22.09.2020

Digital Image Processing (CSE/ECE 478)
Lecture-12: Morphological Operations

Ravi Kiran

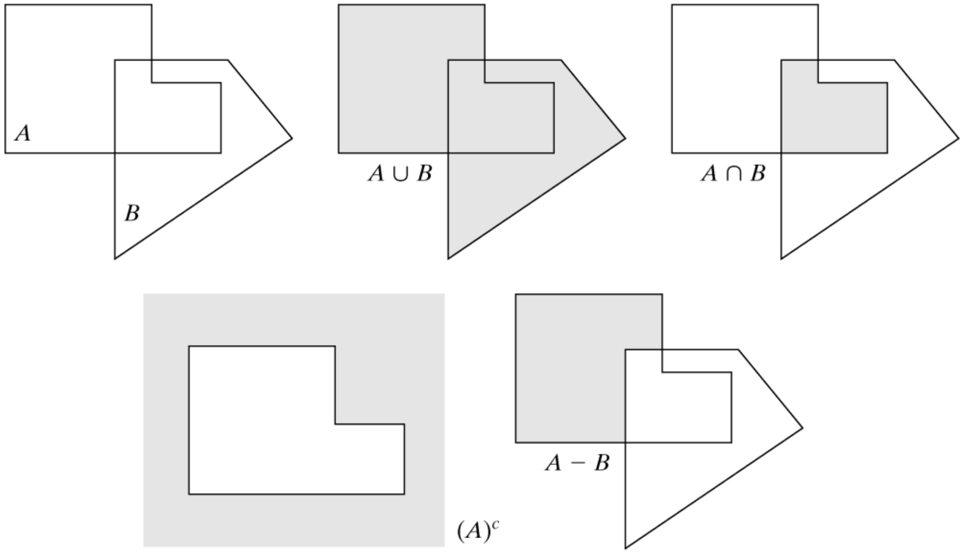
Introduction to Morphological Operators

Image – Set of Pixels

- Basic idea:
  - Object/Region = set of pixels (or coordinates of pixels)

- 0 = background
- 1 = foreground

Object = <u>set of pixels</u> (or coordinates of pixels)



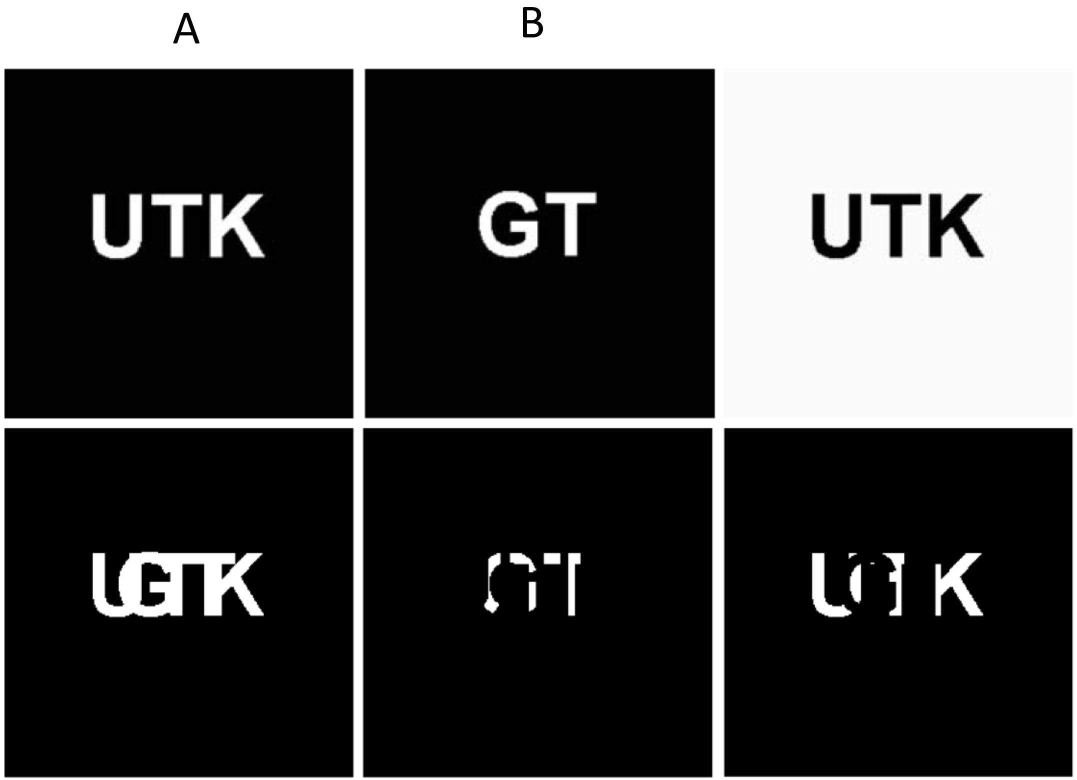
a b c
d e

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.

Basic operations on shapes

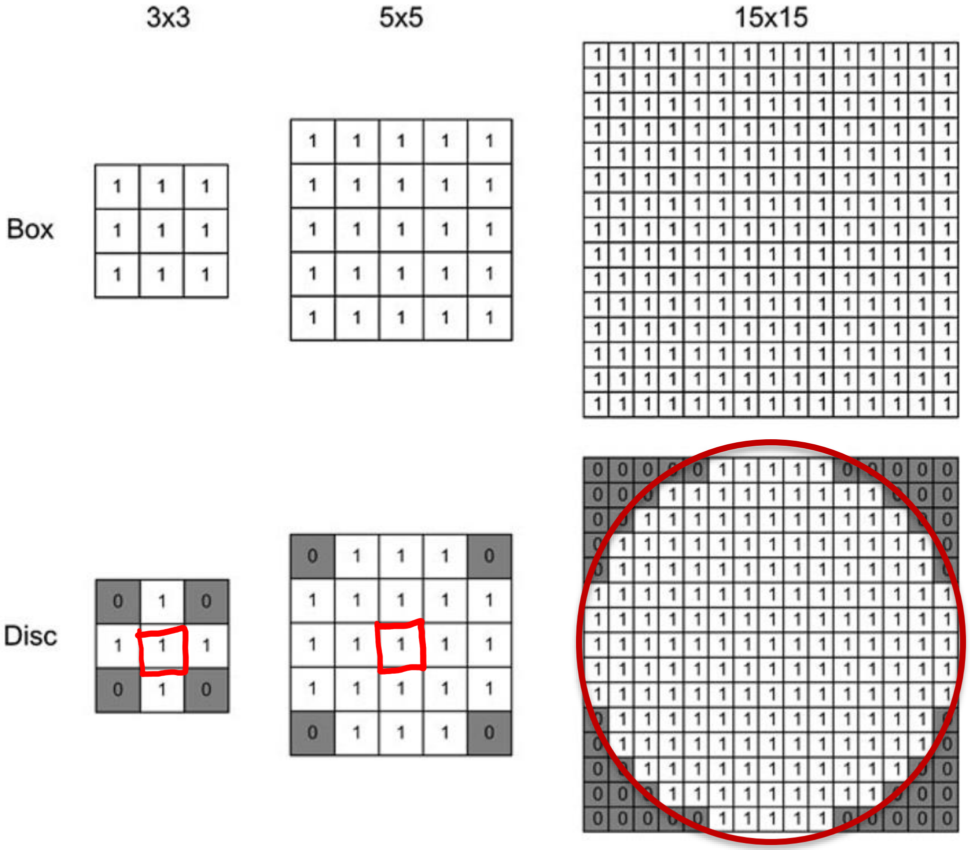
From: Digital Image Processing, Gonzalez, Woods And Eddins

Set Operations on Binary Images



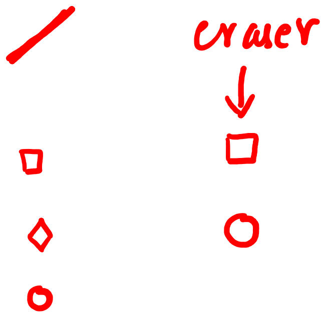
From: Digital Image Processing, Gonzalez,Woods And Eddins

Structuring Element



Structuring Element (Kernel)

- Can have varying sizes
- Have an origin
- Usually, element values are 0,1 and none(!)
  - For thinning, other values are possible
- Empty spots in the Structuring Elements are *don't care*'s!



Box

1	1	1
1	1	1
1	1	1

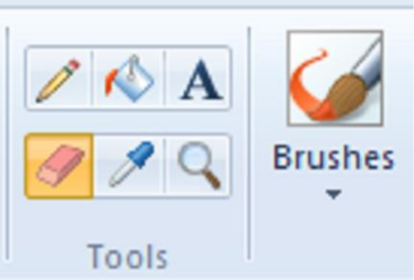
Disc

	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		

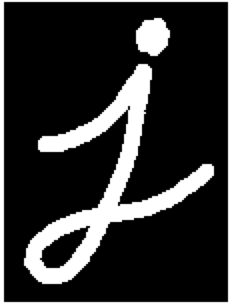
1	1	
1	0	
1		0

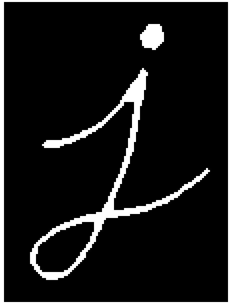
1	1	1
1	10	1
1	1	1



Erosion

Erosion





Erosion : Effect

SE

1	1	1
1	1	1
1	1	1

Set of coordinate points =

{ (-1, -1), (0, -1), (1, -1),

(-1, 0), (0, 0), (1, 0),

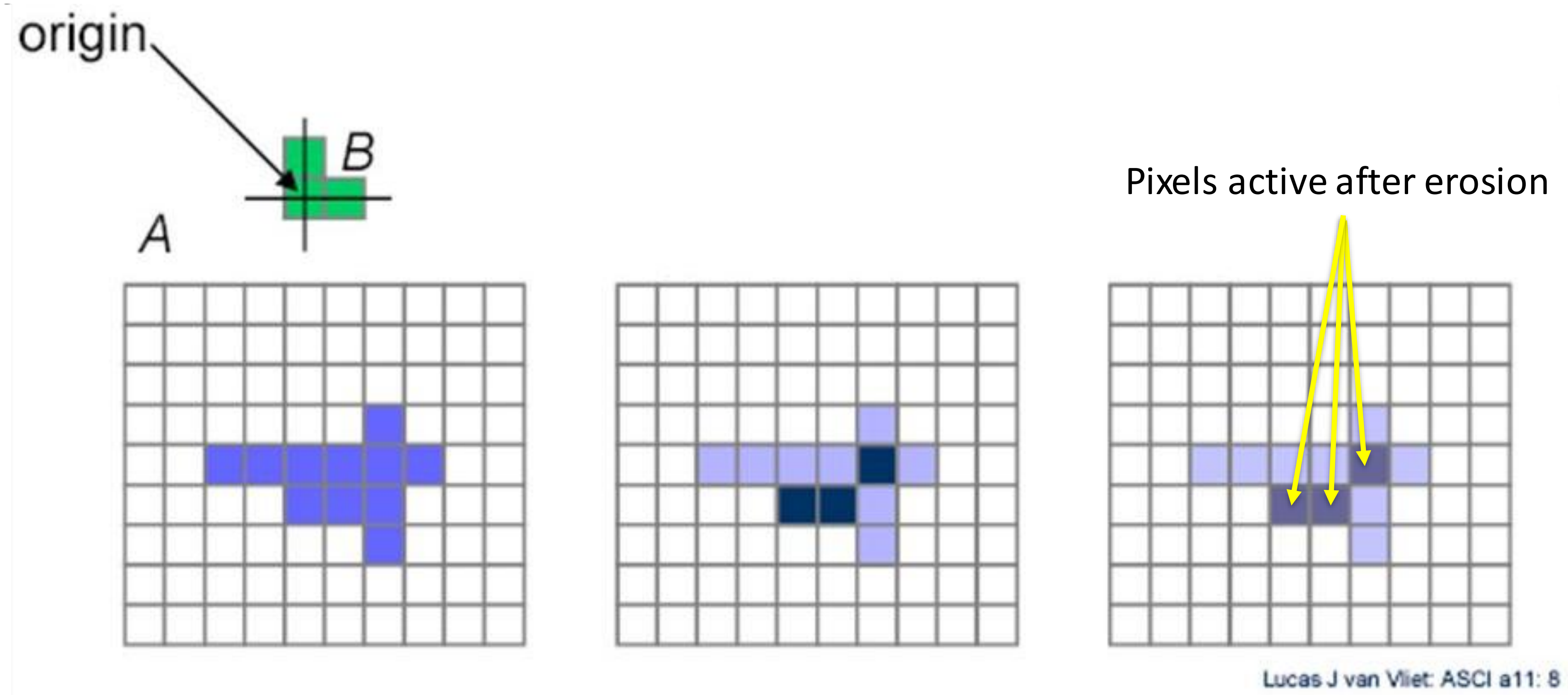
(-1, 1), (0, 1), (1, 1) }

If, for a particular location of Structuring Element (SE) origin, SE lies **fully within the region**, retain the location, else set to 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	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

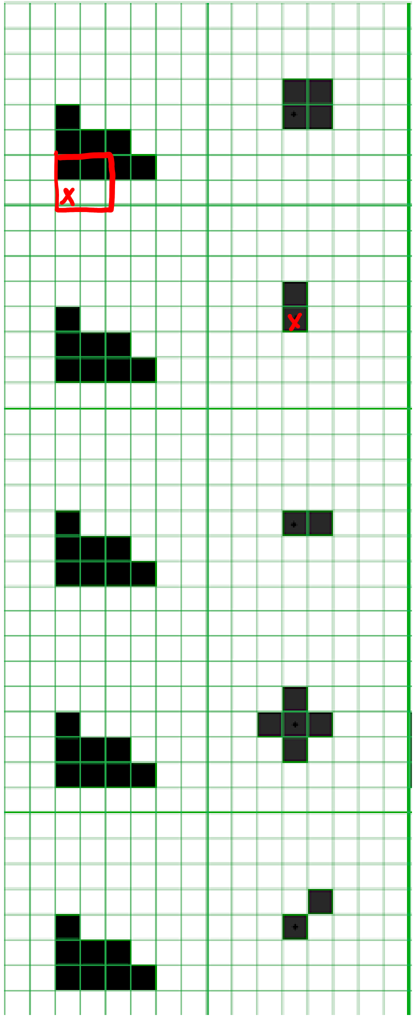
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0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

SEs operate wrt an origin

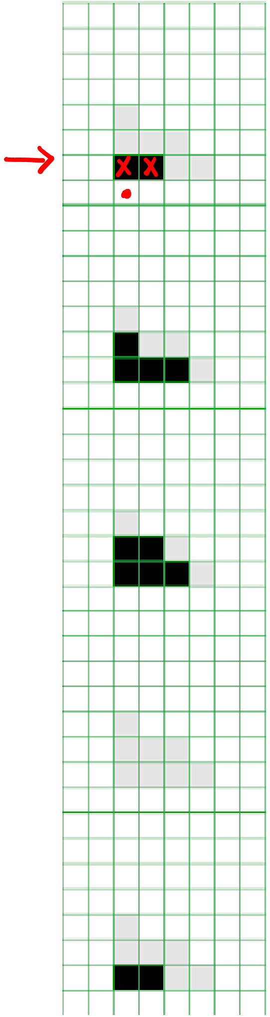


ORIGINAL IMAGE

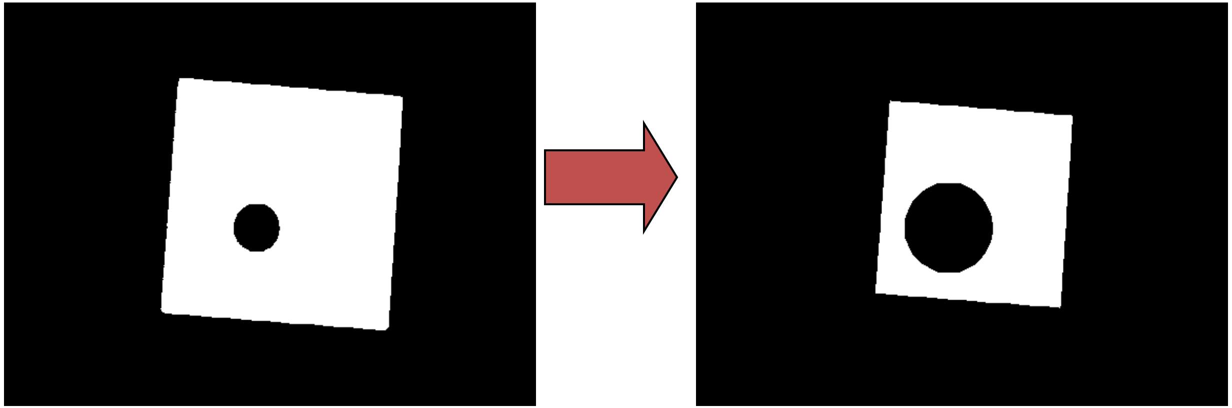
STRUCTURING ELEMENT



EROSION



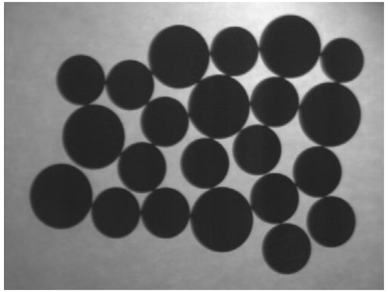
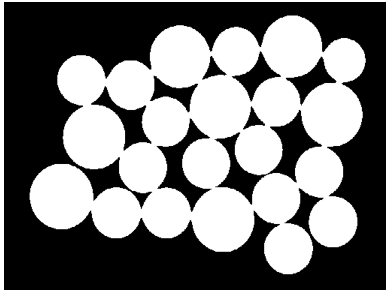
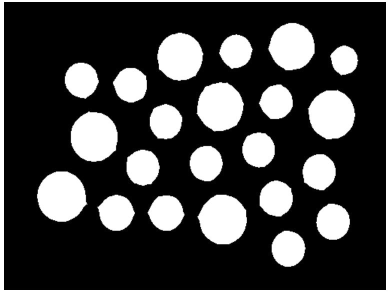
Another example of erosion



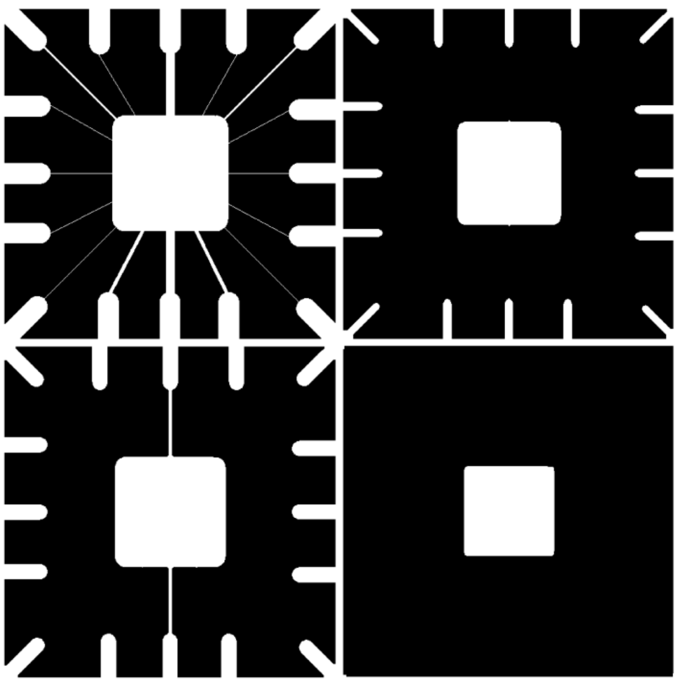
Erosion ➔ Image gets darker

Example: Counting coins

- Difficult because they touch each other!
- Solution: Binarization and Erosion separates them!

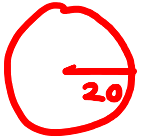




Erosion - example



a b
c d

FIGURE 9.8 An illustration of erosion.
(a) Original image.
(b) Erosion with a disk of radius 10.
(c) Erosion with a disk of radius 5.
(d) Erosion with a disk of radius 20.



From: Digital Image Processing, Gonzalez,Woods And Eddins

Erosion : Operation (min filter)

1	1	1
1	1	1
1	1	1

Set of coordinate points =

{ (-1, -1), (0, -1), (1, -1),

(-1, 0), (0, 0), (1, 0),

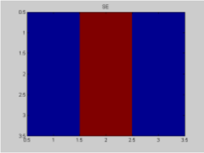
(-1, 1), (0, 1), (1, 1) }

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	1	1	1	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	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

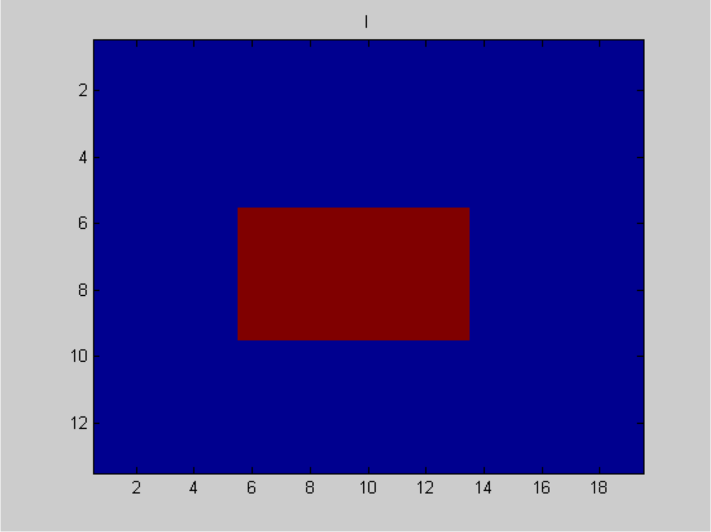
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0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

MATLAB code

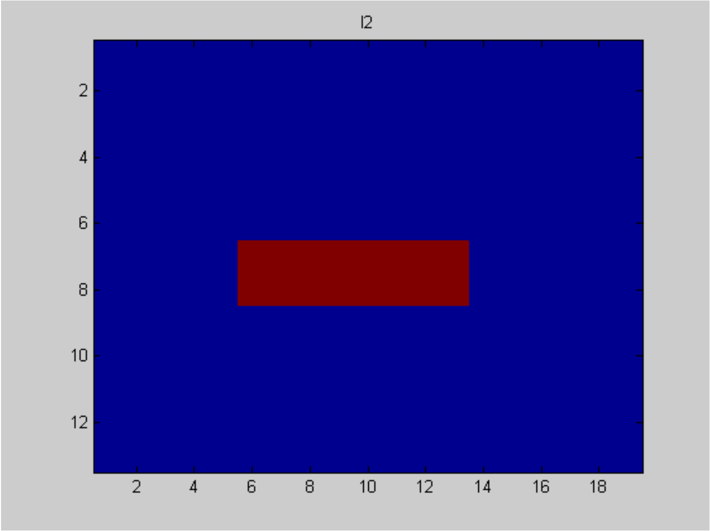
0 | 1 | 0



SE = 3x3



I2



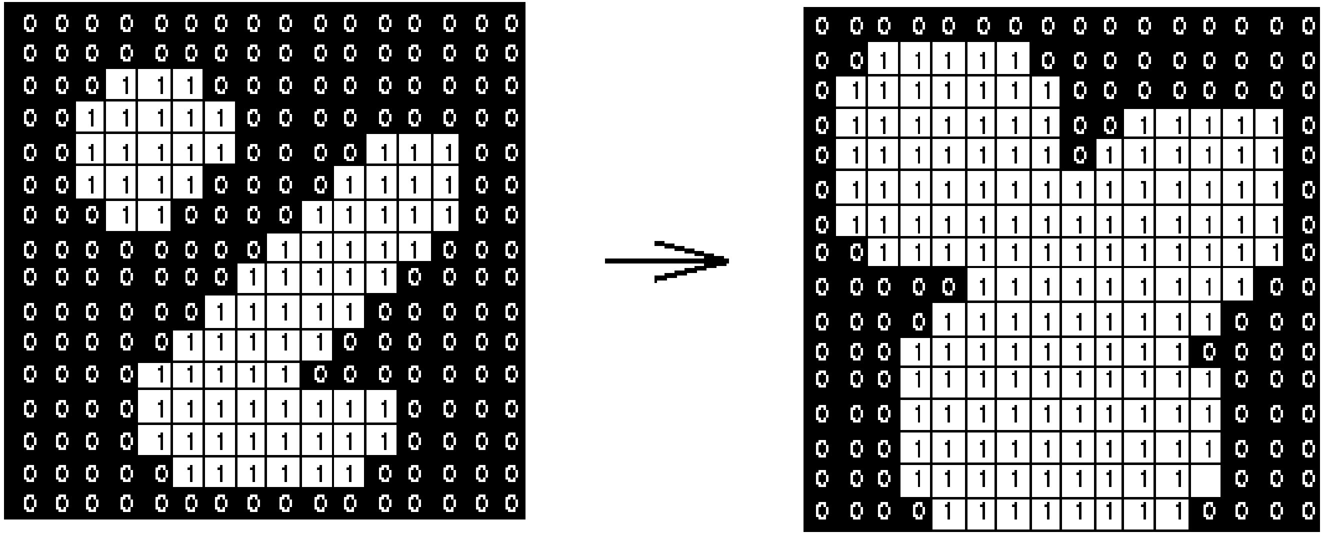
I3=imerode(I2,SE);

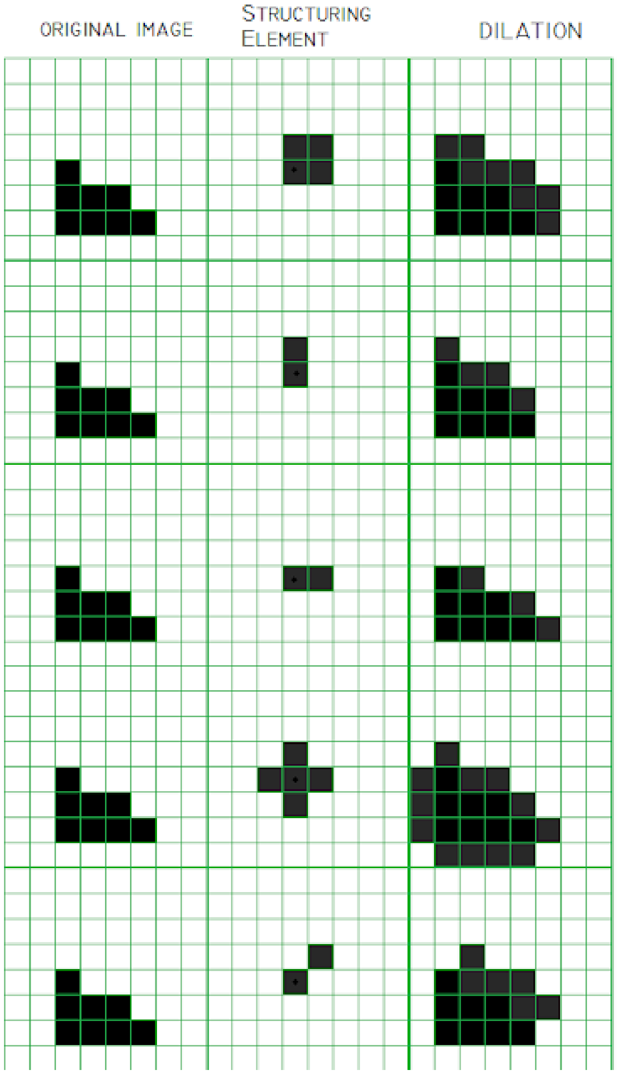
Erosion

- Shrinks foreground objects
- Foreground holes are enlarged
- Small (relative to structuring element size) foreground objects are removed.

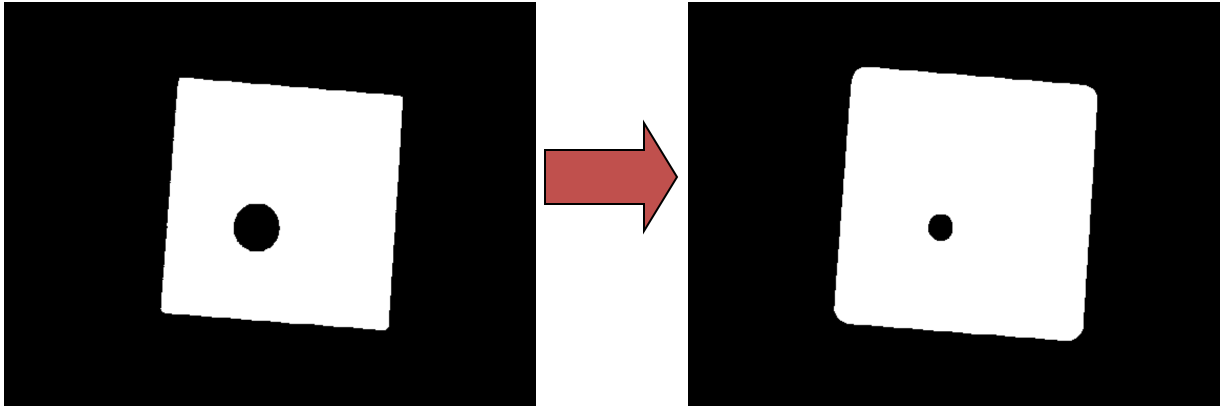
Dilation

1	1	1
1	1	1
1	1	1




ORIGINAL IMAGE
STRUCTURING ELEMENT
DILATION

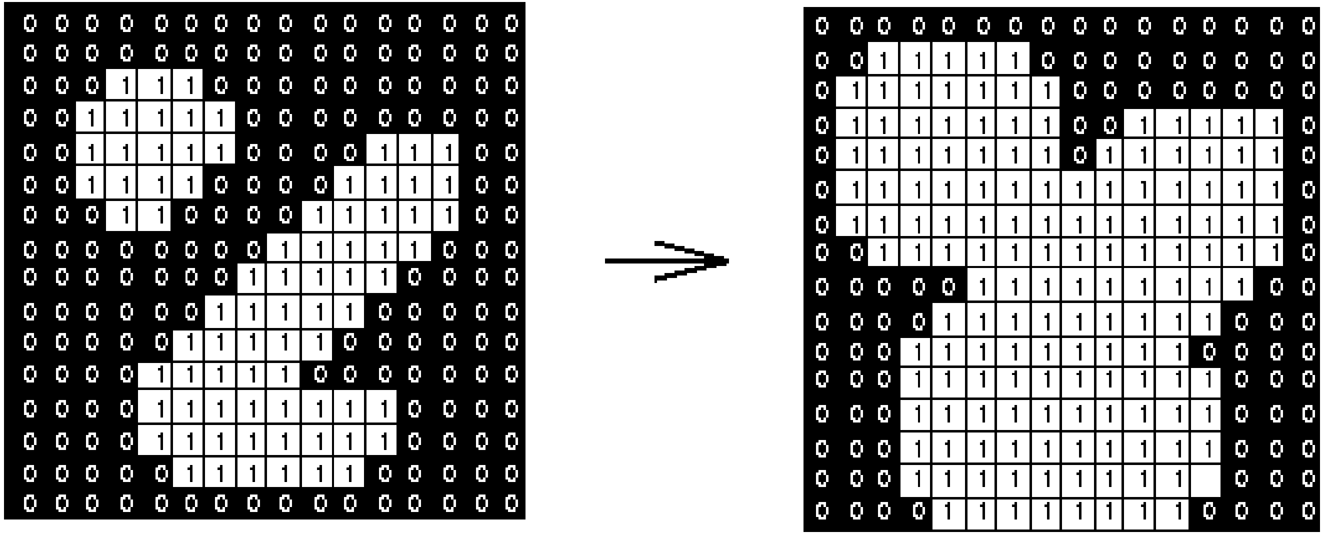
Dilation Example

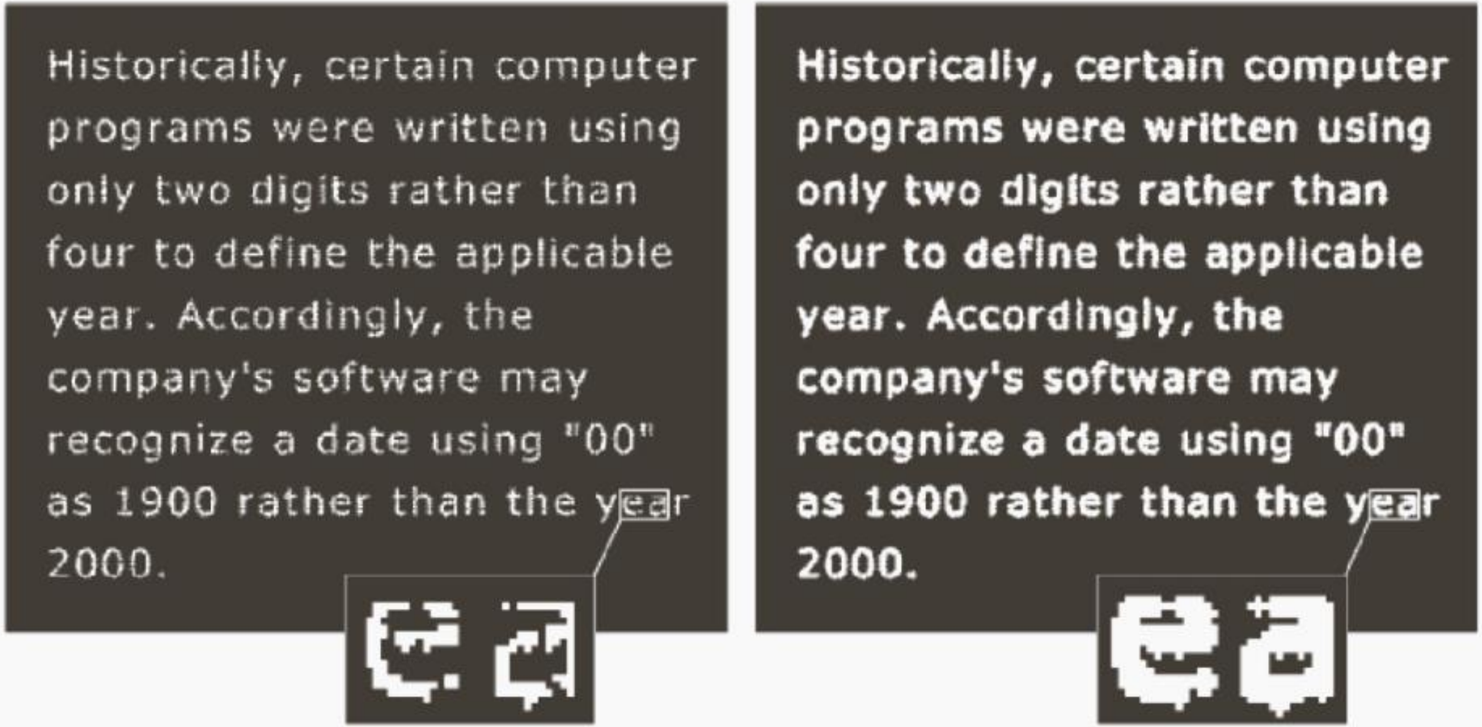


- Image gets lighter, more uniform intensity
- NOTE-1: SE = disk
- NOTE-2: Multiple iterations of dilation

Dilation (max filter)

1	1	1
1	1	1
1	1	1





0	1	0
1	1	1
0	1	0

Figure 1: Image before (left) and after (right) dilation with the structuring element shown at the bottom

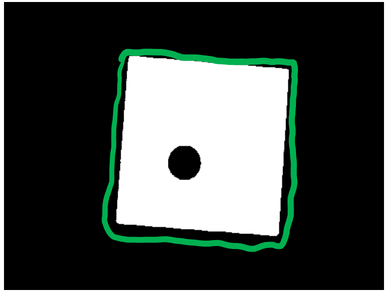
Dilation

- Expands foreground objects
- Foreground holes are shrunk

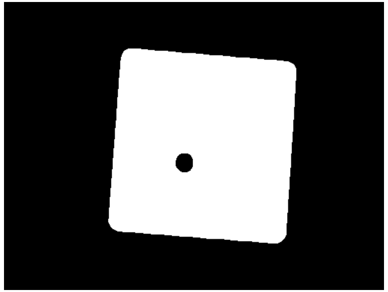
Boundary Detection

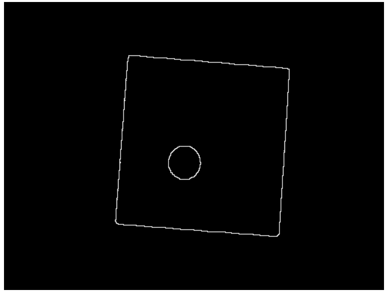
1. Dilate input image
2. Subtract input image from dilated image D - I
3. Boundaries remain!

I



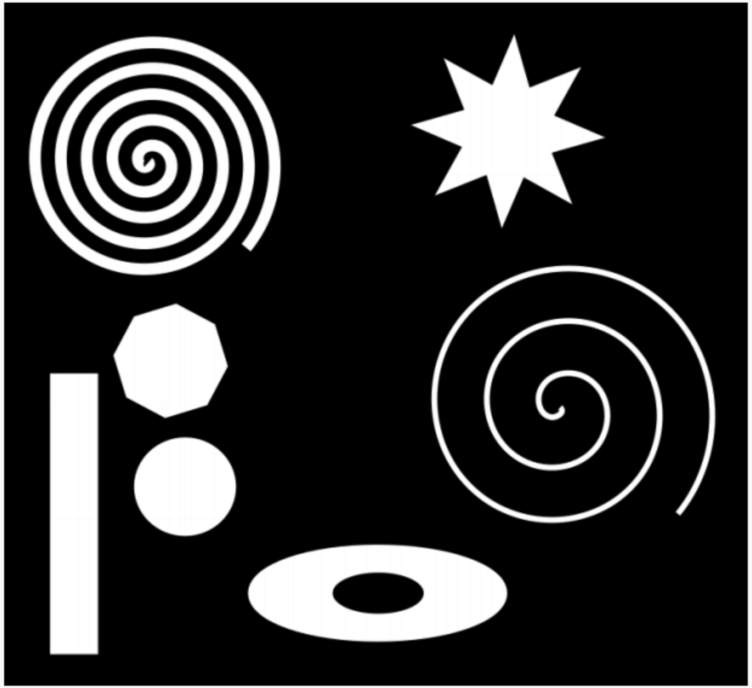
D





Can use erosion also ..

Fig 3: (a) Original Image (linkon.tif) (B) After erosion operation (C) Boundary Extraction with the help of Erosion.

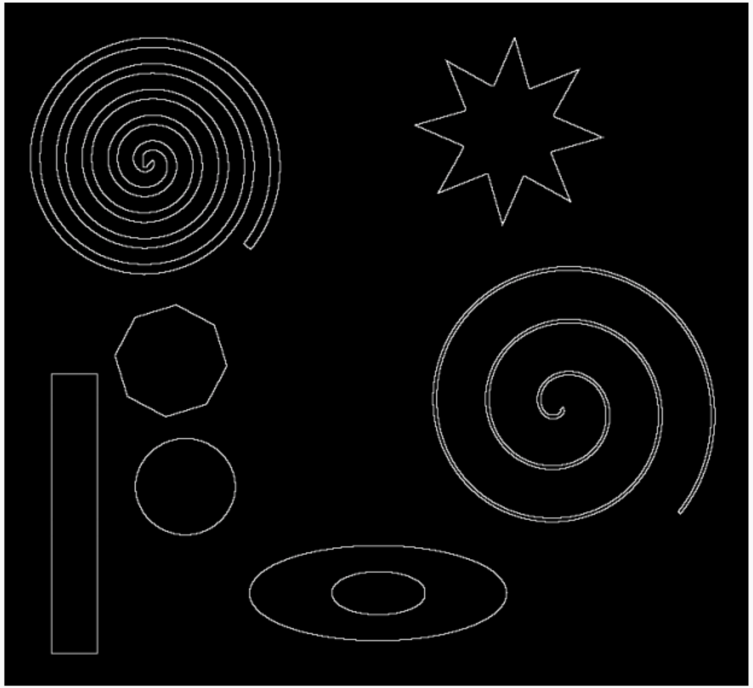


(a) f

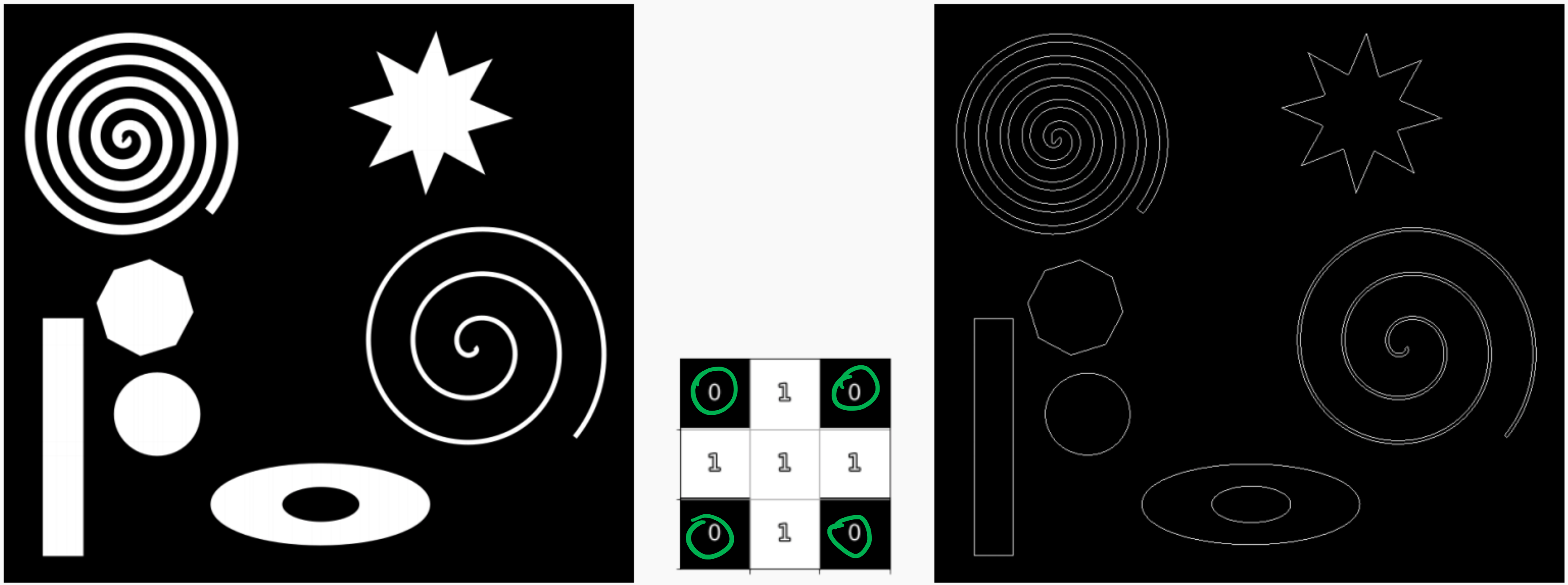
8 - connectivity

1	1	1
1	1	1
1	1	1

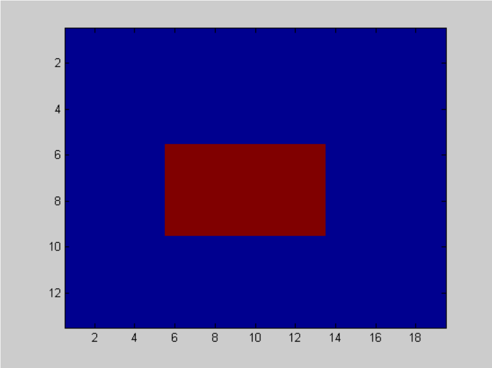
(b) s



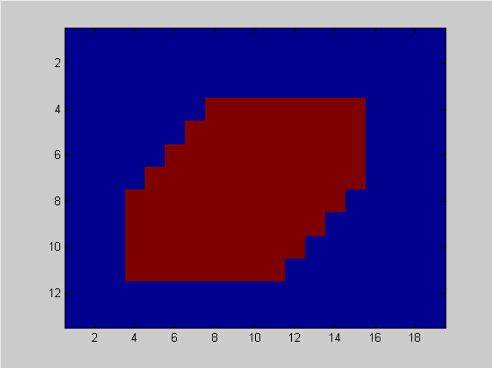
(c) $f - (f \ominus s)$



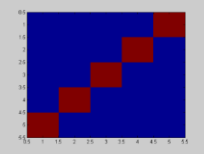
(a) f (b) s (c) $f - (f \ominus s)$



I



I2



SE

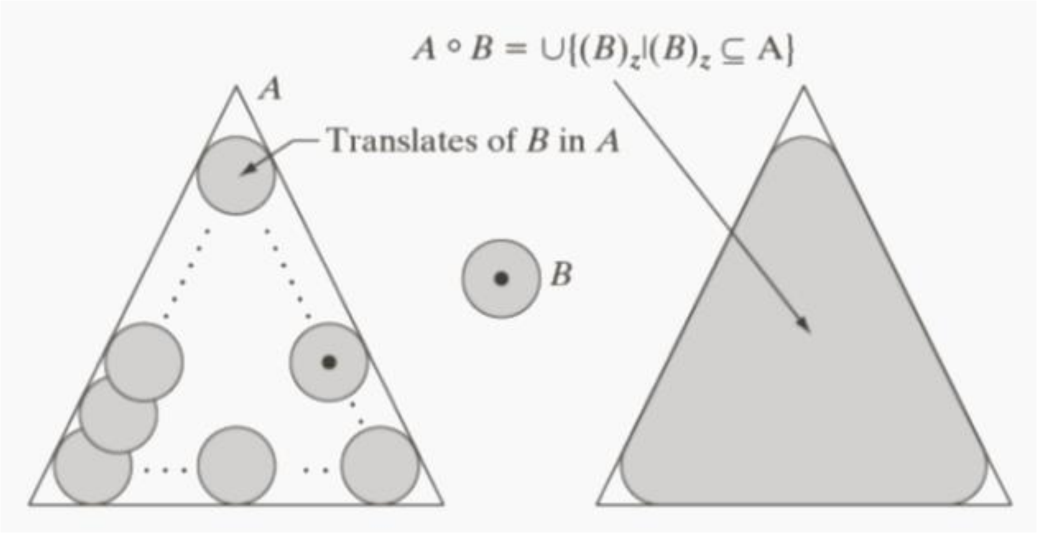
```
>> I(6:9,6:13)=1;
>> figure, imagesc(I)
>> I2=imdilate(I,SE);
>> figure, imagesc(I2)
```

Opening and Closing

- Important operations
- Derived from the fundamental operations
  - Dilation ✓
  - Erosion ✓

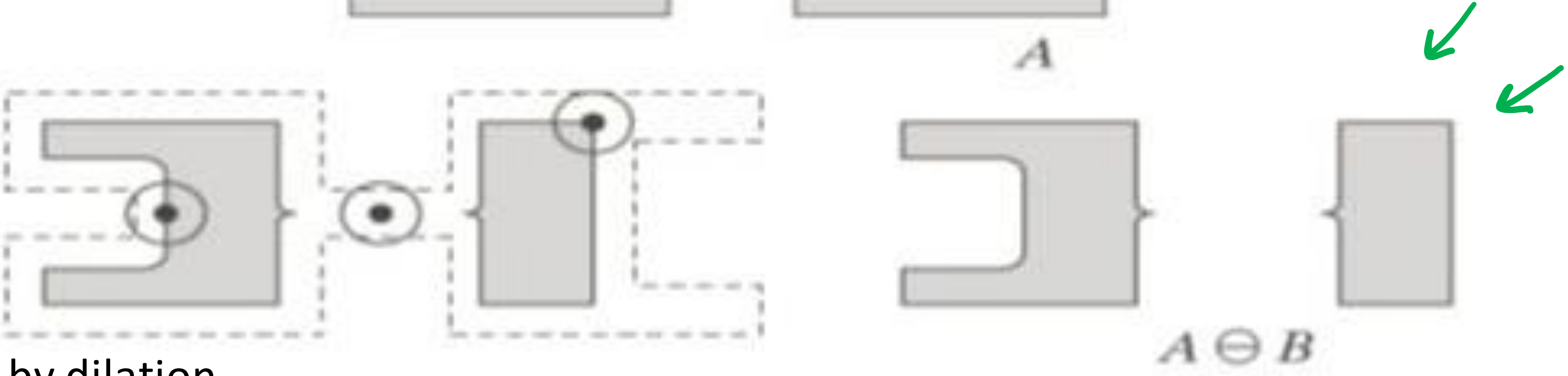
Opening

- Take the structuring element (SE) and <u>slide it around *inside* each foreground region</u>.
  – All pixels which can be covered by the SE with the SE being entirely within the foreground region will be preserved.
  – All foreground pixels which can *not* be reached by the structuring element without lapping over the edge of the foreground object will be eroded away!



Opening

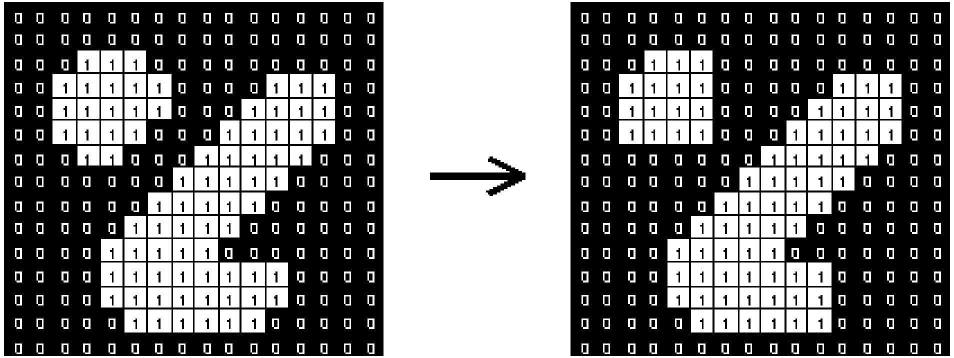
Erosion



Followed by dilation …

Opening

- Structuring element: 3x3 square



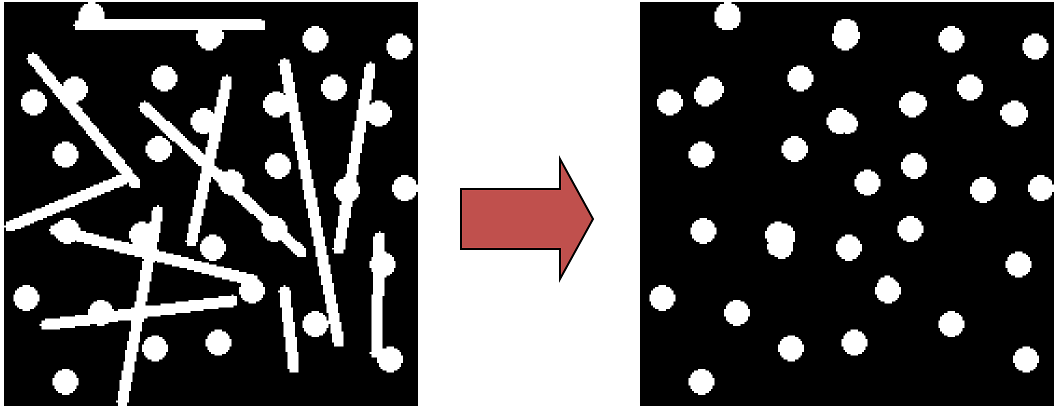
Take the structuring element (SE) and <u>slide it around *inside* (each) foreground region</u>

All foreground pixels which can *not* be reached by the structuring element without lapping over the edge of the foreground object will be eroded away!

All pixels which can be covered by the SE with the SE being entirely within the foreground region will be preserved.

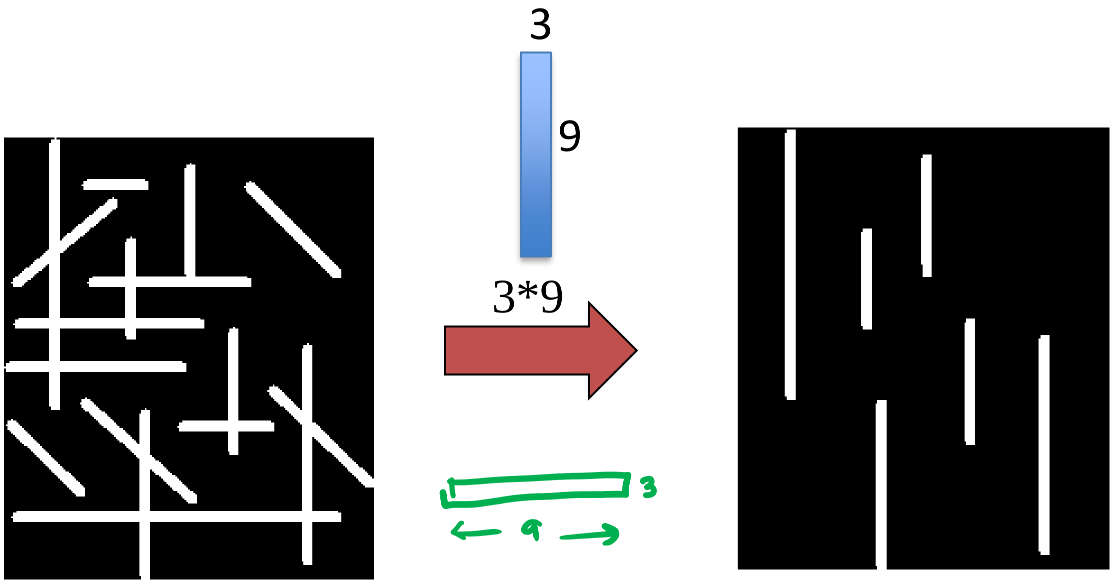
Opening: Example

- Opening with a 11 pixel diameter disc



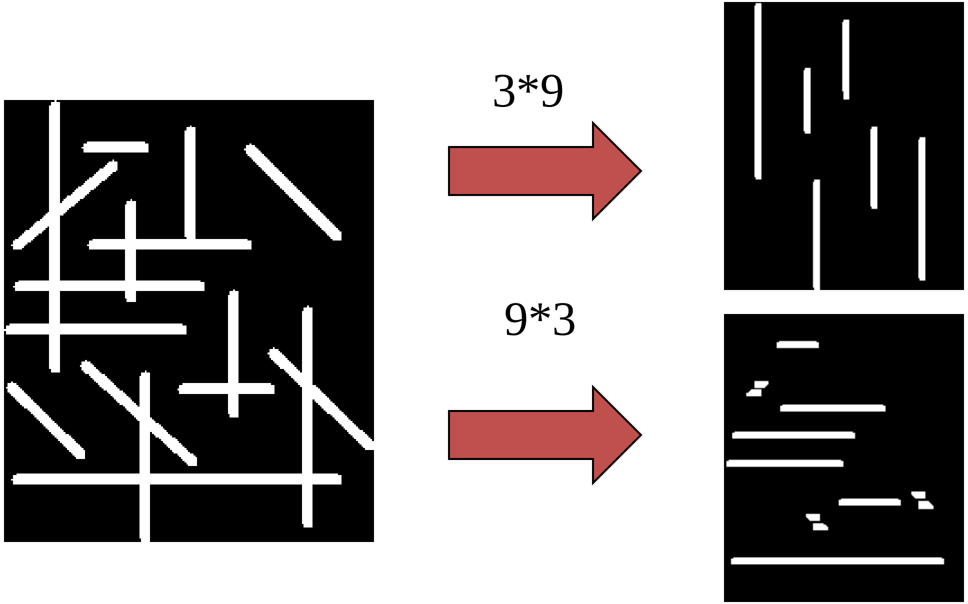
Opening: Another Example

- 3x9 Structuring Element

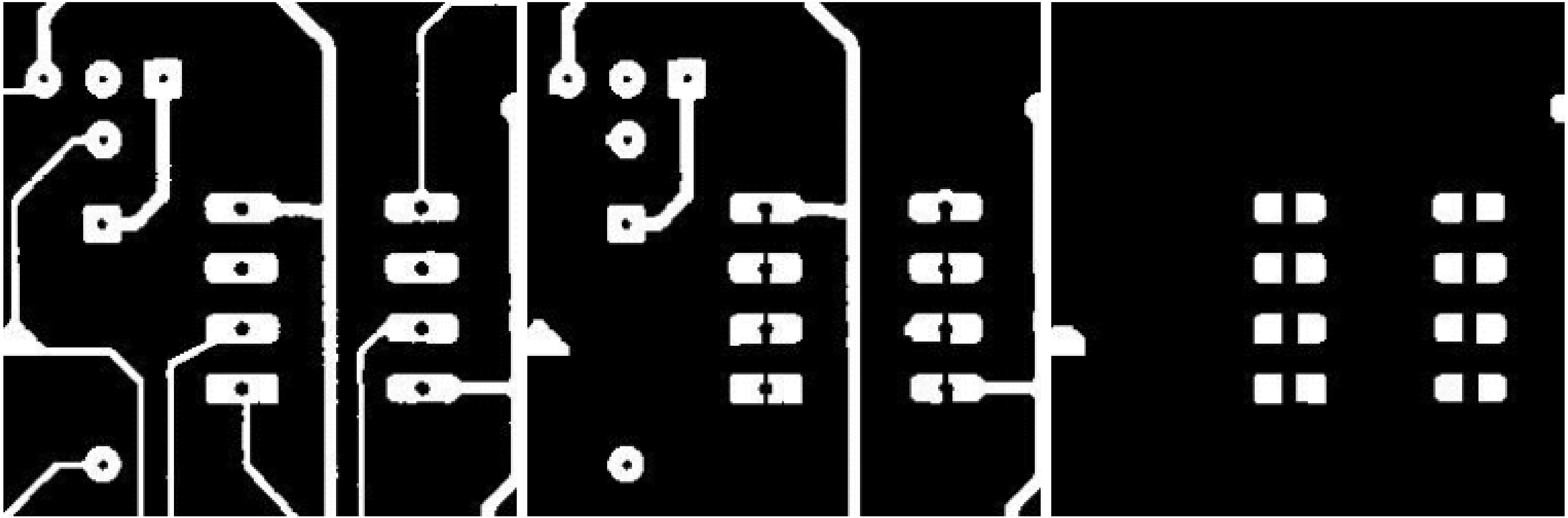


Opening: Another Example

- 3x9 and 9x3 Structuring Element



Opening

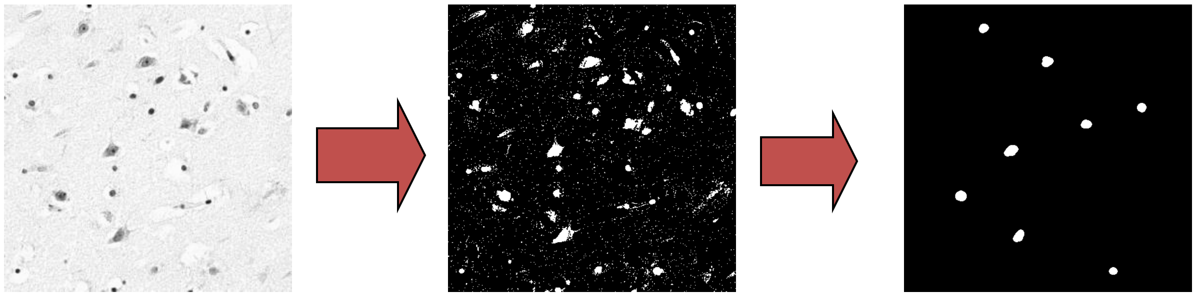


Binary image f $\quad$ $f \circ s$ (5×5 square) $\quad$ $f \circ s$ (9×9 square)

Results of opening with a square structuring element (www.mmorph.com/html/morph/mmopen.html/).

Use Opening for Separating Blobs

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



Opening

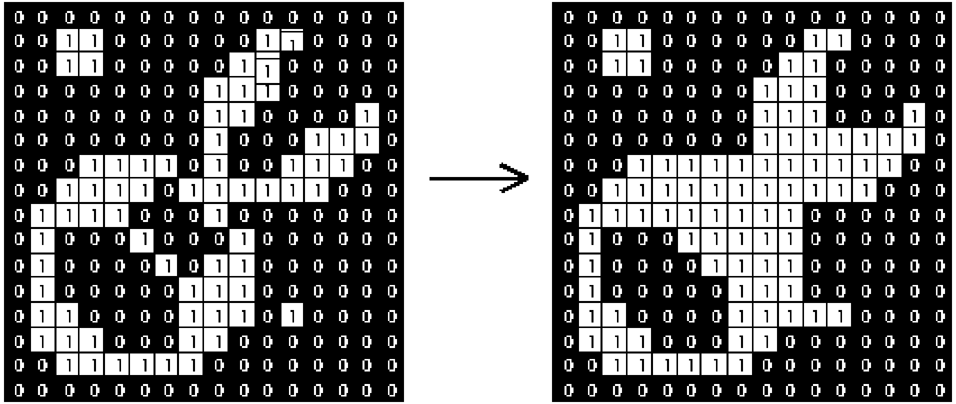
$B = \text{Open}(I)$

$B = \text{Open}(B)$

- Similar to Erosion
  - Spot and noise removal
  - Less destructive
- Erosion followed by Dilation using the same SE
- the *same structuring element for both operations.*
- Opening is **idempotent:** Repeated application has no further effects!

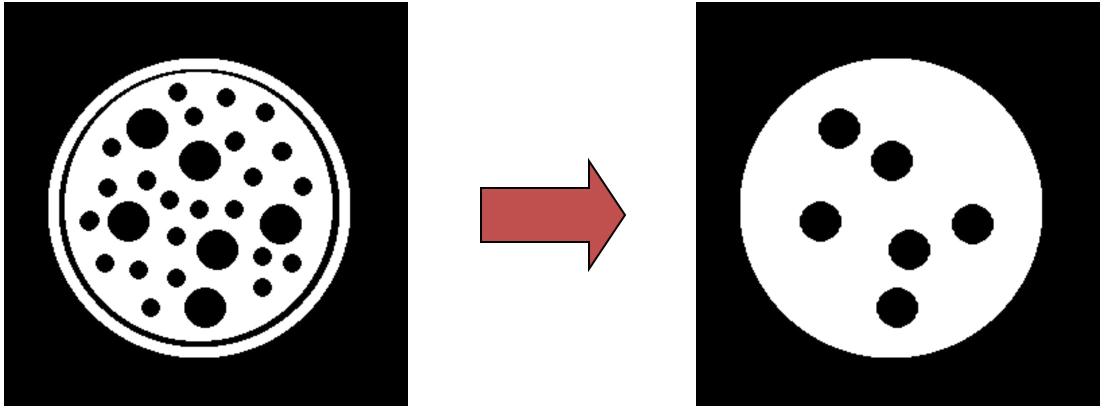
Closing (Dilation then Erosion)

- Structuring element: 3x3 square



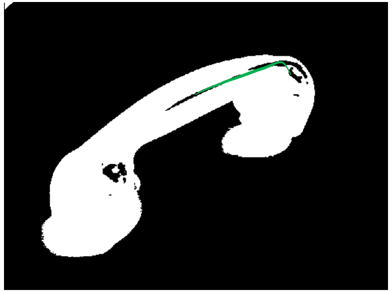
Closing: Example

- Closing operation with a 22 pixel disc
- Closes small holes in the foreground

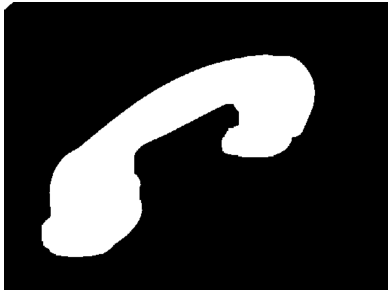


Closing Example 1

1. Threshold
2. Closing with disc of size 20



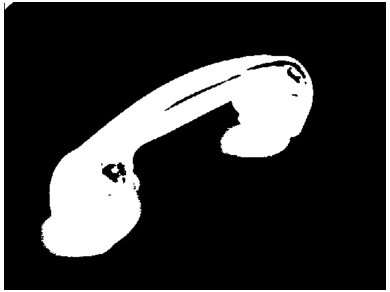
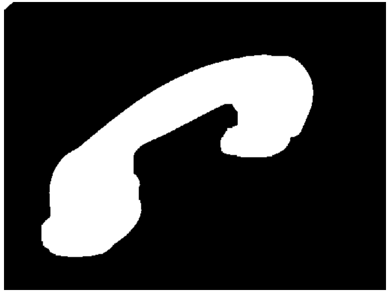
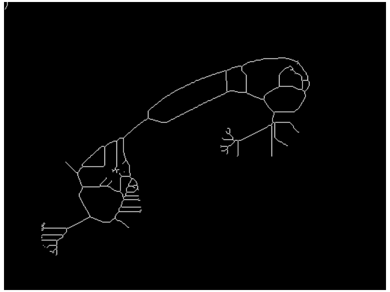
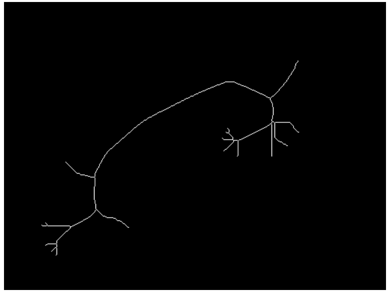
Thresholded



closed

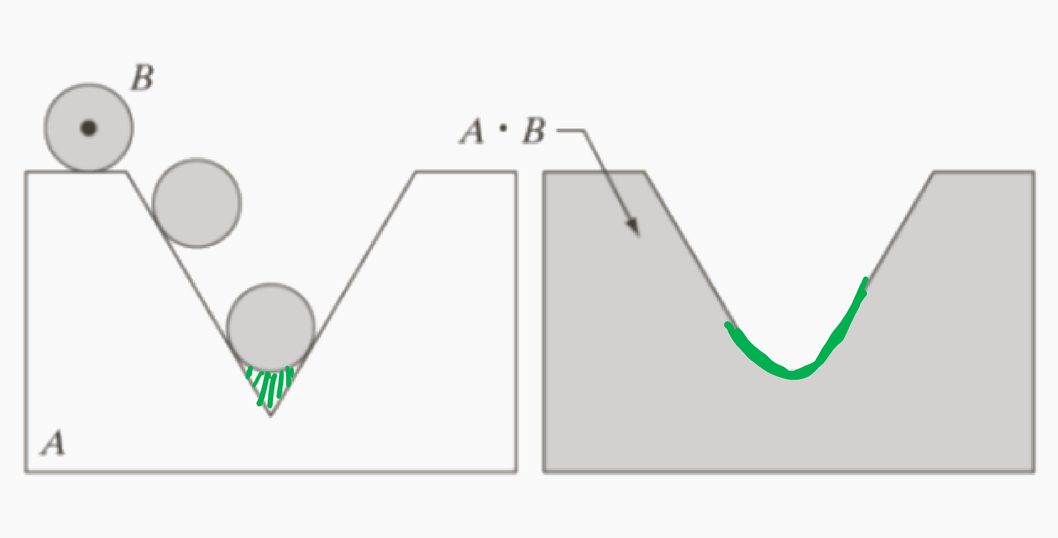
Application of Closing

- Good for further processing: E.g. Skeleton operation looks better for closed image!

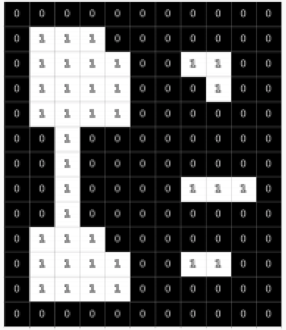
Closing

- Take the structuring element (SE) and slide it around *outside* each foreground region.
- Closing = Regions not covered by the structuring element when doing the above operation.

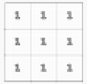


Opening & Closing

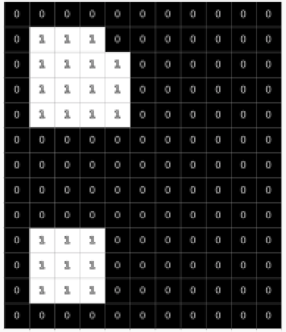
- Opening is the *dual* of closing
- *i.e.* opening the foreground pixels with a particular structuring element
- is equivalent to closing the background pixels with the same element.



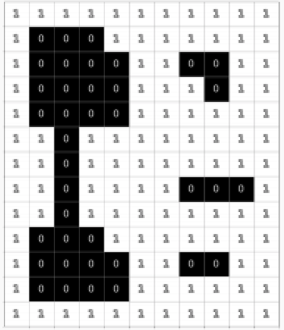
(a) f



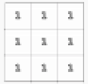
(b) s



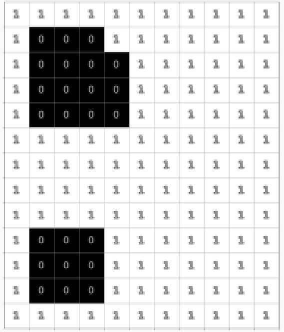
(c) $f \circ s$



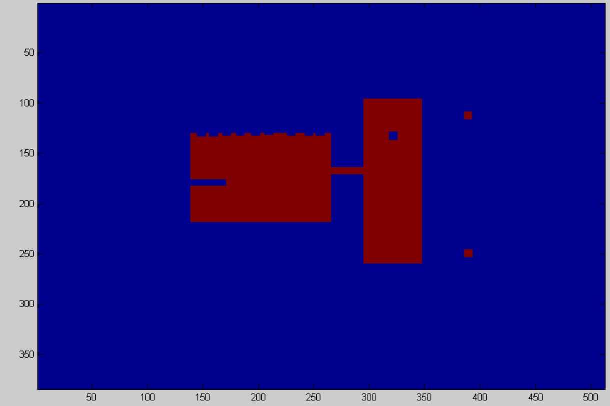
(a) f^c



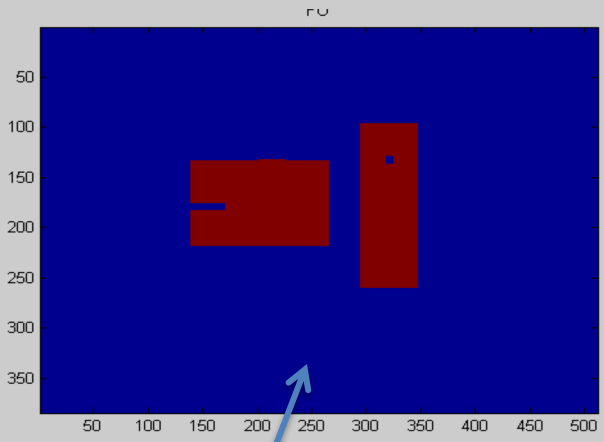
(b) s

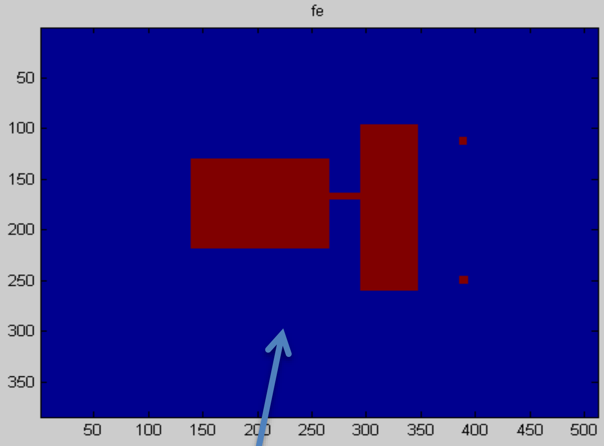


(c) $f^c \bullet s$



ORIGINAL





OPENING

CLOSING

Fingerprint problem

a b c

FIGURE 9.11 (a) Noisy fingerprint image. (b) Opening of image. (c) Opening followed by closing. (Original image courtesy of the National Institute of Standards and Technology.)

From: Digital Image Processing, Gonzalez, Woods And Eddins

MAGNITUDE RELATIONS

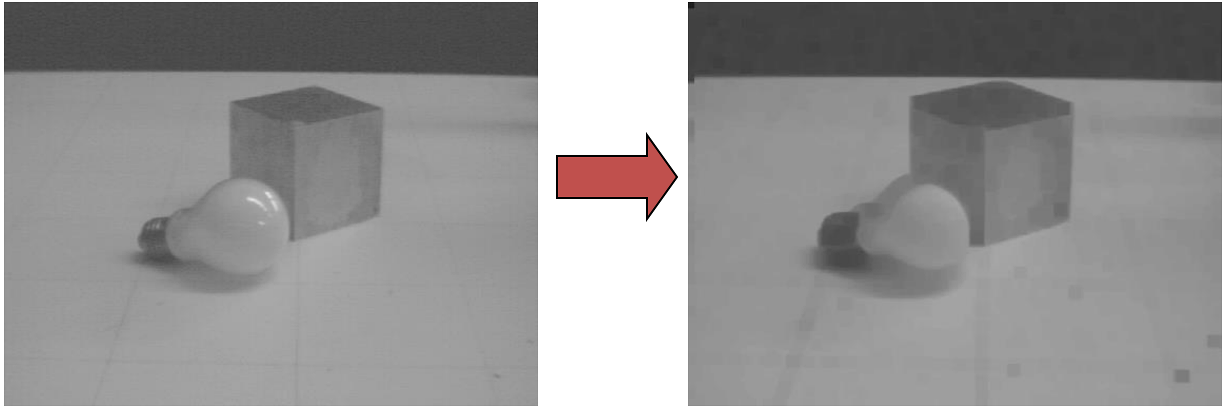
- Dilation and closing are *extending operations*, meaning that foreground pixels are added to the image.
- Erosion and opening are *narrowing operations*, meaning that foreground pixels are removed.
- For a binary image f and a binary structuring element s, we have that

$$\overset{\text{EROSION}}{(f \ominus s)(x)} \leq \overset{\text{OPENING}}{(f \circ s)(x)} \leq f(x) \leq \overset{\text{CLOSING}}{(f \bullet s)(x)} \leq \overset{\text{DILATION}}{(f \oplus s)(x)}$$

- On a similar note, if $F(g)$ is the set of foreground pixels in g,

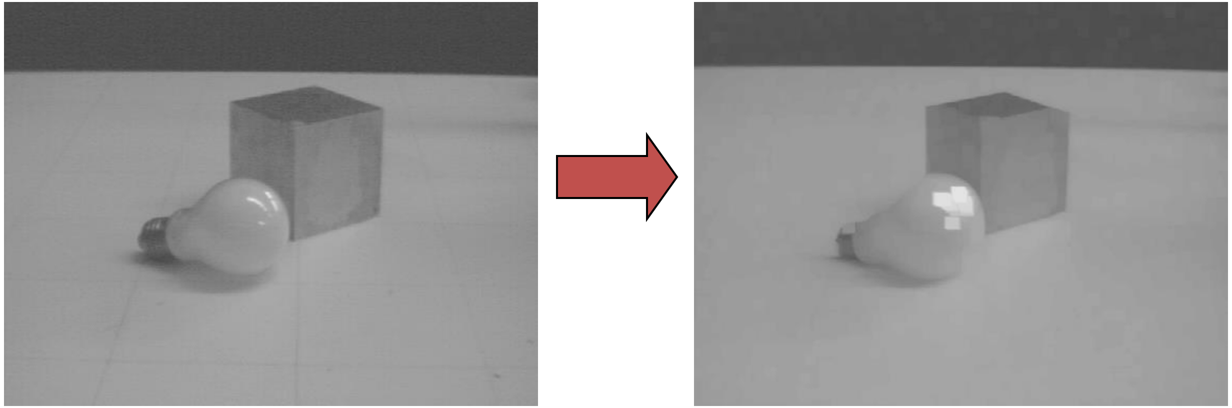
$$F(f \ominus s) \subseteq F(f \circ s) \subseteq F(f) \subseteq F(f \bullet s) \subseteq F(f \oplus s)$$

Erosion on Gray Value Images



- min filter
- Images get darker!

Dilation on Gray Value Images



- max filter
- More uniform intensity

References

- G&W, 3rd Ed., 9.1-9.3, 9.6
- https://in.mathworks.com/help/images/morphological-dilation-and-erosion.html
- https://scikit-image.org/docs/dev/auto_examples/applications/plot_morphology.html#sphx-glr-auto-examples-applications-plot-morphology-py

Scribe List

2018102018
2018102019
2018102022
2018102027
2018102028
2018102031