

# Digital Image Processing

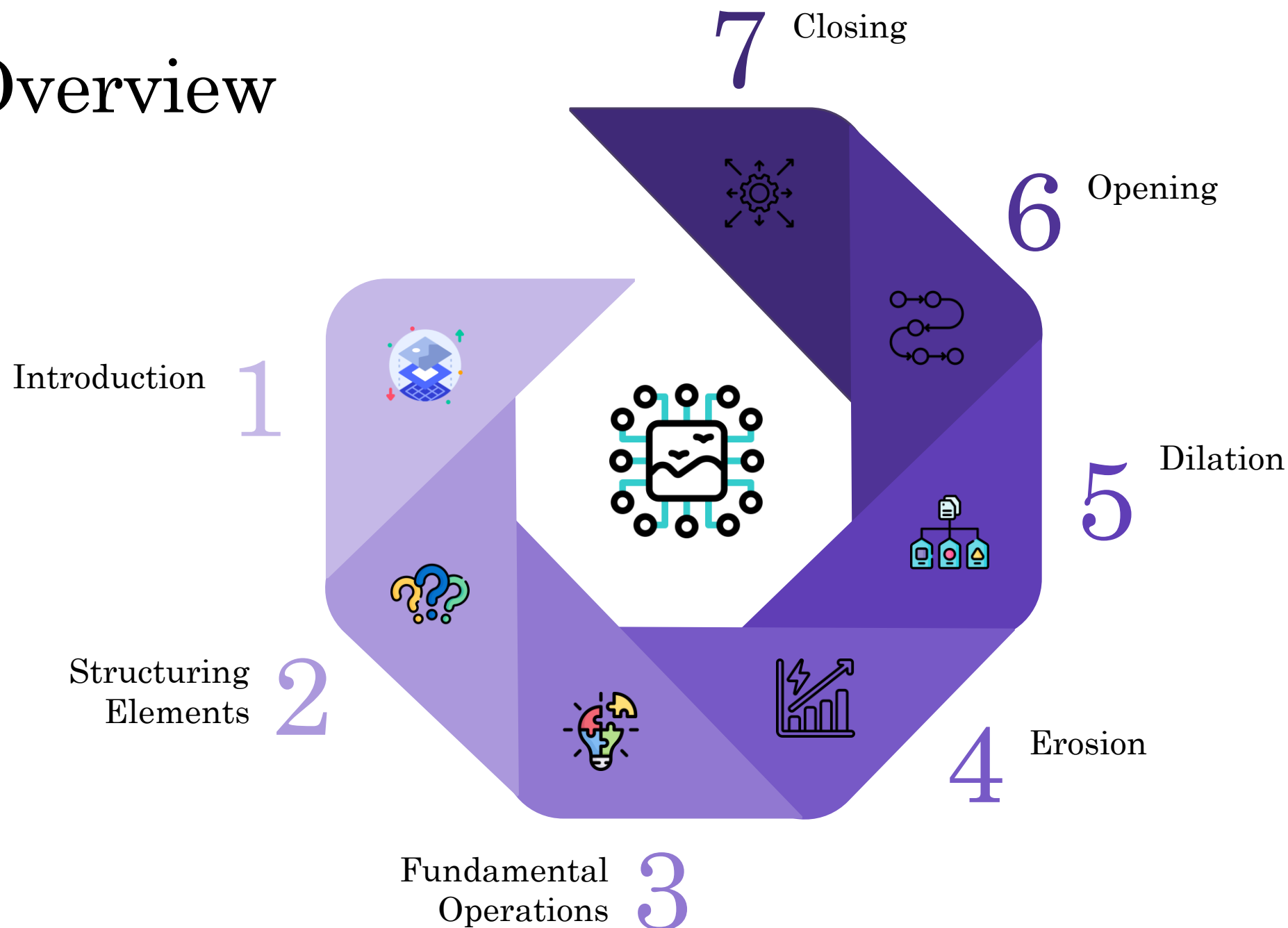


Lecture 05

Morphological Image Processing

Dr. Muhammad Sajjad  
R.A: Asad Ullah

# Overview



# Introduction

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

Morphological image processing is used to extract image components for representation and description of region shape, such as boundaries, skeletons, and the convex hull

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



Image after segmentation



Image after segmentation and  
morphological processing

# Structuring Elements

Structuring elements are small binary images used as shape masks in basic morphological operations.

They can be any shape and size that is digitally representable.

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

Usually, element values are 0 or 1, but other values are possible (including none!)

Empty spots in the Structuring Elements are *don't cares*!

Box →

1	1	1
1	①	1
1	1	1

Disc ↘

	1	
1	①	1
	1	

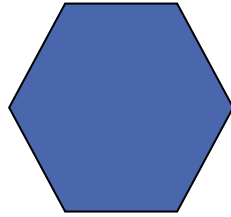
		1	1	1		
	1	1	1	1	1	
1	1	1	1	1	1	1
1	1	1	①	1	1	1
1	1	1	1	1	1	1
	1	1	1	1	1	
		1	1	1		

1	1	
1	①	
1		0

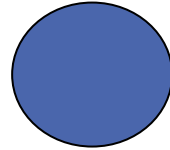
# Structuring Elements Examples



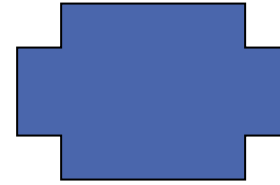
box



hexagon



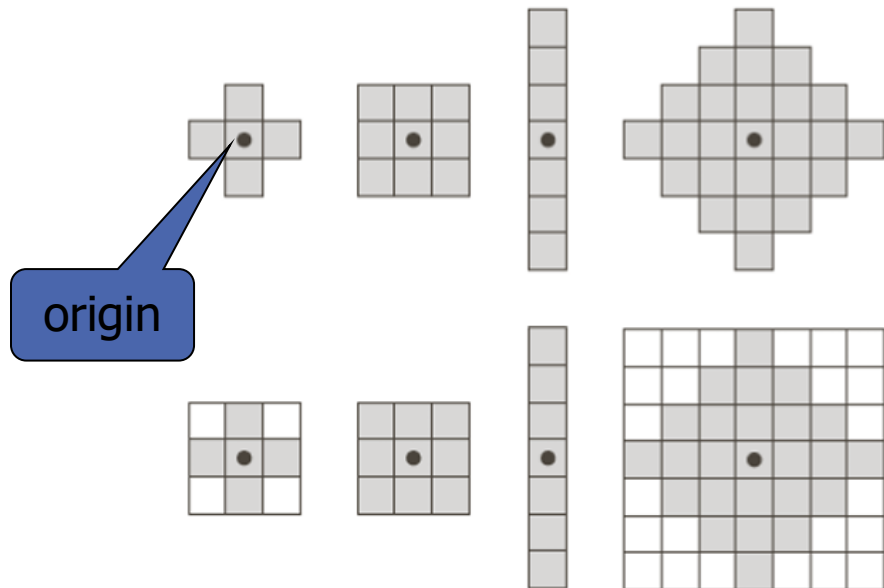
disk



something

box(length,width)

disk(diameter)



1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

a) BOX(3,5)

	1	1	1	
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
	1	1	1	

b) DISK(5)

		1	1	1	
1					1
1					1
1					1
	1	1	1	1	

c) RING(5)

1	1		
1	1		
1	1	1	1
1	1	1	1

d)

1	1	1	1	1	1
1		1	1		1
1		1	1		1
1		1	1		1

e)

1
1
1
1

f)

Figure 3.12: Examples of structuring elements (blanks represent 0's).

# Fundamental Operations

Fundamentally morphological image processing is very like spatial filtering

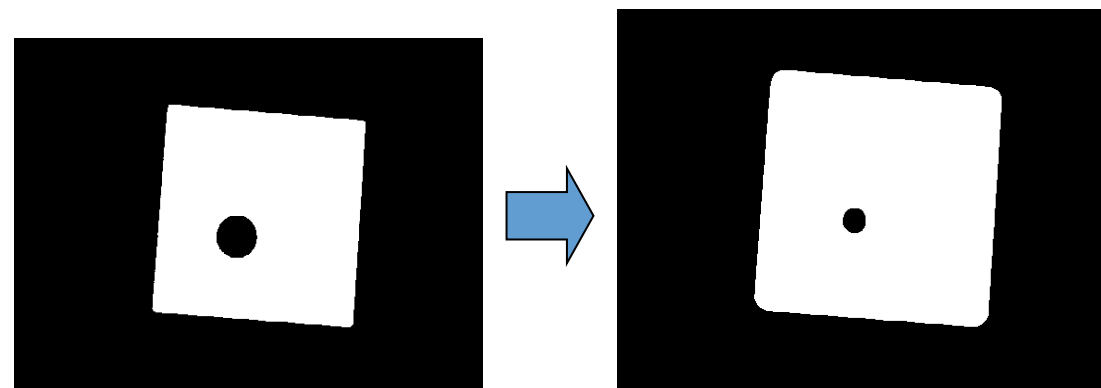
The structuring element is moved across every pixel in the original image to give a pixel in a new processed image

The value of this new pixel depends on the operation performed

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

		1	1	1		
	1	1	1	1	1	
1	1	1	1	1	1	1
1	1	1	①	1	1	1
1	1	1	1	1	1	1
	1	1	1	1	1	
		1	1	1		

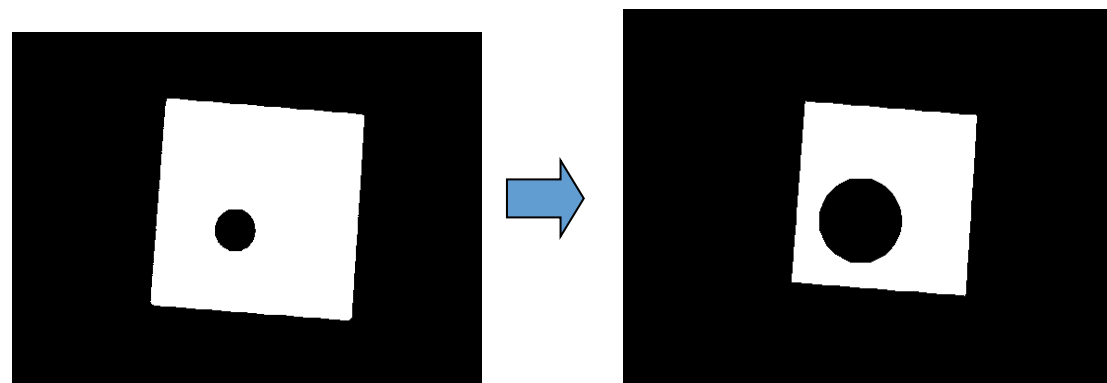
SE



Input

Dilation

Output



Input

Erosion

Output

## **Erosion** (based on “Fit”)

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:

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

# 1D Erosion (based on “Fit”)

Input image

1	0	0	0	1	1	1	0	1	1
---	---	---	---	---	---	---	---	---	---



Structuring Element

1	1	1
---	---	---



Output Image

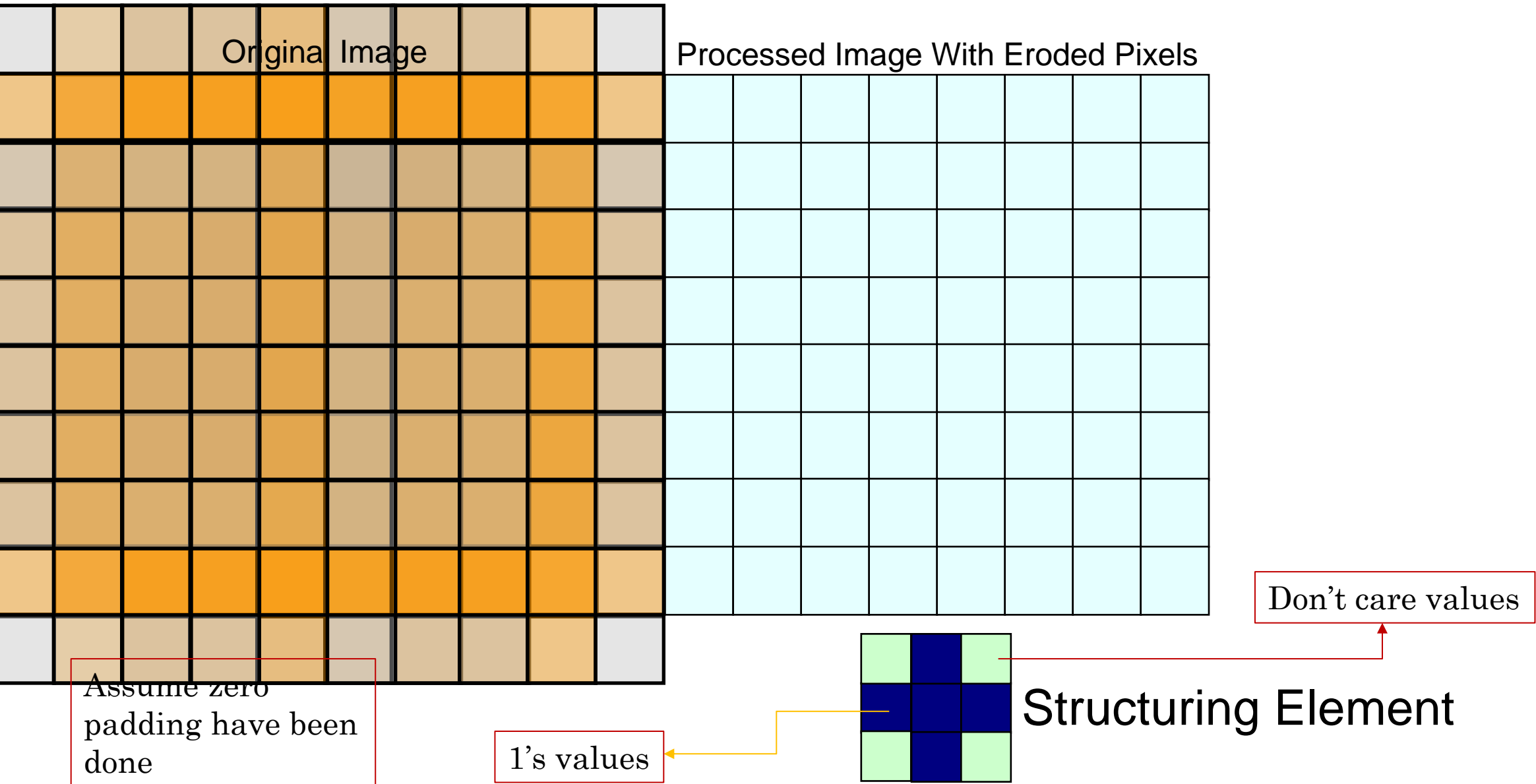
	0	0	0	0	1	0	0	0	
--	---	---	---	---	---	---	---	---	--

$$g(x) = f(x) \ominus s$$

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

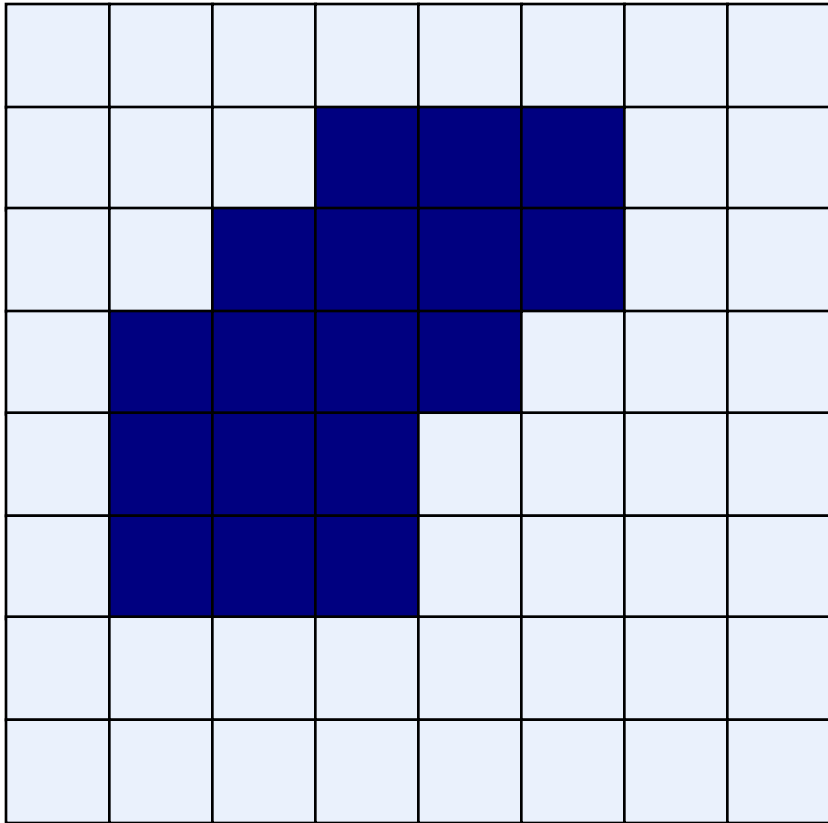


# 2D Erosion (based on Fit)

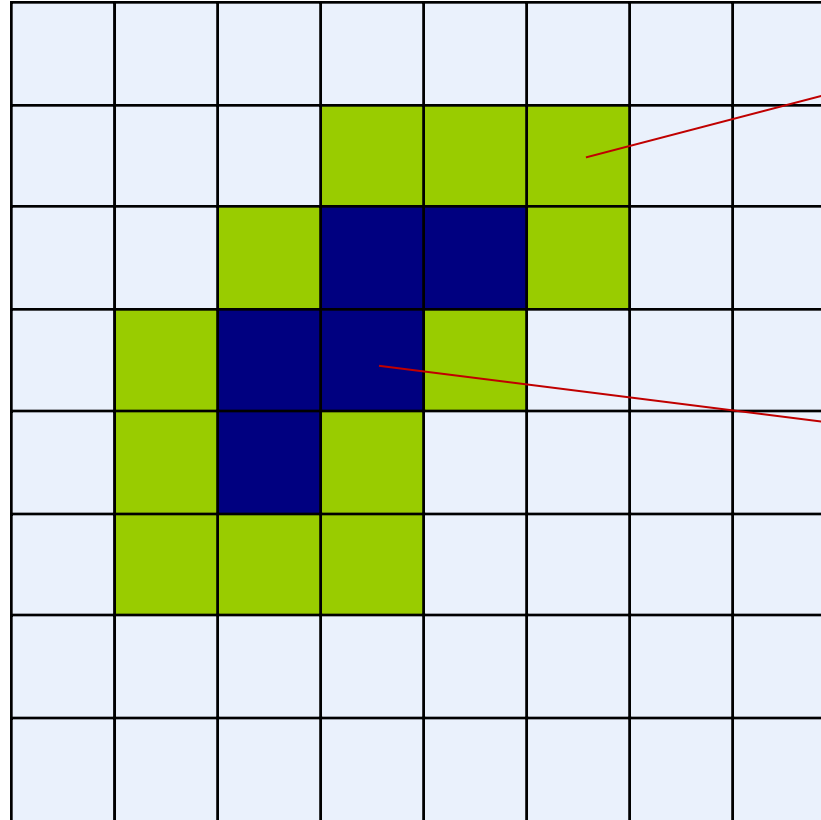


# 2D Erosion (based on Fit)

Original Image



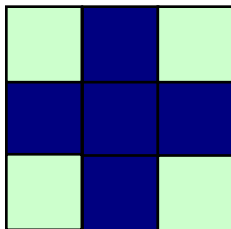
Processed Image



Pixels have been discarded in output which didn't fit

Pixels have been remained in output which fit

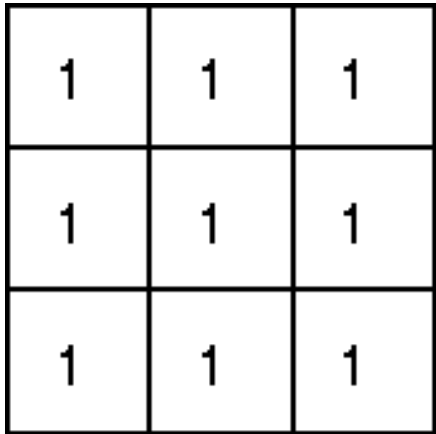
Overall Object get smaller



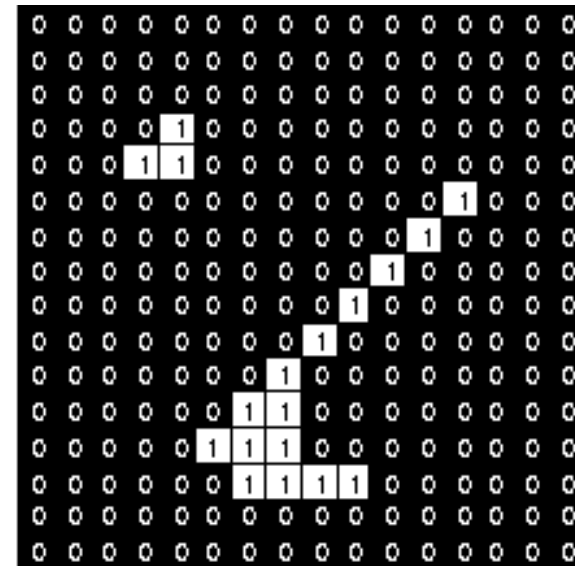
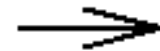
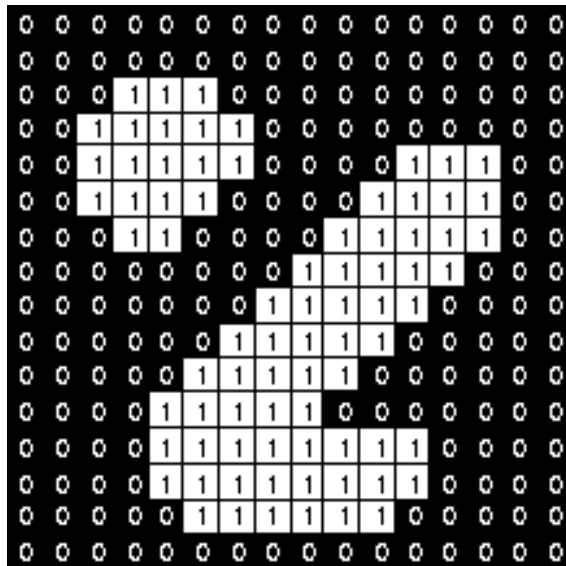
Structuring Element

## 2D Erosion (based on Fit)

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



Structuring element

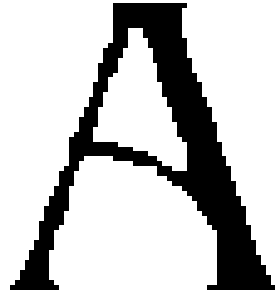


Objects get smaller

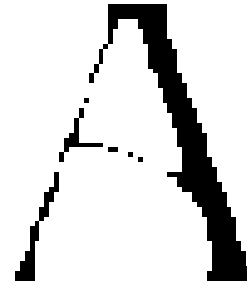
# Erosion Example 1



Original image



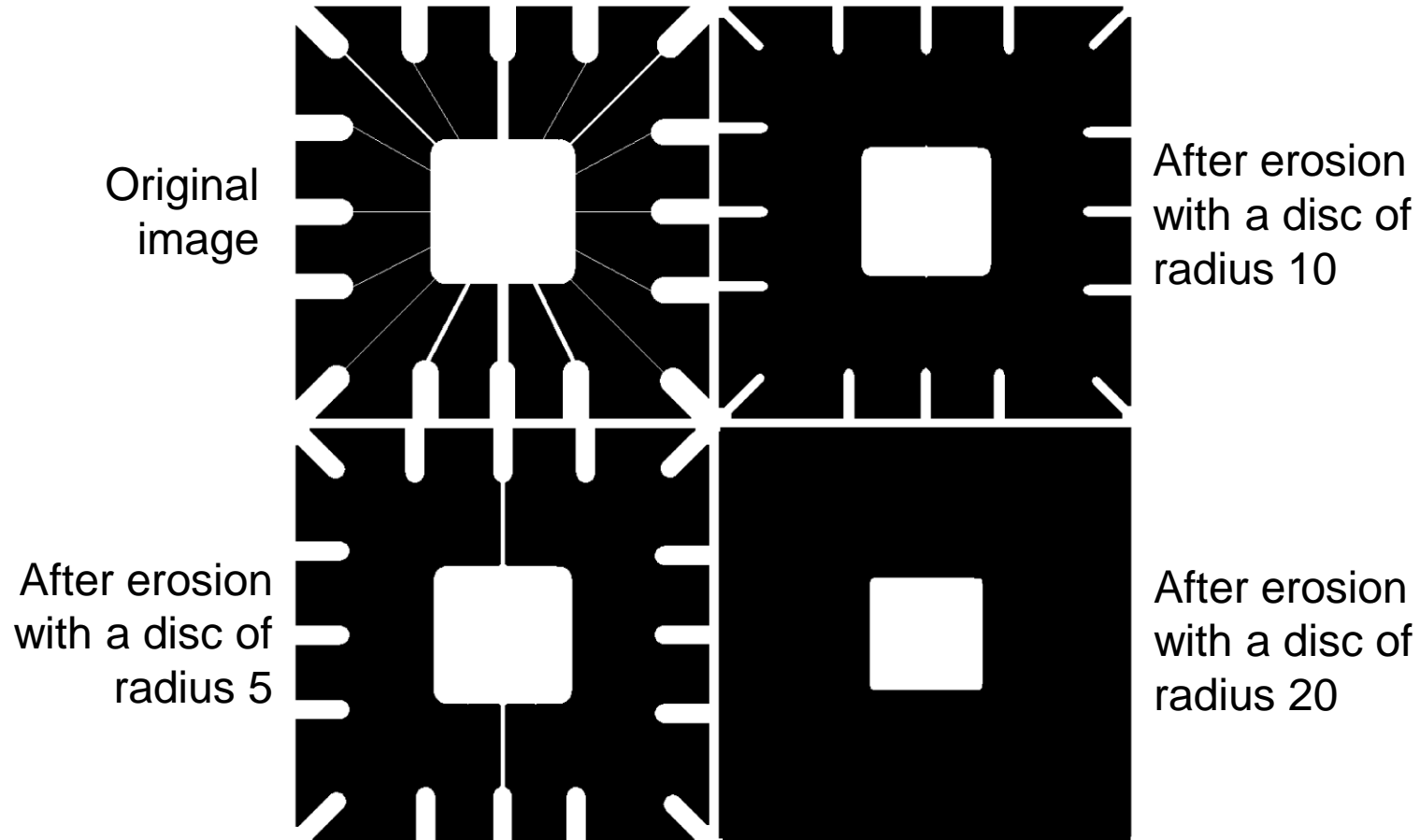
Erosion by 3\*3  
square structuring  
element



Erosion by 5\*5  
square structuring  
element

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

## Erosion Example 2



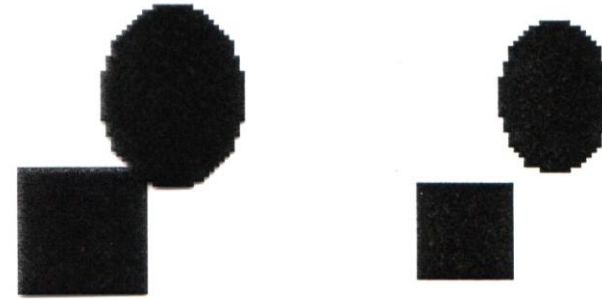
# What Is Erosion For?

Erosion **shrinks** the connected sets of 1s of a binary image.

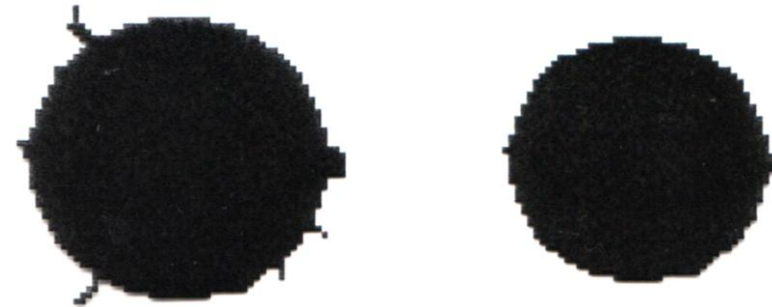
It can be used for

1. shrinking features
2. Removing bridges, branches and small protrusions

**Watch out:** Erosion shrinks objects



Erosion can split apart



## Dilation (based on “Hit”)

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:

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

# 1D Dilation (based on “Hit”)

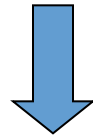
Input image

1	0	0	0	1	1	1	0	1	1
---	---	---	---	---	---	---	---	---	---



Structuring Element

1	1	1
---	---	---



Output Image

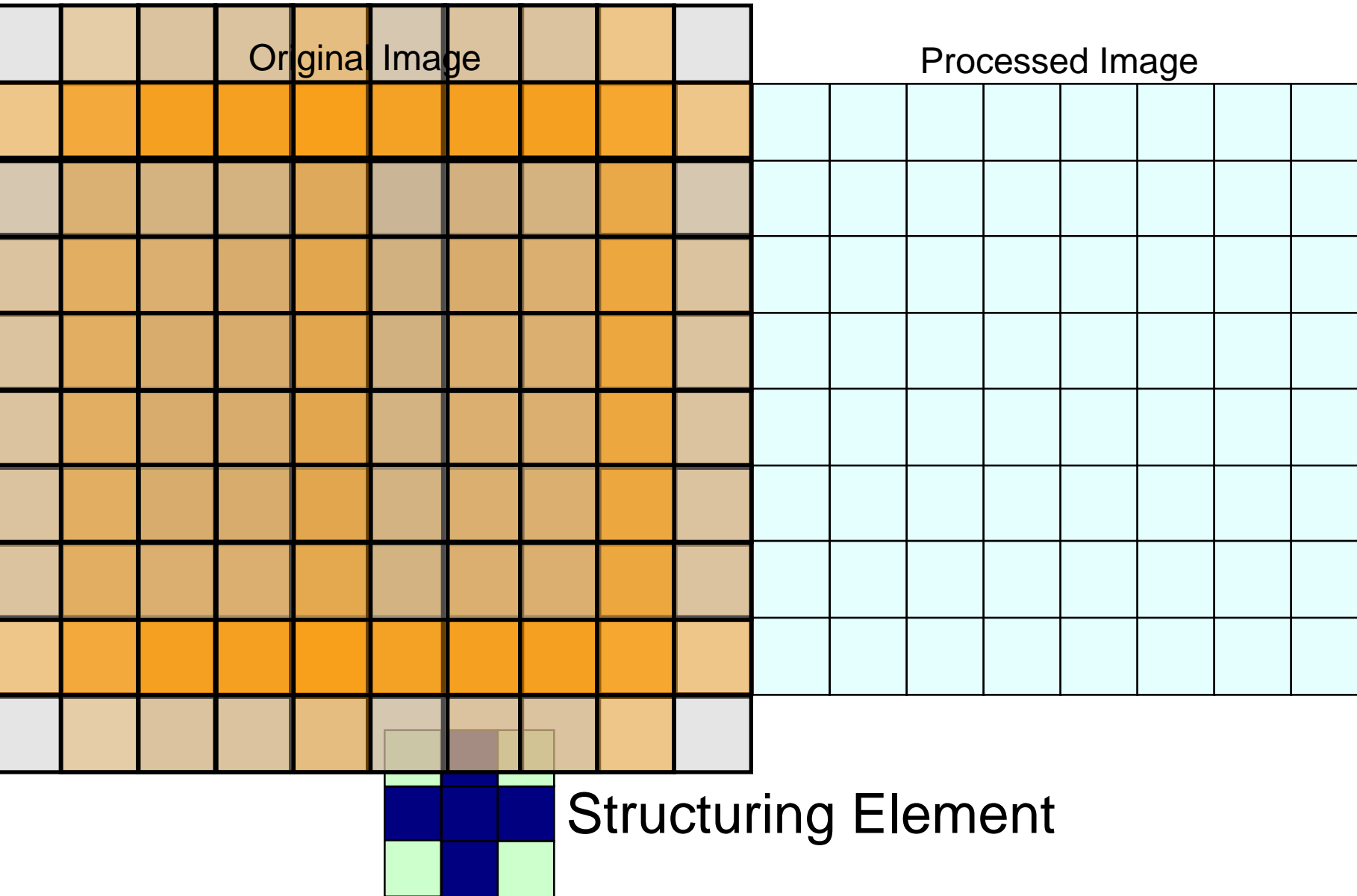
	1	0	1	1	1	1	1	1	
--	---	---	---	---	---	---	---	---	--

$$g(x) = f(x) \oplus S$$

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

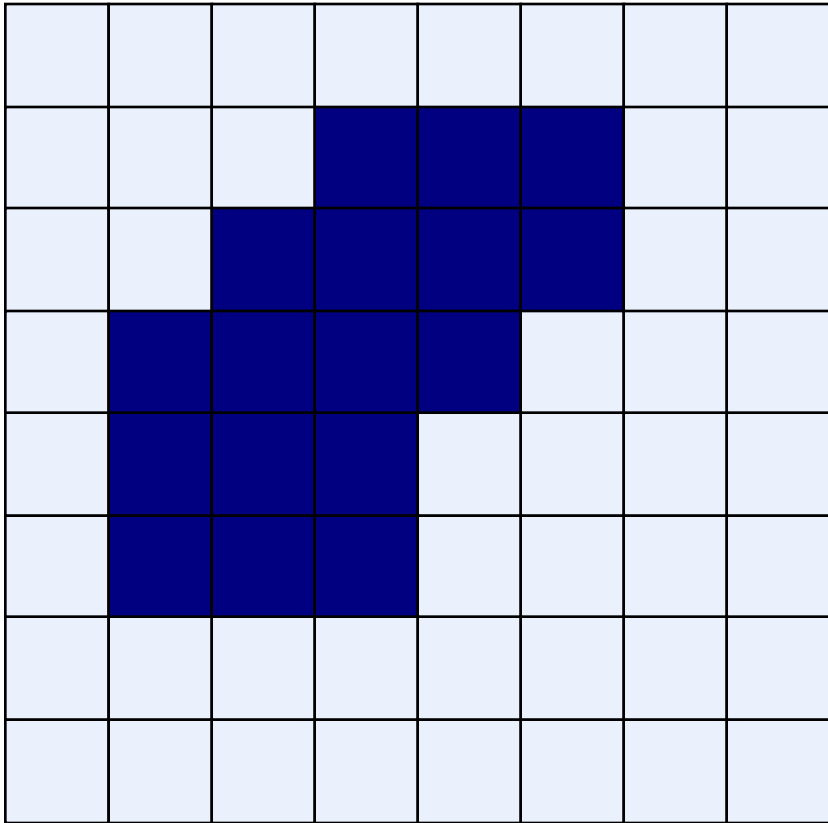


## 2D Dilation (based on “Hit”)

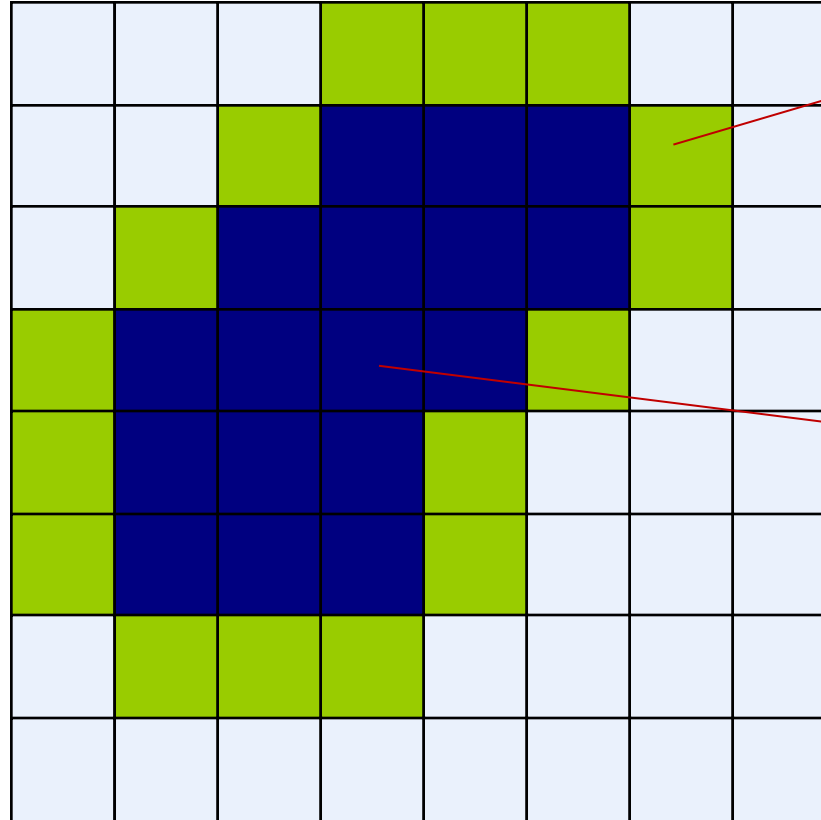


# 2D Dilation (based on “Hit”)

Original Image



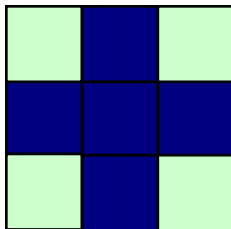
Processed Image With Dilated Pixels



Pixels have been added to the object in output which hit

Pixels have been remained in output which hit

Overall Object get bigger



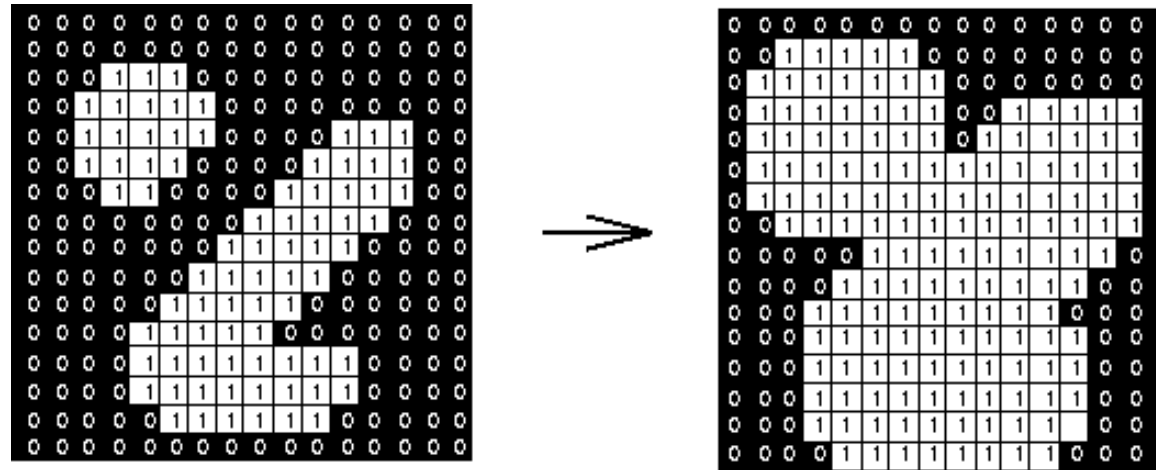
Structuring Element

## 2D Dilation (based on “Hit”)

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

1	1	1
1	1	1
1	1	1

Structuring element



- Objects are merged (holes are filled)
- Sharp corners are preserved

# Dilation Example 1



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!

# Dilation Example 2

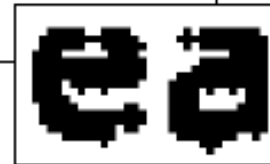
Original image

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.



0	1	0
1	1	1
0	1	0

Structuring element

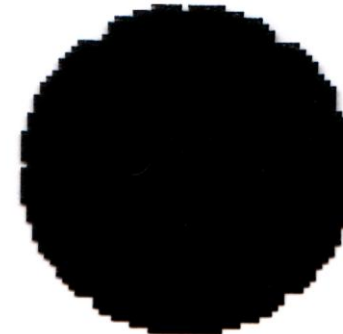
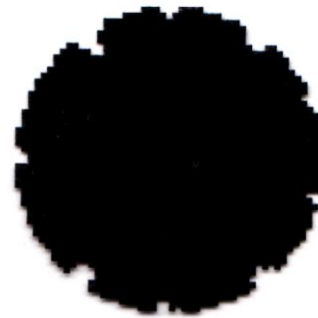
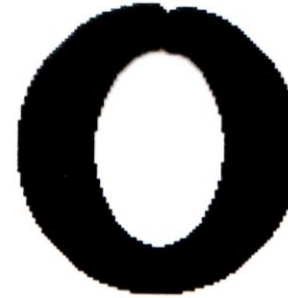
# What Is Dilation For?

Dilation **expands** the connected sets of 1s of a binary image.

It can be used for

1. growing features
2. filling holes and gaps

**Watch out:** Dilation enlarges objects



# Compound operations

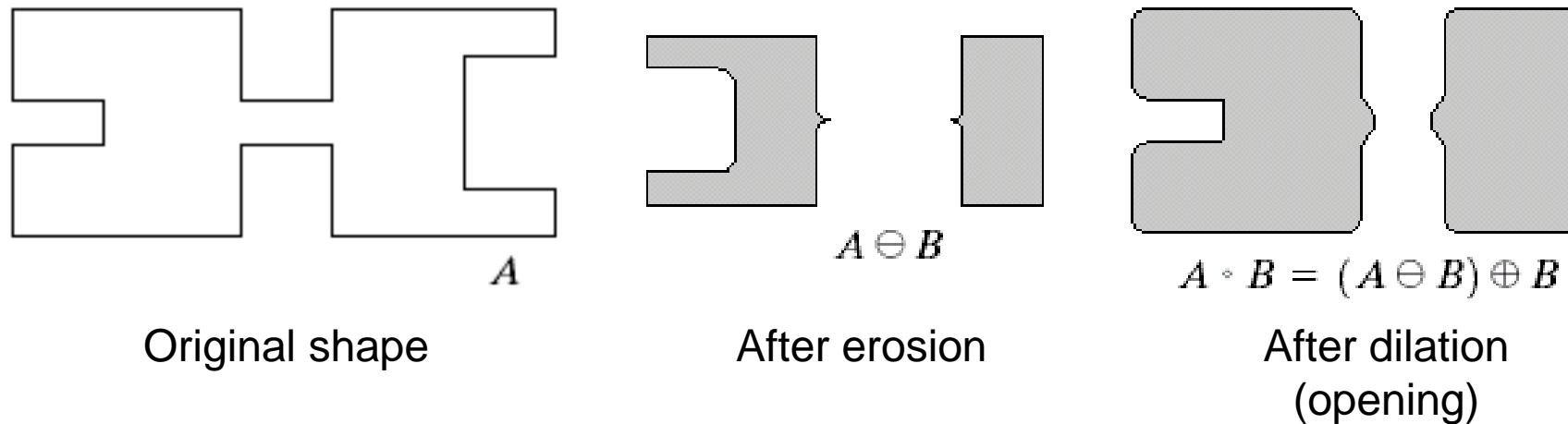
Combining Erosion and Dilation into higher level (more advanced) operations

- Finding the outline
- Opening: Isolate objects and remove small objects (better than Erosion)
- Closing: Fill holes (better than Dilation)
- Others...

# Opening

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

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



Note a disc shaped structuring element is used



# Opening

Motivation: Remove small objects BUT keep original size (and shape)

Erosion + Dilation =  
Opening

- Use the same structuring element!
- Similar to erosion but less destructive

Opening is **idempotent**:  
Repeated operations has no further effects!

Original  
Image



Image  
After  
Opening

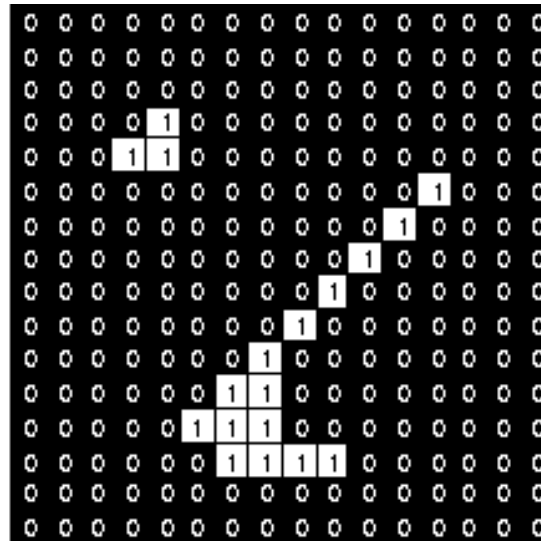
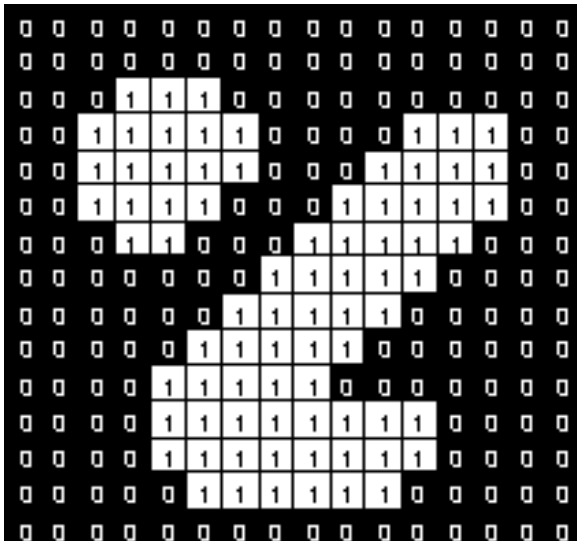


# Opening

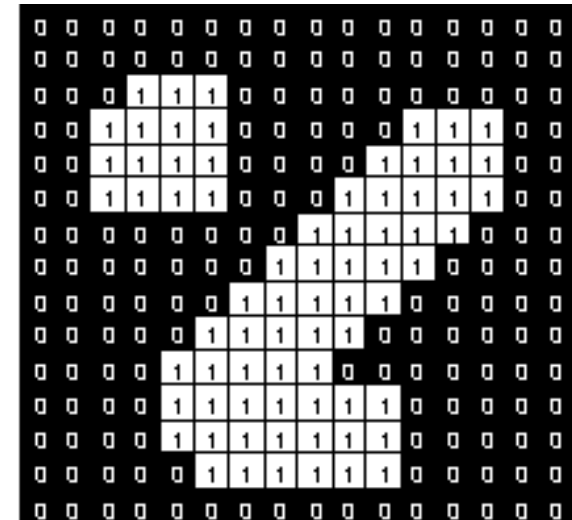
- Structuring element:

1	1	1
1	1	1
1	1	1

Erosion

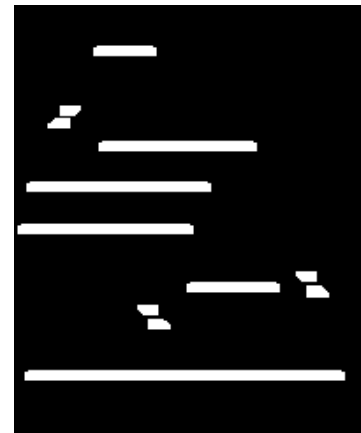
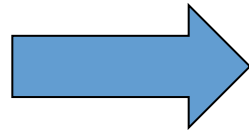
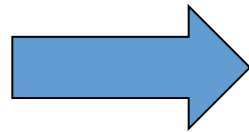
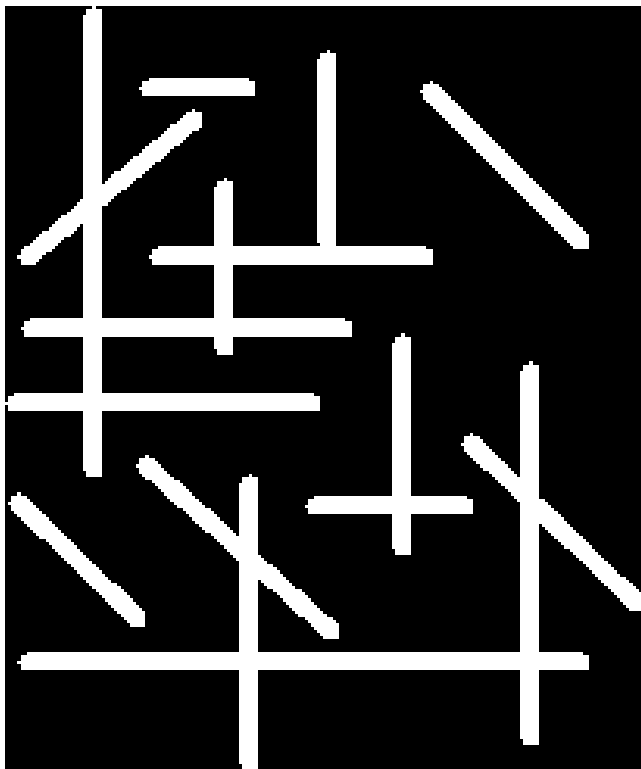


Dilation



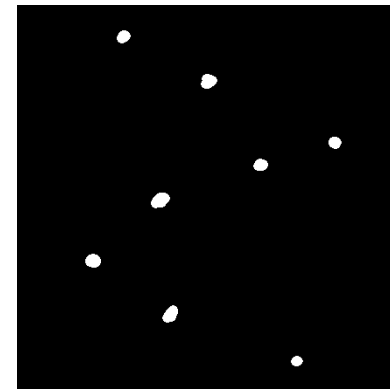
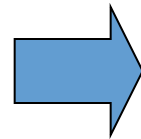
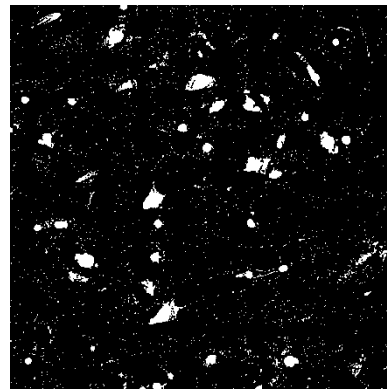
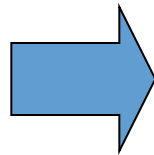
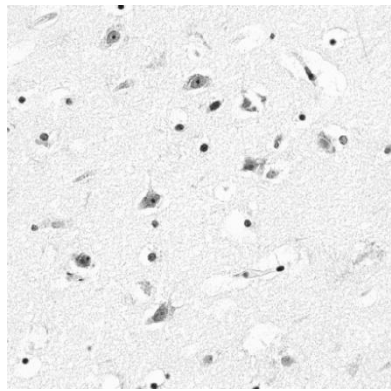
# Opening Example

- 9x3 and 3x9 Structuring Elements



# Opening Example

- Use large structuring element that fits into the big objects
- Structuring Element: 11-pixel disc



Original Image

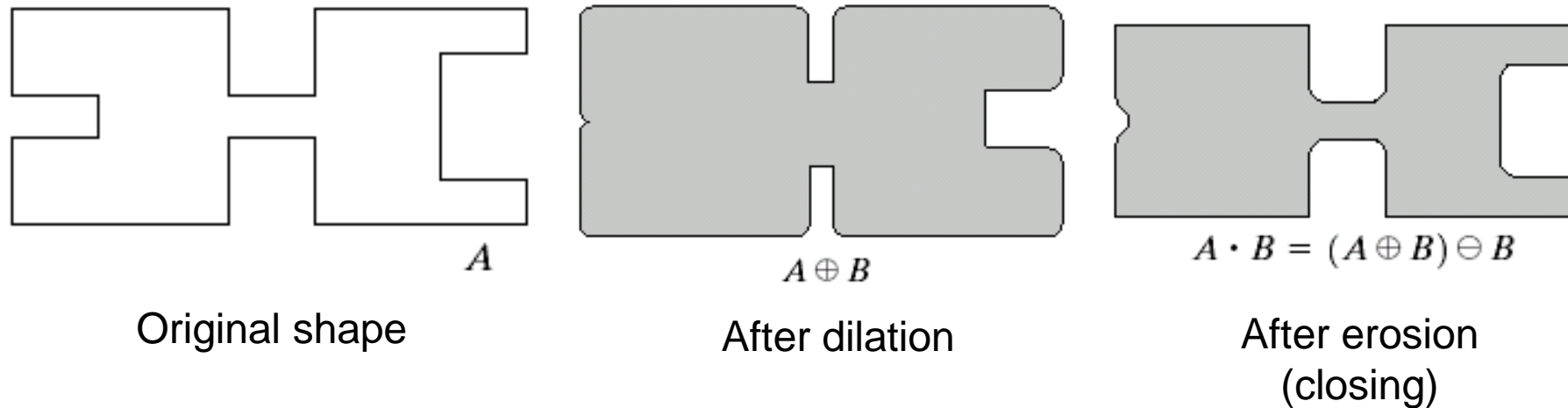
After Thresholding

After Opening Operation

# Closing

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

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



Note a disc shaped structuring element is used

# Closing

Motivation: Fill holes BUT  
keep original size (and shape)

Dilation + Erosion = Closing

- Use the same structuring element
- Similar to Dilation but less destructive

Closing is **idempotent**:  
Repeated operations has no  
further effects!

Original Image

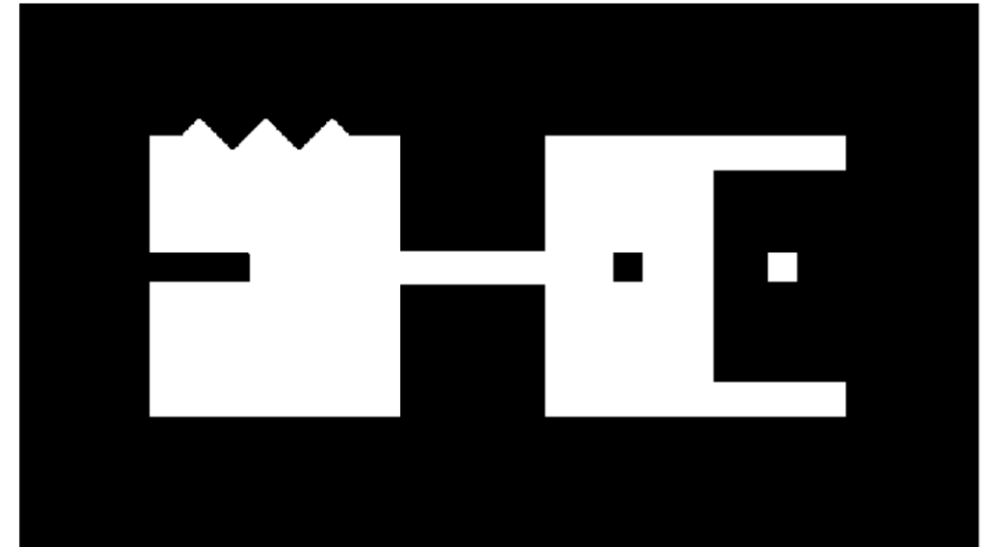
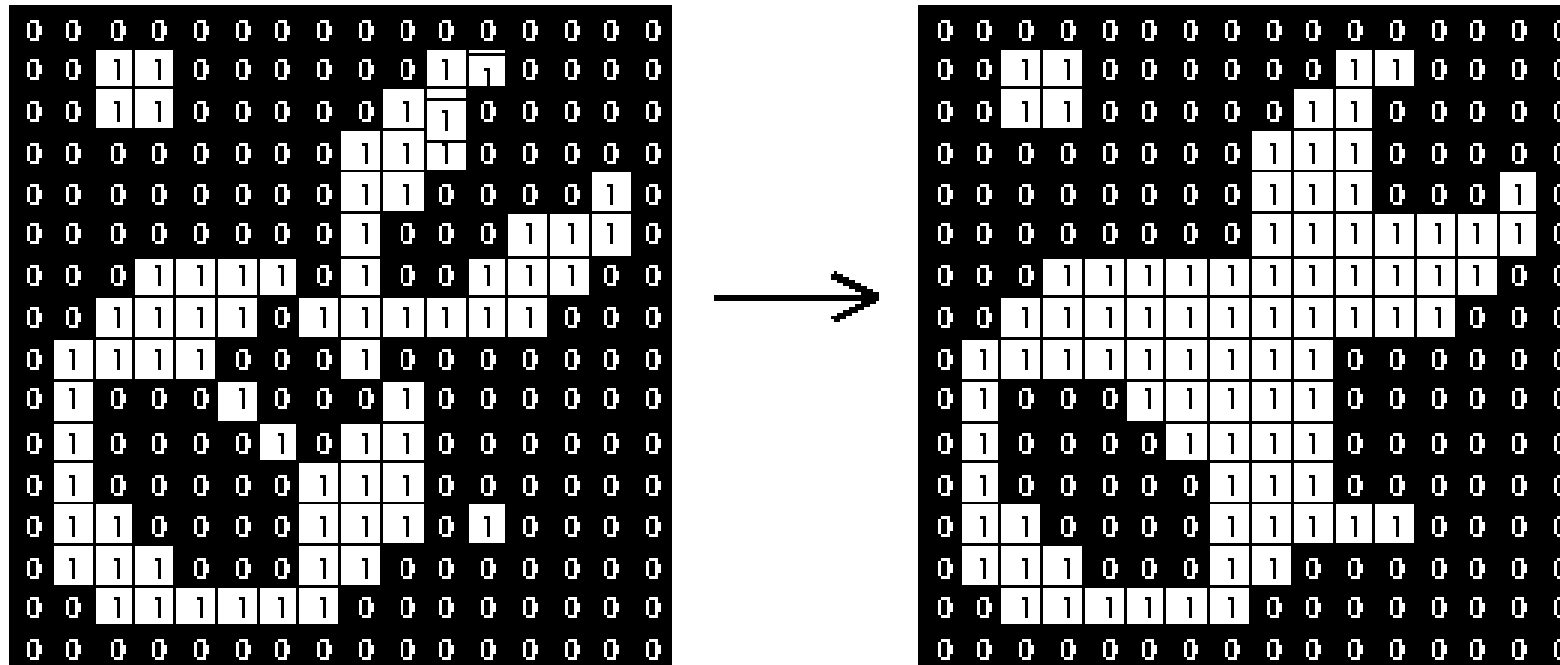


Image After Closing



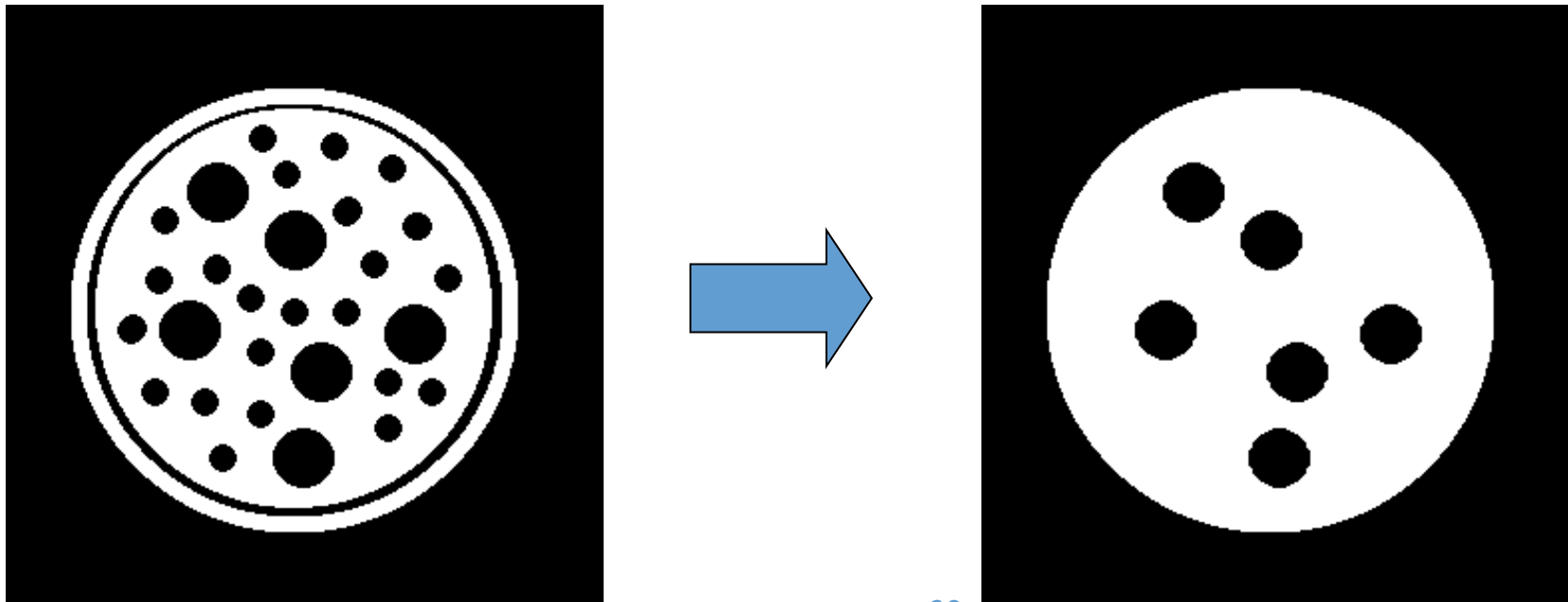
# Closing

- Structuring element: 3x3 square



# Closing Example

- Closing operation with a 22-pixel disc
- Closes small holes



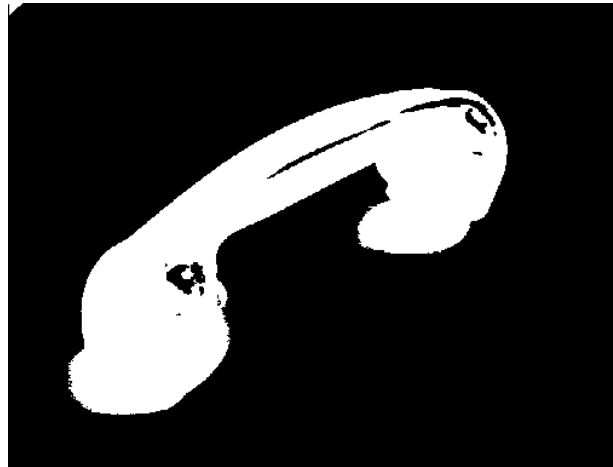


# Closing Example

- Threshold
- Improve segmentation
- Closing with disc of size 20



Original Image

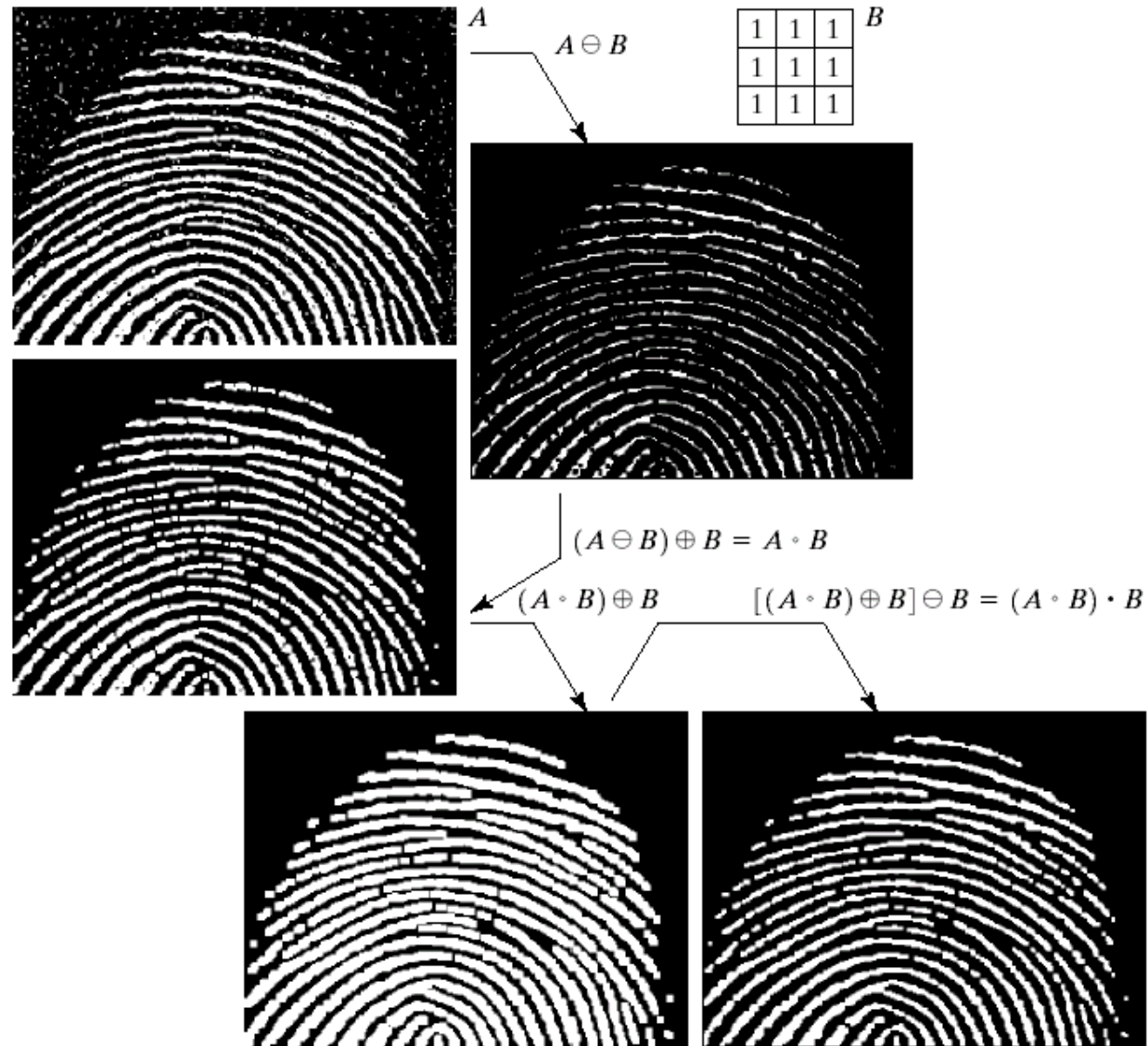


After Thresholding



After Closing Operation

# Morphological Processing Example



# Thank You

For Your Attention