Digital Image Processing

Lecture # 09
Morphological Operations

Image Morphology

Introduction

- Morphology

  A branch of biology which deals with the form and structure of animals and plants

- Mathematical Morphology
  - A tool for extracting image components that are useful in the representation and description of region shapes
  - The language of mathematical morphology is Set Theory

Morphology: Quick Example

Image after segmentation

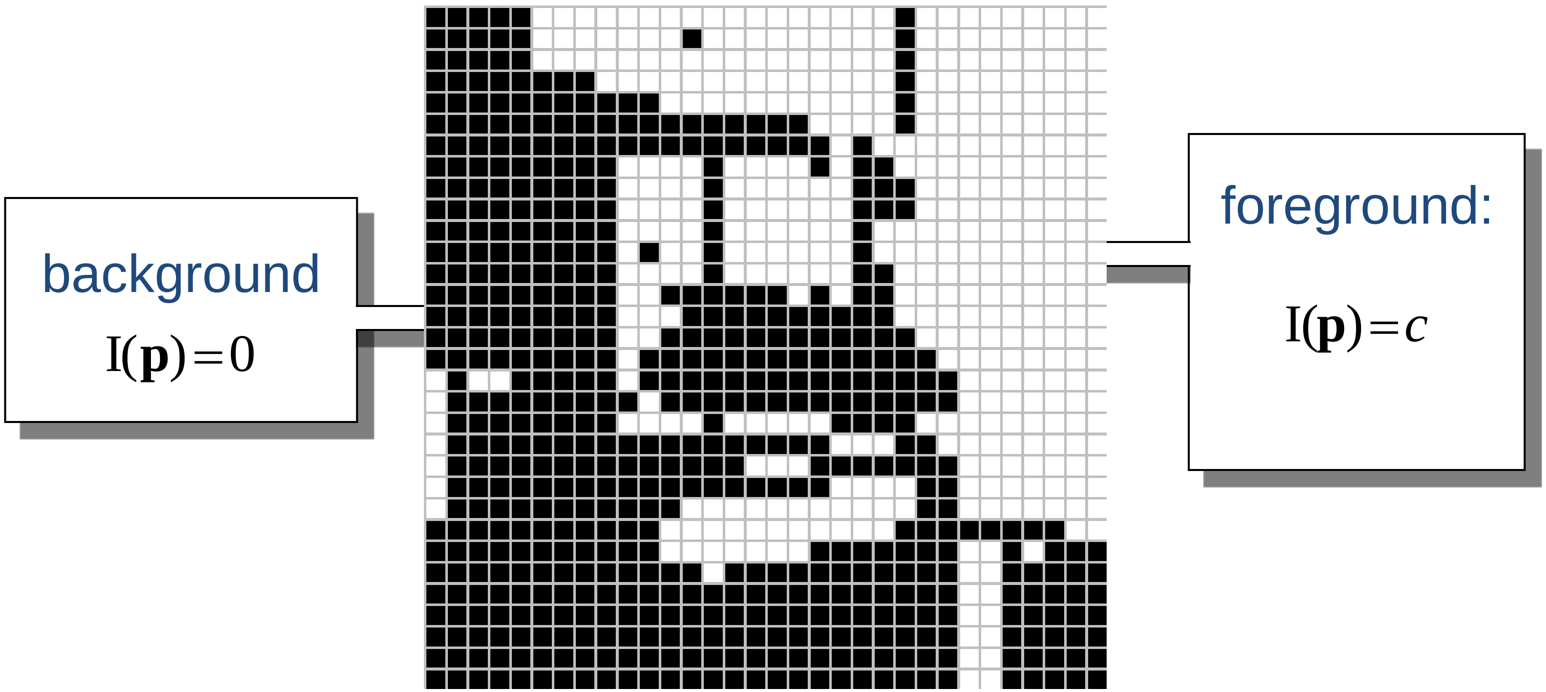
Image after segmentation and morphological processing

Introduction

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

Sets in mathematical morphology represents objects in an image. E.g. Set of all white pixels in a binary image.

Introduction



This represents a digital image. Each square is one pixel.

Set Theory

- The set space of binary image is Z^2
  - Each element of the set is a 2D vector whose coordinates are the *(x,y)* of a black (or white, depending on the convention) pixel in the image
- The set space of gray level image is Z^3
  - Each element of the set is a 3D vector: *(x,y)* and intensity level.

NOTE:

Set Theory and Logical operations are covered in:
Section 9.1, Chapter # 9, 2nd Edition DIP by Gonzalez
Section 2.6.4, Chapter # 2, 3rd Edition DIP by Gonzalez

Set Theory

- Let A be a set in Z^2. if a = (a_1,a_2) is an element of A, then we write

$$a \in A$$

- If a is not an element of A, we write

$$a \notin A$$

- Set representation

$$A = \{(a_1, a_2), (a_3, a_4)\}$$

- Empty or Null set

$$A = \varnothing$$

Set Theory

- Subset: if every element of A is also an element of another set B, the A is said to be a subset of B

$$A \subseteq B$$

- Union: The set of all elements belonging either to A, B or both

$$C = A \bigcup B$$

- Intersection: The set of all elements belonging to both A and B

$$D = A \bigcap B$$

Set Theory

- Two sets A and B are said to be disjoint or mutually exclusive if they have no common element

$$A \bigcap B = \varnothing$$

- Complement: The set of elements not contained in A

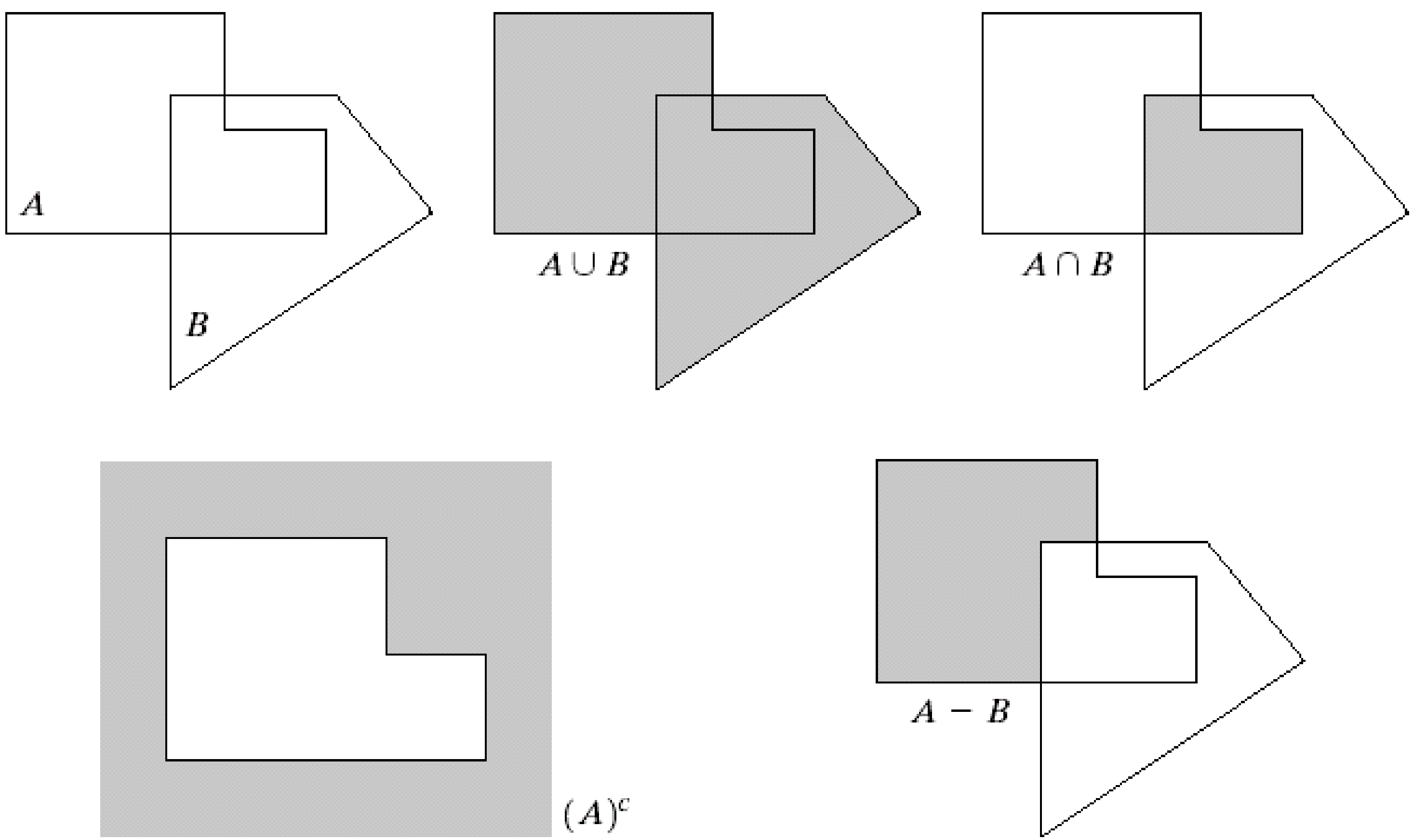
$$A^c = \{w \mid w \notin A\}$$

- Difference of two sets A and B, denoted by A – B, is defined as

$$A - B = \{w \mid w \in A, w \notin B\} = A \bigcap B^c$$

i.e. the set of elements that belong to A, but not to B

Set Theory



a b c
d e

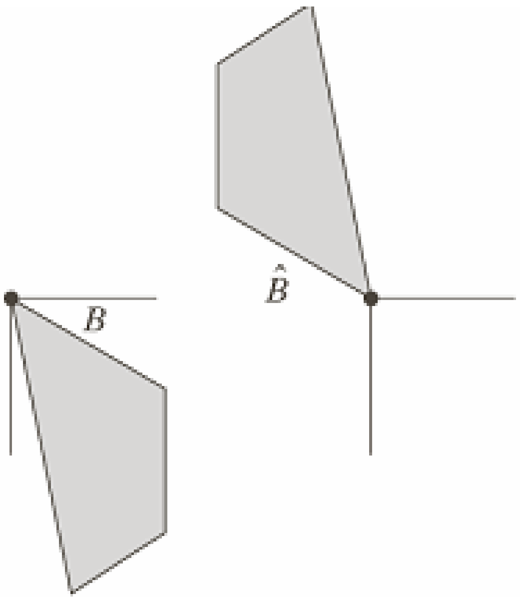
FIGURE 9.1
(a) Two sets *A* and *B*. (b) The union of *A* and *B*. (c) The intersection of *A* and *B*. (d) The complement of *A*. (e) The difference between *A* and *B*.

Set Theory

- Reflection of set B

$$B = \{w \mid w = -b, \quad \text{for } b \in B\}$$

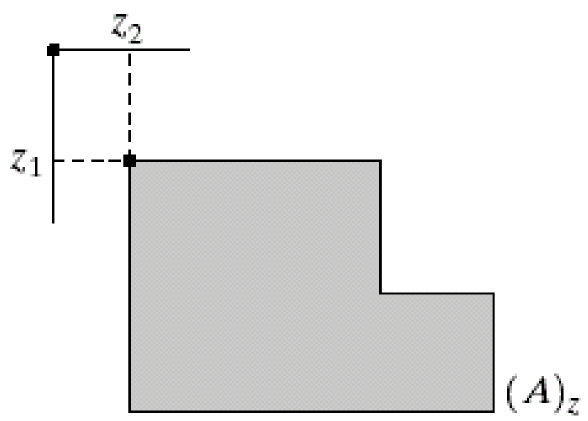
i.e. the set of element *w*, such that *w* is formed by multiplying each of two coordinates of all the elements of set *B* by -1



Set Theory

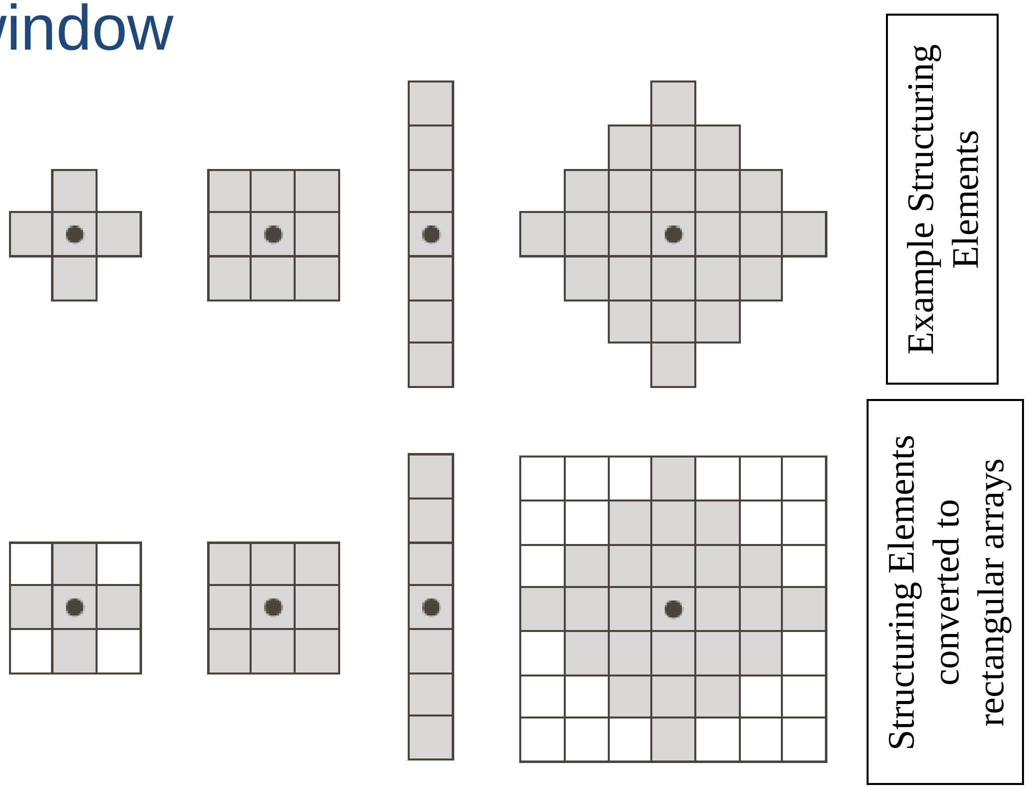
- Translation of set A by point $z = (z_1,z_2)$, denoted $(A)_z$, is defined as

$$(A)_z = \{w \mid w = a + z, \quad \text{for } a \in A\}$$



Structuring Element

A structuring element is a small image – used as a moving window



Structuring Element

For simplicity we will use rectangular structuring elements with their origin at the middle pixel

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

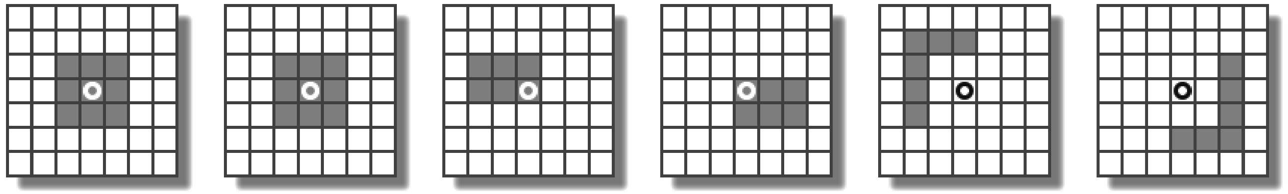
0	1	0
1	**1**	1
0	1	0

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

Structuring Element: Reflection

$$\widehat{B}(x, y) = B(-x, -y)$$

$\widehat{B}$ is B rotated by 180º around its origin.



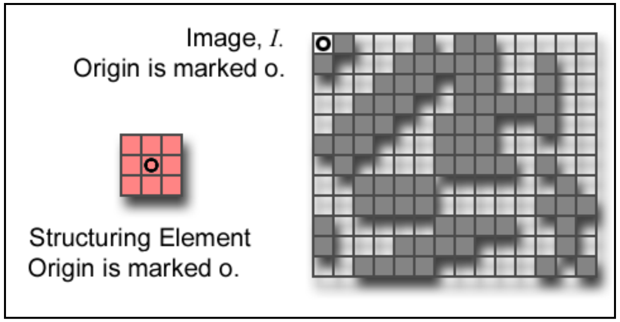
B_1 $\widehat{B}_1$ B_2 $\widehat{B}_2$ B_3 $\widehat{B}_3$

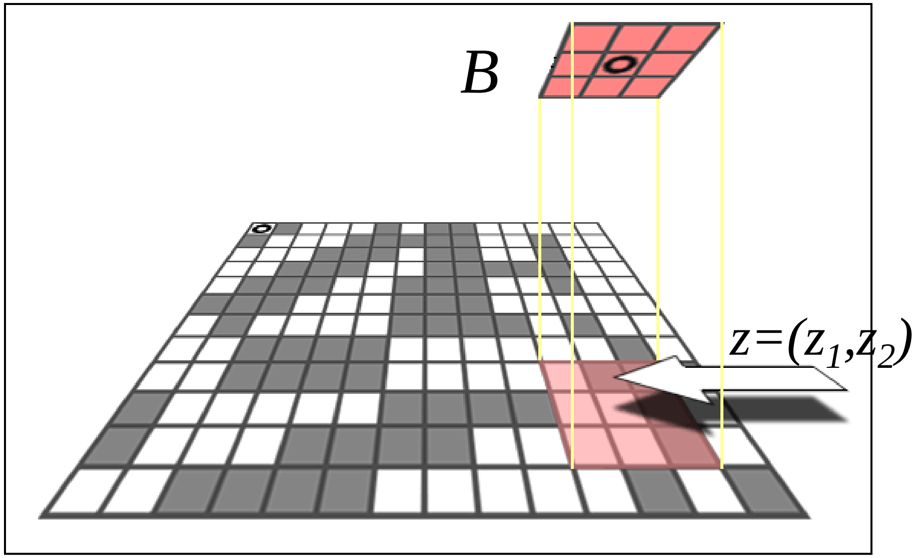
Structuring Element: Translation

Let I be an image and *B* a SE.

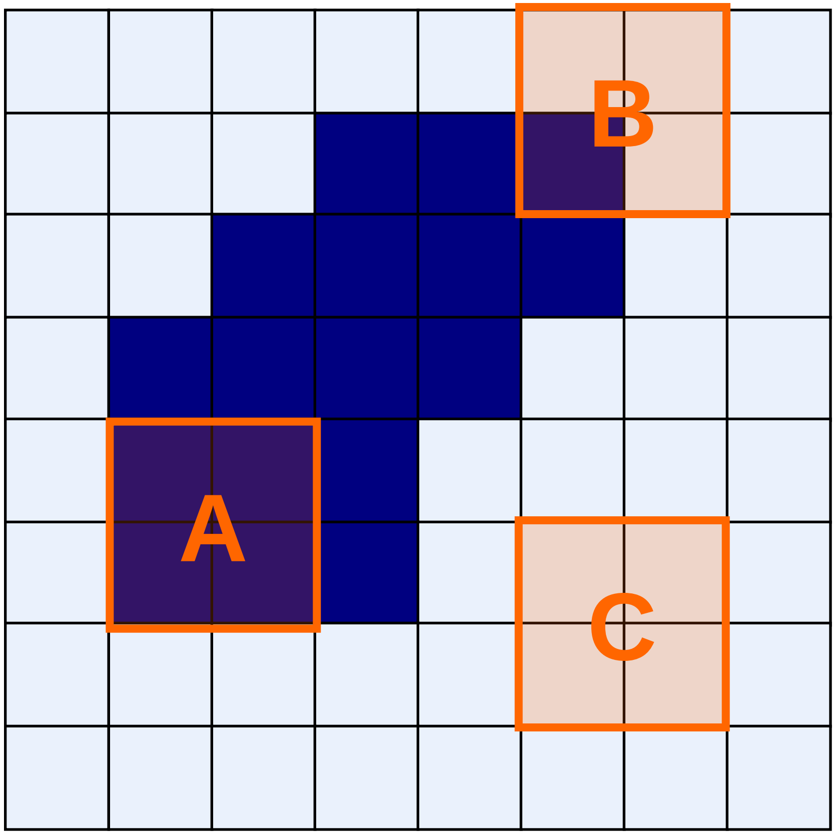
(B)$_z$ means that *B* is moved so that its origin coincides with location z in image *I*.

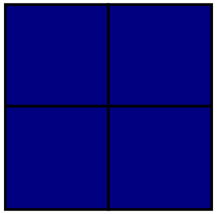
(B)$_z$ is the *translate* of *B* to location *z* in *I*.





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

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	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	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	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

B C A

1	1	1
1	1	1
1	1	1

Structuring Element 1

0	1	0
1	1	1
0	1	0

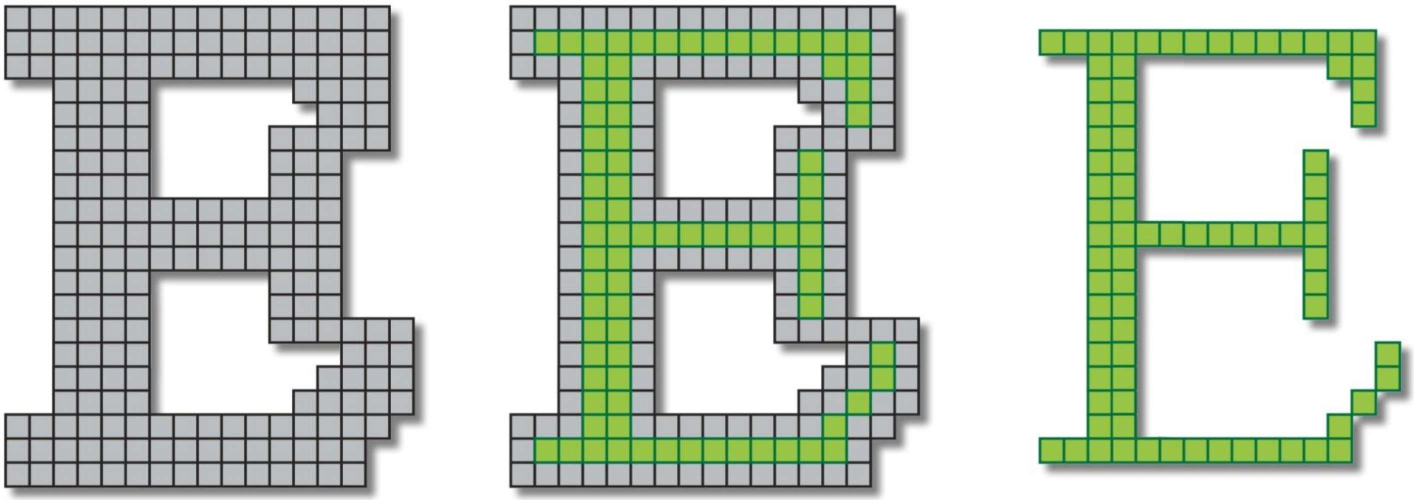
Structuring Element 2

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**

Erosion



Erosion

Definition 1:

The erosion of two sets A and B is defined as:

$$A\,!\;\; B = \{z \mid (B)_z \subseteq A\}$$

i.e. The Erosion of *A* by *B* is the set of all points *z*, such that B, translated by *z*, is contained in *A*

Erosion

Definition 2:

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:

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

Erosion – How to compute

- For each foreground pixel (which we will call the *input pixel*)
  - Superimpose the structuring element on top of the input image so that the origin of the structuring element coincides with the input pixel position.
  - If *for every* pixel in the structuring element, the corresponding pixel in the image underneath is a foreground pixel, then the input pixel is left as it is.
  - If any of the corresponding pixels in the image are background, however, the input pixel is also set to background value.

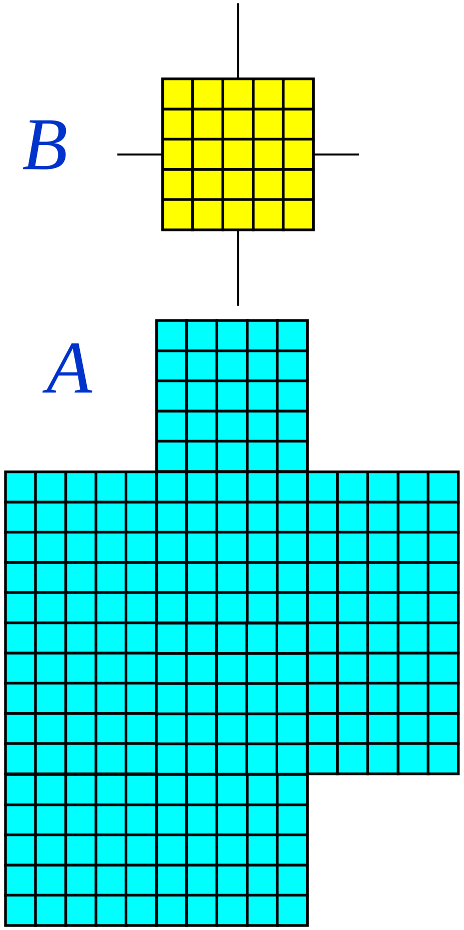
Erosion – How to compute

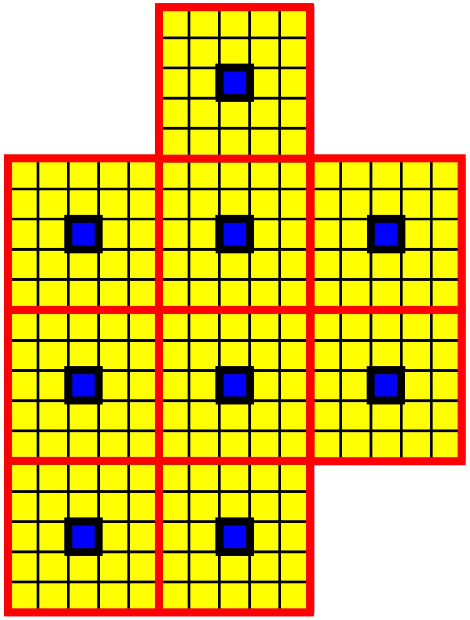
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	1	1	1	0	0
0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
0	0	0	1	1	0	0	0	0	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	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	0	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	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	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	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	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

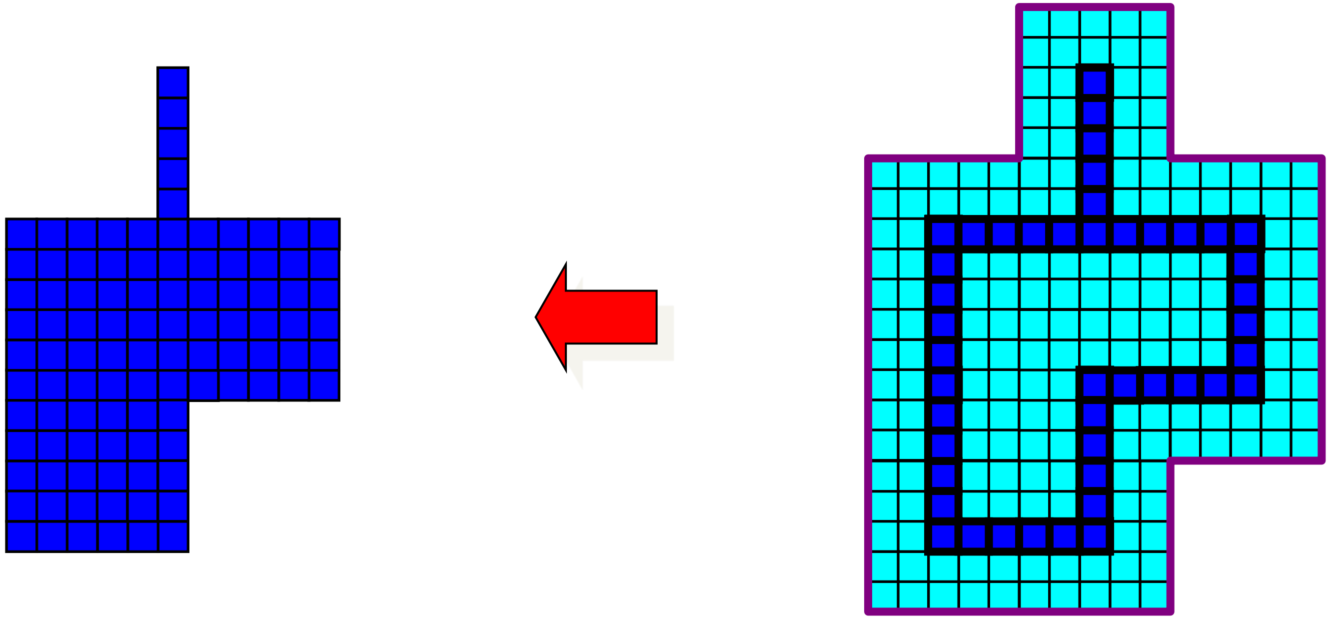
Erosion with a structuring element of size 3x3

Erosion

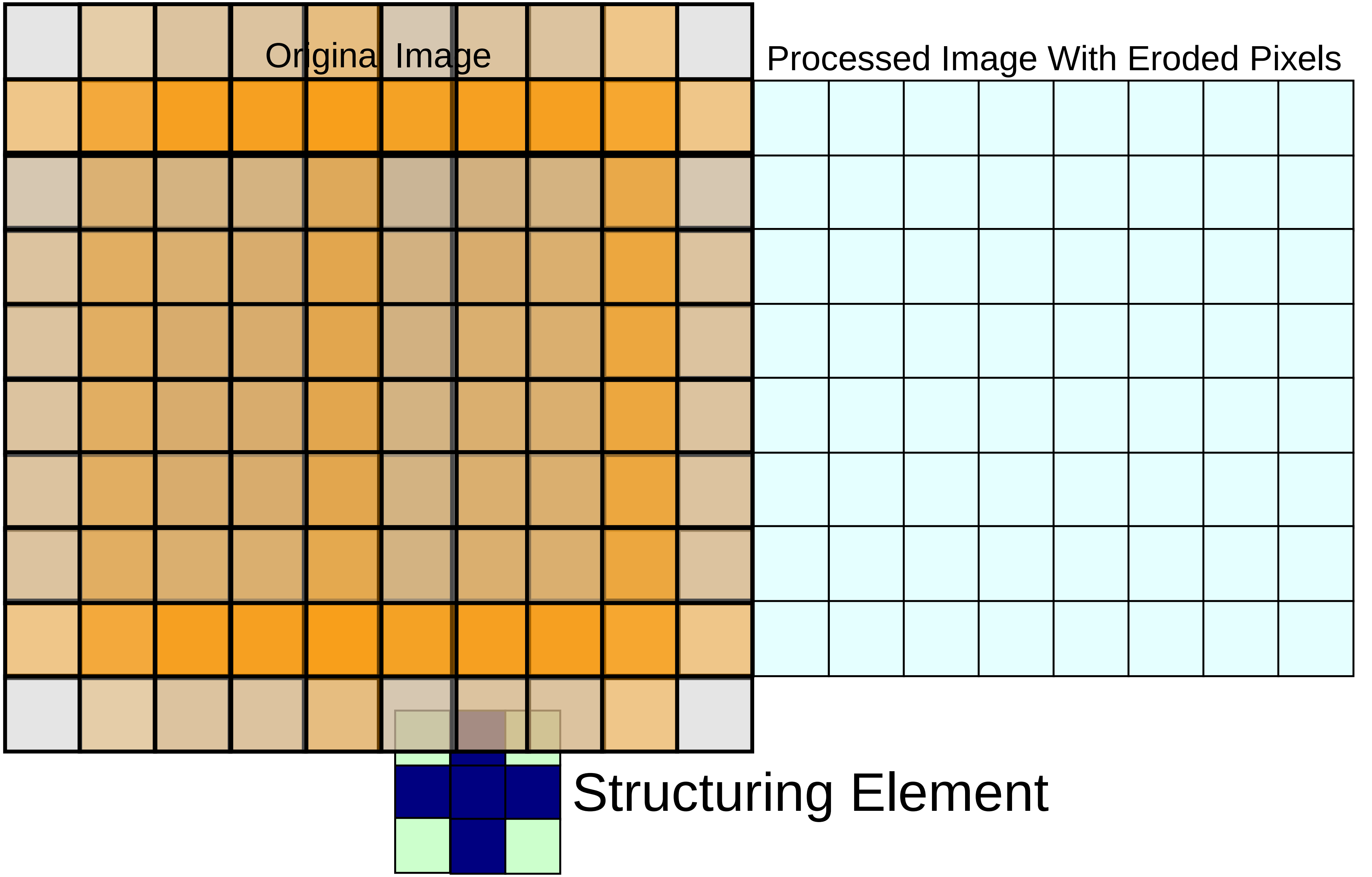




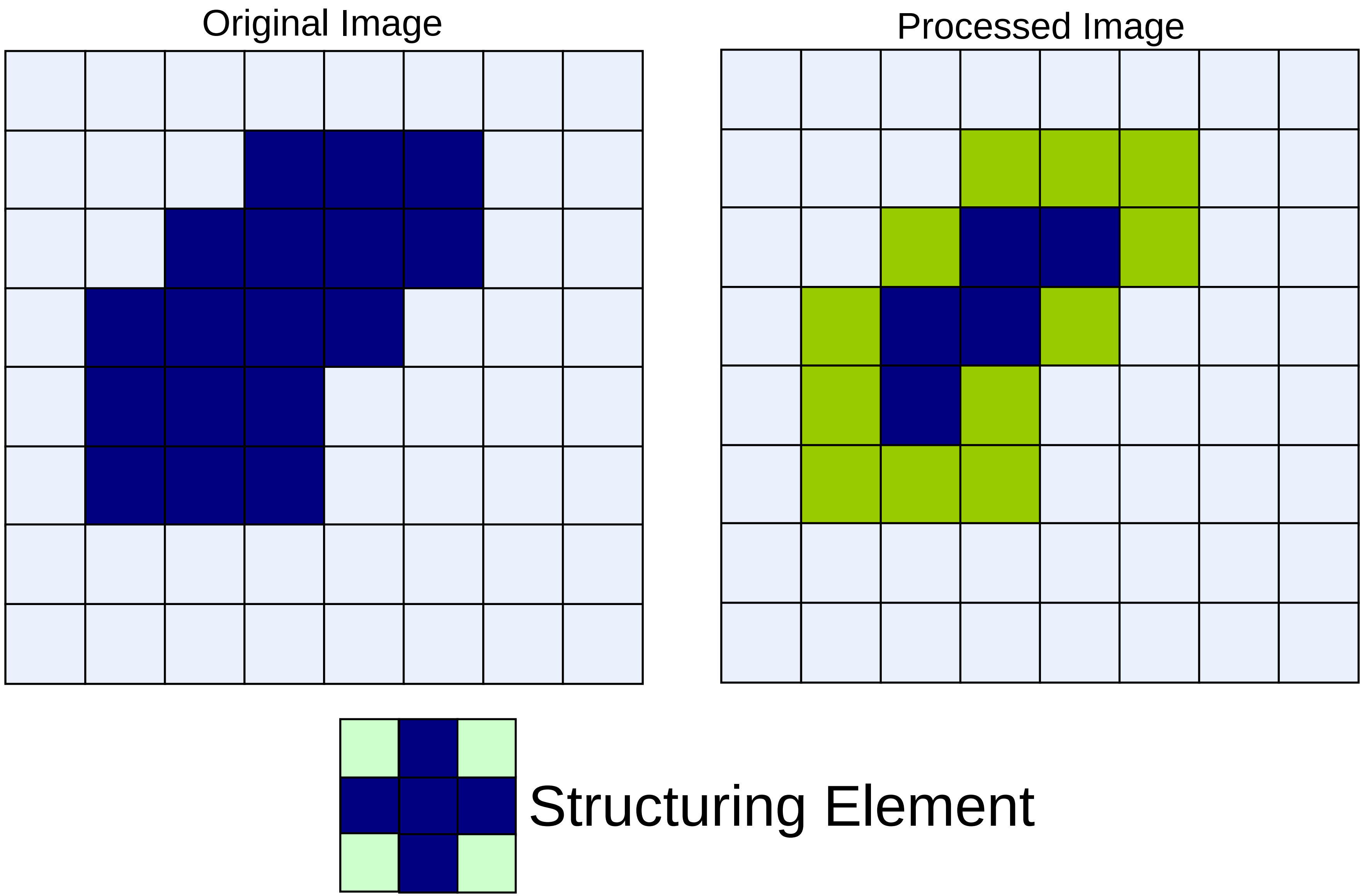
Erosion



Erosion: Example



Erosion: Example



Erosion

- Effects
  - Shrinks the size of foreground (1-valued) objects
  - Smoothes object boundaries
  - Removes small objects

- Rule for Erosion

  In a binary image, if any of the pixel (in the neighborhood defined by structuring element) is 0, then output is 0

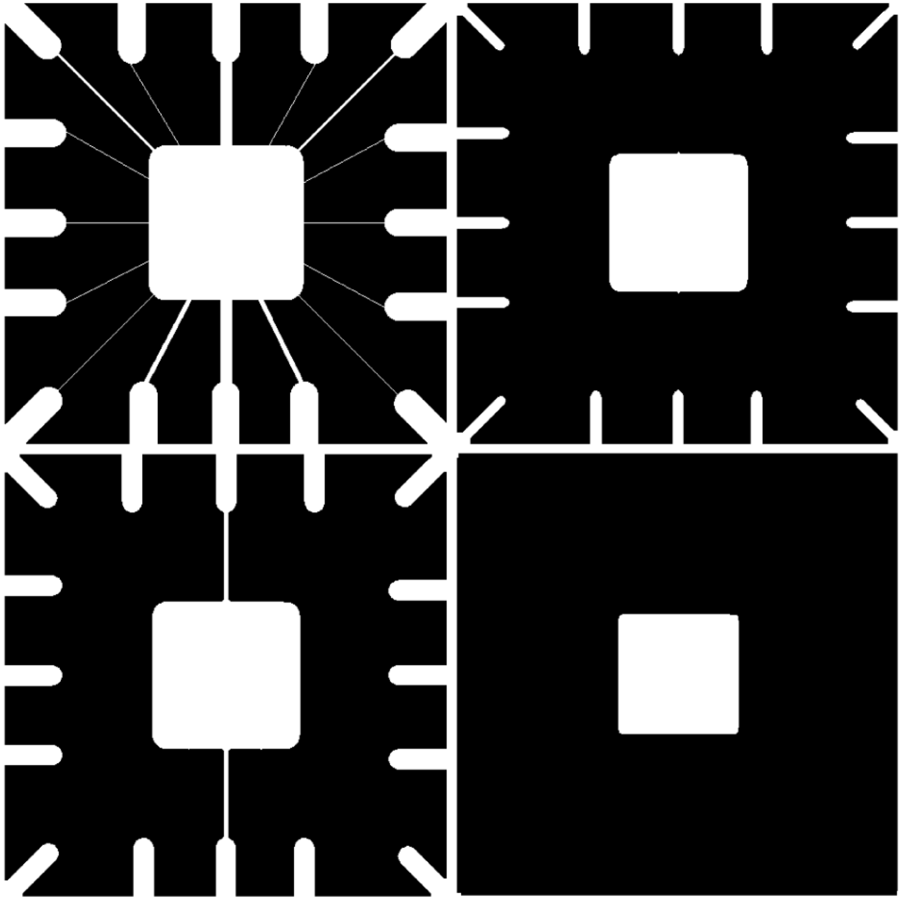
Erosion: Example 1

Original image

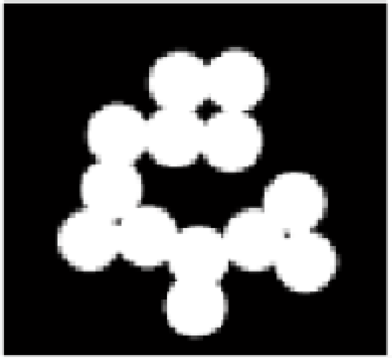
After erosion with a disc of radius 10

After erosion with a disc of radius 5

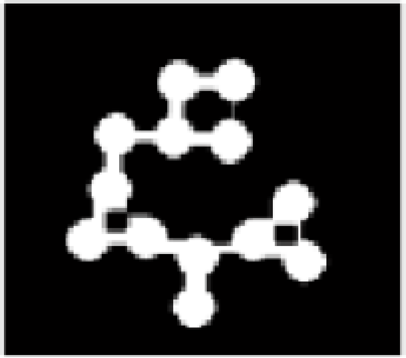
After erosion with a disc of radius 20



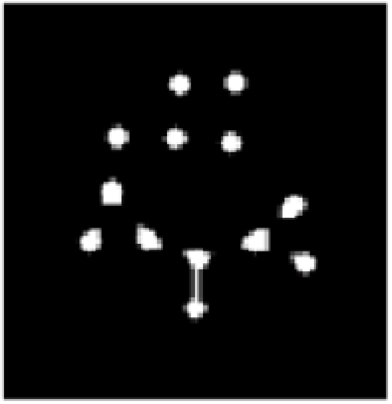
Erosion: Example 2



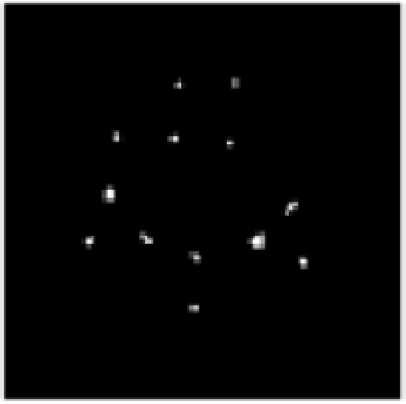
Original binary image *circles*



Erosion by 11x11 structuring element



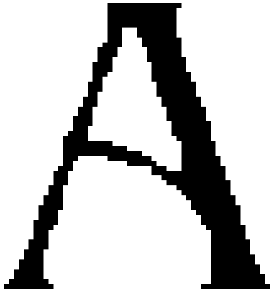
Erosion by 21x21 structuring element



Erosion by 27x27 structuring element

Erosion: Example 3

Original image



Erosion by 3*3 square structuring element

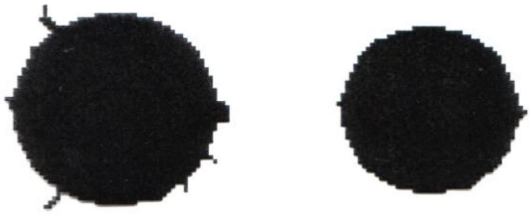
Erosion by 5*5 square structuring element

Note: In these examples a 1 refers to a black pixel!

Erosion

Erosion can split apart joined objects

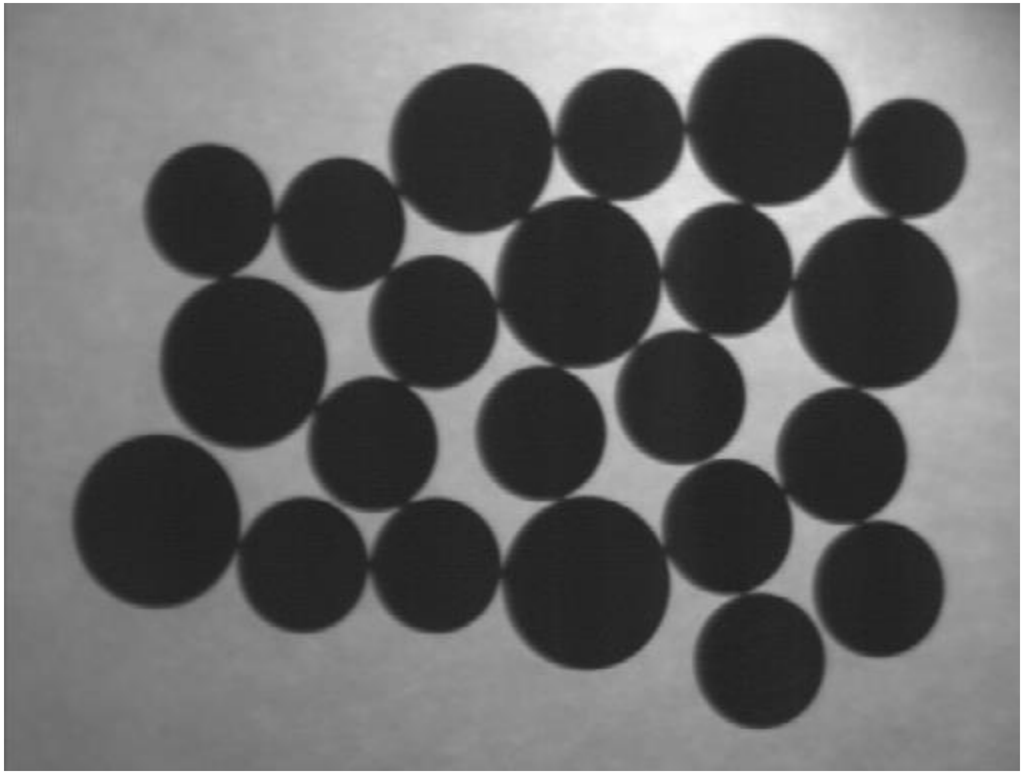
Erosion can strip away extrusions



Watch out: Erosion shrinks objects

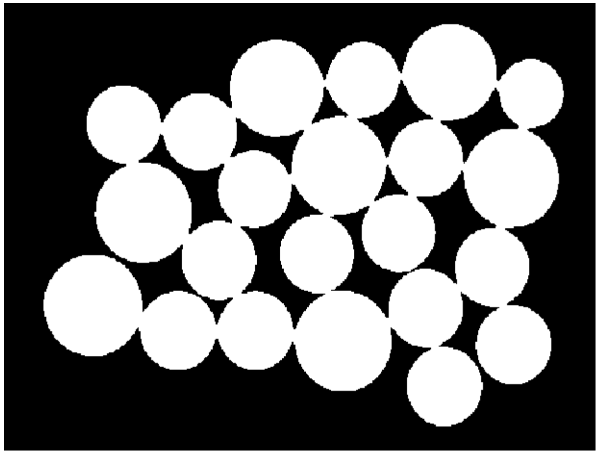
Exercise

Count the number of coins in the given image

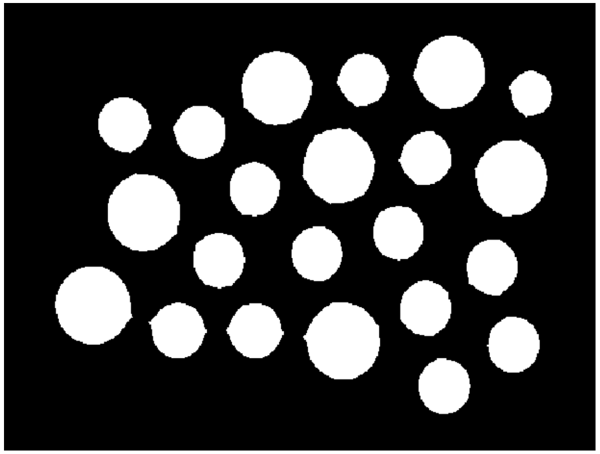


Exercise: Solution

Binarize the image

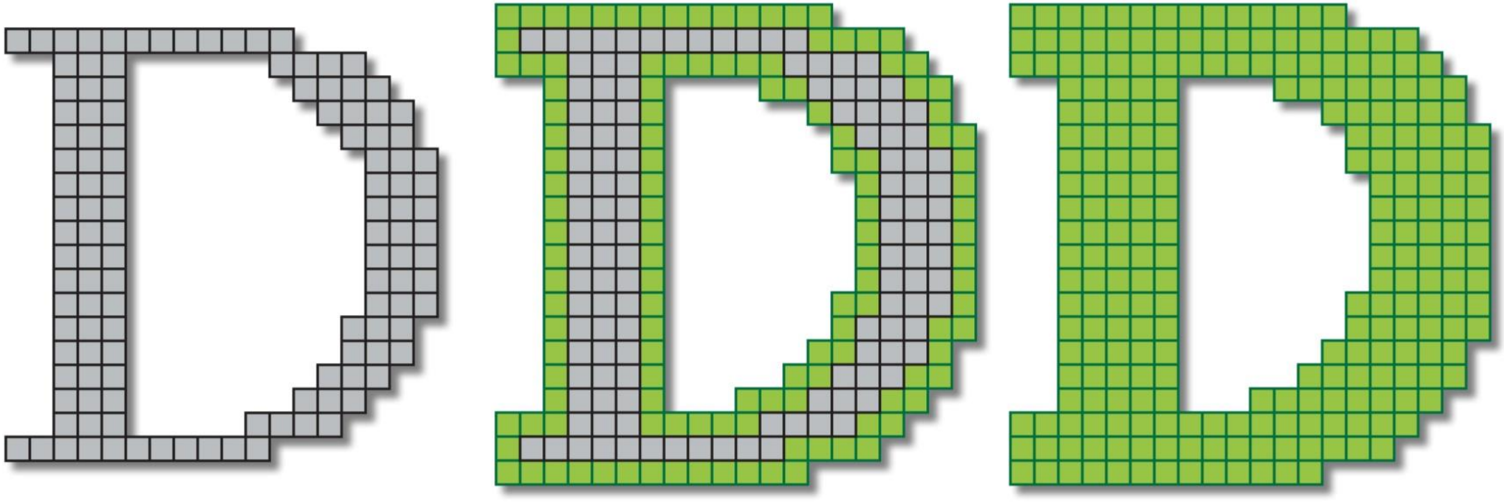


Perform Erosion



Use connected component labeling to count the number of coins

Dilation



Dilation

Definition 1:

The dilation of two sets A and B is defined as:

$$A \oplus B = \{z \mid (B)_z \cap A \neq \varnothing\}$$

i.e. when the reflection of set *B* about its origin is shifted by *z*, the dilation of *A* by *B* is the set of all displacements such that overlap *A* by at least one element

We will only consider symmetric SEs so reflection will have no effect

Dilation

Definition 2:

Dilation of image *f* by structuring element *s* is given by

$f \oplus s$

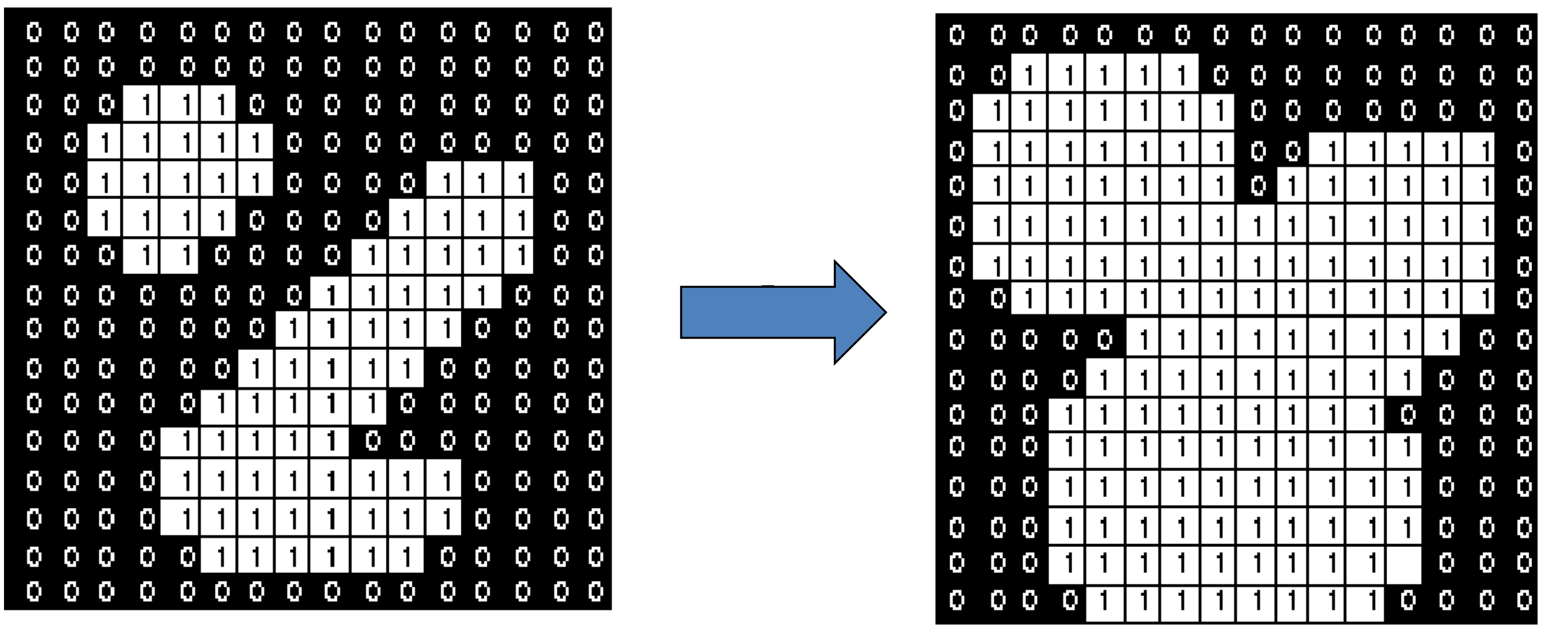
The structuring element s is positioned with its origin at *(x, y)* and the new pixel value is determined using the rule:

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

Dilation – How to compute

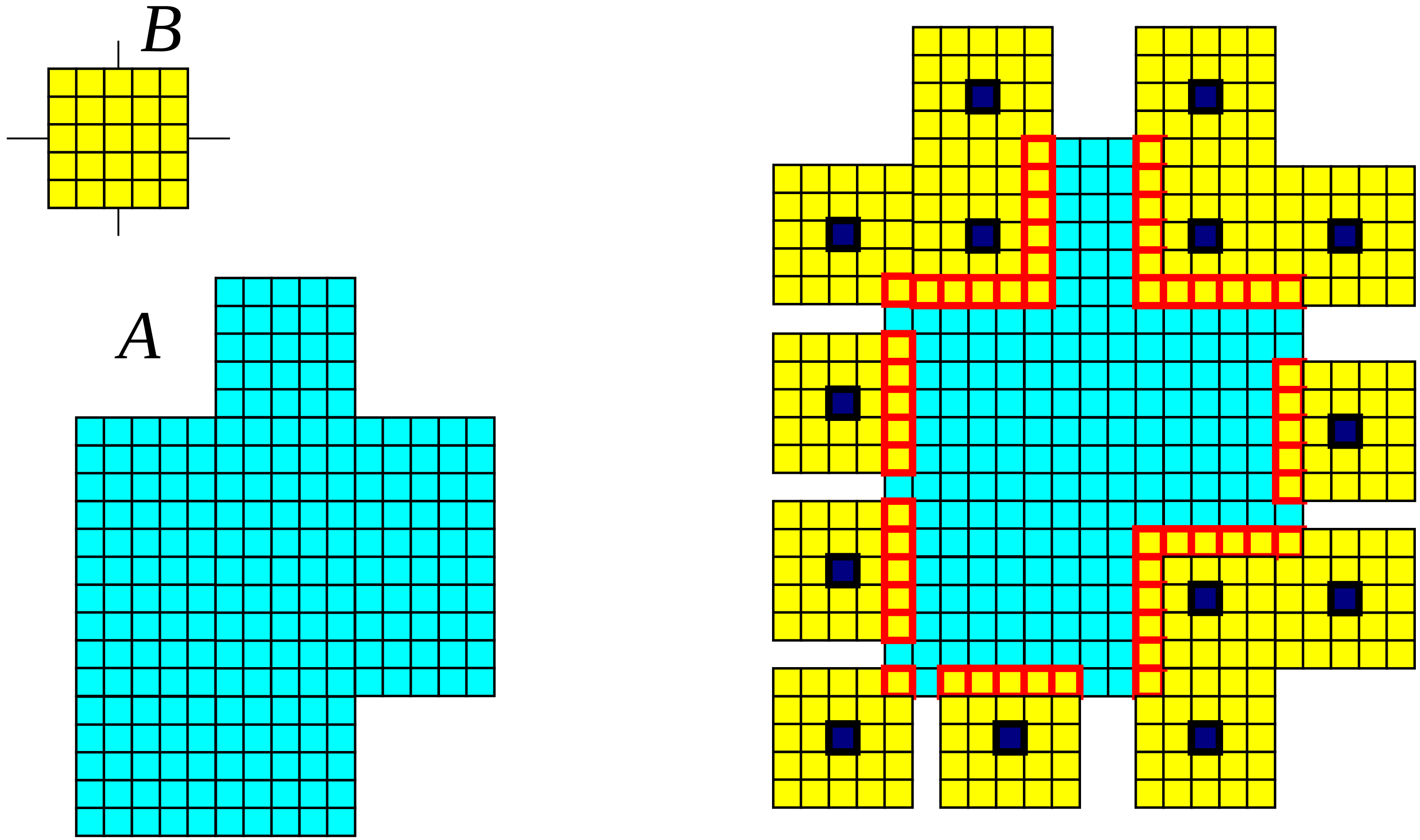
- For each background pixel (which we will call the *input pixel*)
  - Superimpose the structuring element on top of the input image so that the origin of the structuring element coincides with the input pixel position
  - If *at least one* pixel in the structuring element coincides with a foreground pixel in the image underneath, then the input pixel is set to the foreground value
  - If all the corresponding pixels in the image are background, however, the input pixel is left at the background value

Dilation: Example

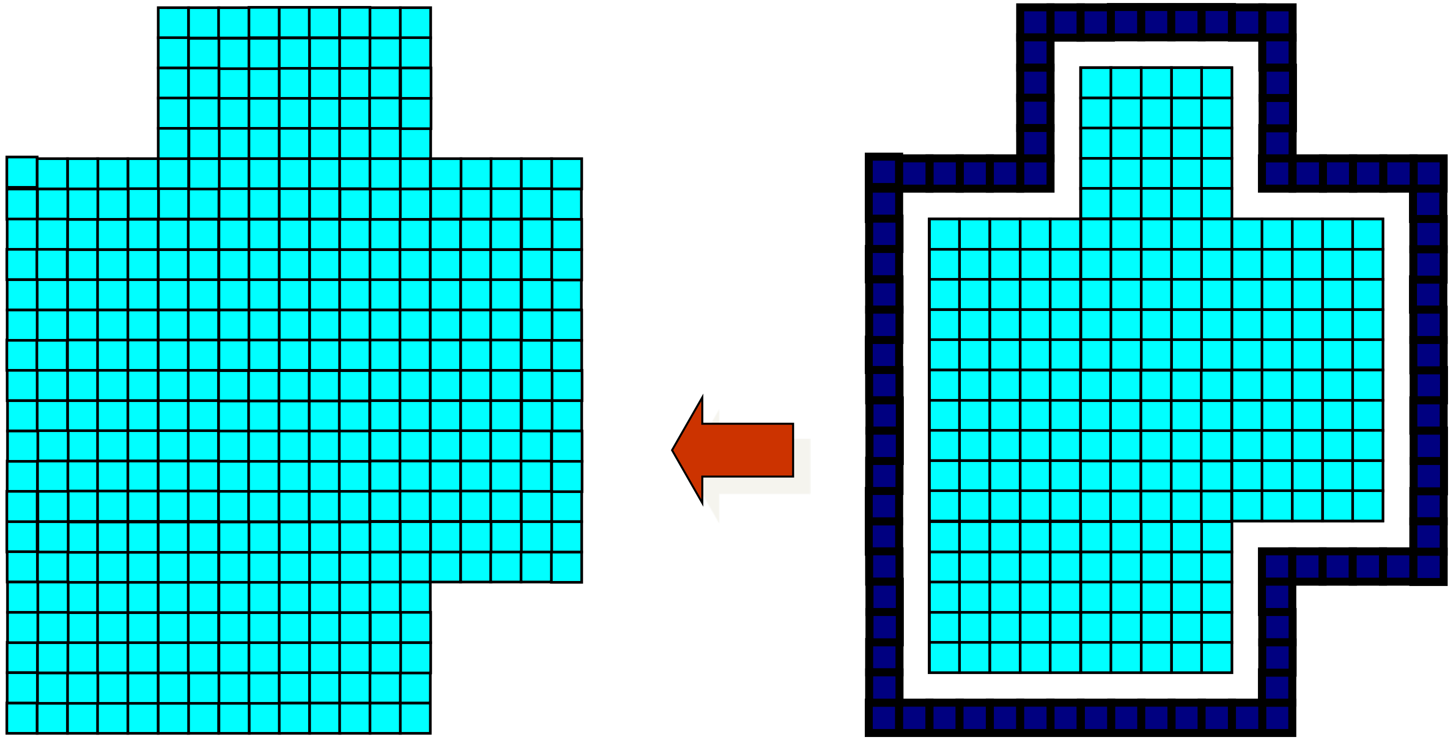


Effect of dilation using a 3×3 square structuring element

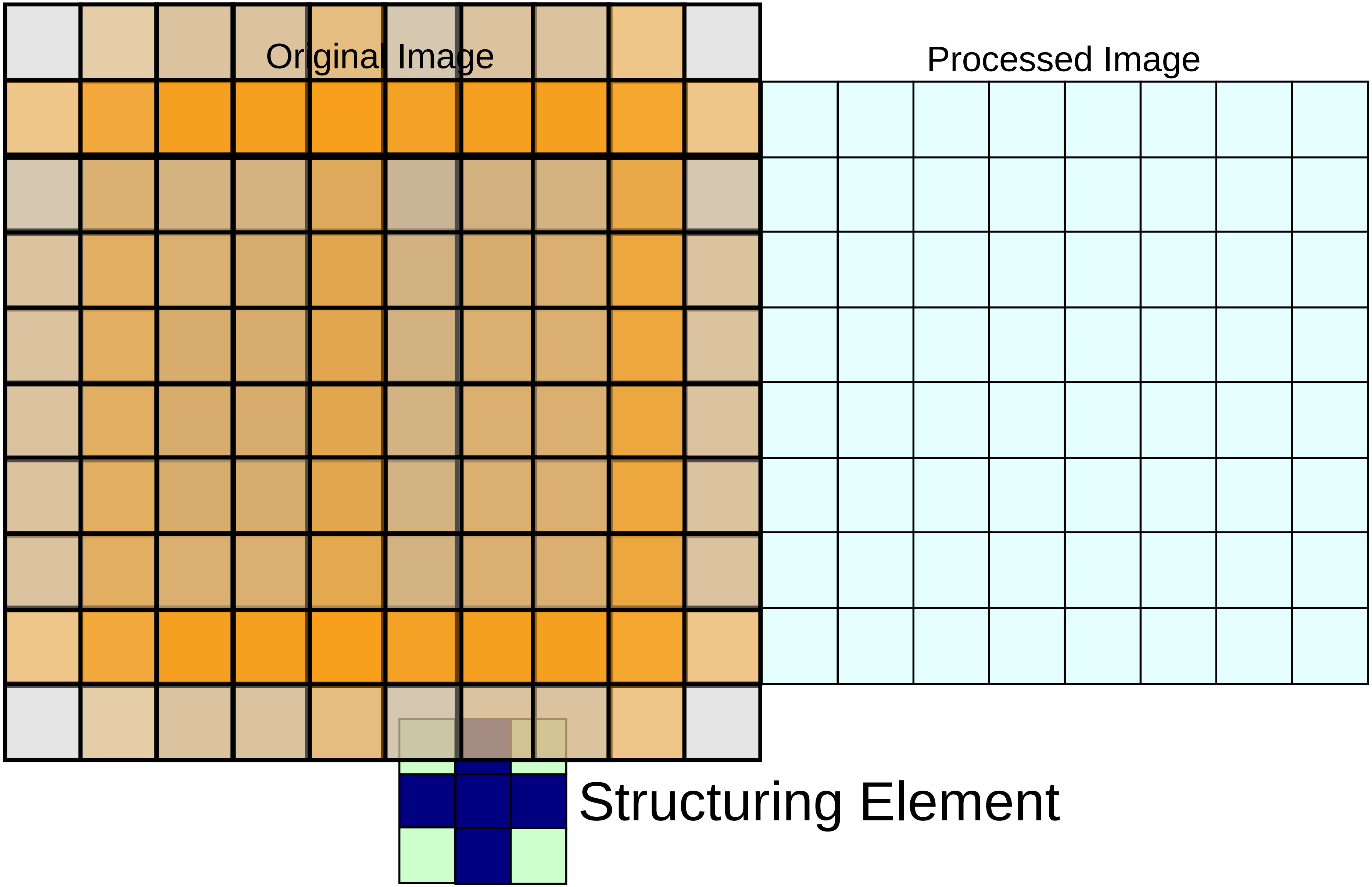
Dilation



Dilation

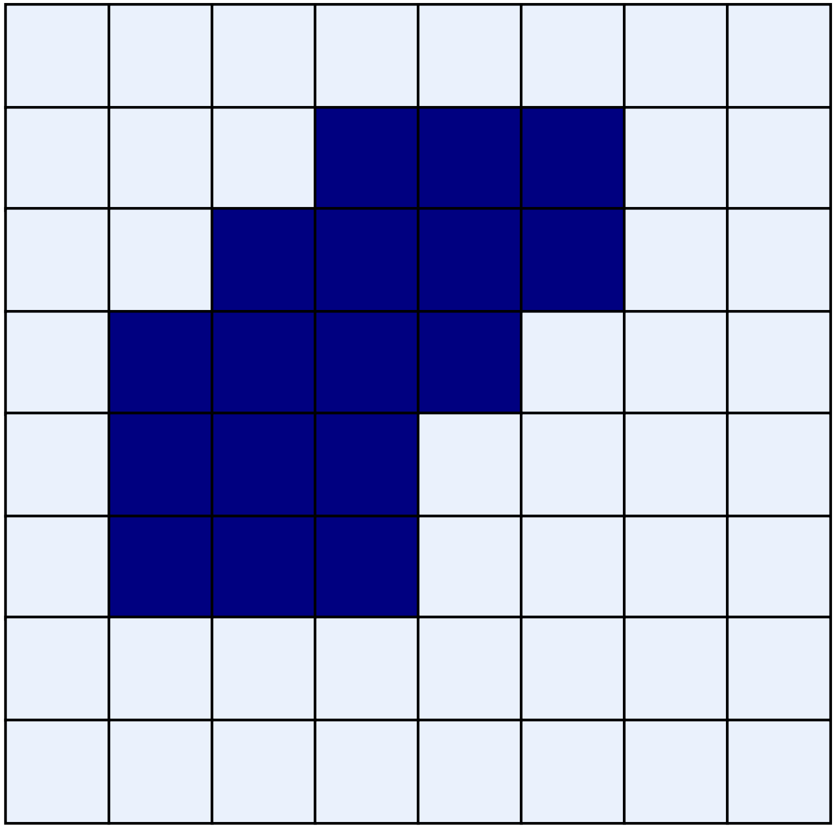


Dilation: Example

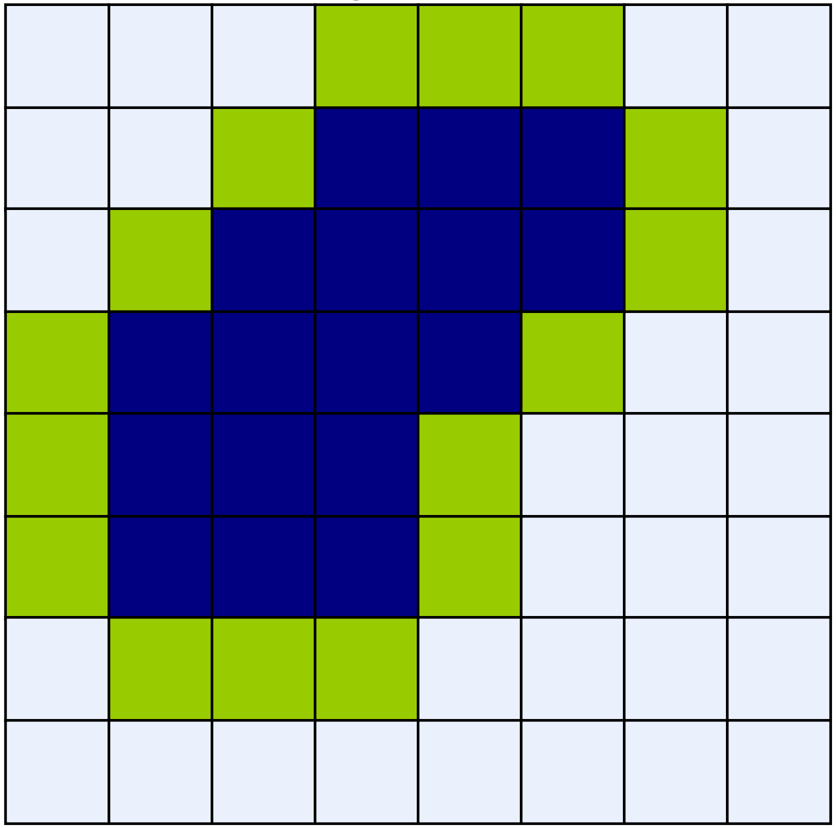


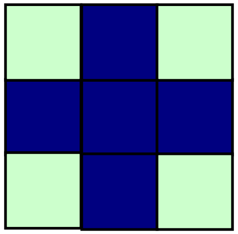
Dilation: Example

Original Image



Processed Image With Dilated Pixels





Structuring Element

Dilation

- Effects
  - Expands the size of foreground(1-valued) objects
  - Smoothes object boundaries
  - Closes holes and gaps

- Rule for Dilation

  In a binary image, if any of the pixel (in the neighborhood defined by structuring element) is 1, then output is 1

Dilation: Example 1

Original image

Dilation by 3*3 square structuring element

Dilation by 5*5 square structuring element

Note: In these examples a 1 refers to a black pixel!

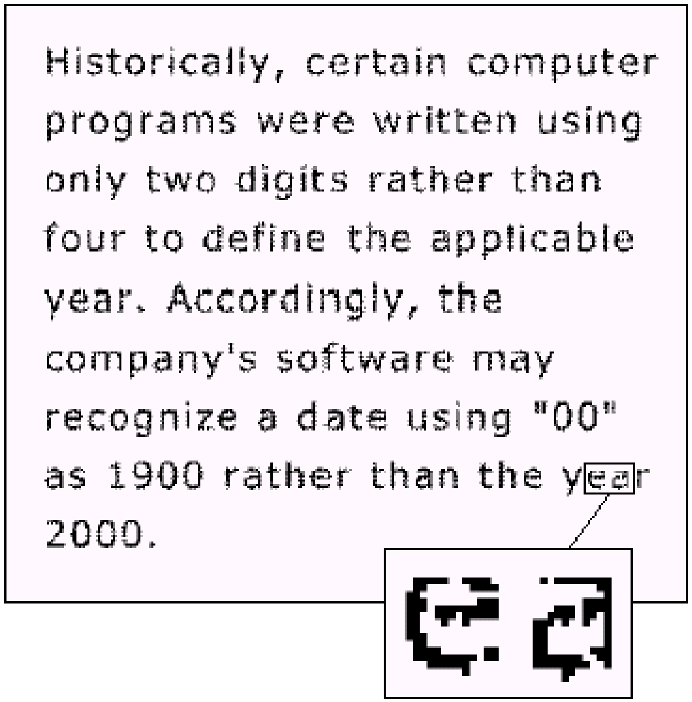
Dilation: Example 2

Original (178x178)

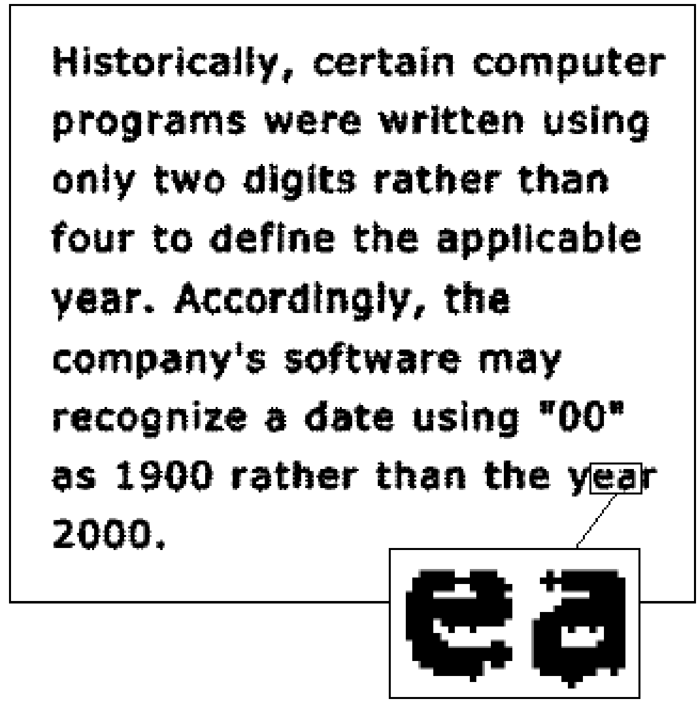
dilation with
3x3 structuring element

dilation with
7x7 structuring element

Dilation: Example 3



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.



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.

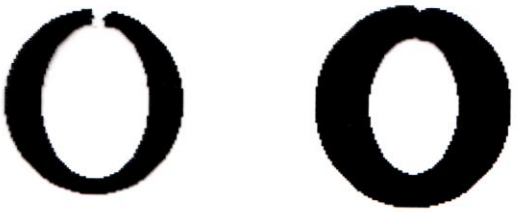
a c
b

FIGURE 9.5
(a) Sample text of poor resolution with broken characters (magnified view).
(b) Structuring element.
(c) Dilation of (a) by (b). Broken segments were joined.

0	1	0
1	1	1
0	1	0

Dilation

Dilation can repair breaks



Dilation can repair intrusions

Watch out: Dilation enlarges objects

Duality relationship between Dilation and Erosion

- Dilation and erosion are duals of each other:

$$(A!\ \ B)^c = A^c \oplus B$$

- For a symmetric structuring element:

$$(A!\ \ B)^c = A^c \oplus B$$

It means that we can obtain erosion of an image A by B simply by dilating its background (i.e. A^c) with the same structuring element and complementing the result.

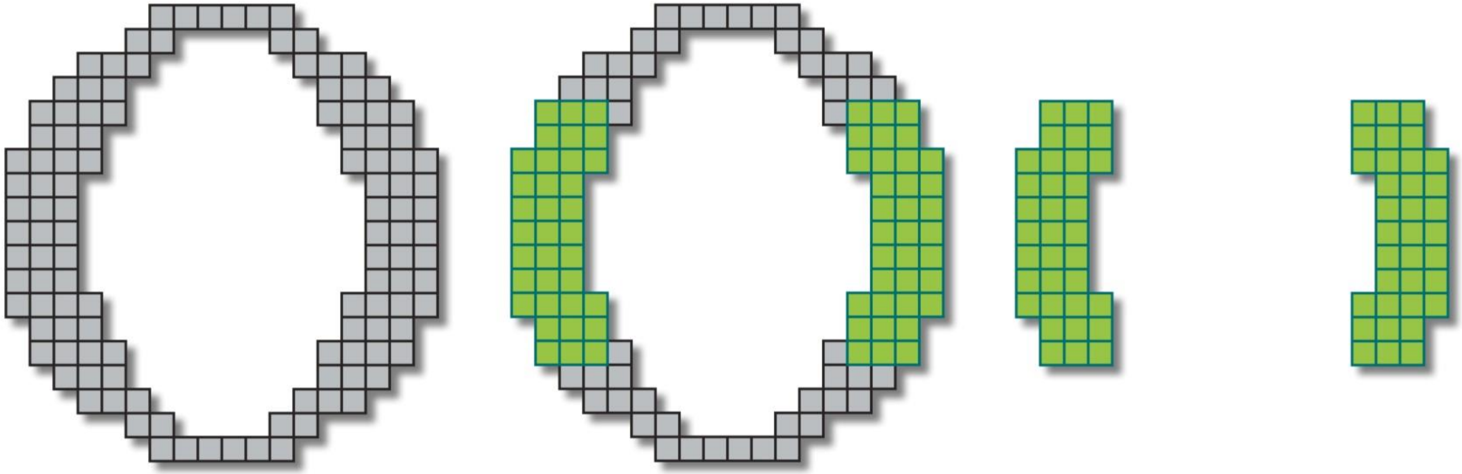
Compound Operations

- 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

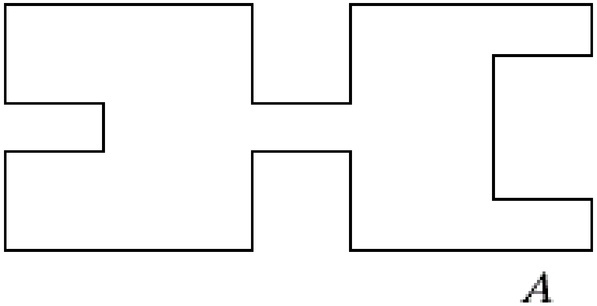
Opening



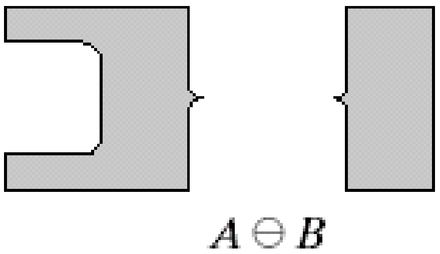
Opening

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

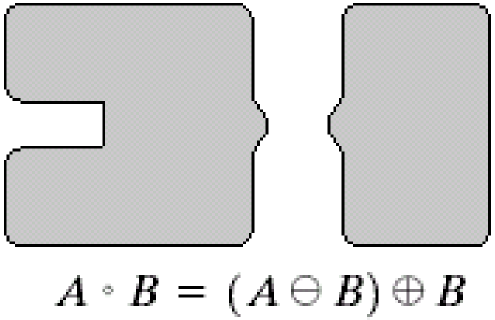
$$f \circ s = (f \ominus s) \oplus s$$



Original shape



After erosion



After dilation (opening)

Opening: Example

Original Image

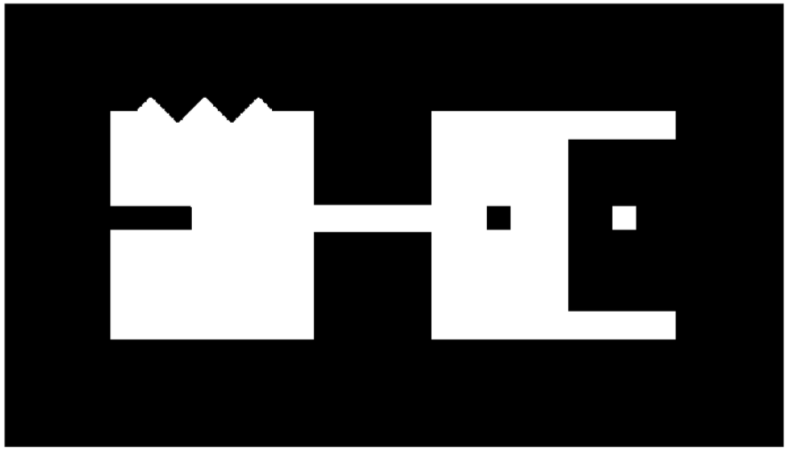
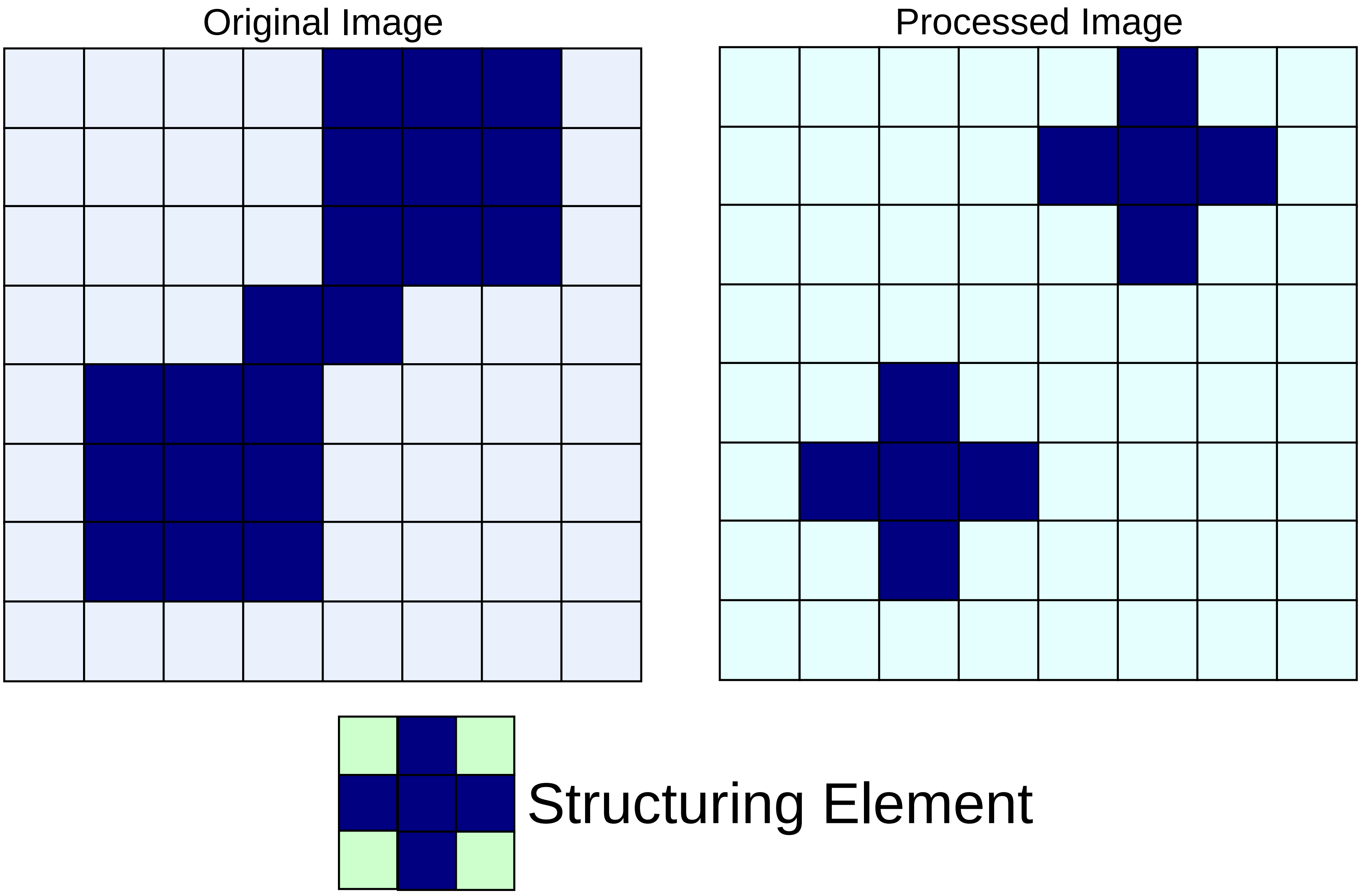


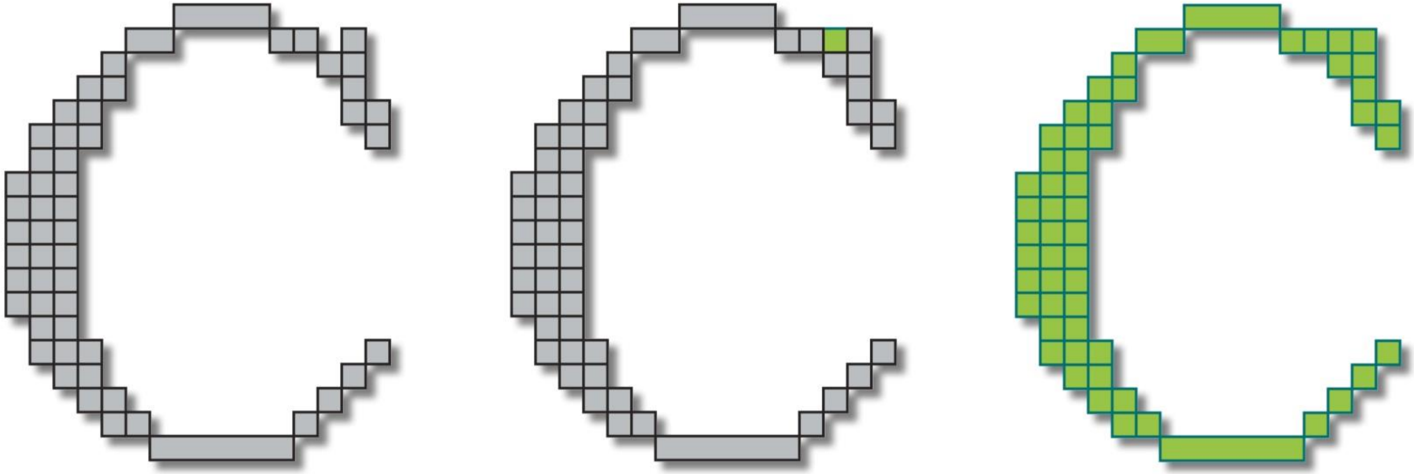
Image After Opening

Opening
Breaks narrow joints
Removes ‘Salt’ noise

Opening: Example



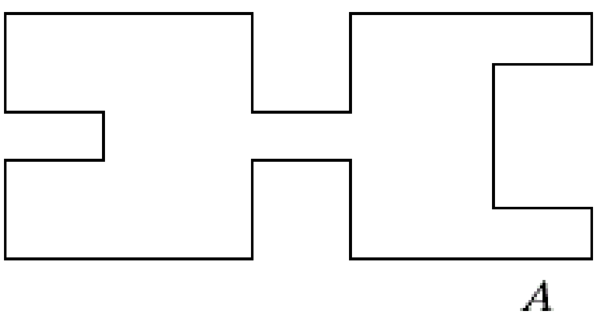
Closing



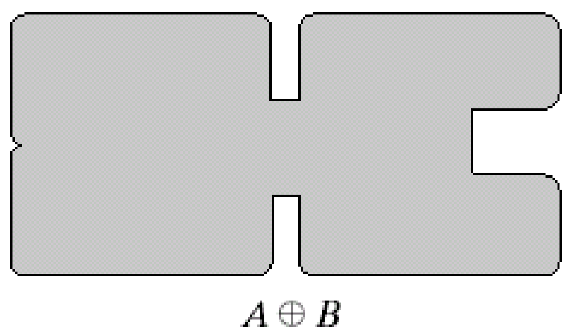
Closing

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

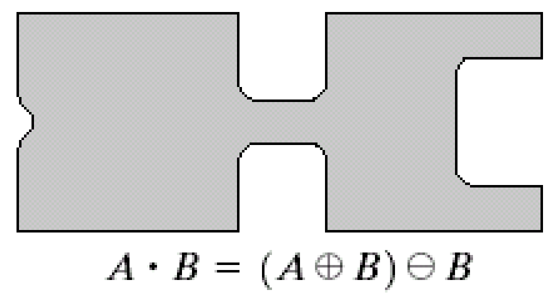
$$f \bullet s = (f \oplus s) \ominus s$$



Original shape



After dilation



After erosion (closing)

Closing: Example

Original Image

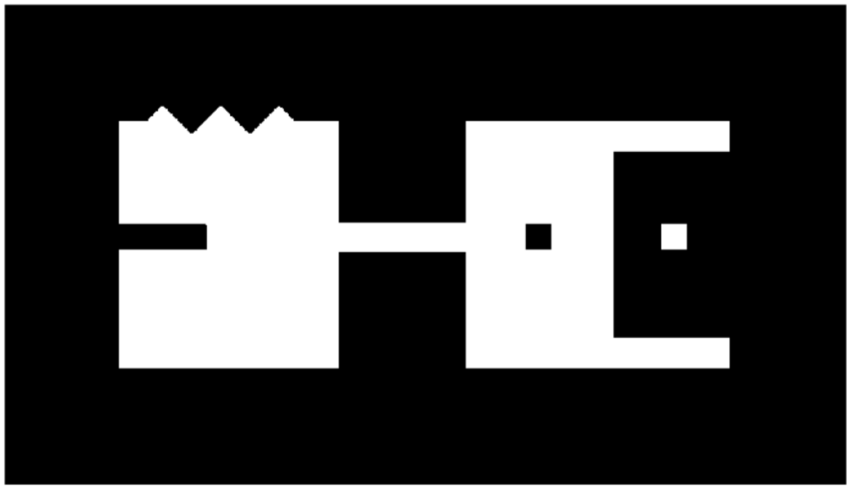
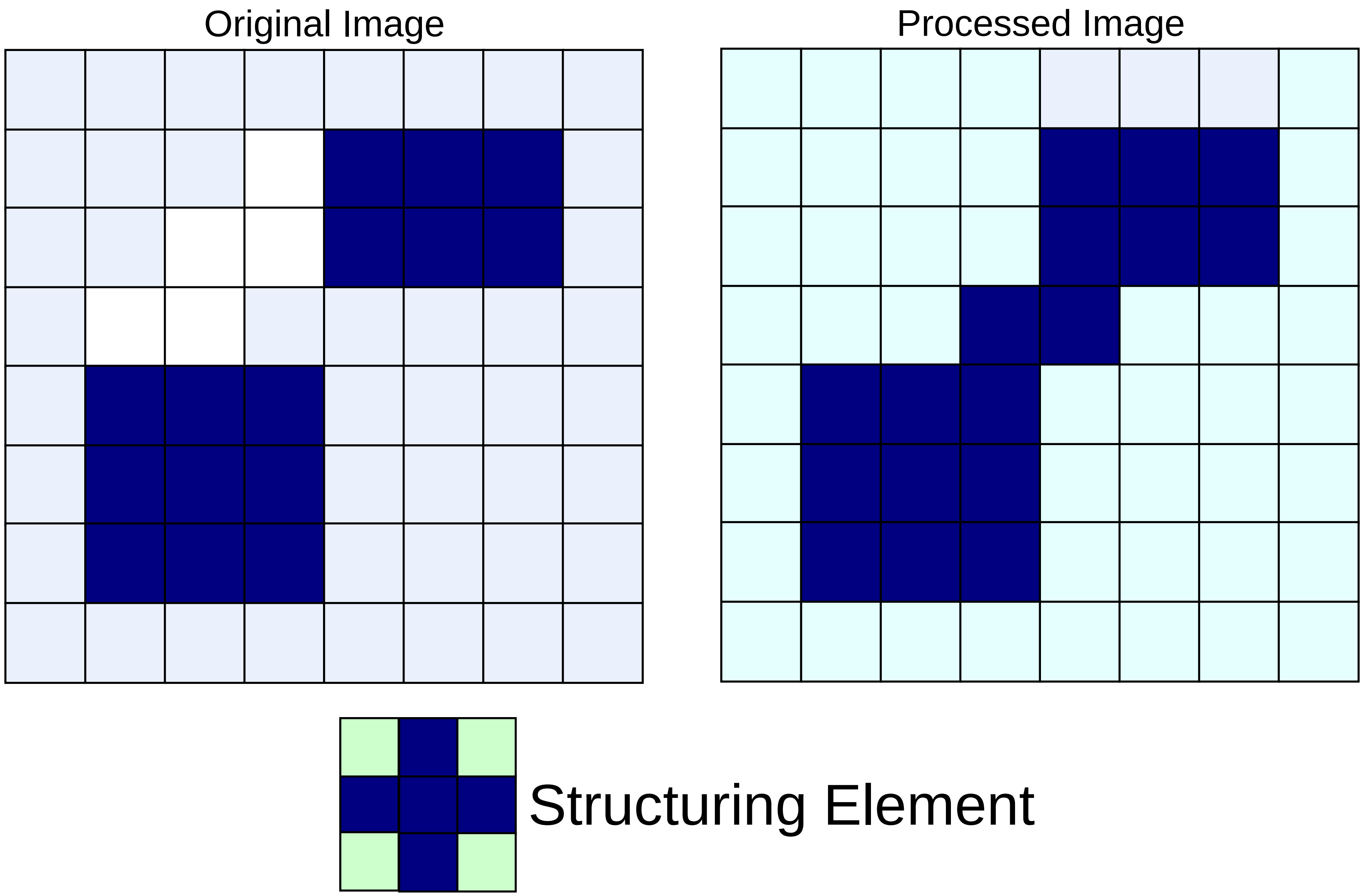


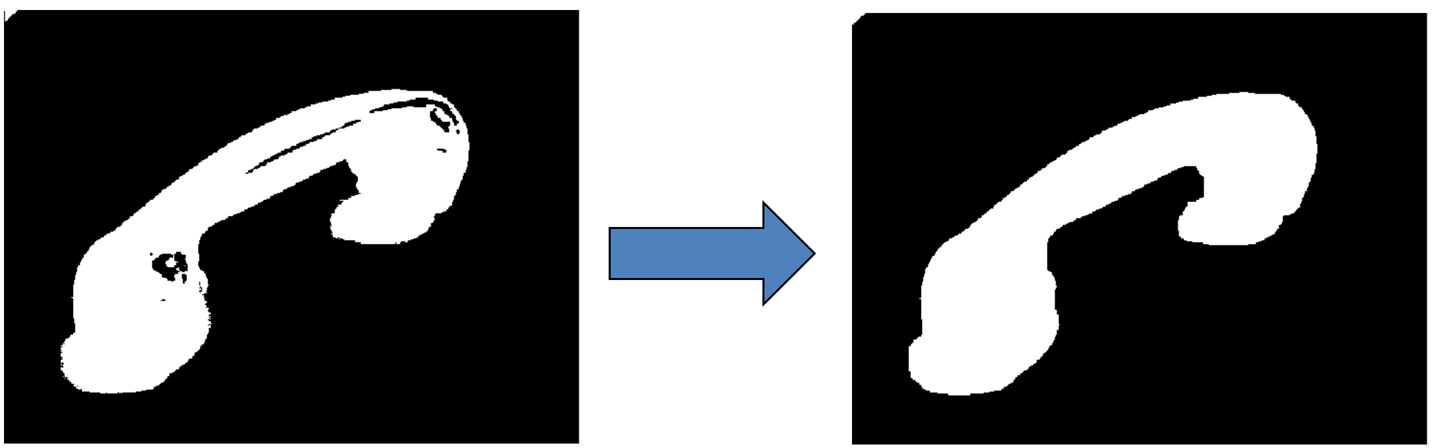
Image After Closing

Closing
- Eliminates small holes
- Fills gaps
- Removes 'Pepper' noise

Closing: Example



Closing



Opening & Closing

- Opening and closing are duals of each others

$$(A \bullet B)^c = (A^c \circ B)$$

$$(A \circ B)^c = (A^c \bullet B)$$

Morphological Processing Example

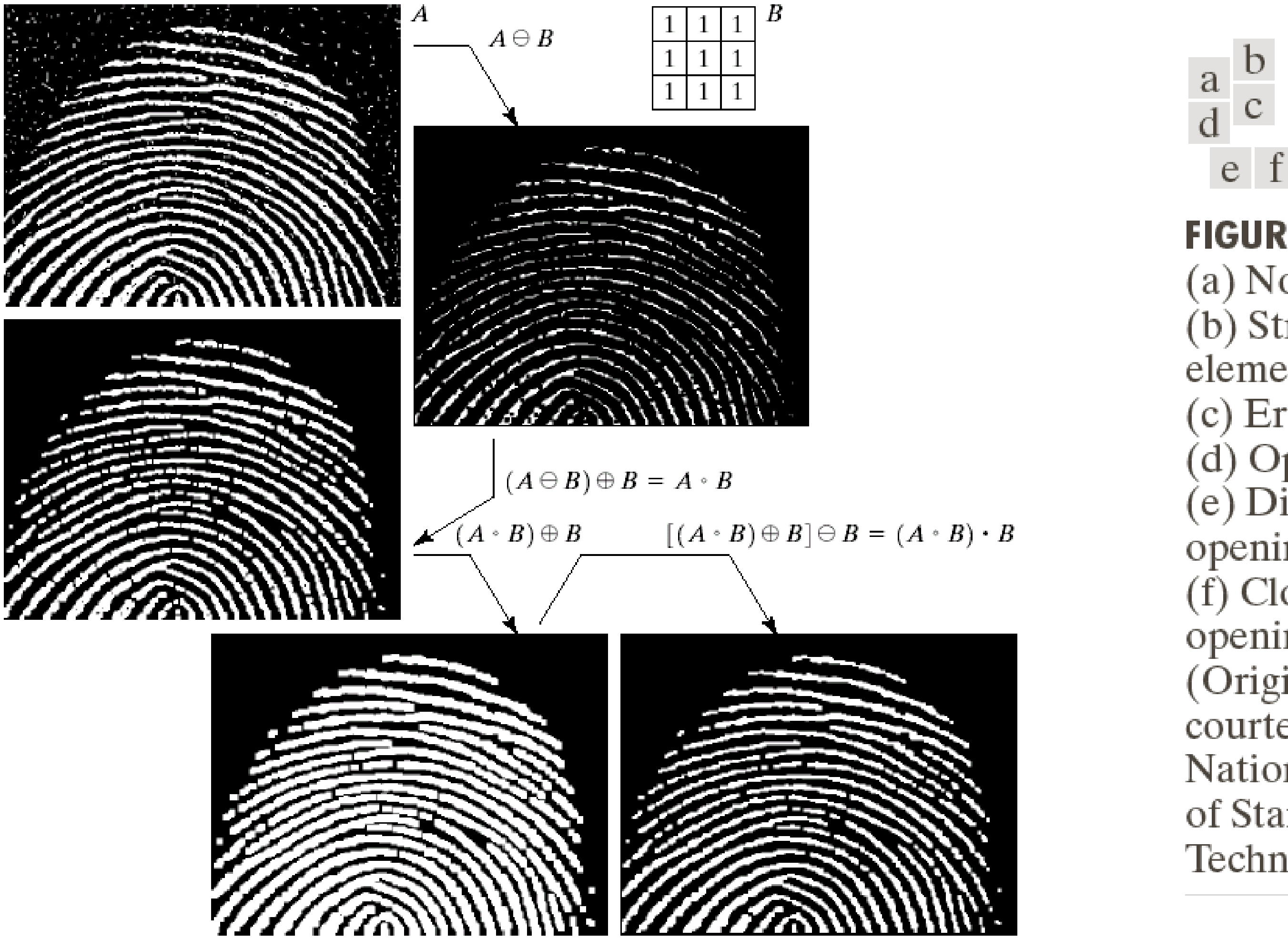


FIGURE 9.11
(a) Noisy image.
(b) Structuring element.
(c) Eroded image.
(d) Opening of A.
(e) Dilation of the opening.
(f) Closing of the opening.
(Original image courtesy of the National Institute of Standards and Technology.)

Morphological Algorithms

Using the simple technique we have looked at so far we can begin to consider some more interesting morphological algorithms

We will look at:

- Boundary extraction
- Region filling
- Extraction of connected components

There are lots of others as well though:

- Thinning/thickening
- Skeletonization

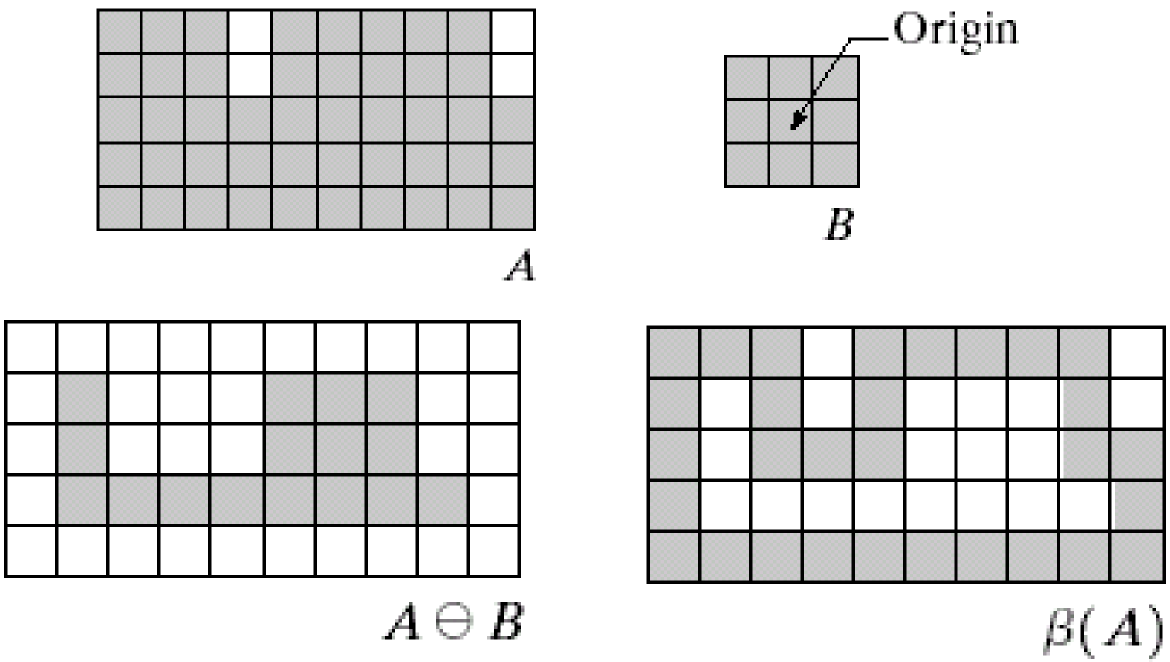
Boundary Extraction

The boundary of set A denoted by *β(A)* is obtained by first eroding *A* by a suitable structuring element *B* and then taking the difference between *A* and its erosion.

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

Boundary Extraction

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



Boundary Extraction

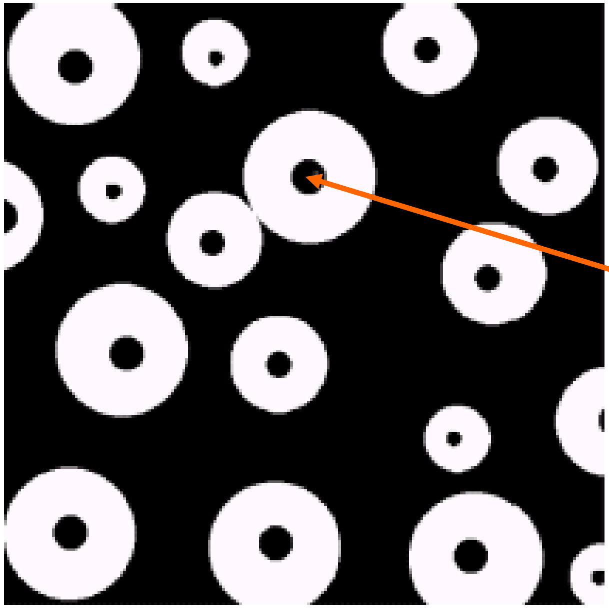
A simple image and the result of performing boundary extraction using a square 3*3 structuring element

Original Image

Extracted Boundary

Region (hole) Filling

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

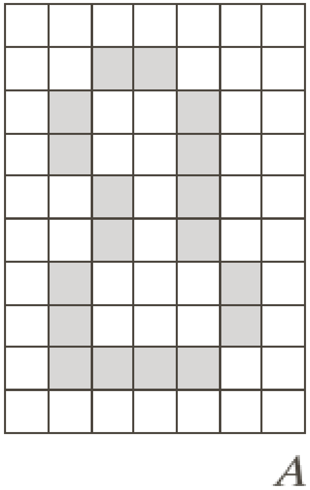


Given a point inside here, can we fill the whole circle?

Region Filling

Let A is a set containing a subset whose elements are 8-connected boundary points of a region, enclosing a background region i.e. hole

If all boundary points are labeled 1 and non boundary points are labeled 0, the following procedure fills the region:



Inside the boundary

- Start from a known point p and taking $X_0 = p$,
- Then taking the next values of X_k as:

$$X_k = (X_k \oplus B) \cap A^c \qquad k = 1, 2, 3, \cdots$$

B is suitable structuring element

- Terminate iterations if $X_{k+1} = X_k$
- The set union of X_k and A contains the filled set and its boundaries.

Region Filling

0	0	0	0	0	0	0
0	1	1	1	1	1	0
0	1	0	0	0	1	0
0	1	0	0	0	1	0
0	1	0	0	0	1	0
0	1	1	1	1	1	0
0	0	0	0	0	0	0

A

1	1	1	1	1	1	1
1	0	0	0	0	0	1
1	0	1	1	1	0	1
1	0	1	1	1	0	1
1	0	1	1	1	0	1
1	0	0	0	0	0	1
1	1	1	1	1	1	1

A^c

0	1	0
1	1	1
0	1	0

B

Region Filling

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

X_0

0	1	0
1	1	1
0	1	0

B

0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	1	1	0	0	0
0	0	1	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

$(X_0 \oplus B)$

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

Region Filling

0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	1	1	0	0	0
0	0	1	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

$X_1 = (X_0 \oplus B)$

1	1	1	1	1	1	1
1	0	0	0	0	0	1
1	0	1	1	1	0	1
1	0	1	1	1	0	1
1	0	1	1	1	0	1
1	0	0	0	0	0	1
1	1	1	1	1	1	1

A^c

0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	1	1	0	0	0
0	0	1	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

$X_1 = (X_0 \oplus B) \bigcap A^c$

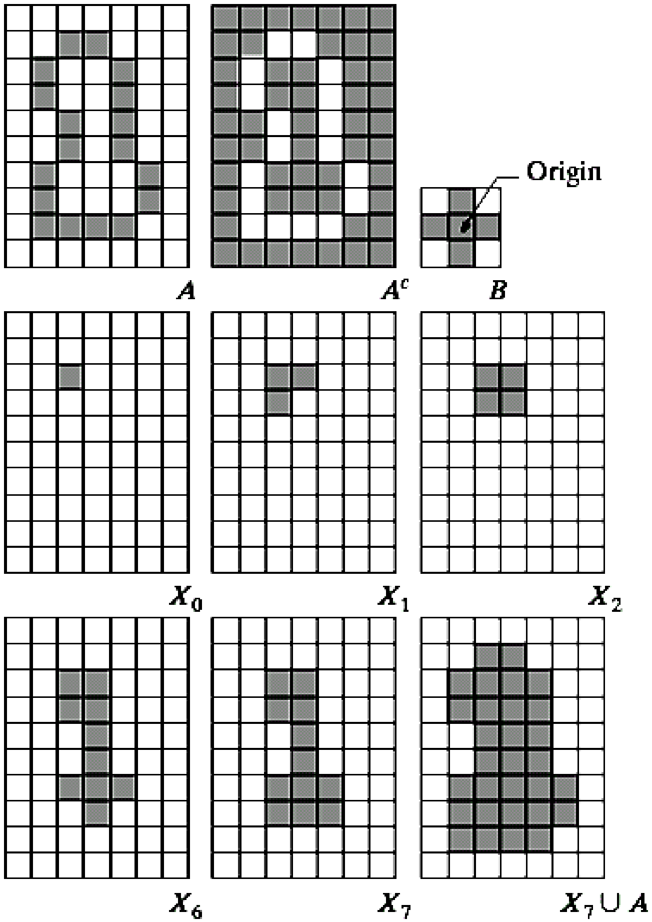
$$X_{k+1} = (X_k \oplus B) \bigcap A^c$$

$$k = 1, 2, 3, \cdots$$

Region Filling

$$X_{k+1} = (X_k \oplus B) \bigcap A^c$$
$$k = 1, 2, 3, \cdots$$

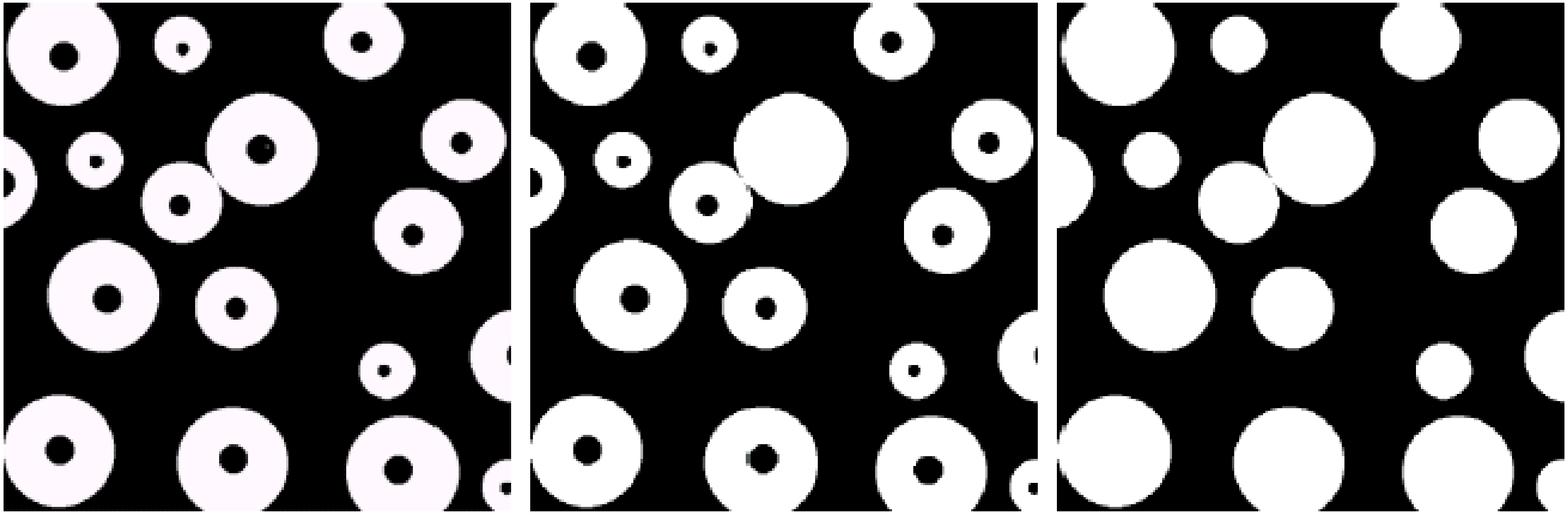
NOTE:
The intersection of dilation and the complement of A limits the result to inside the region of interest



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: Example



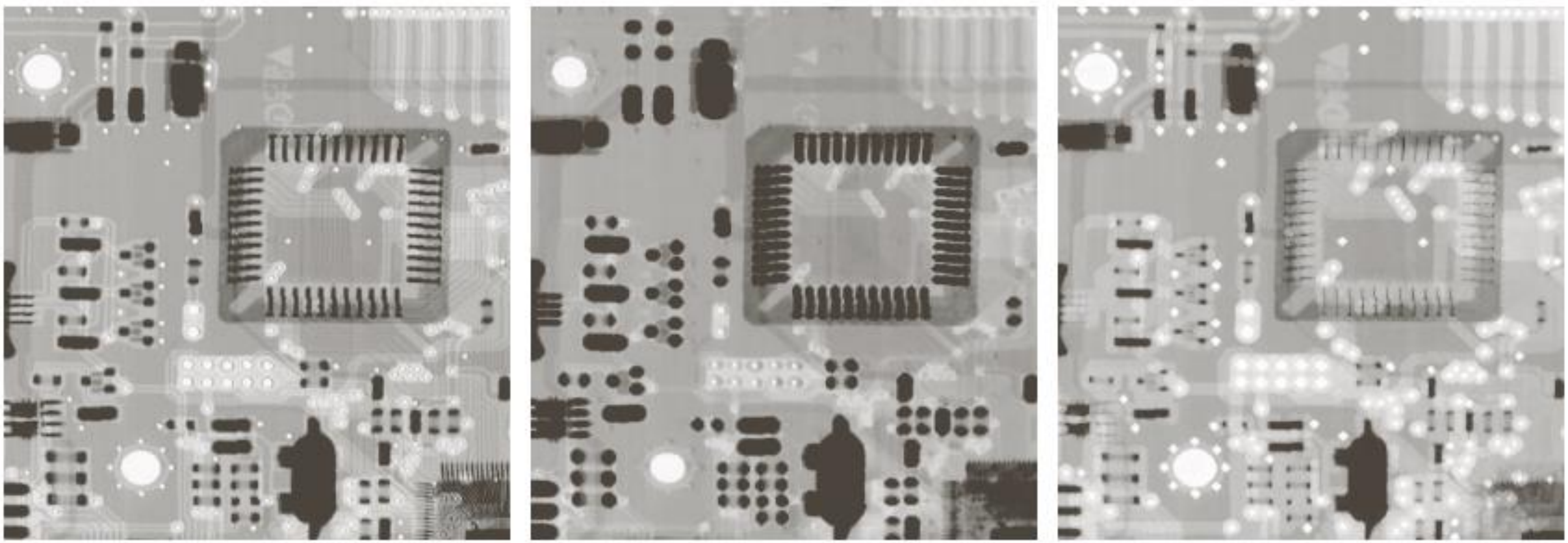
Original Image | One Region Filled | All Regions Filled

Gray Level Image Morphological Operations

Dilation & Erosion

$$(f \oplus b)(s,t) = max\{f(s-x, t-y)\}$$

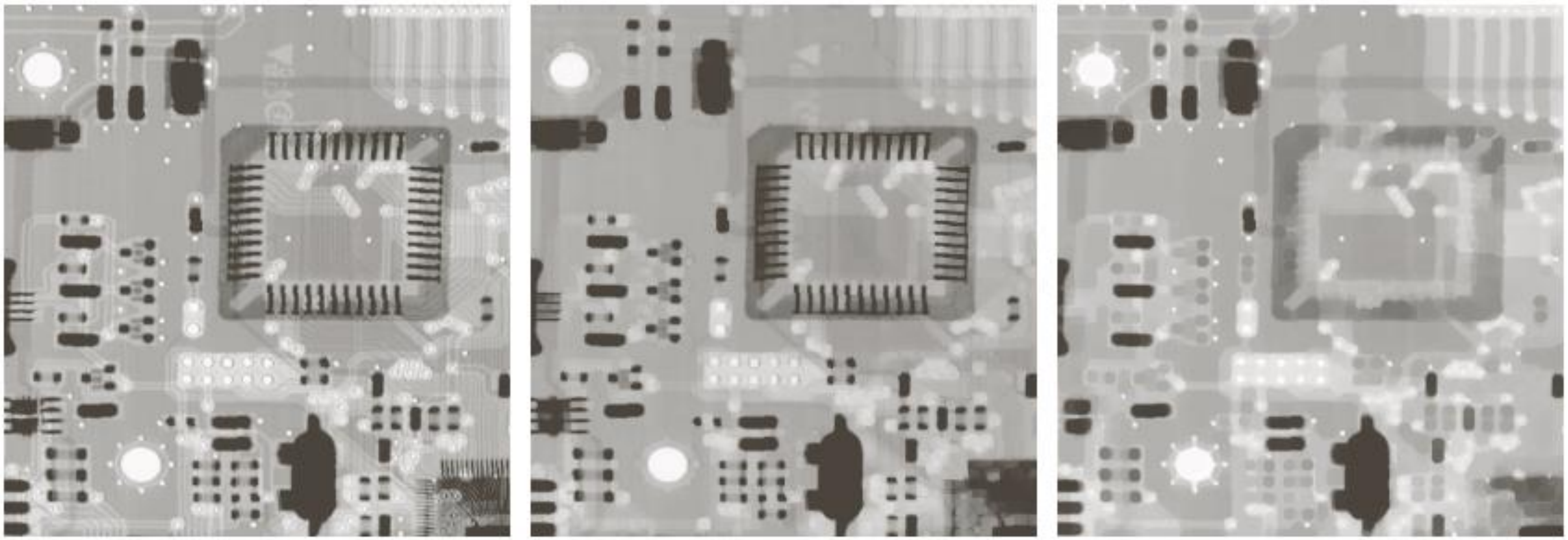
$$(f \ominus b)(s,t) = min\{f(s-x, t-y)\}$$



a b c

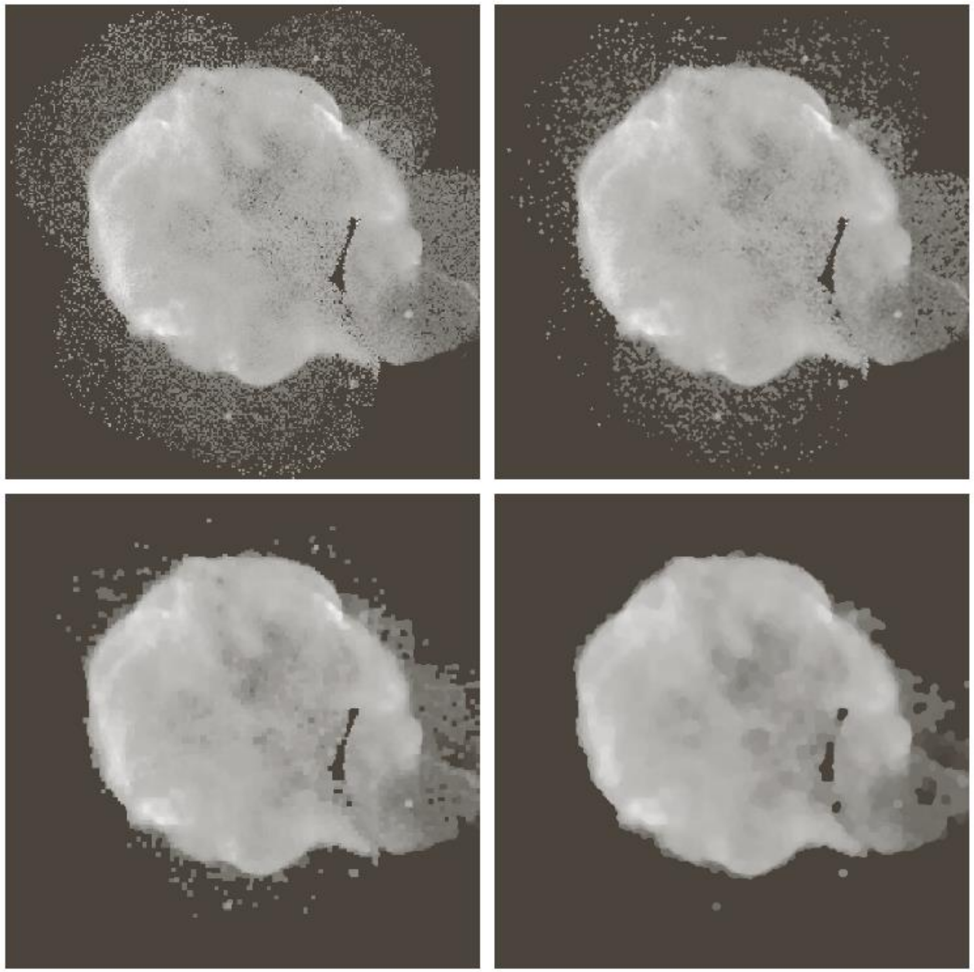
FIGURE 9.35 (a) A gray-scale X-ray image of size 448 × 425 pixels. (b) Erosion using a flat disk SE with a radius of two pixels. (c) Dilation using the same SE. (Original image courtesy of Lixi, Inc.)

Opening & Closing



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.

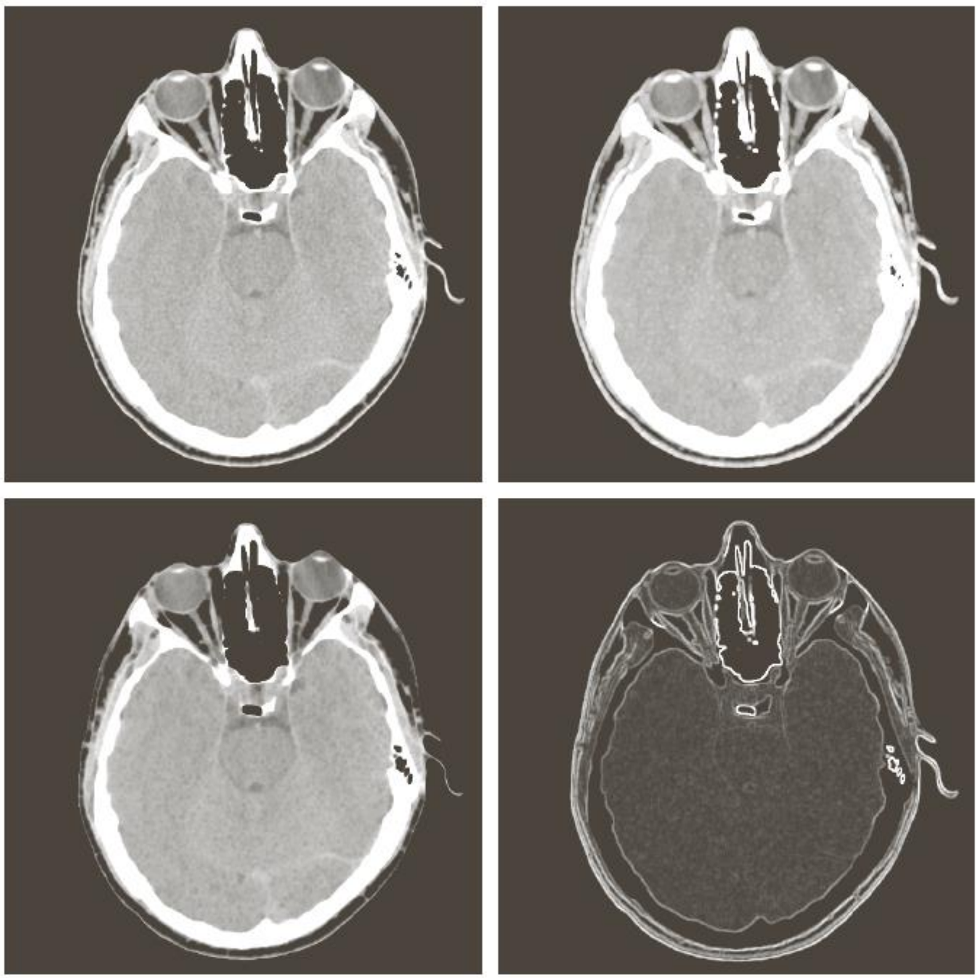


a b
c d

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.)

Morphological Gradient

$$g = (f \oplus b) - (f \ominus b)$$

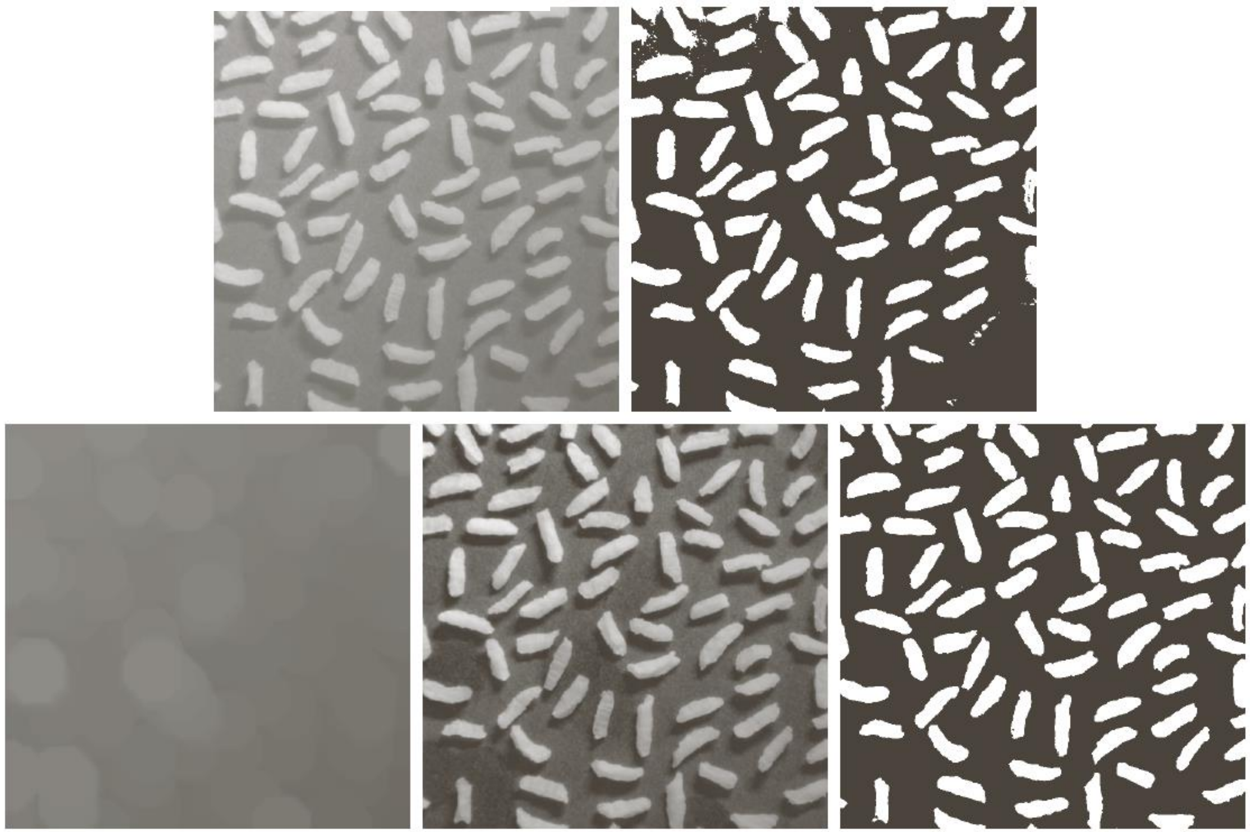


a b
c d

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 & Bottom Hat Transformations

$$g_{top} = f - (f \circ b) \qquad g_{bot} = f - (f \bullet b)$$



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.

Readings from Book (3rd Edn.)

- Morphological Operations (Chapter – 9)
- Reading Assignment
  - Connected Component

Acknowledgements

Material in these slides has been taken from, the following resources

- Digital Image Processing", Rafael C. Gonzalez & Richard E. Woods, Addison-Wesley, 2002
- Computer Vision for Computer Graphics, Mark Borg