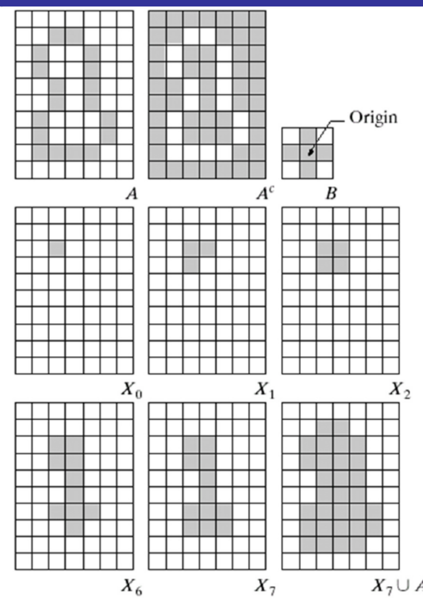
Region/Hole Filling

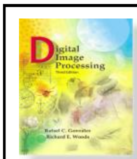
Region filling:

$$X_0 = P$$
$$\text{while } X_k \neq X_{k-1} \text{ do}$$
$$\quad X_k = (X_{k-1} \oplus B) \cap A^C$$
$$X_F = X_k \cup A$$

The dilation would fill the whole area were it not for the intersection with Ac

→**Conditional dilation**





Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

Chapter 9

Morphological Image Processing

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.

Note that - Current algorithm requires the knowledge of whether the black points are background points or sphere inner points

Extraction of Connected Components

Let A be a set containing one or more connected components, and form an array X_0 (with the same size as A) whose elements are 0 (background), except at each location known to correspond to a point in each connected component in A, which is set to 1 (foreground)

The following iterative procedure starts with X_0 and find all the connected components

$$X_k = (X_{k-1} \oplus B) \cap A \quad k = 1, 2, 3, \dots,$$

where B is a suitable structuring element. When $X_k = X_{k-1}$, with X_k containing all the connected components, the procedure terminates

This algorithm is applicable to any finite number of sets of connected components contained in A, assuming that a point is known in each connected component

Digital Image Processing, 3rd ed.

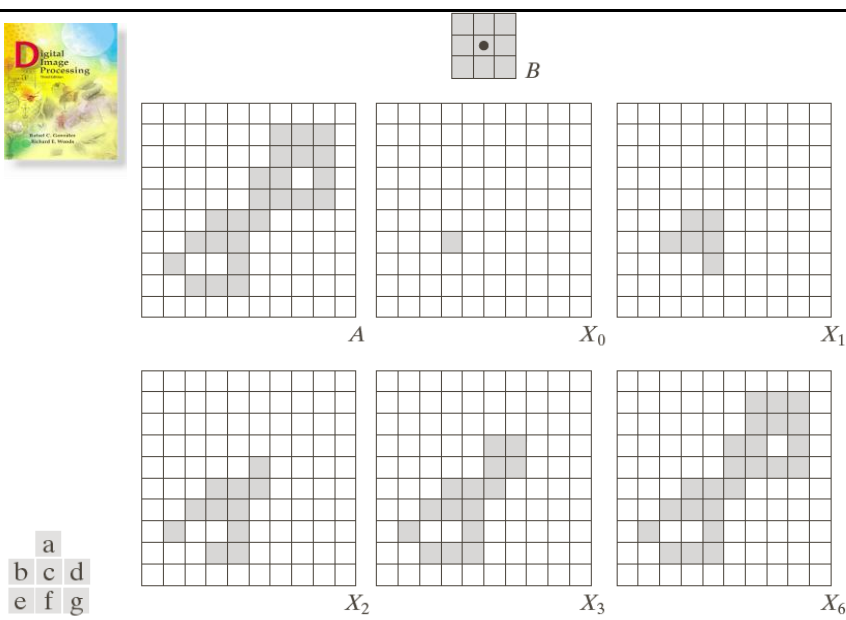


FIGURE 9.17 Extracting connected components. (a) Structuring element. (b) Array containing a set with one connected component. (c) Initial array containing a 1 in the region of the connected component. (d)–(g) Various steps in the iteration of Eq. (9.5-3).

Digital Image Processing, 3rd ed.
Gonzalez & Woods
www.ImageProcessingPlace.com

Extraction of Connected Components

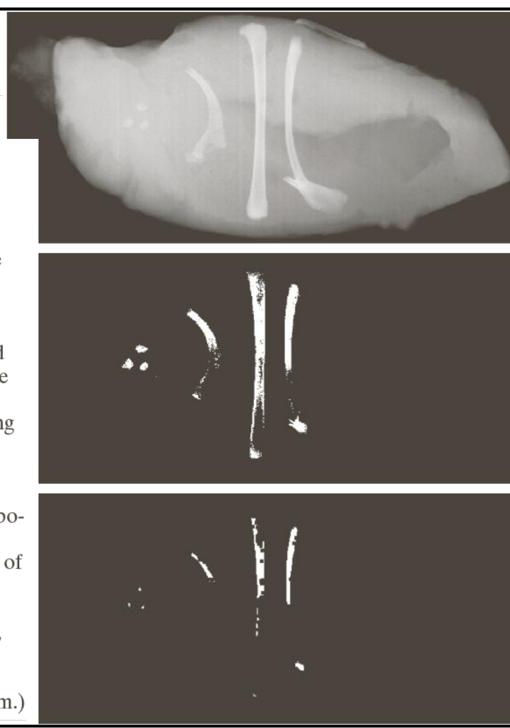
Hole filling

$$X_0 = P$$
$$while\ X_k \neq X_{k-1}\ do$$
$$X_k = (X_{k-1} \oplus B) \cap A^C$$
$$X_F = X_k \cup A$$

Extraction of connected components

$$X_0 = P$$
$$while\ X_k \neq X_{k-1}\ do$$
$$X_k = (X_{k-1} \oplus B) \cap A$$

- **Intersection with A (not A^c) :** As we are looking for foreground points in CC-Extraction, but in Region-Filling, we were looking for background points
- Shape of structuring element used is based on 8-connectivity between pixels.
- This algorithm also assumes knowledge of the point within the connected component



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

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
- H-S is called the convex deficiency of S
- The convex hull and convex deficiency are useful for object description, in some applications

Convex Hull

Morphological algorithm for obtaining the convex hull, $C(A)$, of a set A...

Let B^1, B^2, B^3 and B^4 represent the four structuring elements in Fig 9.19 (a), and then implement the equation ...

$$X_k^i = (X_{k-1} \circledast B^i) \cup A, \; i = 1,2,3,4, \; k = 1,2,\ldots, \; X_0^i = A$$

Now let $D^i = X^i_{\text{conv}}$, where "conv" indicates convergence in the sense that $X_k^i = X_{k-1}^i$. Then the convex hull of A is

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

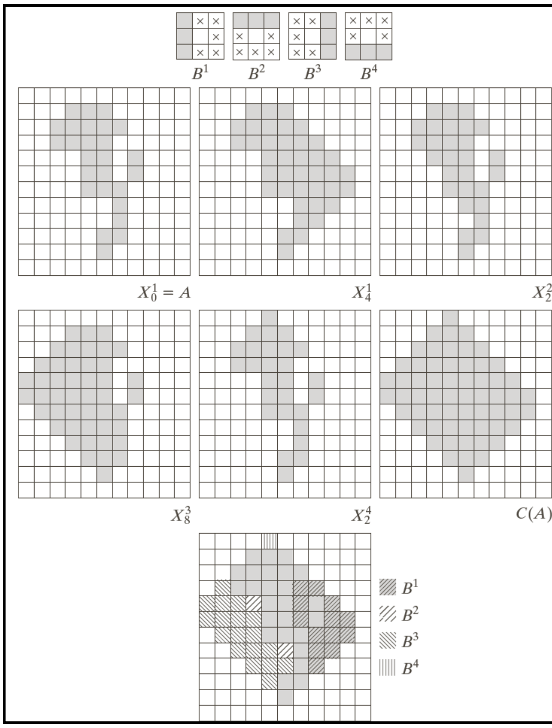
Here we are using simplified "Hit or miss" transform (no background match) i.e. erosion

Procedure illustrated in Fig 9.19: $\times$ entries indicate "don't care" conditions

Shortcoming of above algorithm: convex hull can grow beyond the minimum dimensions required to guarantee convexity

Possible solution: Limit growth so that it does not extend past the vertical and horizontal dimensions of the original set of points

Boundaries of greater complexity can be used to limit growth even further in images with more detail



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.

Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

Chapter 3
Intensity Transformations & Spatial Filtering

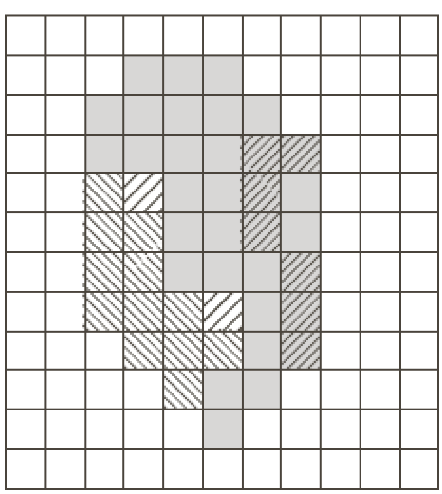


FIGURE 9.20 Result of limiting growth of the convex hull algorithm to the maximum dimensions of the original set of points along the vertical and horizontal directions.

Thinning

- Used to remove selected foreground pixels from binary images, somewhat like erosion or opening.
- Particularly useful for skeletonization. In this mode, it is commonly used to tidy up the output of edge detectors by reducing all lines to single pixel thickness.
- Most common use is to reduce the thresholded output of an edge detector such as the Sobel operator, to lines of a single pixel thickness, while preserving the full length of those lines (i.e. pixels at the extreme ends of lines should not be affected)
- Thinning is normally only applied to binary images, and produces another binary image as output.

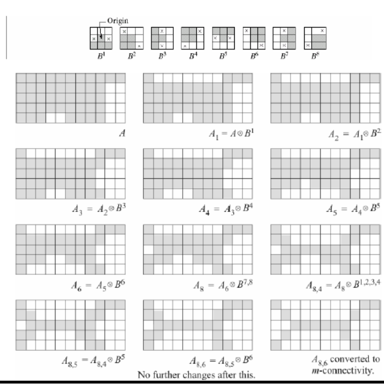
Thinning

9.5.5 Thinning: The thinning of a set A by a structuring element B:

$$A \otimes B = A - (A \circledast B) = A \cap (A \circledast B)^c$$

Symmetric thinning: Sequence of SEs, $\{B\} = \{B^1, B^2, B^3, \ldots, B^n\}$, where B^i is a rotated version of B^{i-1}

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



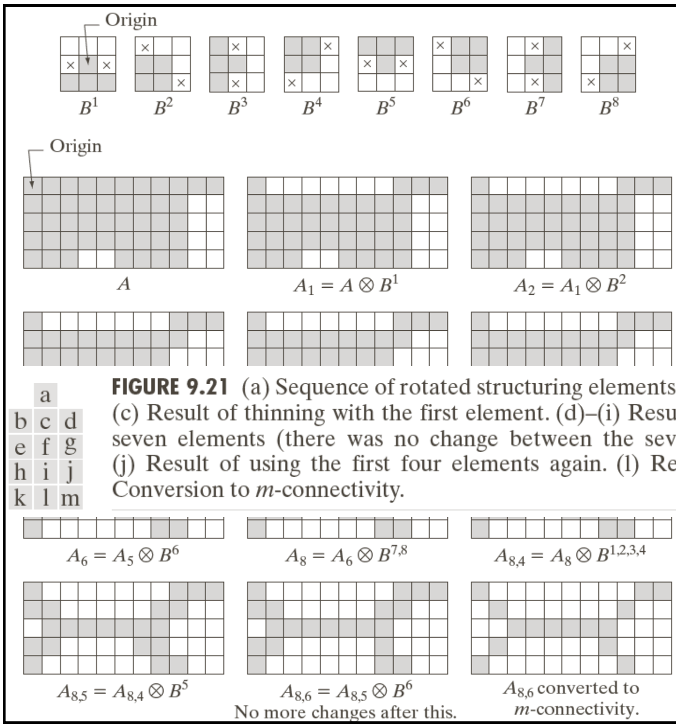
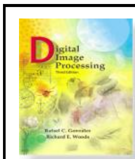


FIGURE 9.21 (a) Sequence of rotated structuring elements used for thinning. (b) Set A. (c) Result of thinning with the first element. (d)–(i) Results of thinning with the next seven elements (there was no change between the seventh and eighth elements). (j) Result of using the first four elements again.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



Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

Thickening

- Used to grow selected regions of foreground pixels in binary images, somewhat like dilation or closing.
- It has several applications, including determining the approximate convex hull of a shape, and determining the skeleton by zone of influence.
- Thickening is normally only applied to binary images, and produces another binary image as output.

Thickening

9.5.6 Thickening: Thickening is the morphological dual of thinning and is defined by:

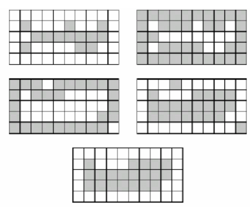
$$A \odot B = A \cup (A \circledast B),$$

where B is a structuring element

Similar to thinning: $A \odot \{B\} = ((\ldots((A \odot B^1) \odot B^2)\ldots) \odot B^n)$

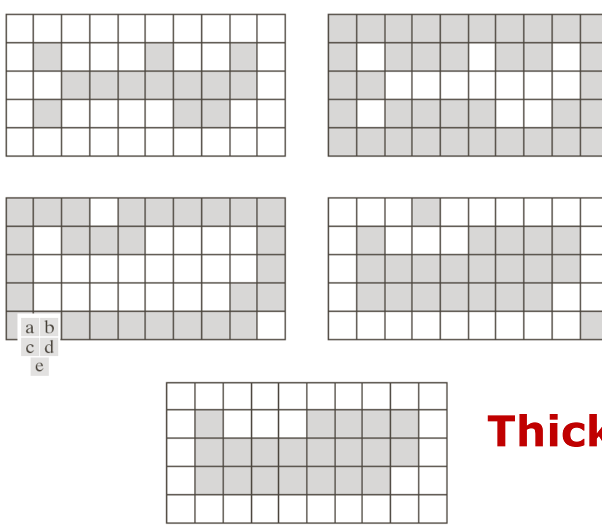
Structuring elements for thickening are similar to those of Fig 9.21 (a), but with all 1's and 0's interchanged

A separate algorithm for thickening is seldom used in practice – we thin the background instead, and then complement the result



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
e

Thickening

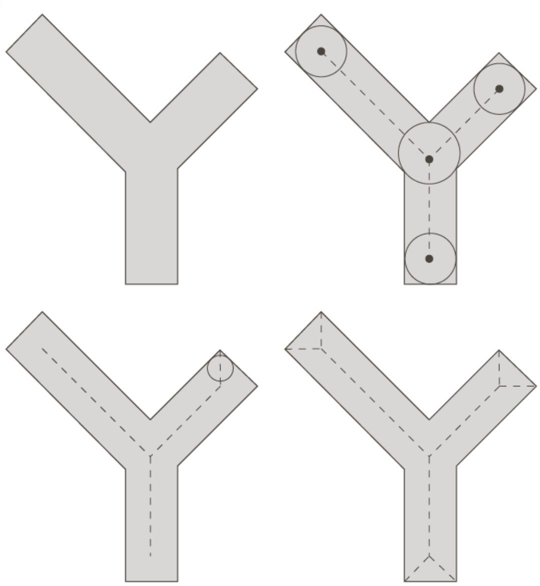
FIGURE 9.22 (a) Set A. (b) Complement of A. (c) Result of thinning the complement of A. (d) Thickened set obtained by complementing (c). (e) Final result, with no disconnected points.

Skeletons

(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.

Digital Image Processing, 3rd ed.
Gonzalez & Woods
www.ImageProcessingPlace.com



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.

Skeletons

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

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

with

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

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

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

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

$$K = \max\{k \mid (A \ominus kB) \neq \varnothing\} \qquad (9.5\text{-}14)$$

Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

Skeletons

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

with

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

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

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

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

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

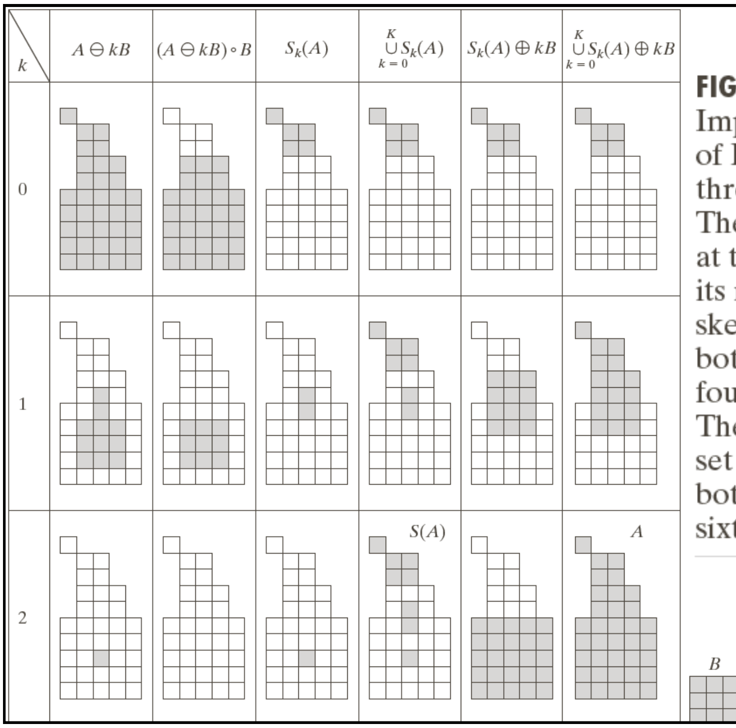


FIGURE 9.24 Implementation of Eqs. (9.5-11) through (9.5-15). The original set is at the top left, and its morphological skeleton is at the bottom of the fourth column. The reconstructed set is at the bottom of the sixth column.

Digital Image Processing, 3rd ed.

Gonzalez & Woods

www.ImageProcessingPlace.com

Pruning

9.5.8 Pruning

- Cleans up "parasitic" components left by thinning and skeletonization
- Use combination of morphological techniques

Illustrative problem: Hand-printed character recognition

- Analyze shape of skeleton of character
- Skeletons characterized by spurs ("parasitic" components)
- Spurs caused during erosion of non-uniformities in strokes
- We assume that the length of a parasitic component does not exceed a specified number of pixels

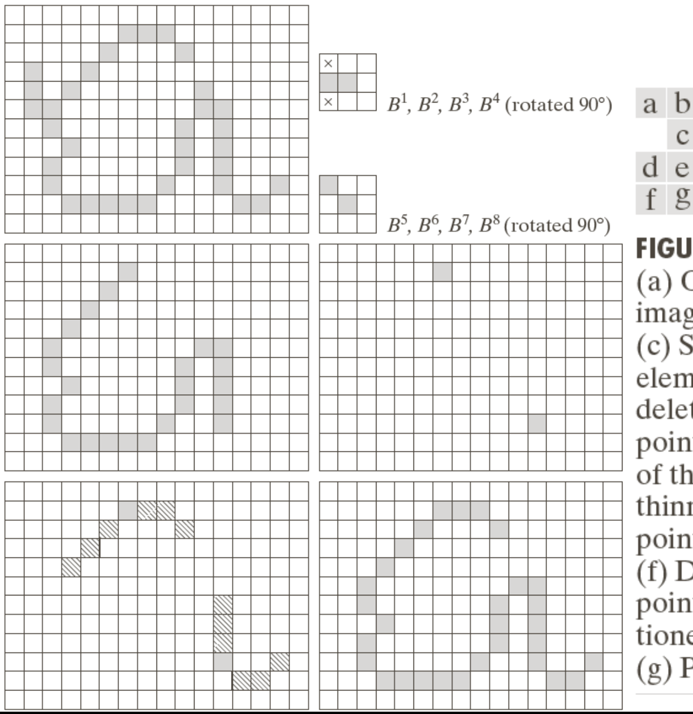


FIGURE 9.25 (a) Original image. (b) and (c) Structuring elements used for deleting end points. (d) Result of three cycles of thinning. (e) End points of (d). (f) Dilation of end points conditioned on (a). (g) Pruned image.

Pruning

Any branch with three or less pixels is to be eliminated

(1) Three iterations of:

$$X_1 = A \otimes \{B\}$$

(2) Find all the end points in X_1:

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

(3) Dilate end points three times, using A as a delimiter:

$$X_3 = (X_2 \oplus H) \cap A, \quad H = \begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

(4) Finally:

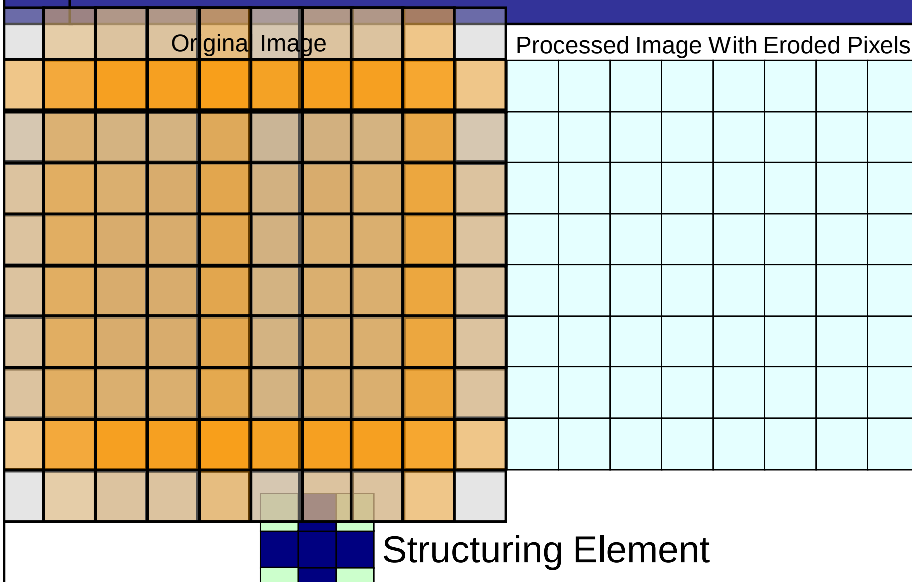
$$X_4 = X_1 \cup X_3$$

Appendix

Extraction of Connected Components

85

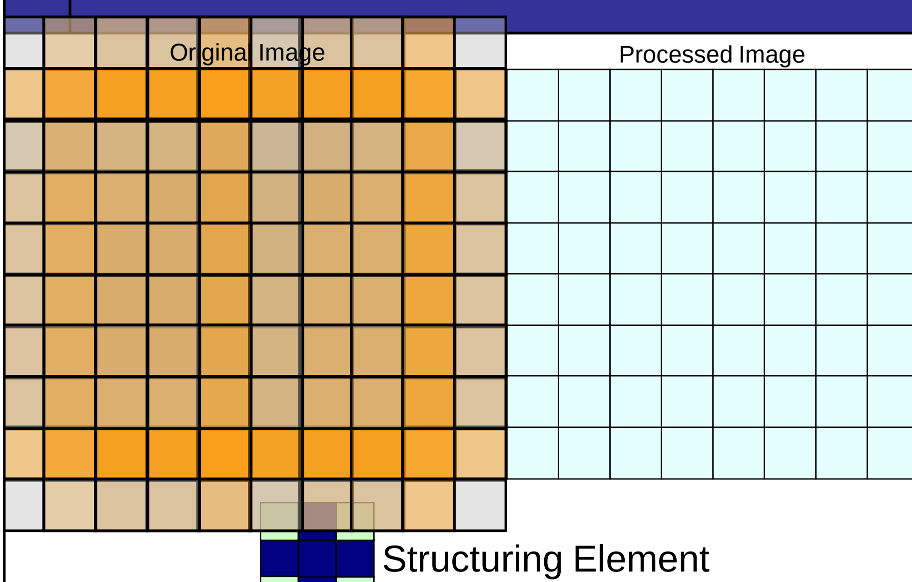
Erosion Example



Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

86

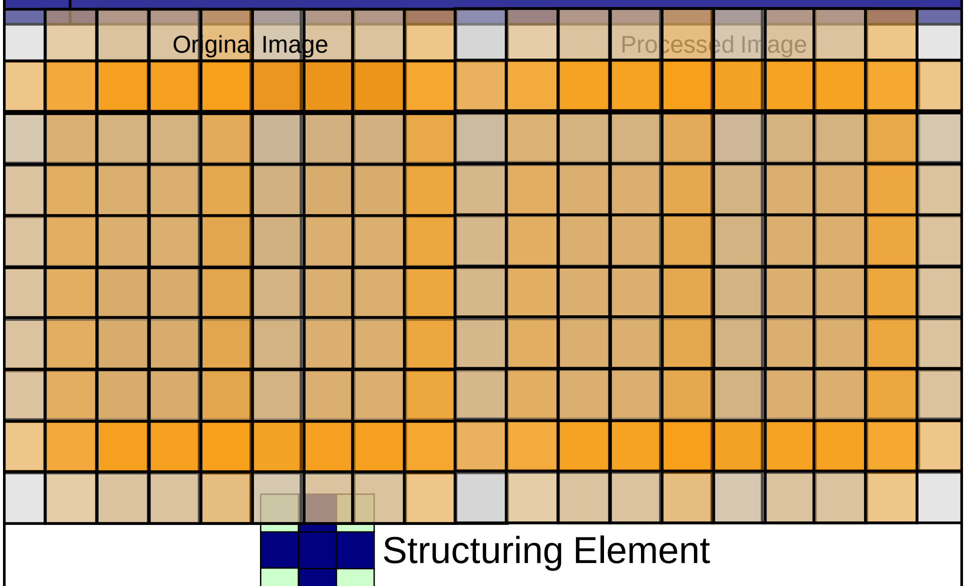
Dilation Example



Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

87

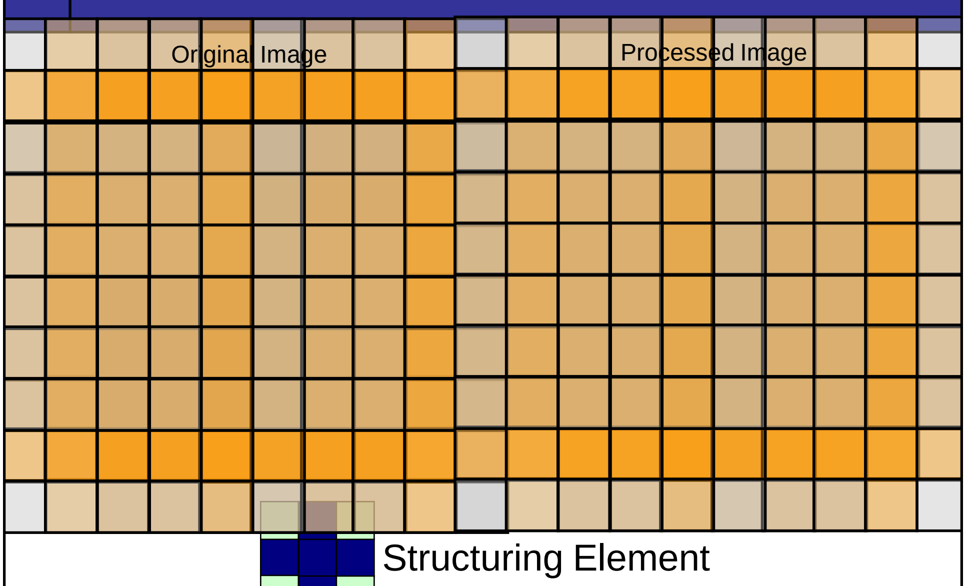
Opening Example



Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland

88

Closing Example



Credits: Dr. Brian Mac, Dublin Institute of Technology, Ireland