Morphological Image Processing

Introduction

- *‘Morphology’* denotes a branch of biology

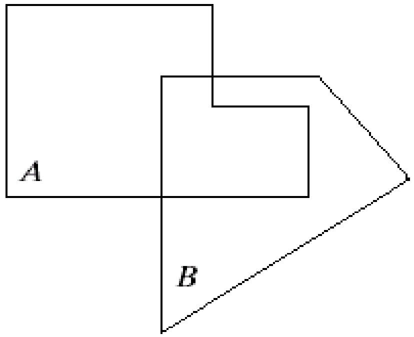
 that deals with the form and structure of animals and plants.

- For images, morphological operations change the shape of the object

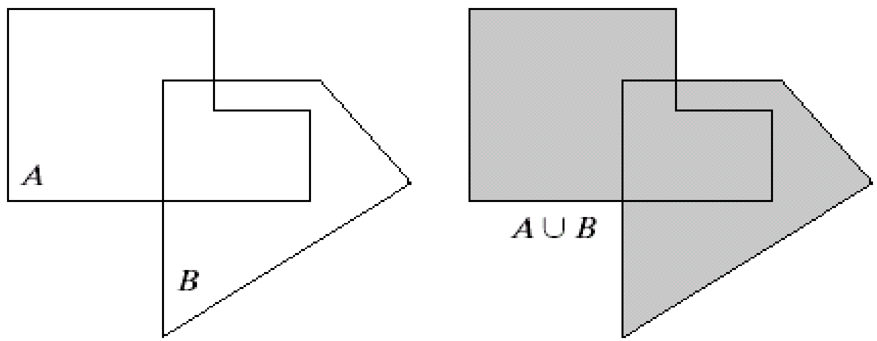
Basic Principle

- Extraction of geometrical information from an unknown image through transformations
- Use a well-defined, set known as Structuring Element (SE) for extraction
- Design of SEs, their shape and size, is crucial to the success of the morphological operations

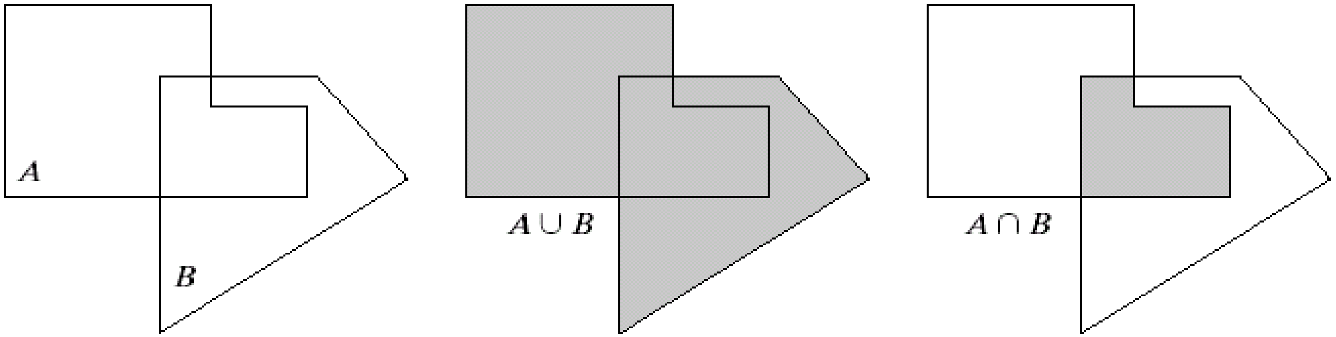
Operators



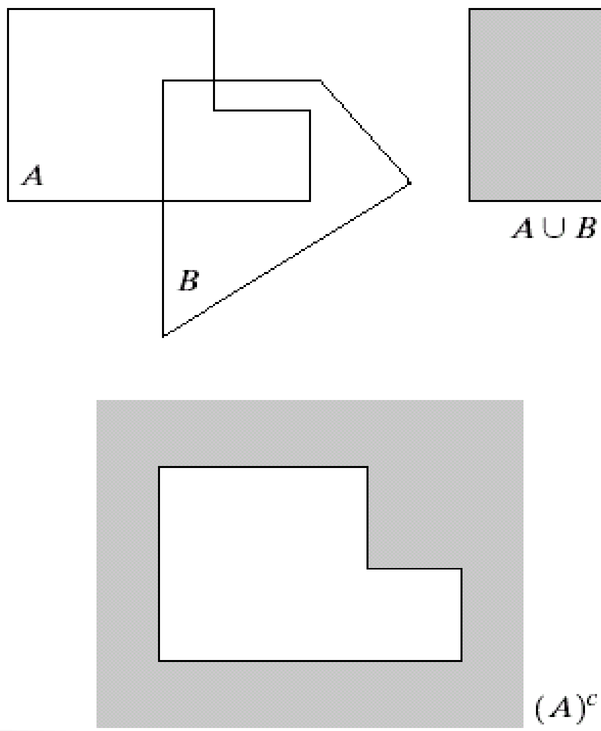
Operators



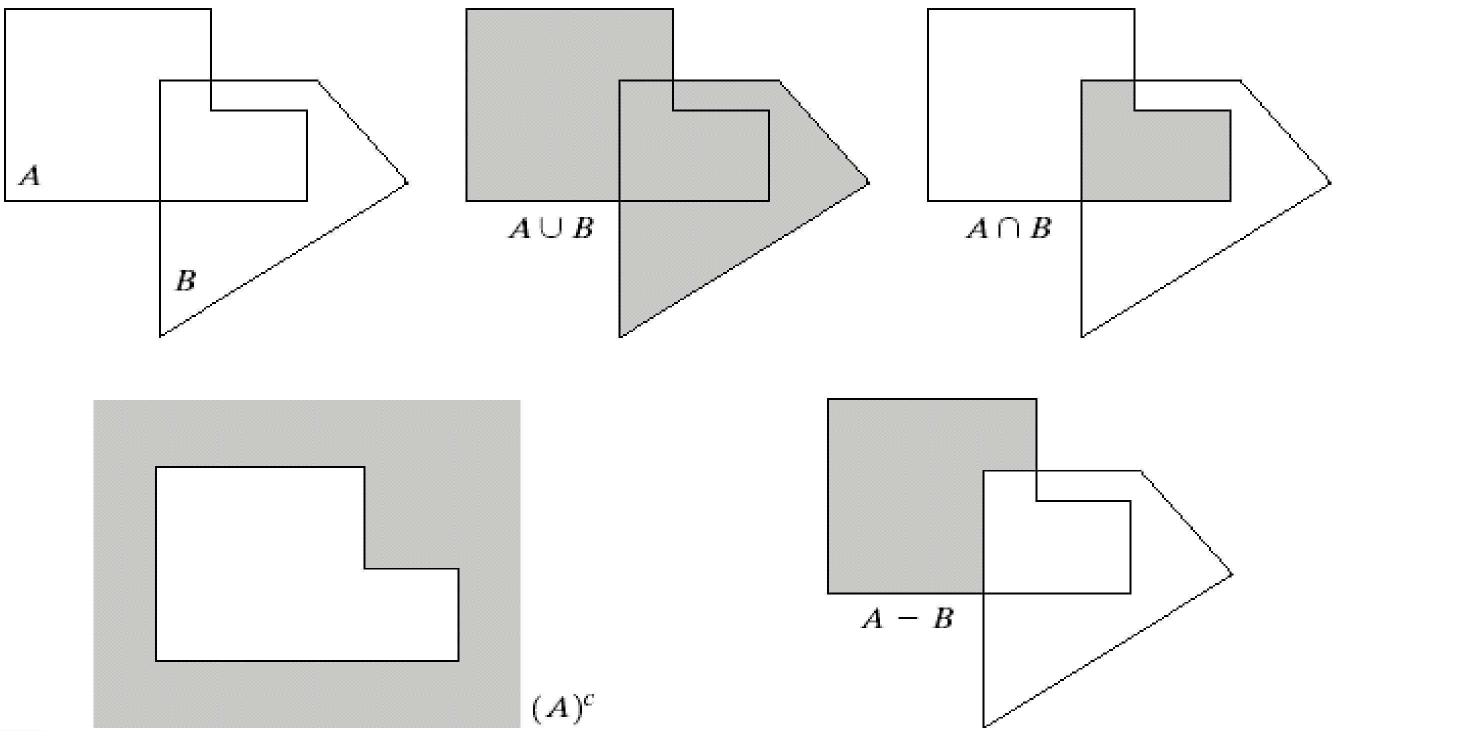
Operators



Operators

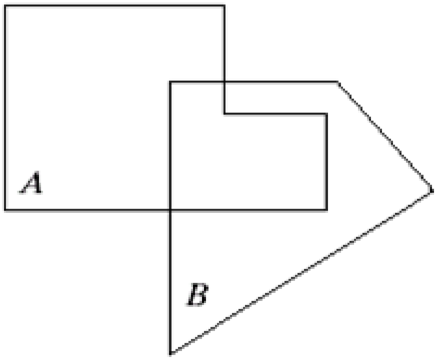


Operators



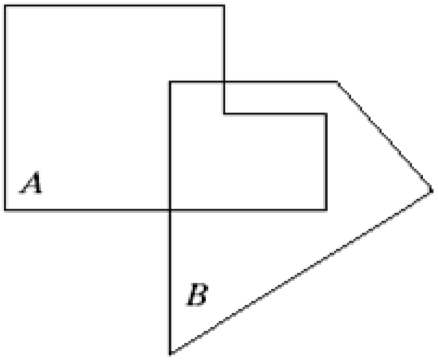
Reflection and Translation

Original image

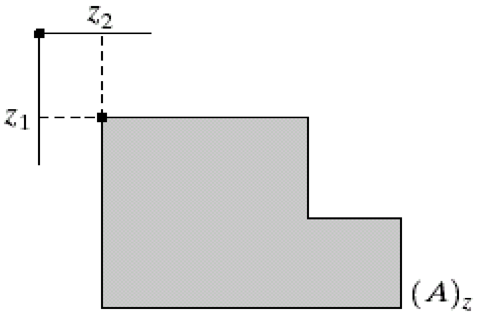


Translation and Reflection

Original image



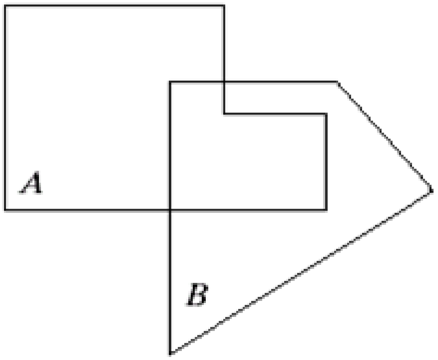
Translation of A by z = (z_1,z_2)units



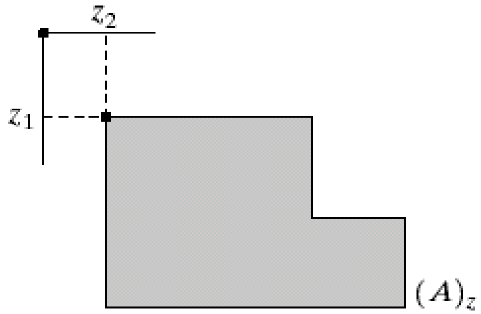
$$(A)_z = \{c \mid c = a + z, for\ a \in A\}$$

Translation and Reflection

Original image



Translation of A by z = (z_1,z_2)units

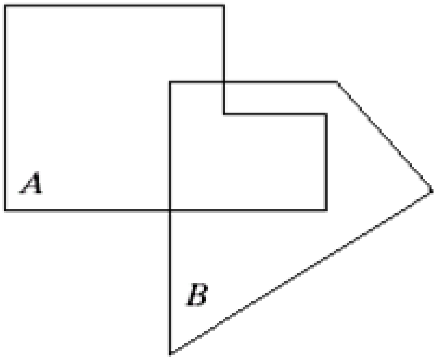


Reflection (flipping) of B

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

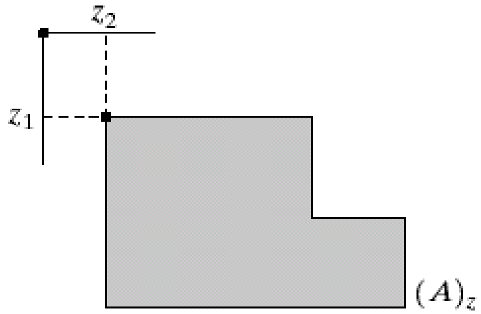
Translation and Reflection

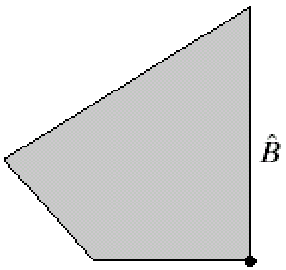
Original image



Translation of A by z = (z_1,z_2)units

Reflection (flipping) of B

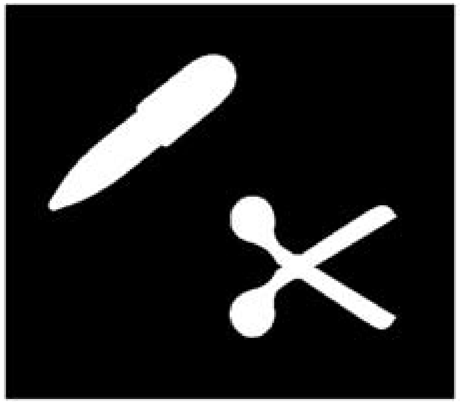




$(A)_z = \{c|c = a + z, for\ a \in A\}$ $\hat{B} = \{w|w = -b, for\ b \in B\}$

Binary operations

(a) Set A



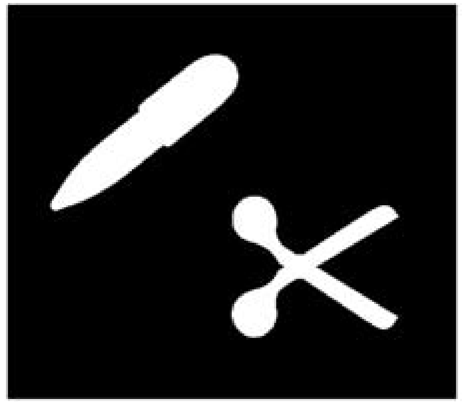
(b) Set B

Binary operations

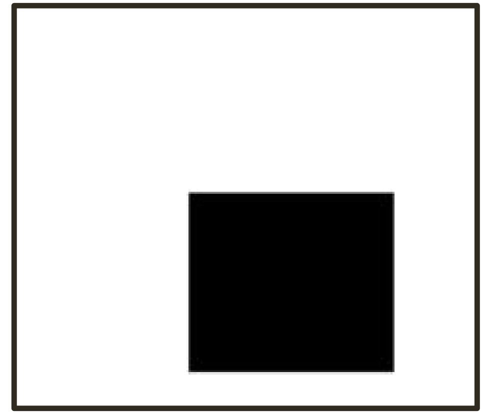
(a) Set A (b) Set B (c) Set A^c

Binary operations

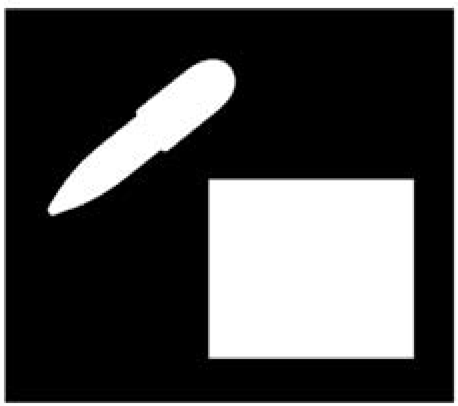
(a) Set A



(b) Set B

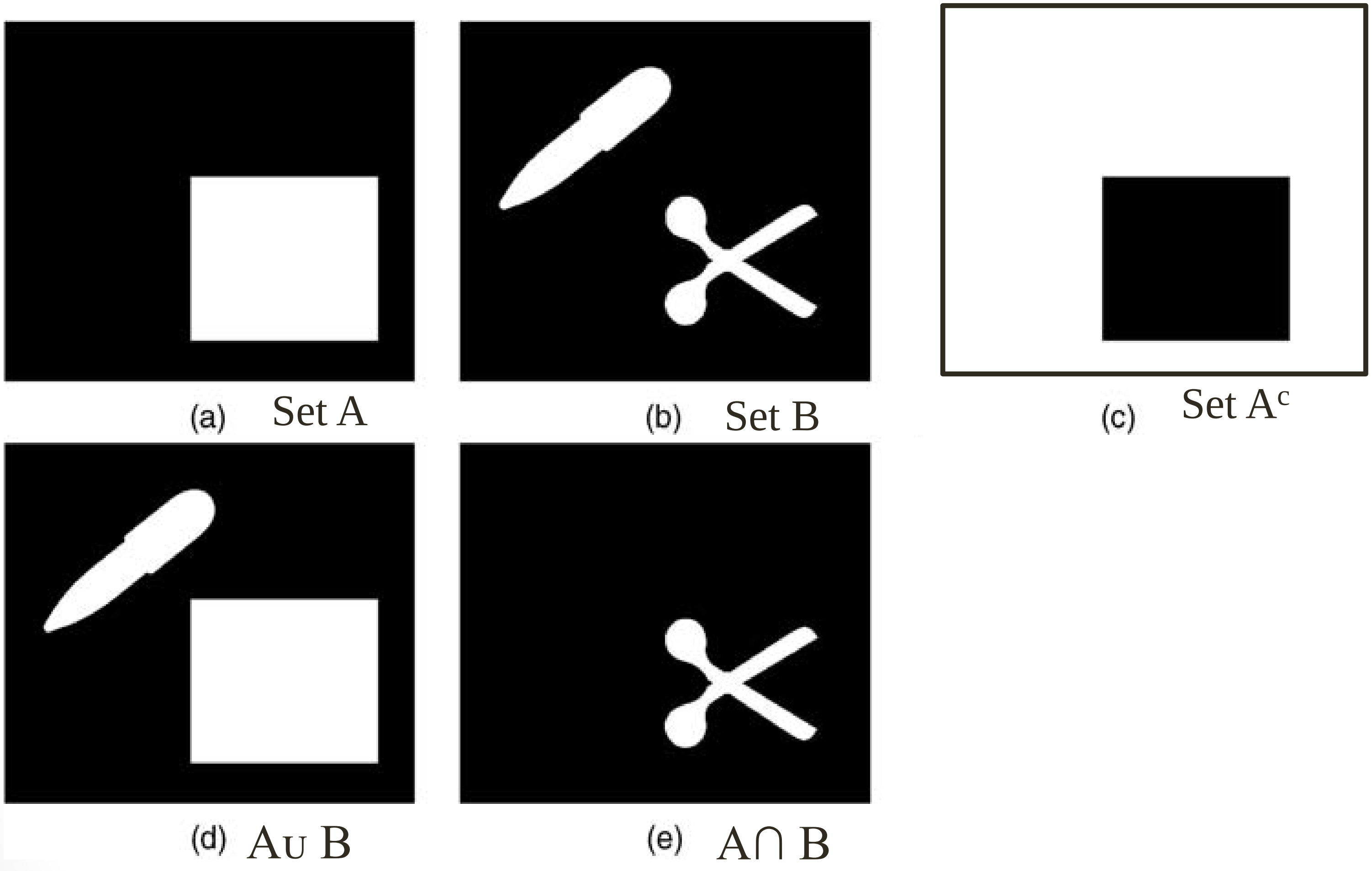


(c) Set A^c

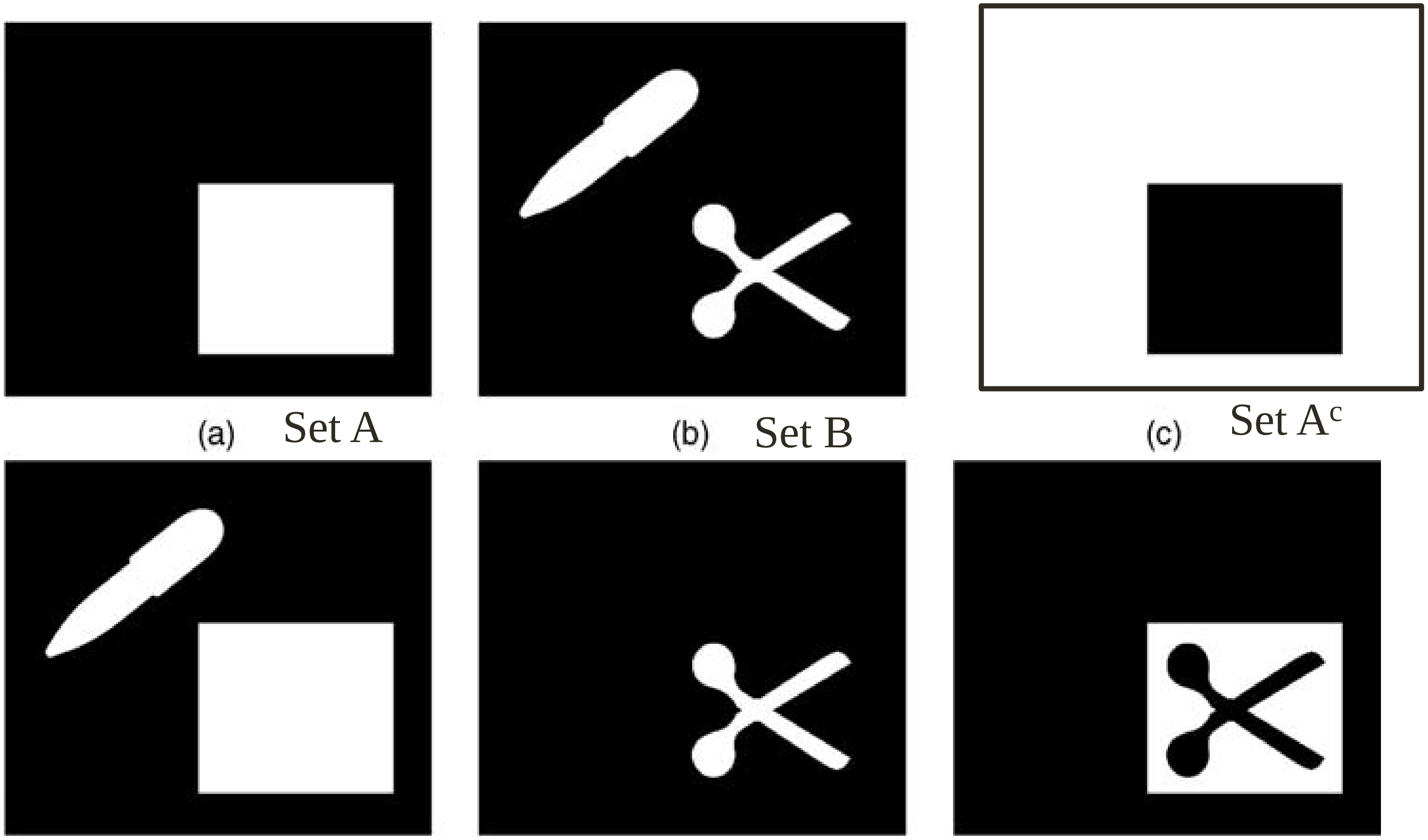


(d) $A \cup B$

Binary operations



Binary operations

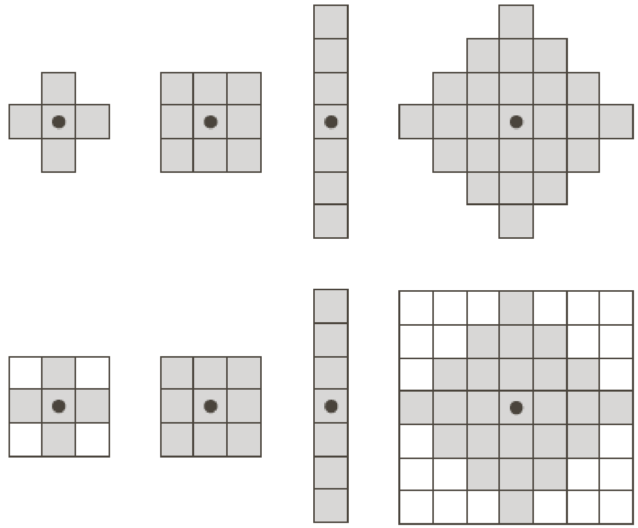


(a) Set A (b) Set B (c) Set A^c

(d) $A \cup B$ (e) $A \cap B$ (f) A - B

Structuring Element

- Small sets or subimages
- Used to probe an image to study region of interest
- Ex:Grey square (foreground) shows true (‘1’)
- Ex: White square (background) shows false (‘0’)



Some structuring elements

Some Morphological Operations

- Erosion
- Dilation
- Opening
- Closing

Erosion

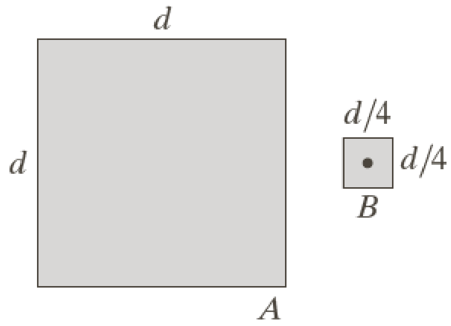
- Erosion is used for shrinking element A by using element B
- Set B is a structuring element
  - Erosion for Sets A and B is defined by

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

- Erosion of A by B is the set of all points, z such that B, translated by z, is contained in A
- That is eroded image contains center of structuring element after translation

Erosion of A by Structuring Element B

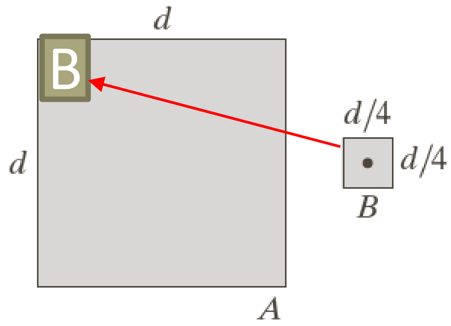
Image SE1



Erosion of A by Structuring Element B

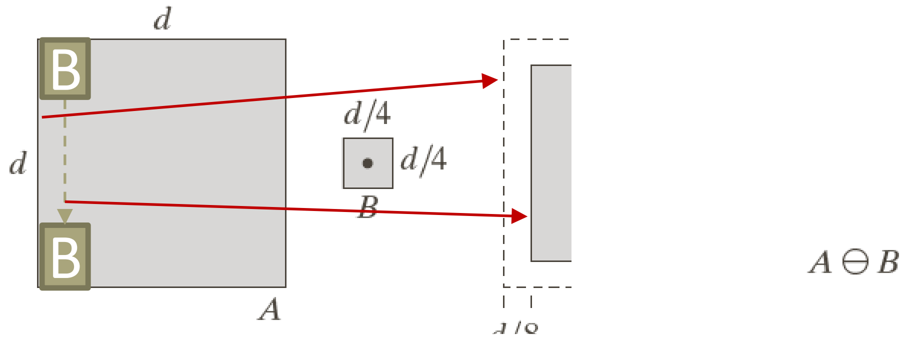
SE1 on image SE1

Eroded image with SE1

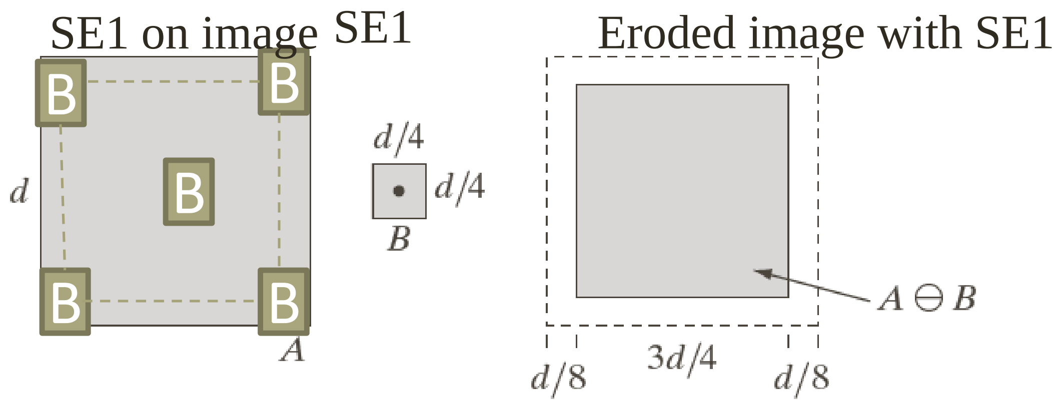


Erosion of A by Structuring Element B

SE1 on image SE1 Eroded image with SE1

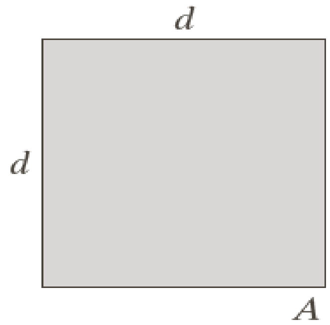


Erosion of A by Structuring Element B

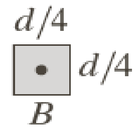


Erosion of A by Structuring Element B

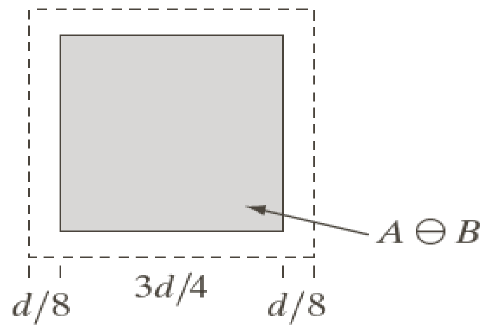
Image

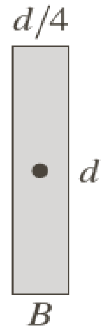


SE1



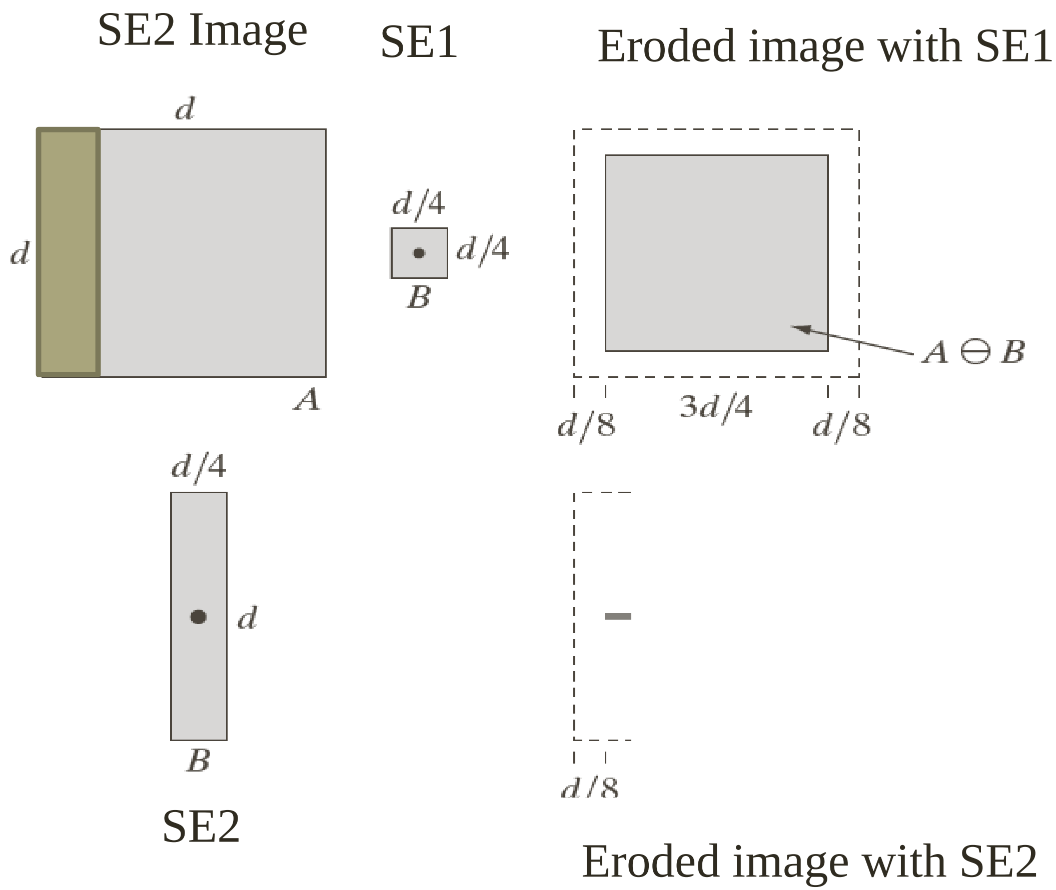
Eroded image with SE1





SE2

Erosion of A by Structuring Element B

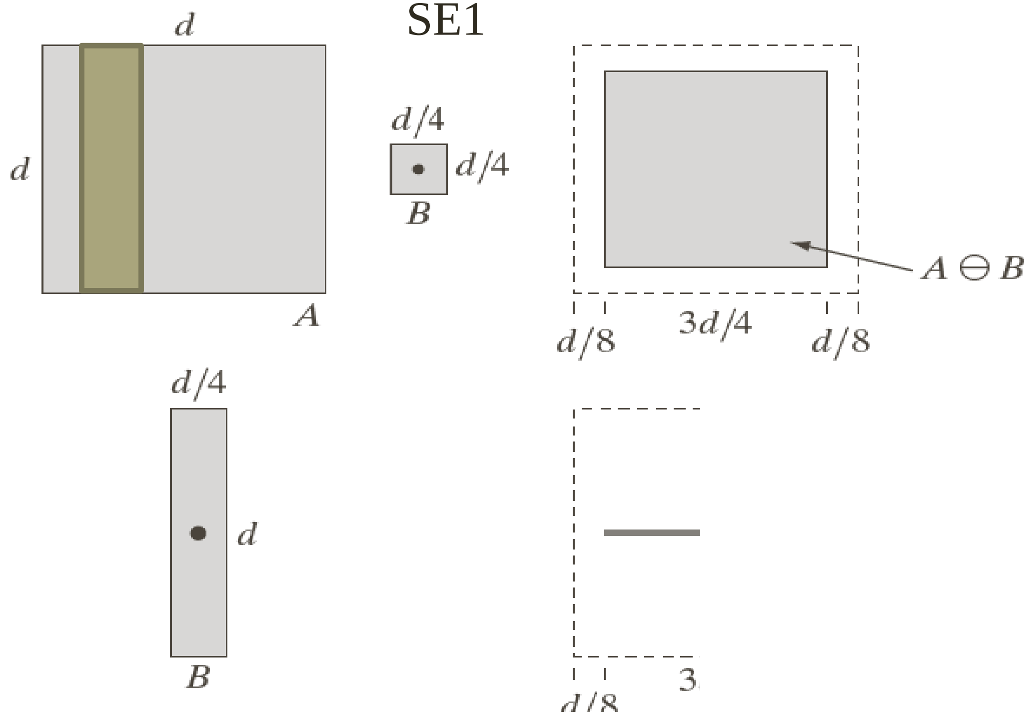


Erosion of A by Structuring Element B

SE2 on Image

Eroded image with SE1

SE1

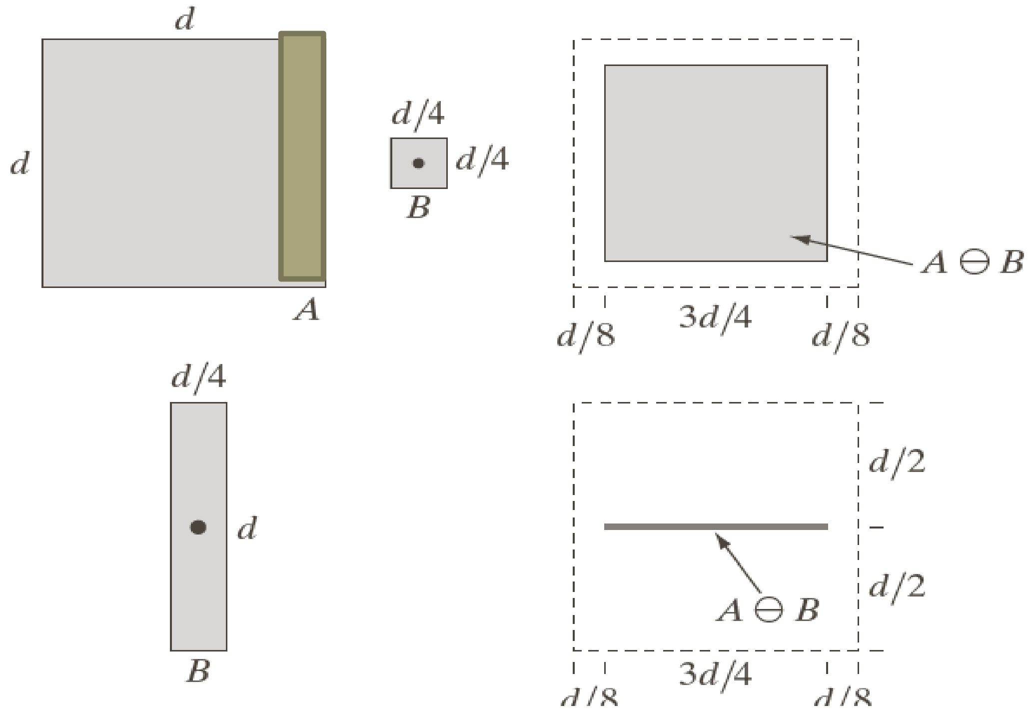


SE2

Eroded image with SE2

Erosion of A by Structuring Element B

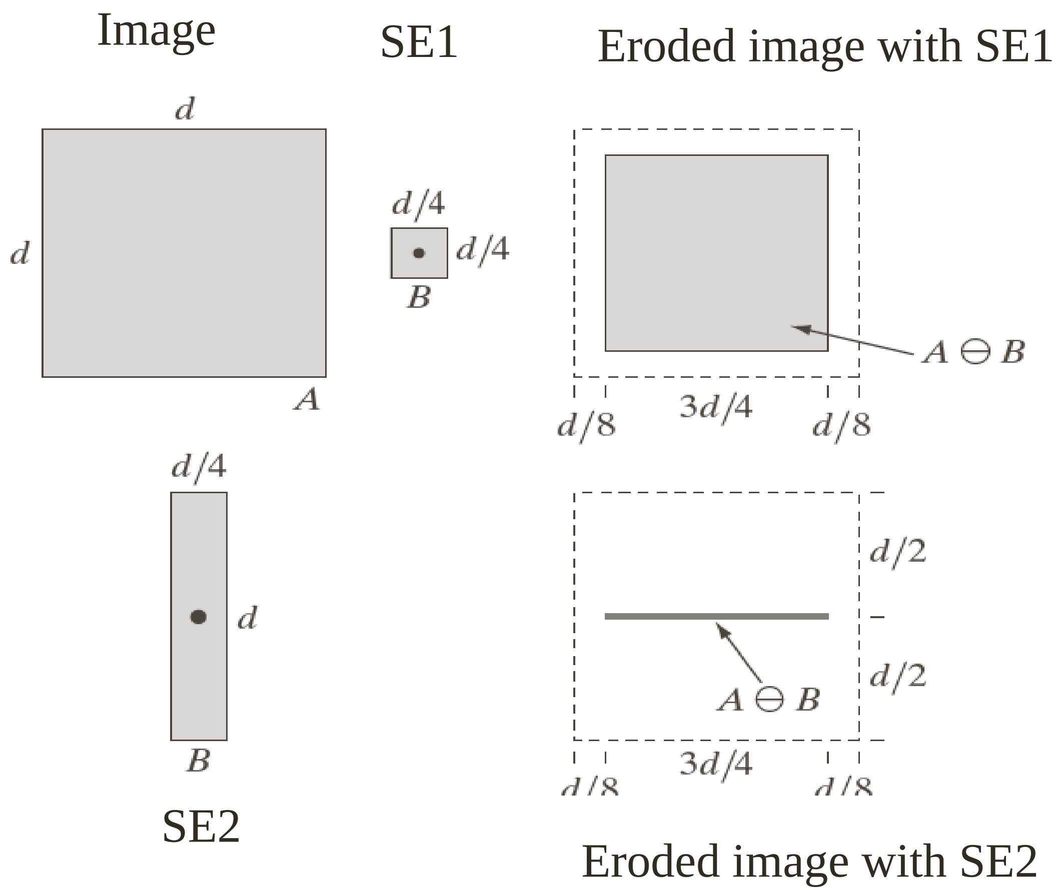
SE2 on Image SE1 Eroded image with SE1



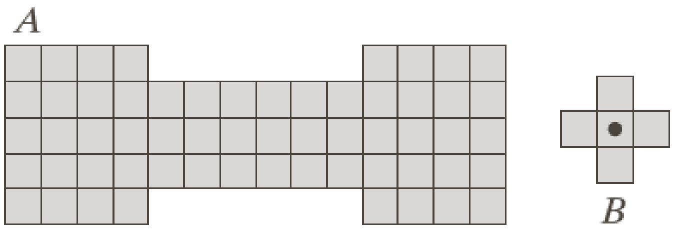
SE2

Eroded image with SE2

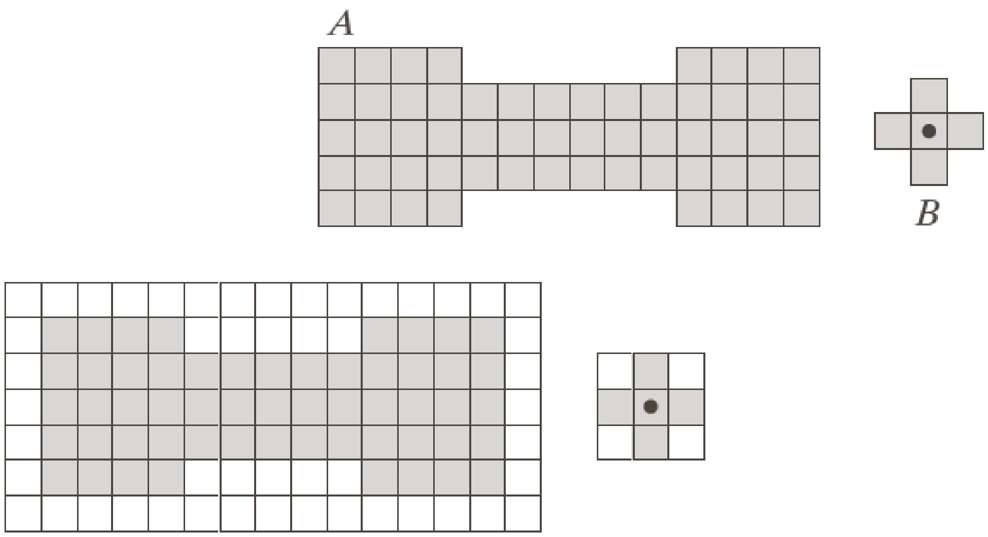
Erosion of A by Structuring Element B



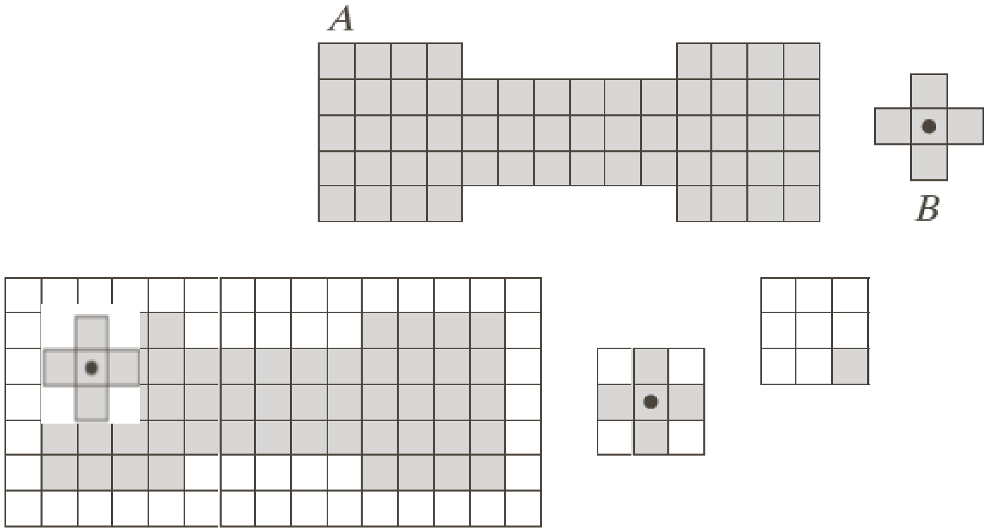
Example: Erosion



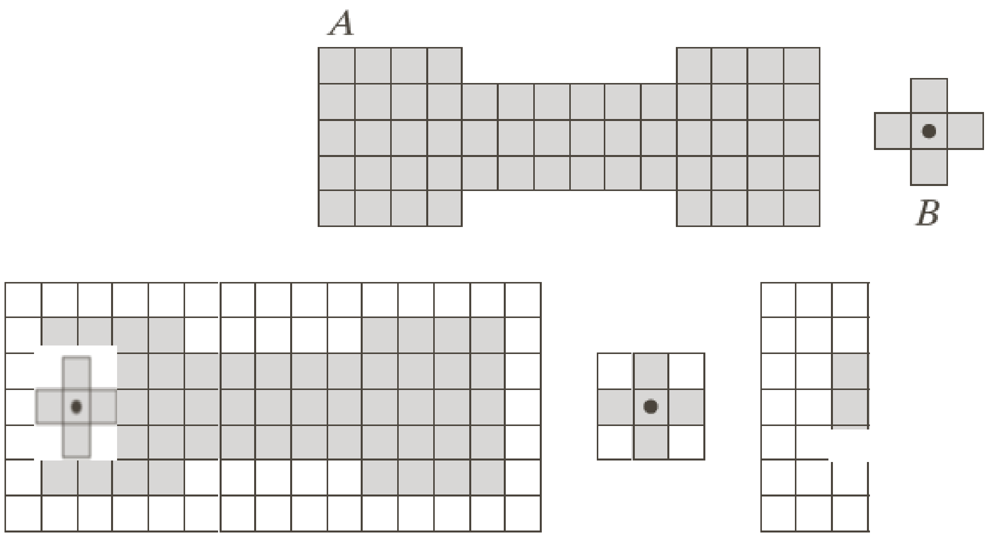
Example: Erosion



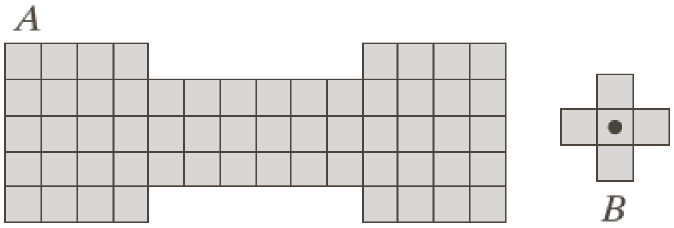
Example: Erosion

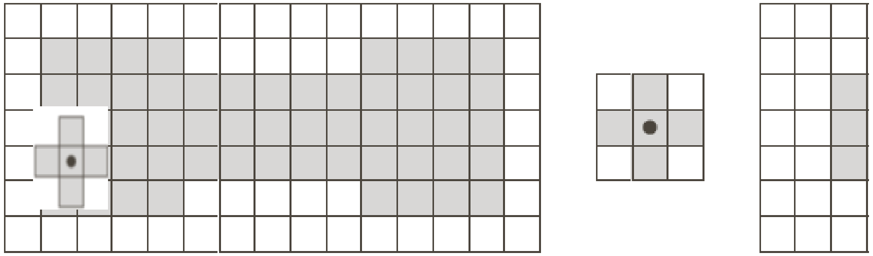


Example: Erosion

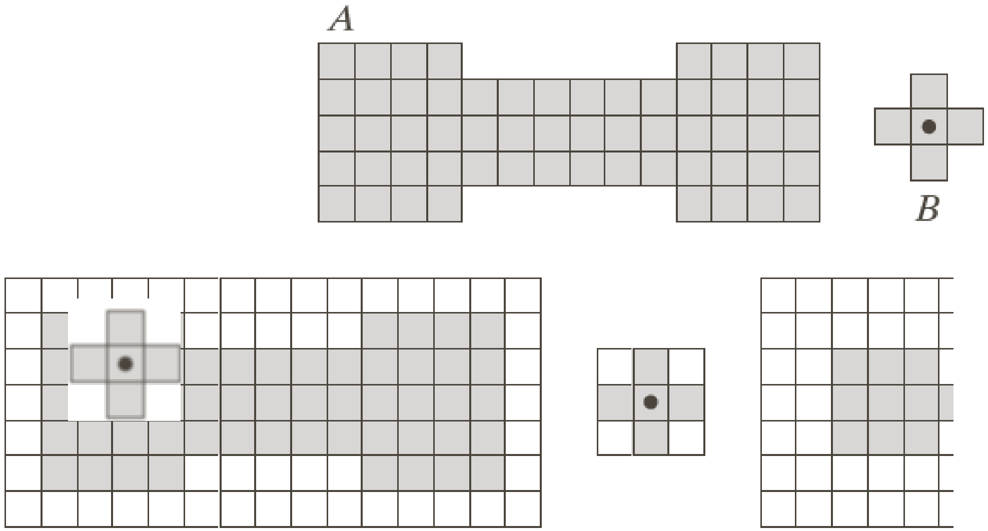


Example: Erosion

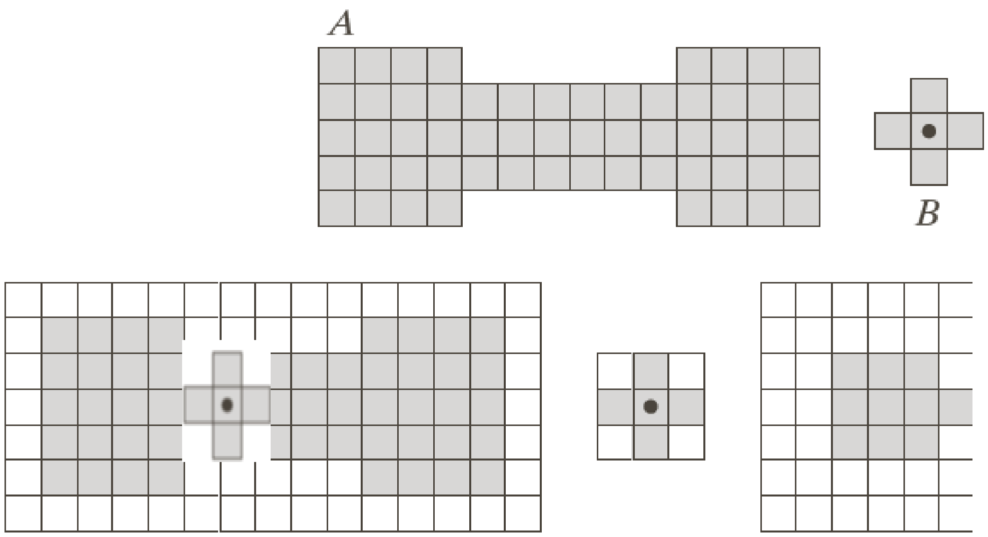




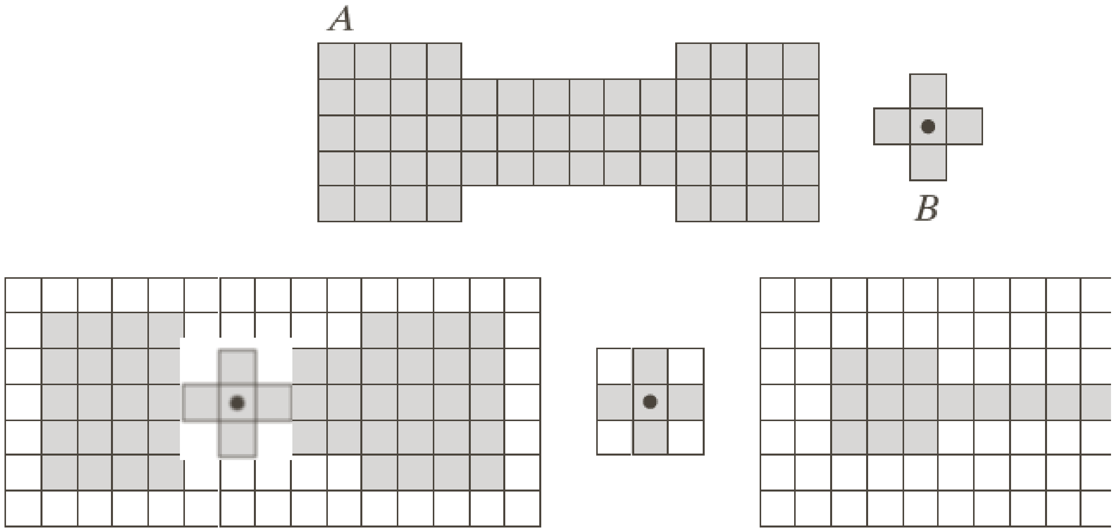
Example: Erosion



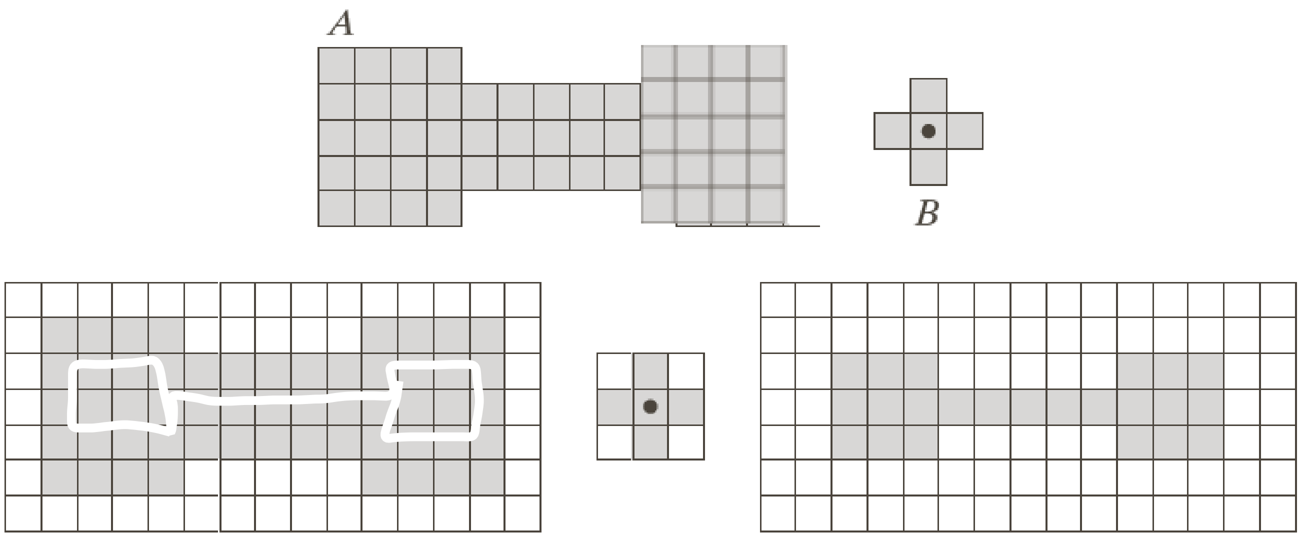
Example: Erosion



Set and Structuring Element



Set and Structuring Element



Pixels of SE and the eroded objects of image have same pixel intensity

Erosion example

Erosion with structuring elements

SE:11×11, white image

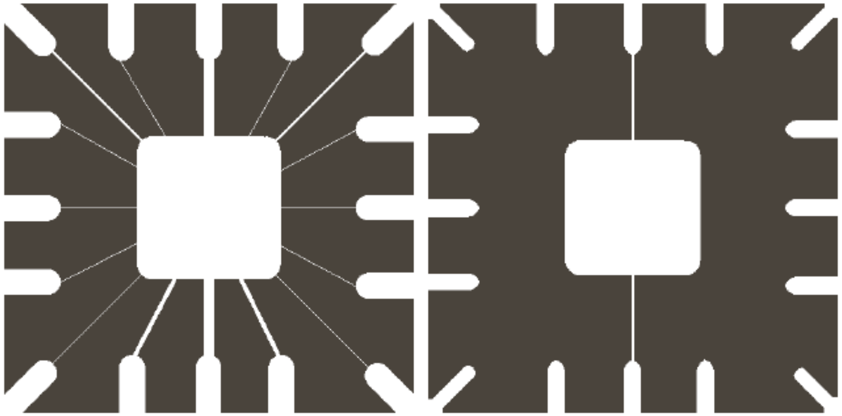
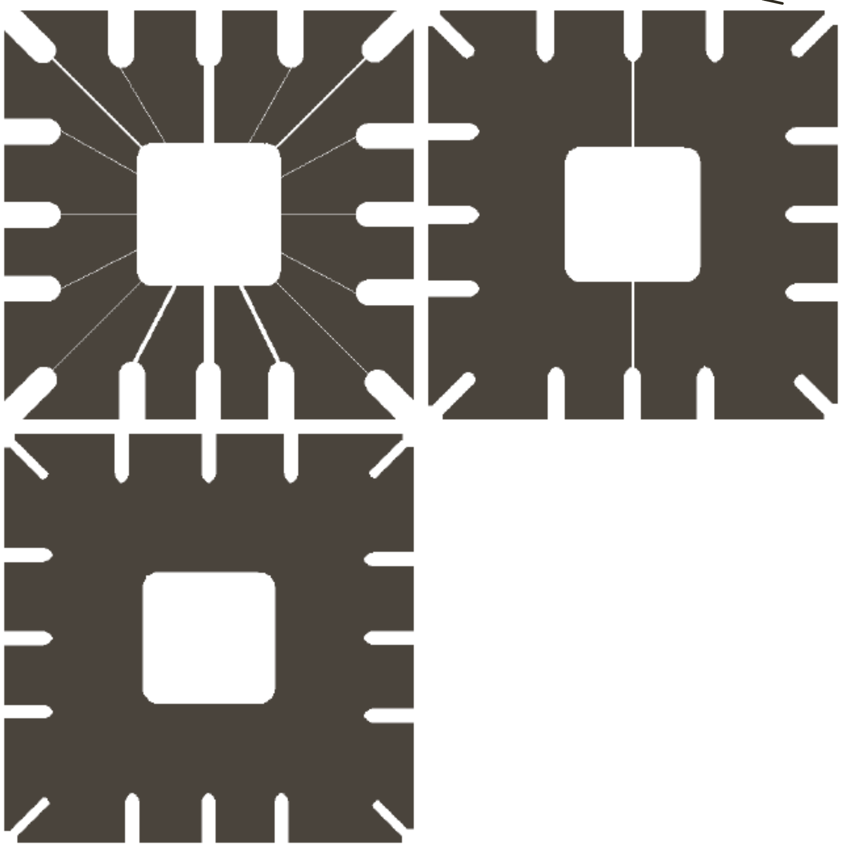


Image details smaller than the structuring element are removed

Erosion with structuring elements

SE:11×11 white image

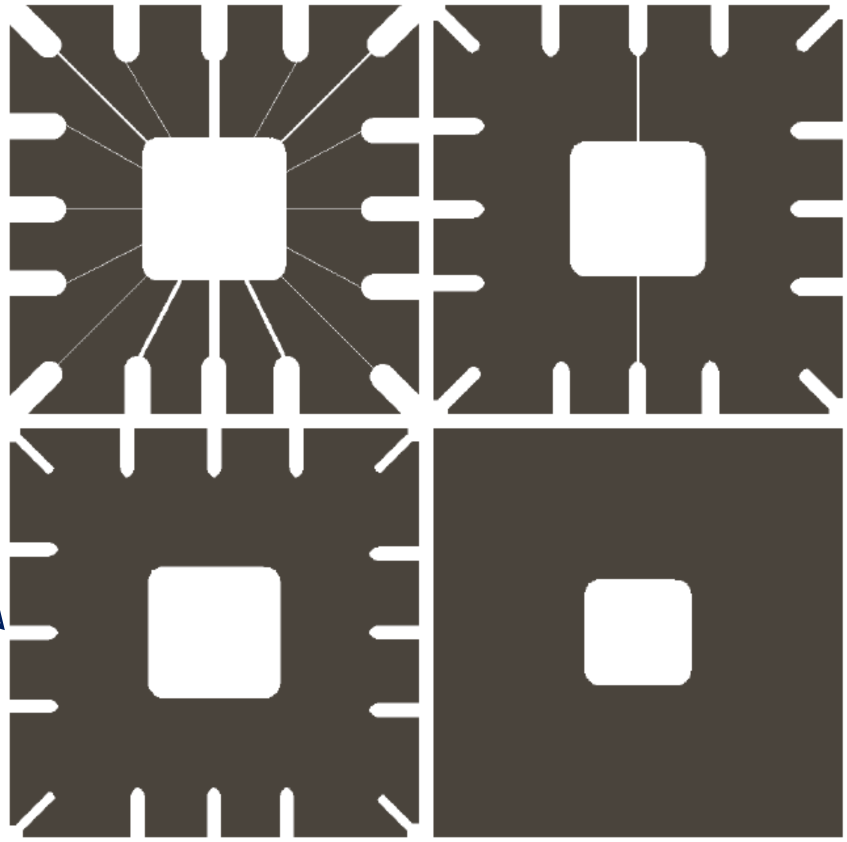


SE:15×15 white image

Image details smaller than the structuring element are removed

Erosion with structuring elements

SE:11×11 white image



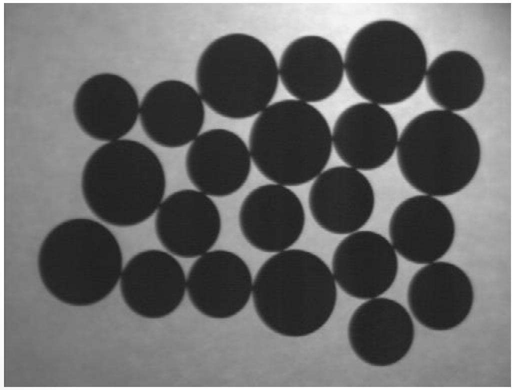
SE:15×15 white image

SE:45×45 white image

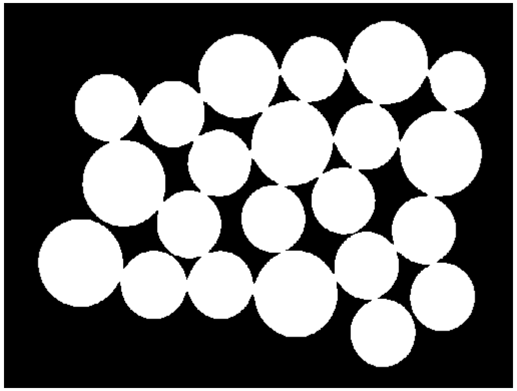
Image details smaller than the structuring element are removed

Counting coins

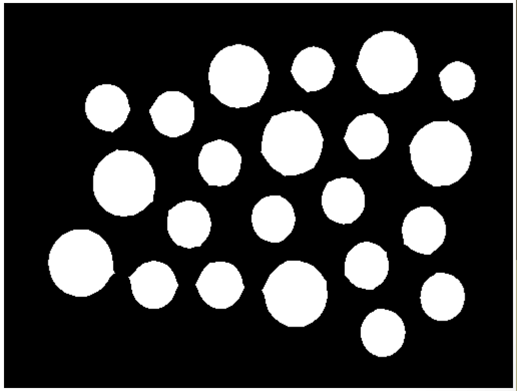
- Counting coins is difficult because they touch each other
- Solution: Binarization using thresholding and Erosion separates them
- Apply Structuring element of circular shape with size smaller than smallest coin



Gray Image



Binary Image



Eroded Binary Image

Dilation

- Dilation is used for expanding an element A by using structuring element B

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

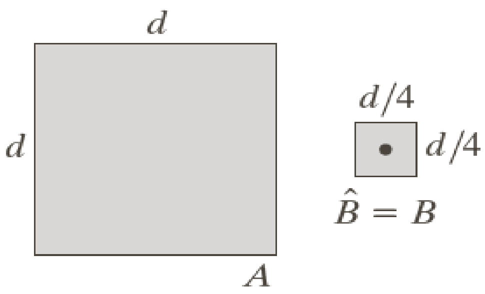
- Based on obtaining the reflection of B about its origin and shifting the reflection by z
- The dilation of A by B is the set of all displacements z, such that reflection of B and A overlap by at least one element

$$A \oplus B = \{z \mid [(\hat{B})_z \cap A] \subset A\}$$

Dilation

Image

SE1

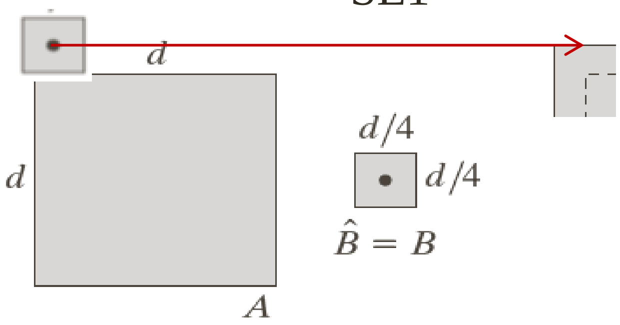


Dilation

Image with SE1

Dilation of Image with SE1

SE1

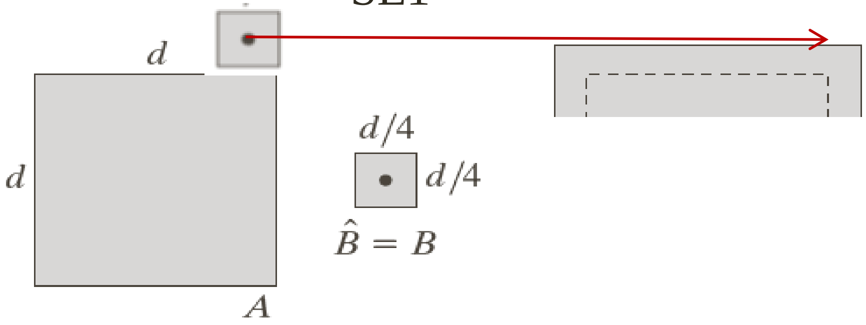


Dilation

Image with SE1

Dilation of Image with SE1

SE1

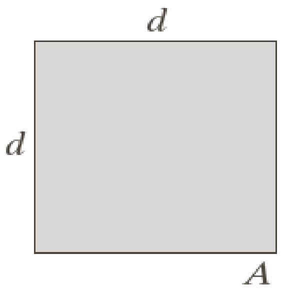


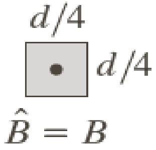
Dilation

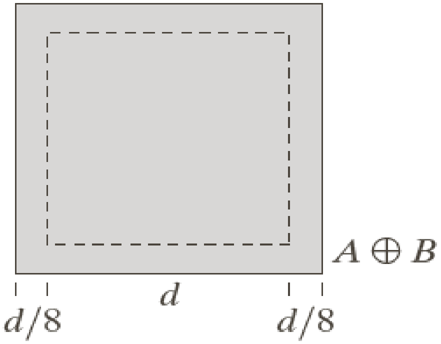
Image

SE1

Dilation of Image with SE1





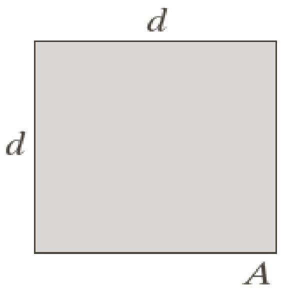


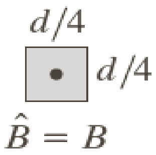
Dilation

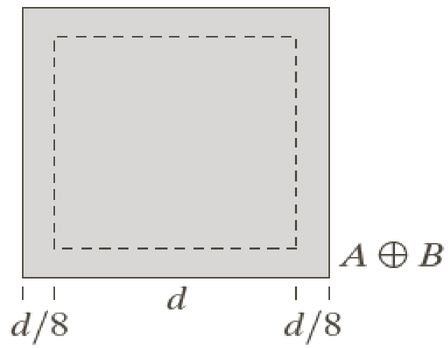
Image

SE1

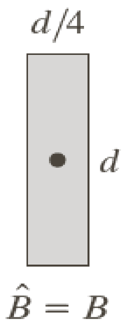
Dilation of Image with SE1





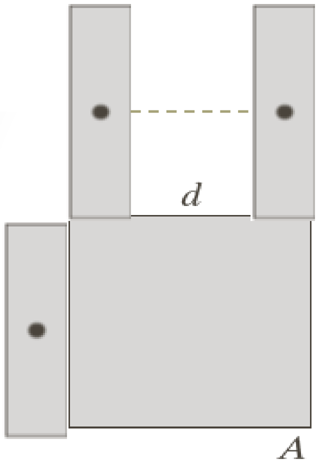


SE2

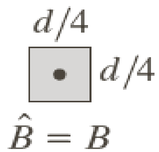


Dilation

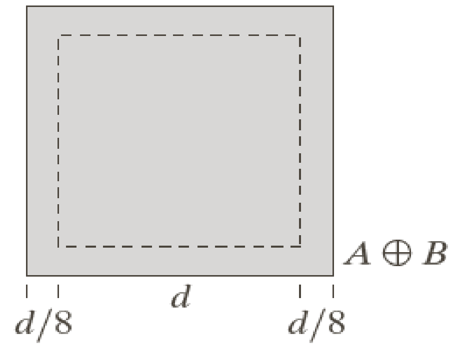
Image with SE2



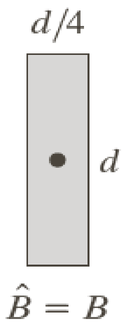
SE1

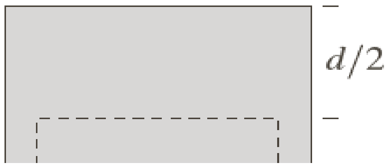


Dilation of Image with SE1



SE2

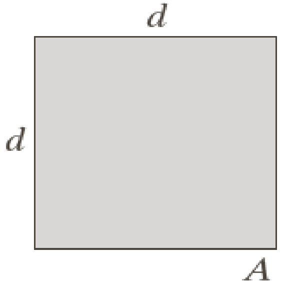




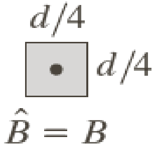
Dilation of Image with SE2

Dilation

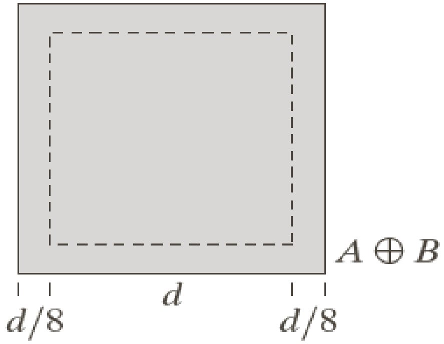
Image



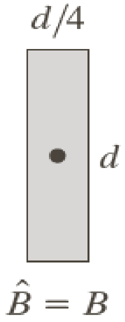
SE1

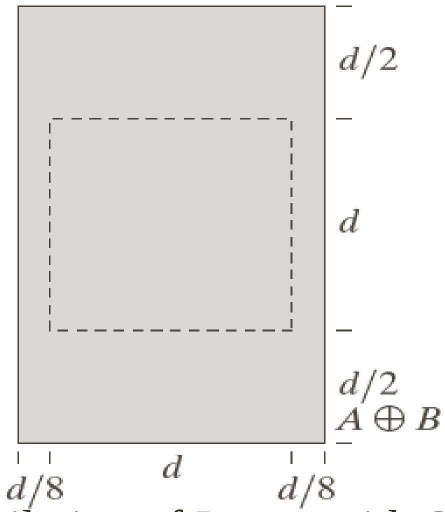


Dilation of Image with SE1



SE2





Dilation of Image with SE2

Example: Dilation

1	1	1
1	1	1
1	1	1

Structuring element

Example: Dilation

1	1	1
1	1	1
1	1	1

Structuring element

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

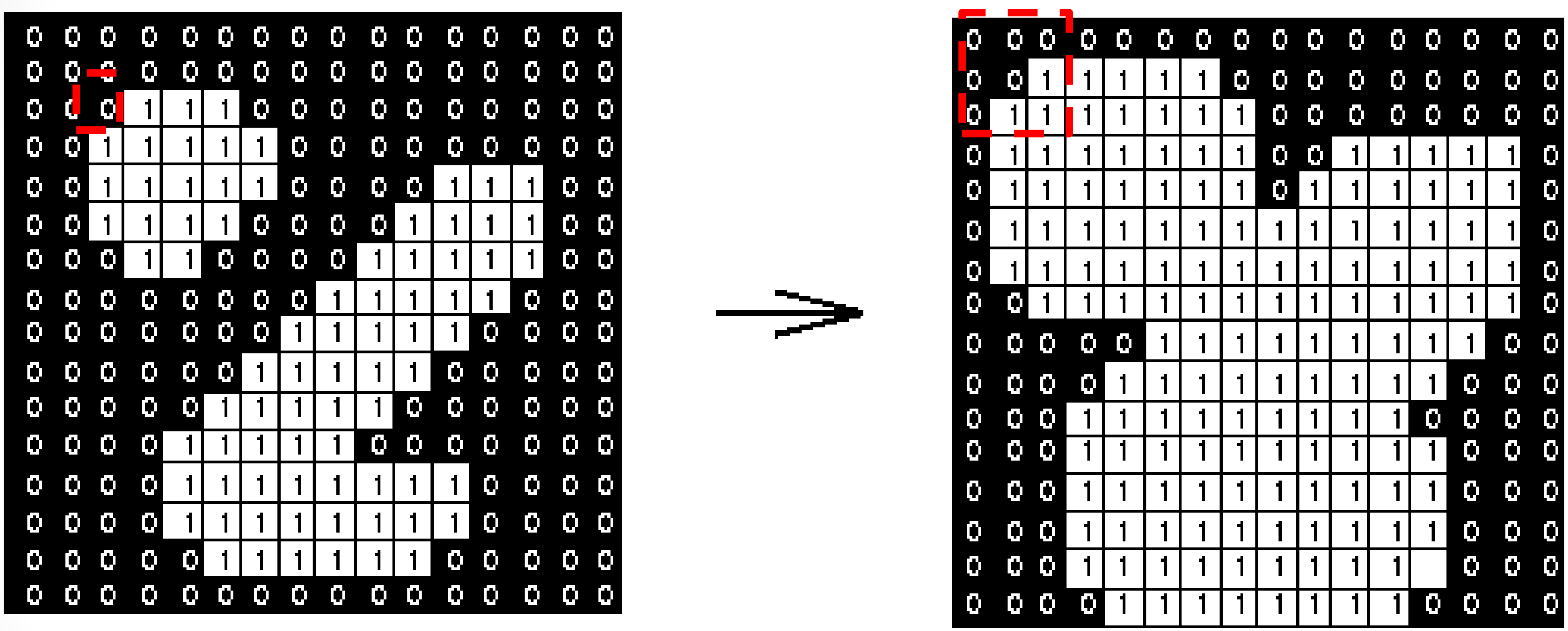
Example: Dilation

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

1	1	1
1	1	1
1	1	1

Structuring element

Example: Dilation

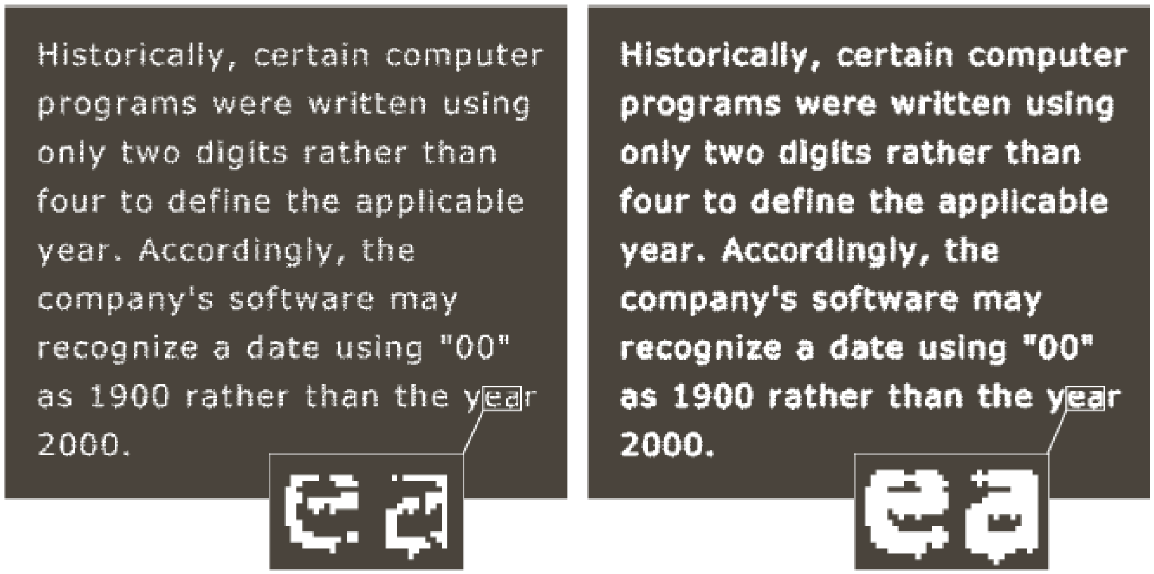


append one white pixel to border pixels of object

1	1	1
1	1	1
1	1	1

Structuring element

Dilation



0	1	0
1	1	1
0	1	0

Erosion and Dilation

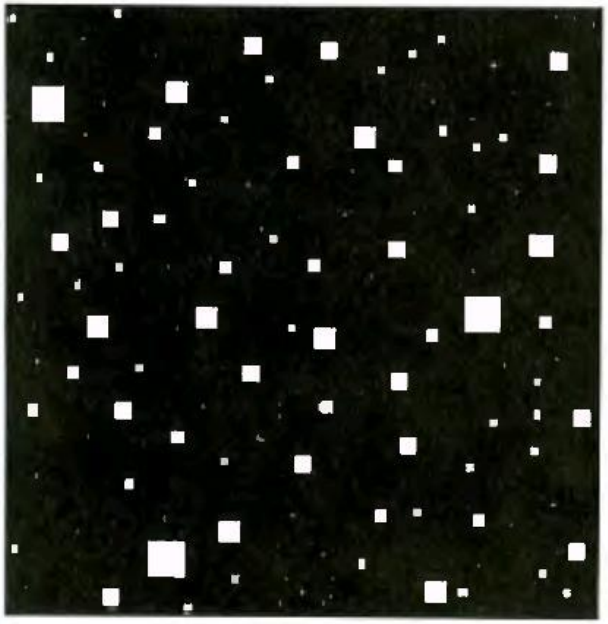


Image with squares of 1,3,5,7,9 and 15 pixels

Erosion

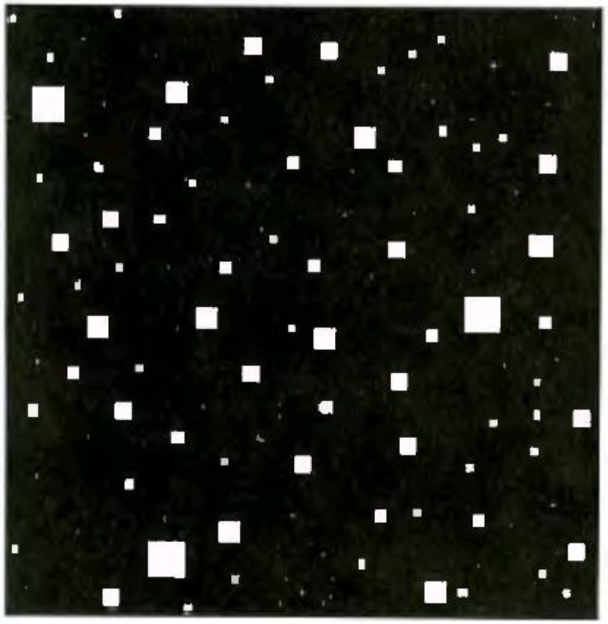


Image with squares of 1,3,5,7,9 and 15 pixels

Erosion with square SE of size 13 pixels

Erosion and Dilation

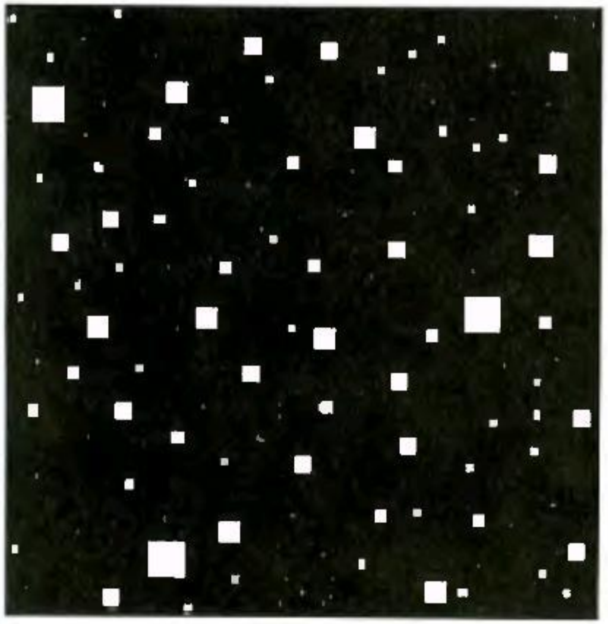
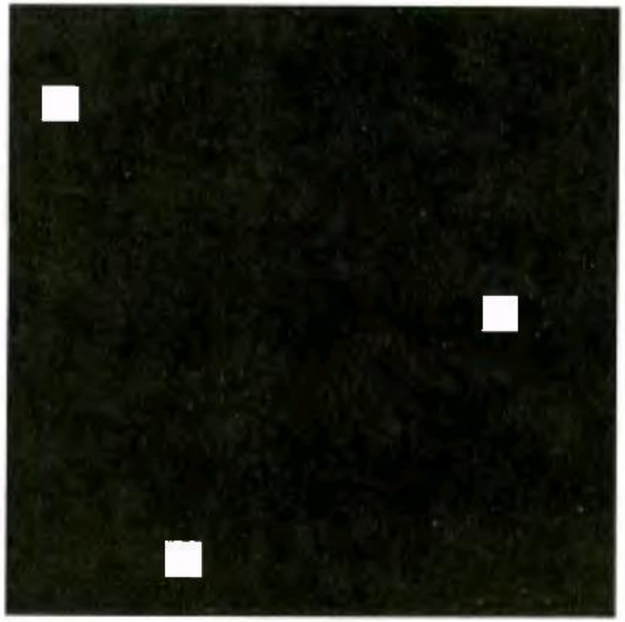


Image with squares of 1,3,5,7,9 and 15 pixels

Erosion with square SE of size 13 pixels



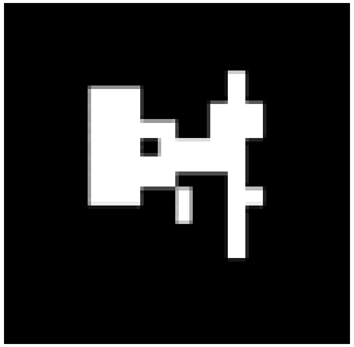
Dilationwith square SE of size 13 pixels

Duality of Erosion and dilation

$$(A\ e\ B)^c = (A^c\ d\ \hat{B})$$

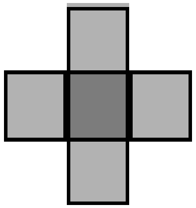
$$(A\ d\ B)^c = (A^c\ e\ \hat{B})$$

Duality of dilation and erosion

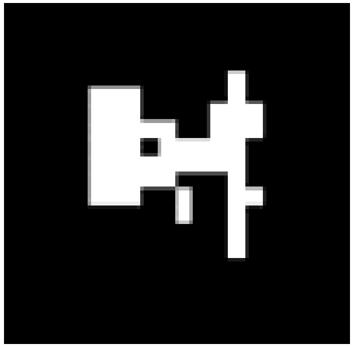


A

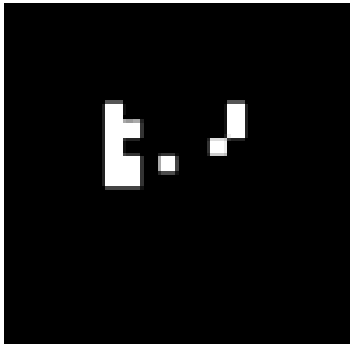
$B = \hat{B}$



Duality of dilation and erosion

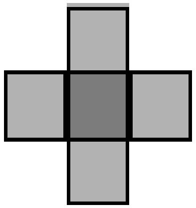


A

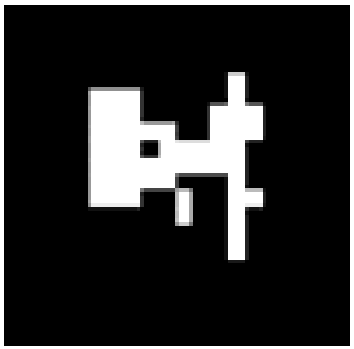


$A \ominus B$

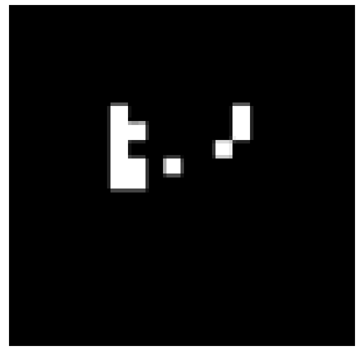
$$B = \hat{B}$$



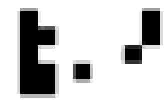
Duality of dilation and erosion



A

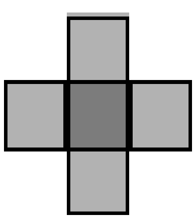


A⊖B

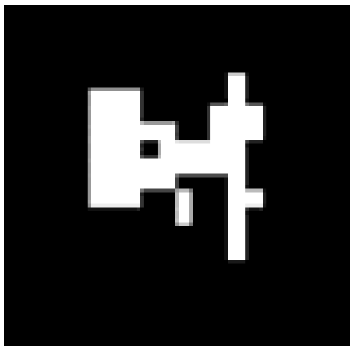


$(A \ominus B)^C$

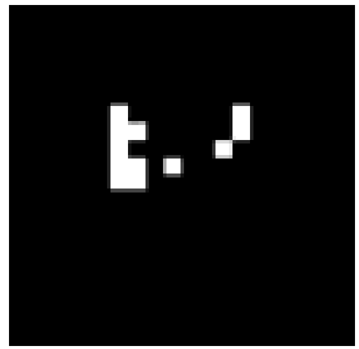
$$B = \hat{B}$$



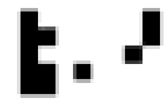
Duality of dilation and erosion



A

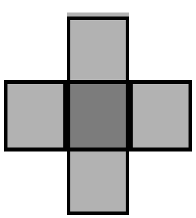


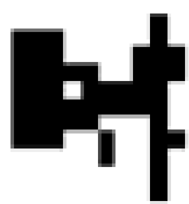
$A \ominus B$



$(A \ominus B)^C$

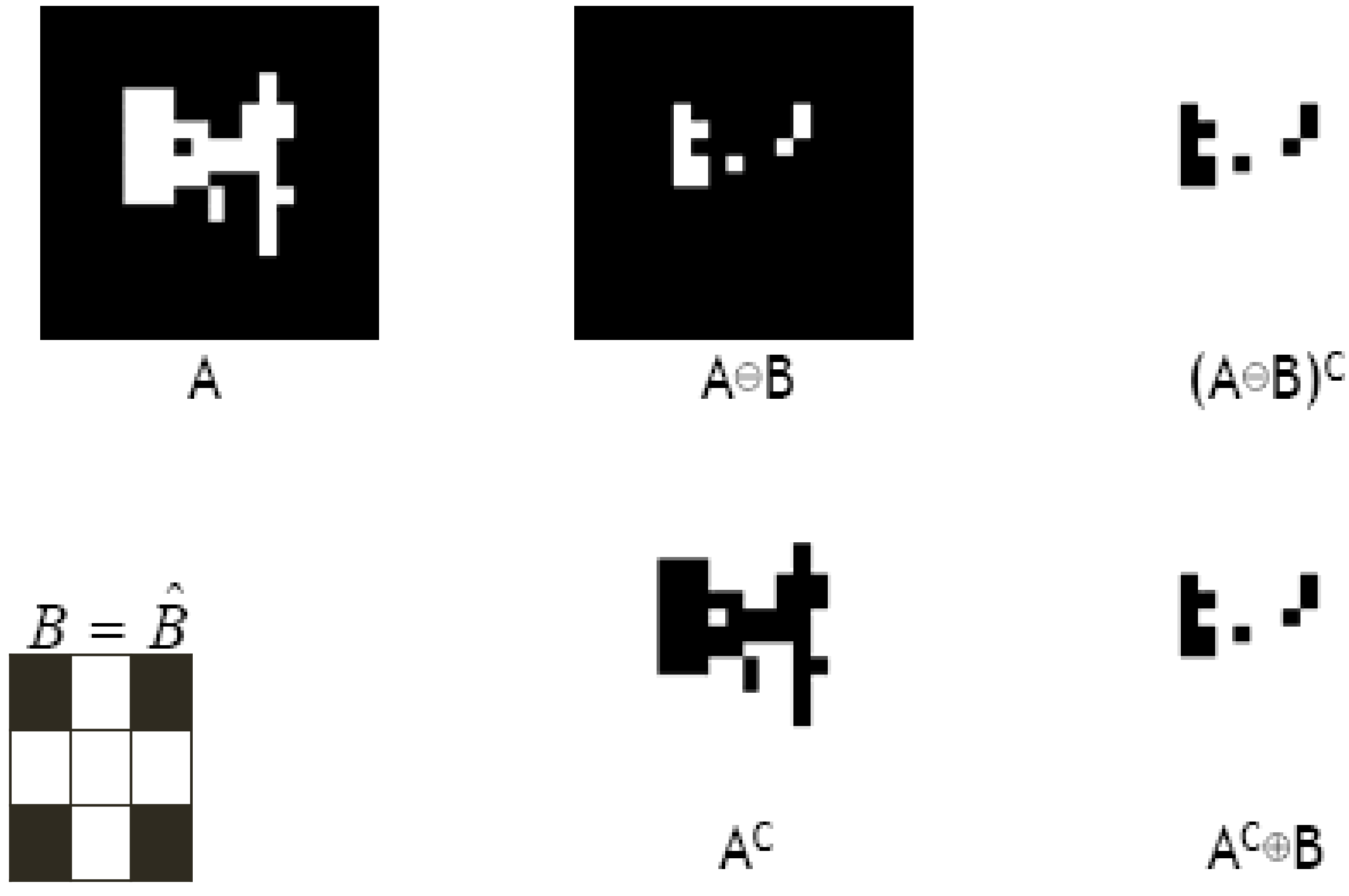
$B = \hat{B}$





A^C

Duality of dilation and erosion



Opening

- erosion followed by dilation
- denoted by ∘

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

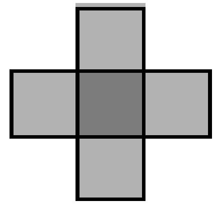
- eliminates protrusions
- breaks necks
- smoothens contour

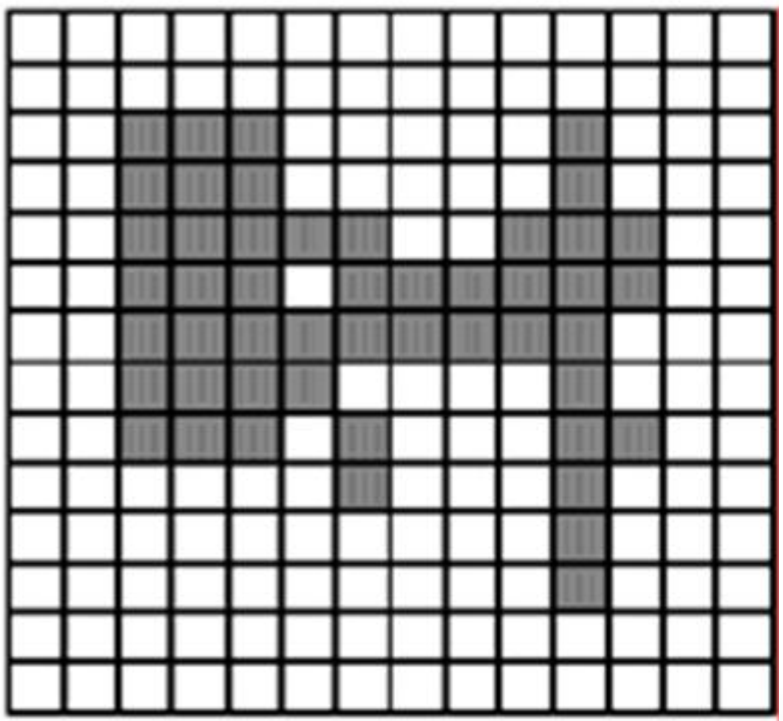
Opening

- once an image is opened with a certain SE
- subsequent applications of the opening algorithm with the same SE will not cause any effect on the image.
- Mathematically,

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

Erode

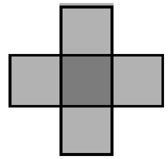
B= 

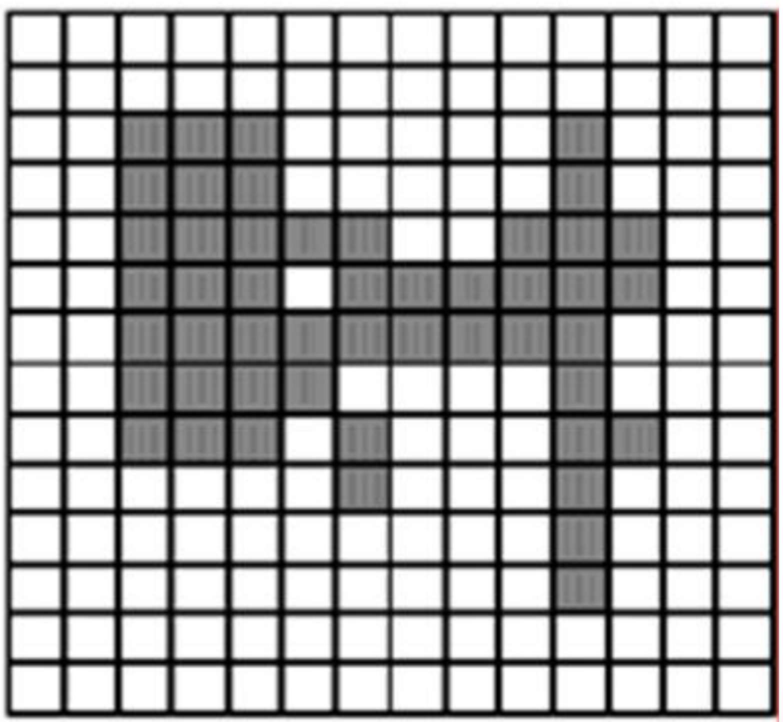


A

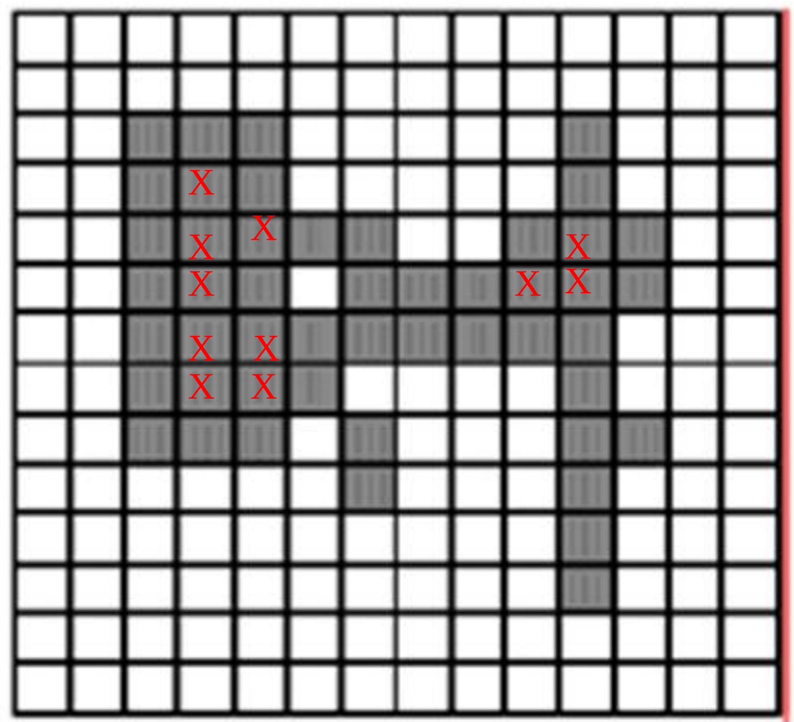
A e B

Erode

B= 



A

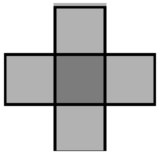


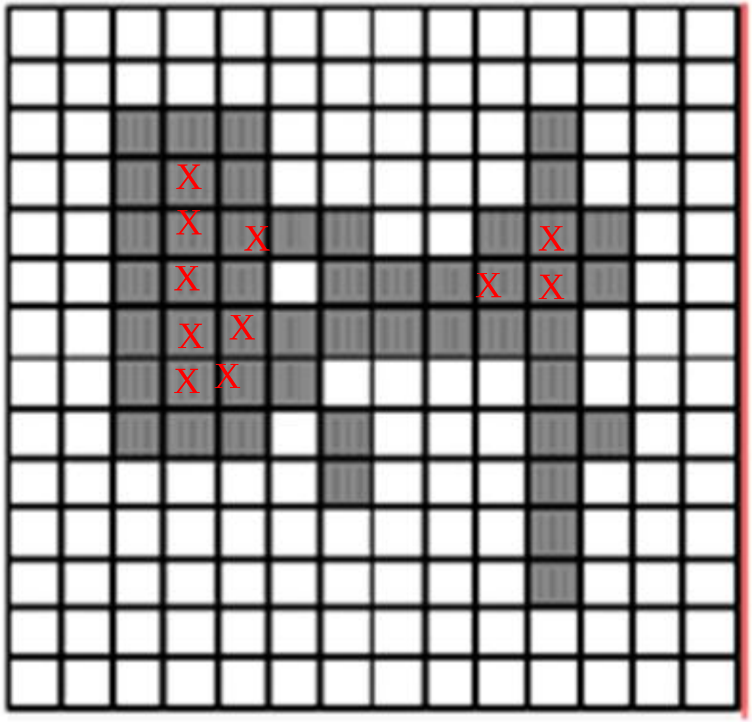
A e B

Red pixels are of eroded image

Dilate Eroded Image

B=

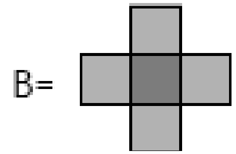


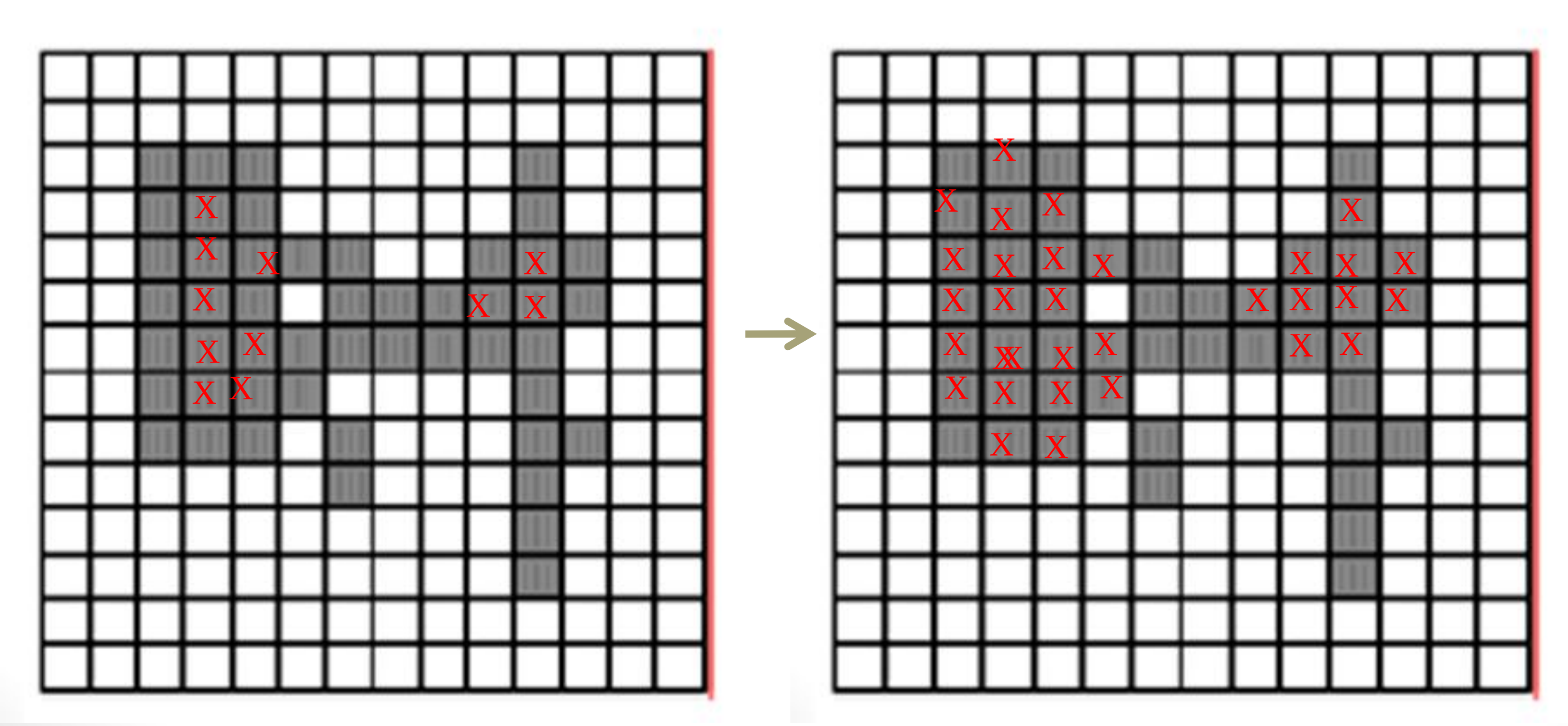


A e B

(A e B)d B

Dilate Eroded Image

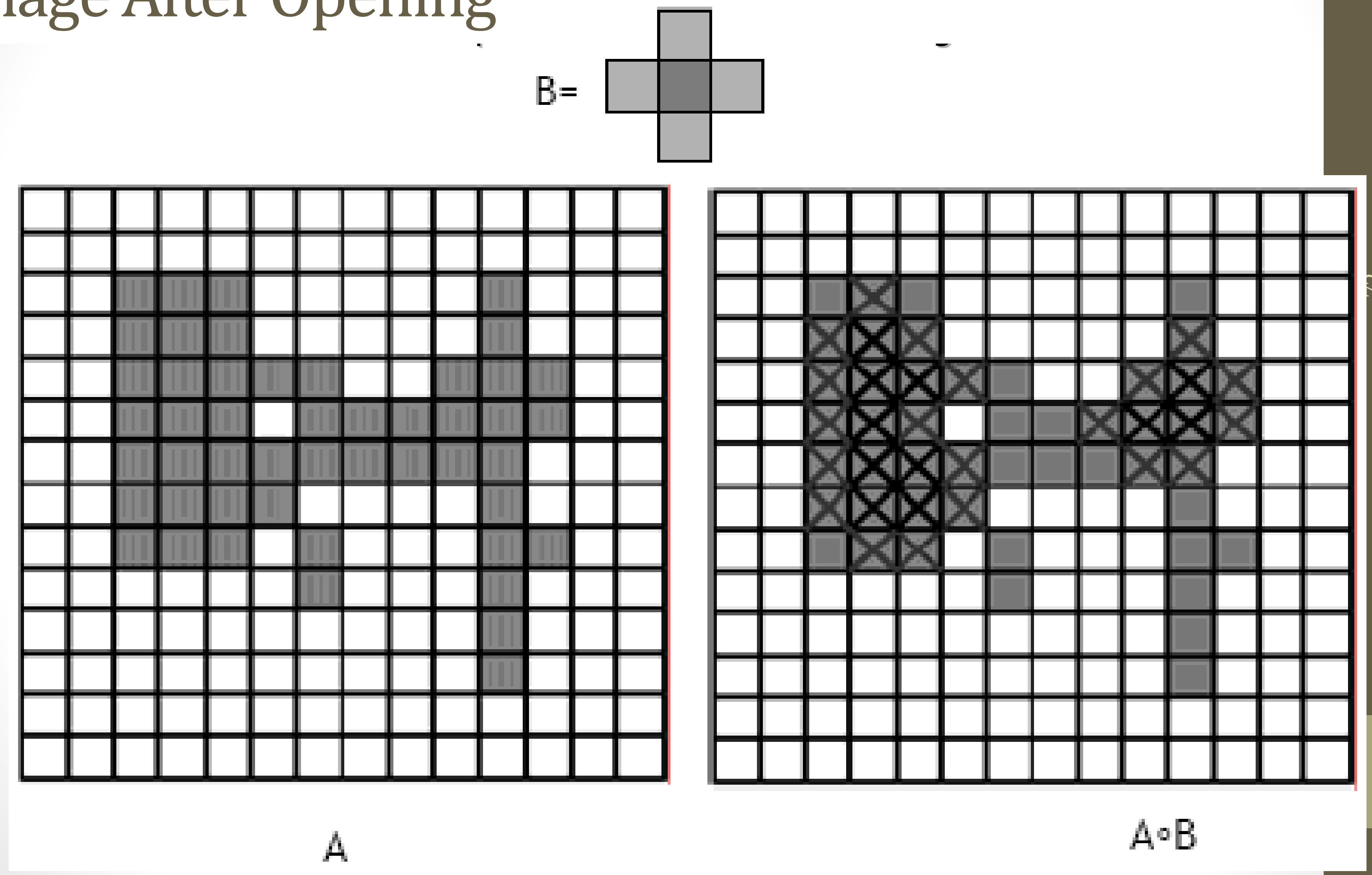




A e B

(A e B)d B

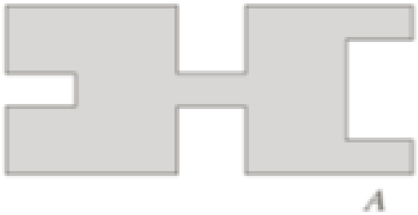
Image After Opening



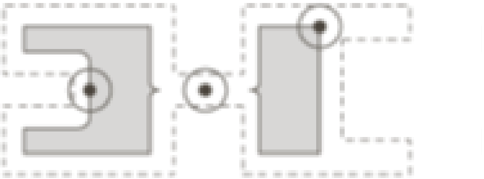
Opening

- typically used to remove thin protrusions from objects
- and to open up a gap between objects connected by a thin bridge
- without shrinking the objects
- Because after erosion dilation is used
- Also causes a smoothening of the object’s boundary

Opening

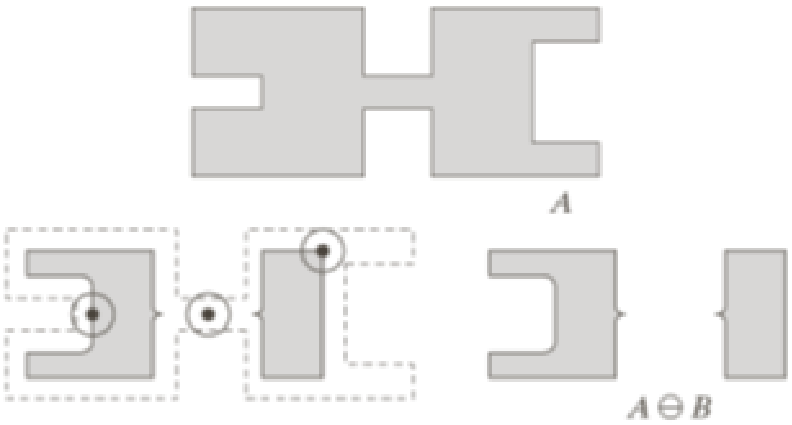


Erosion

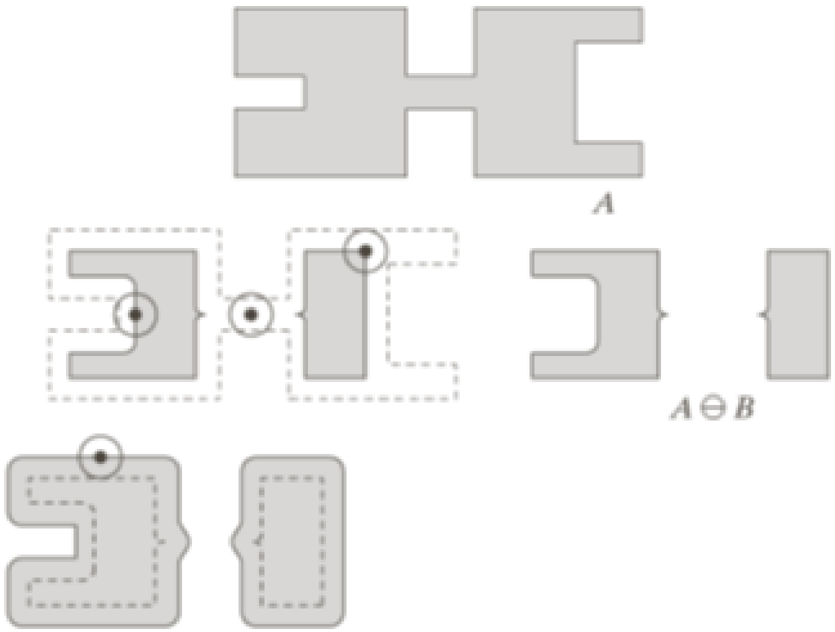


Opening

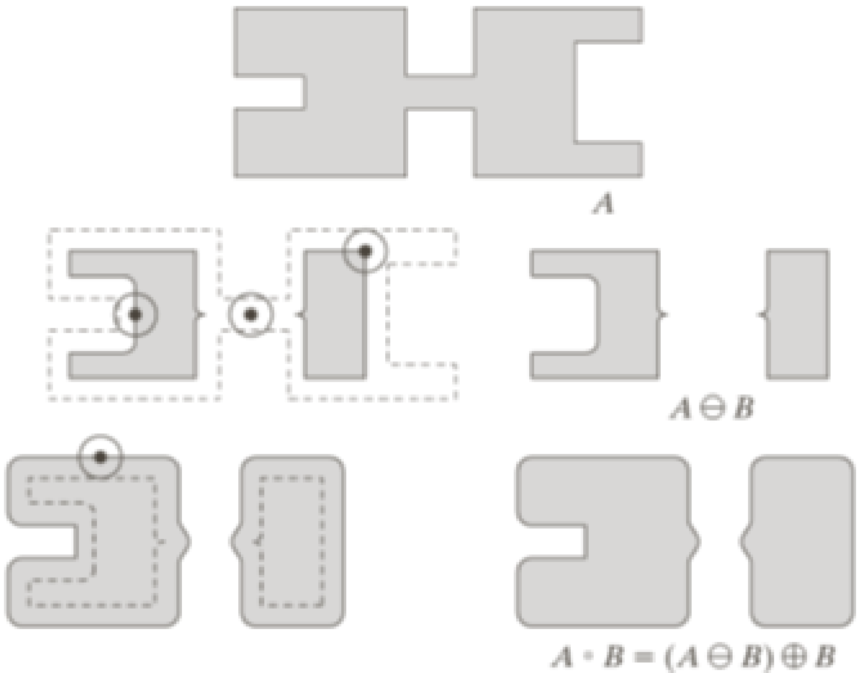
Erosion



Opening

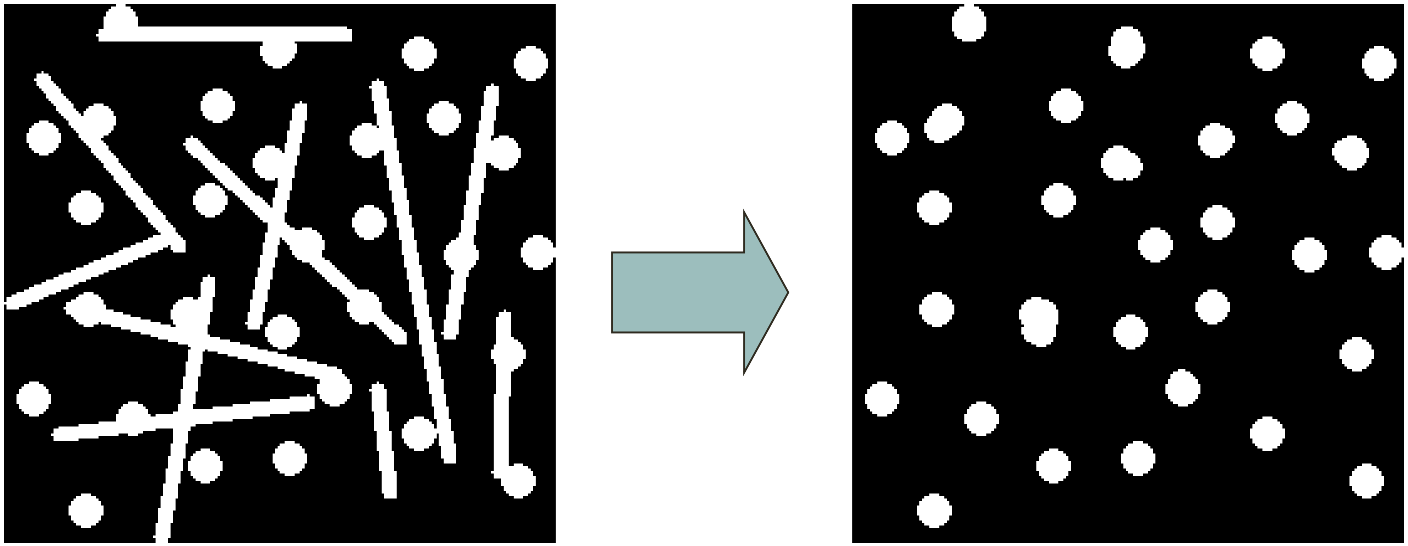


Opening



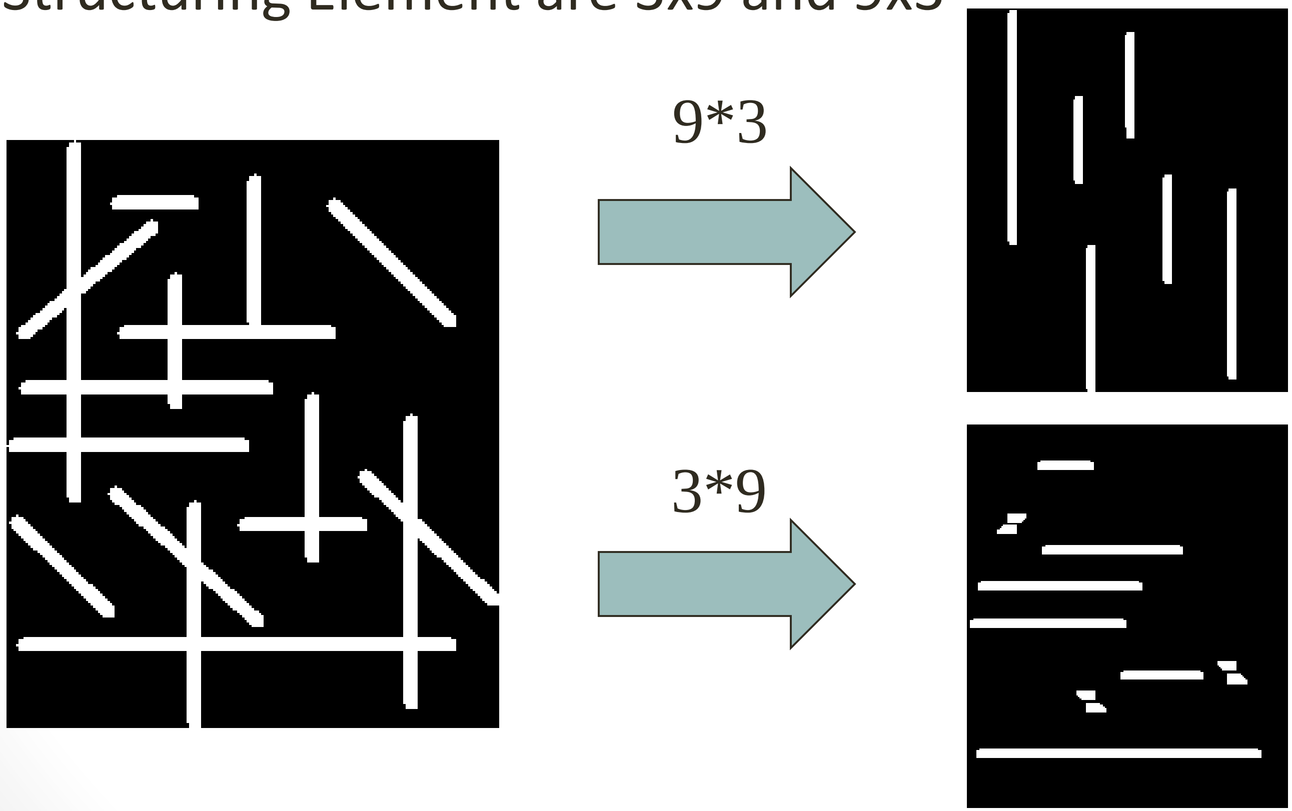
opening

- Opening with a 11 pixel diameter disc



opening

- Assume that image has vertical bars of size greater than 3x9 and 9x3
- Structuring Element are 3x9 and 9x3



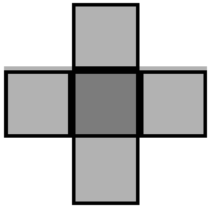
Closing

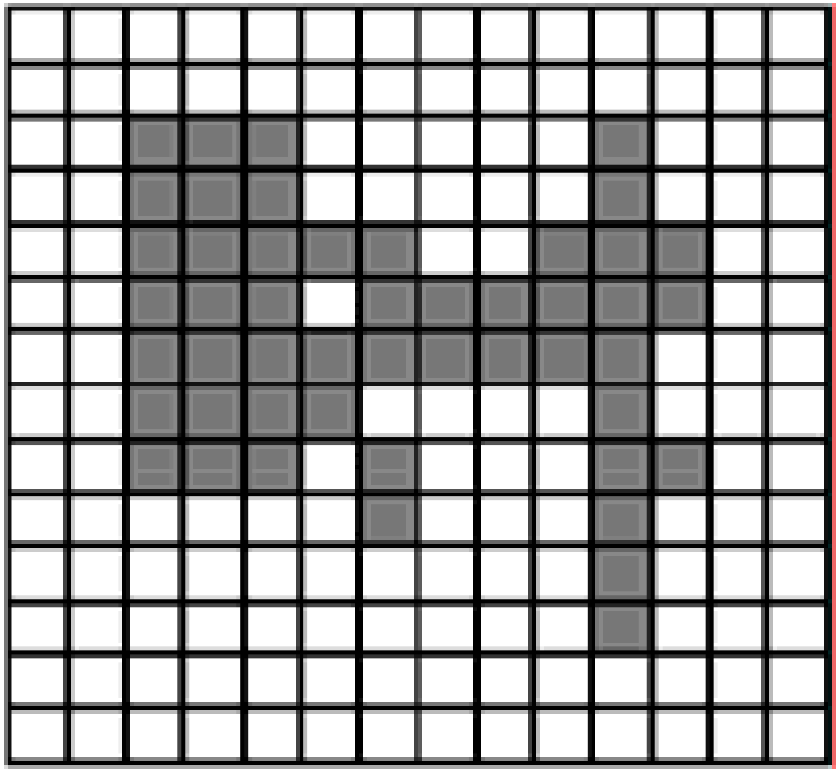
- dilation followed by erosion

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

- smooth contour
- fuse narrow breaks and long thin gulfs
- eliminate small holes
- fill gaps in the contour

Closing

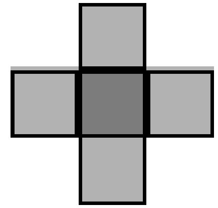
B=




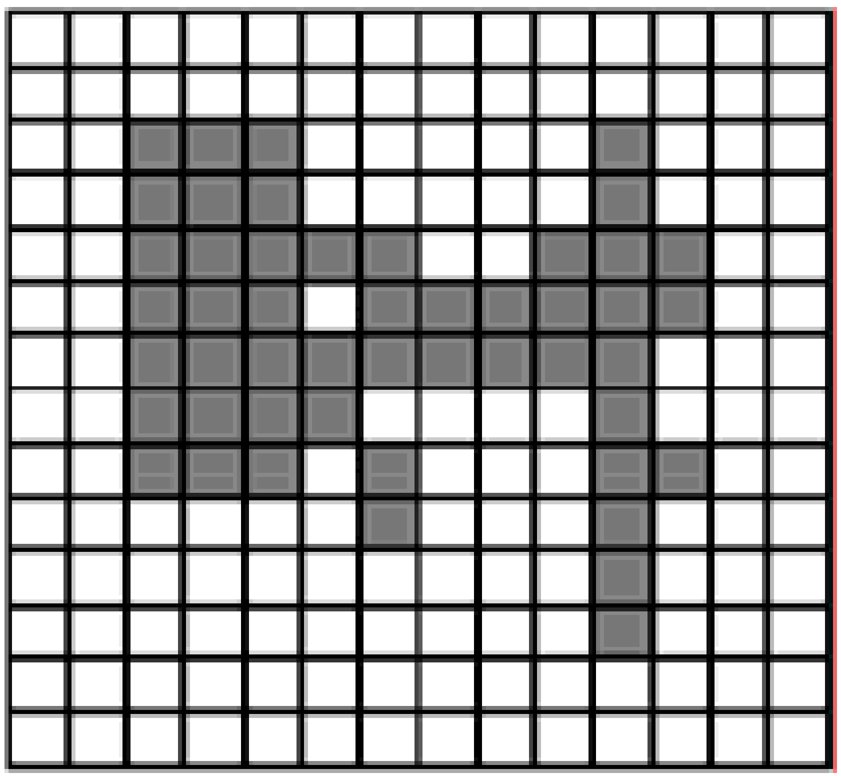
A

A • B

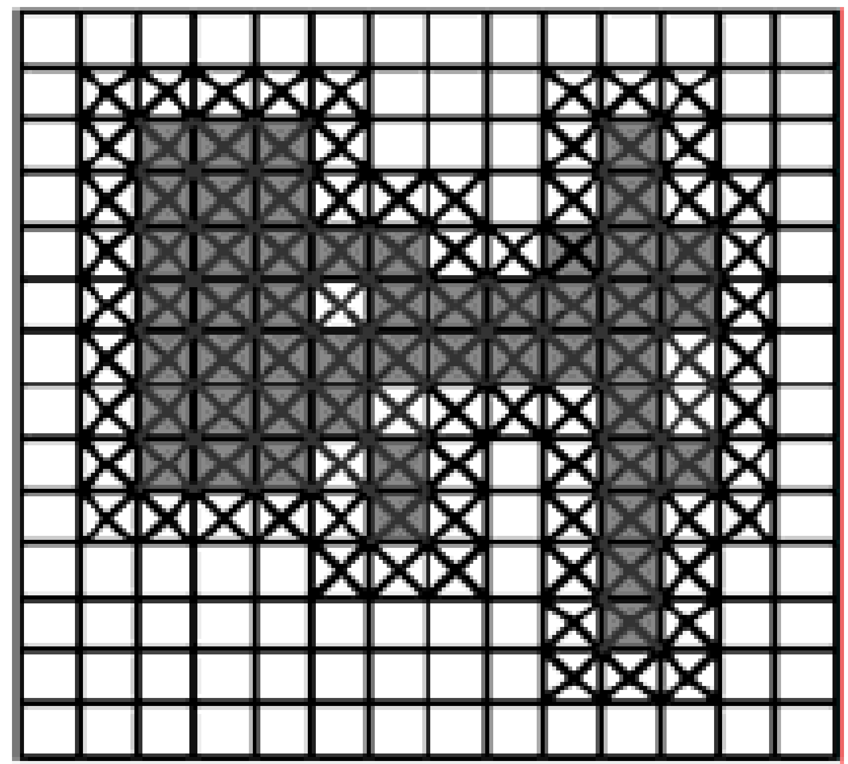
Closing

B= 

- Dilation

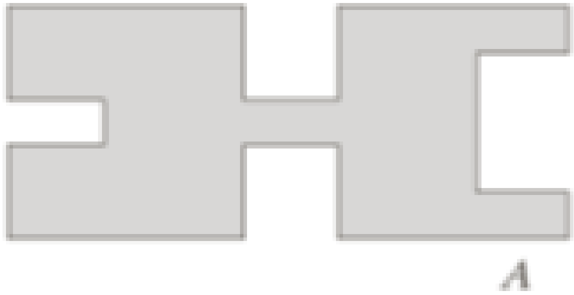


A

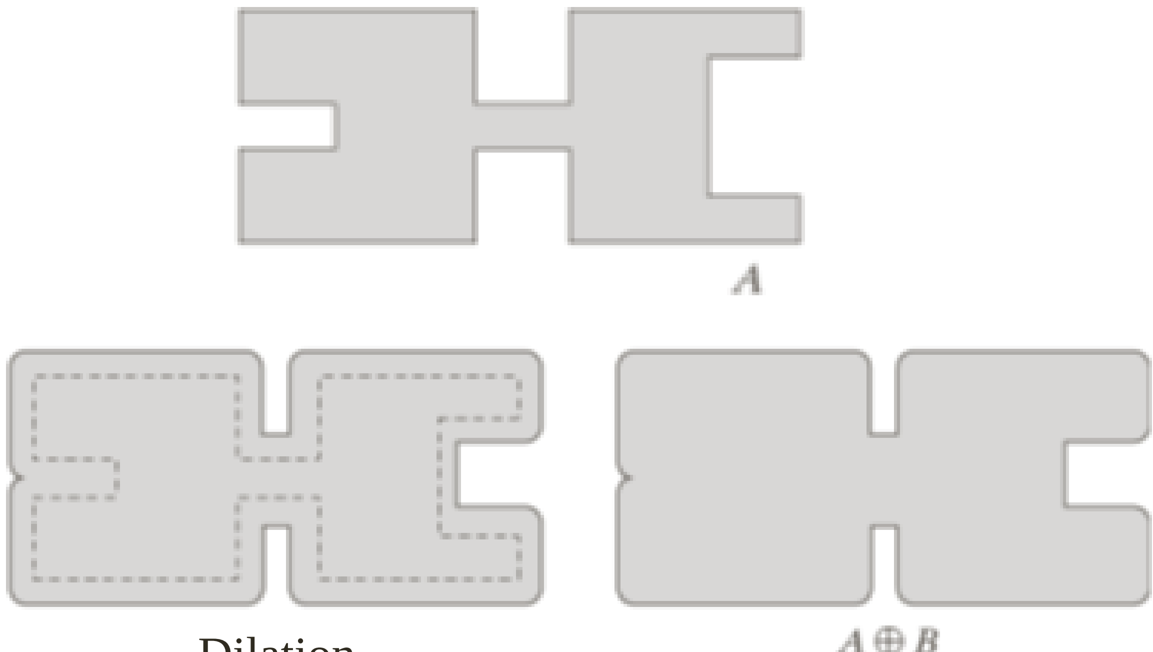


A • B

Closing

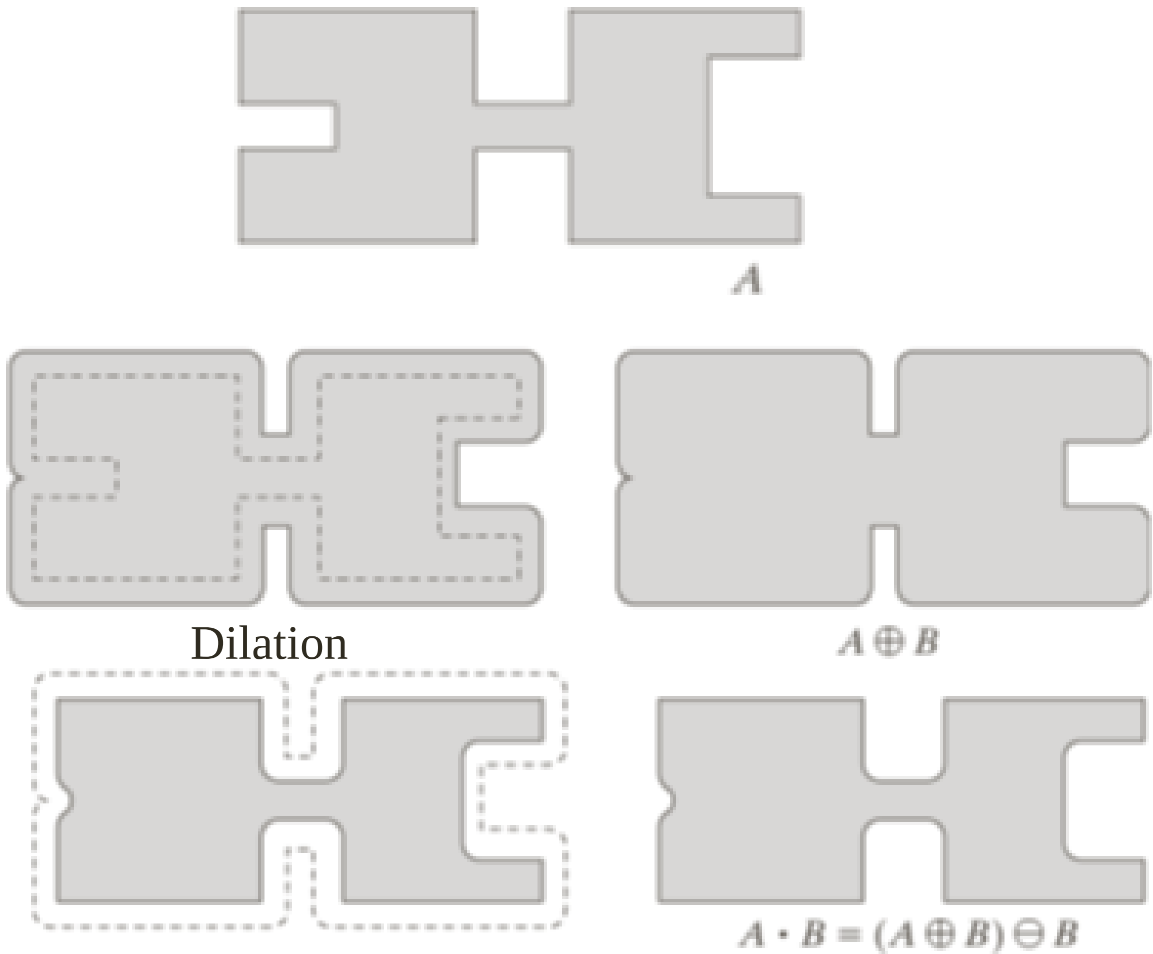


Closing



Dilation

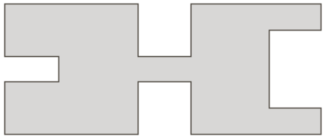
Closing



Dilation

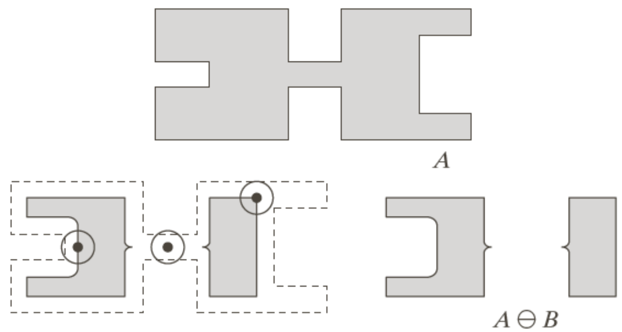
Erosion of dilate image

Erosion, opening, dilation and closing

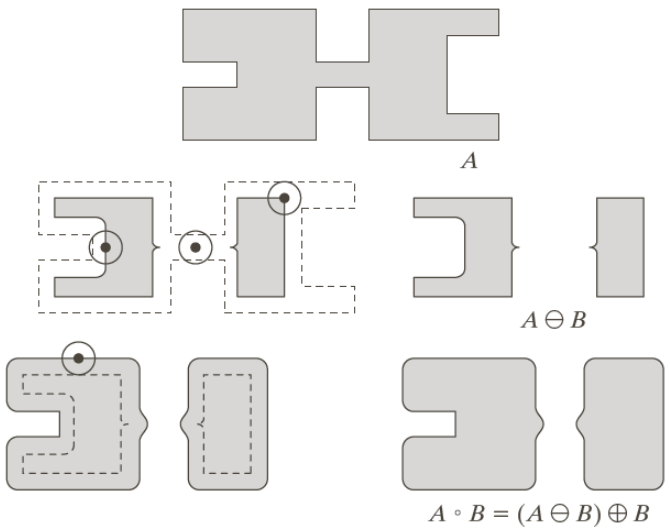


A

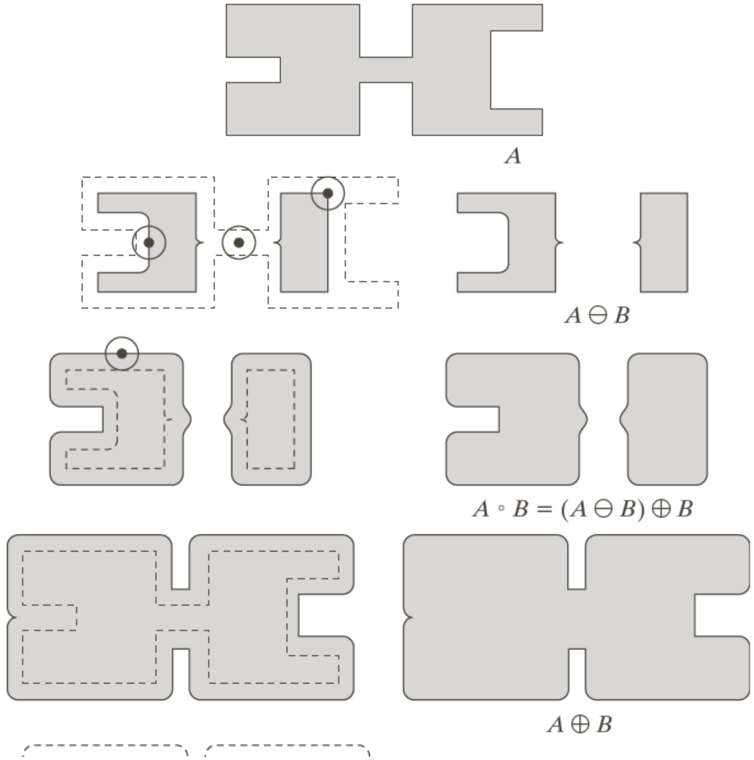
Erosion, opening, dilation and closing



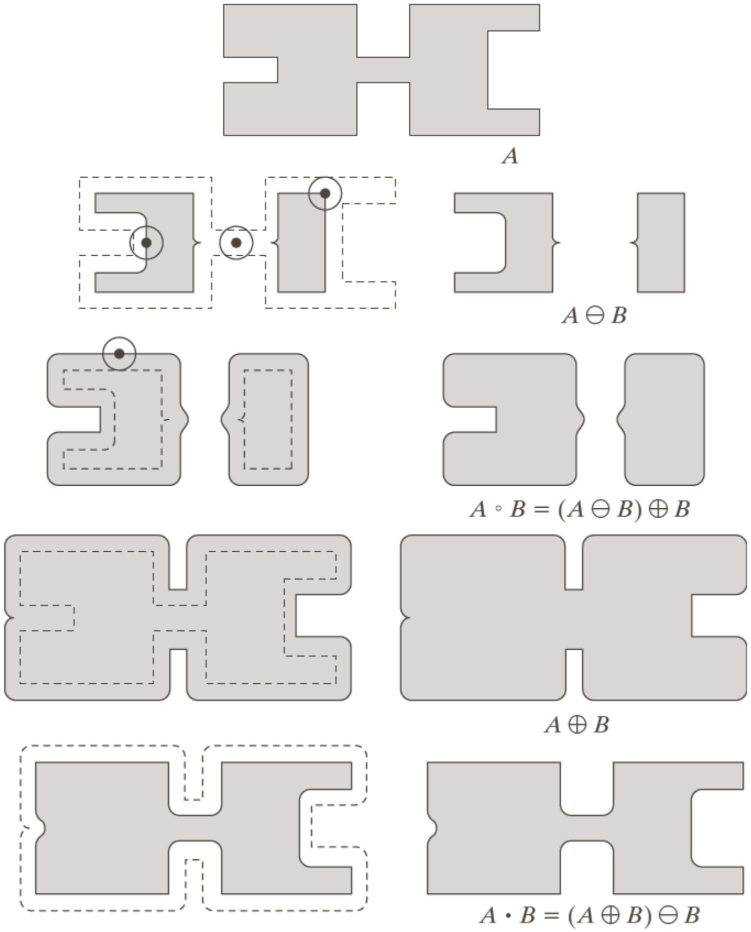
Erosion, opening, dilation and closing



Erosion, opening, dilation and closing



Erosion, opening, dilation and closing



Erosion, opening, dilation and closing

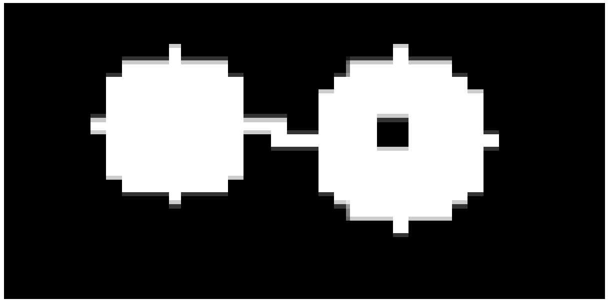
$$(A\ e\ B)^c = (A^c\ d\ \hat{B})$$

$$(A\ d\ B)^c = (A^c\ e\ \hat{B})$$

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

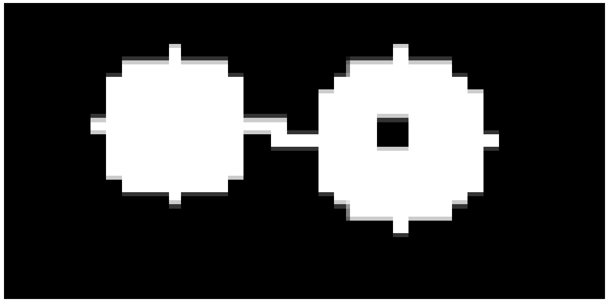
$$(A\ o\ B)^c = (A^c\ c\ \hat{B})$$

Example: opening & closing



A

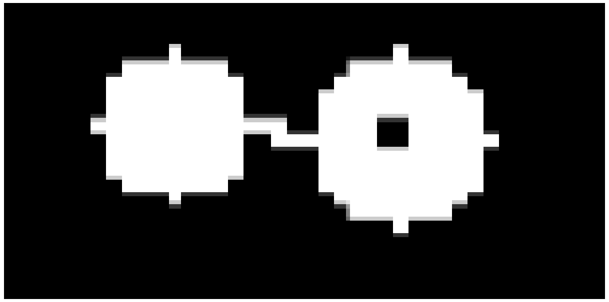
Example: opening & closing



A

opening of A
→removal of small protrusions, thin connections, ...

Example: opening & closing



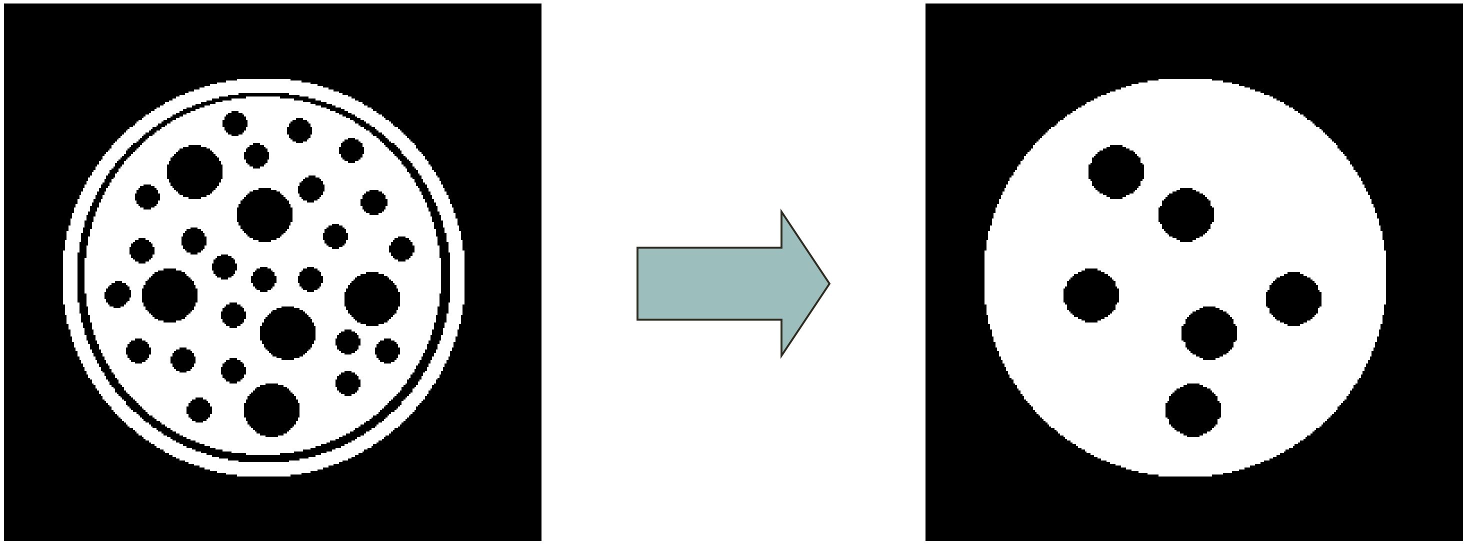
A

opening of A
→removal of small protrusions, thin connections, …

closing of A
→ removal of holes

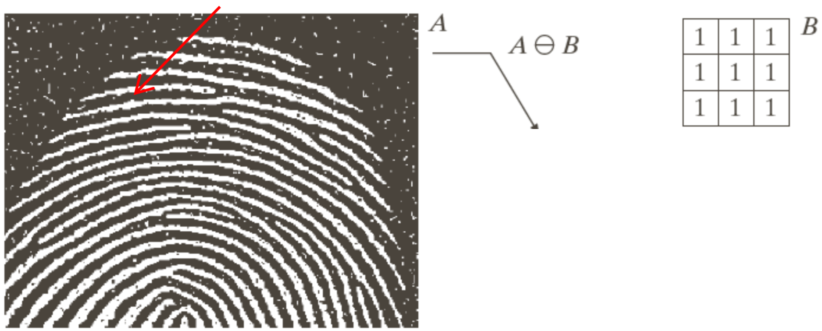
Closing

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



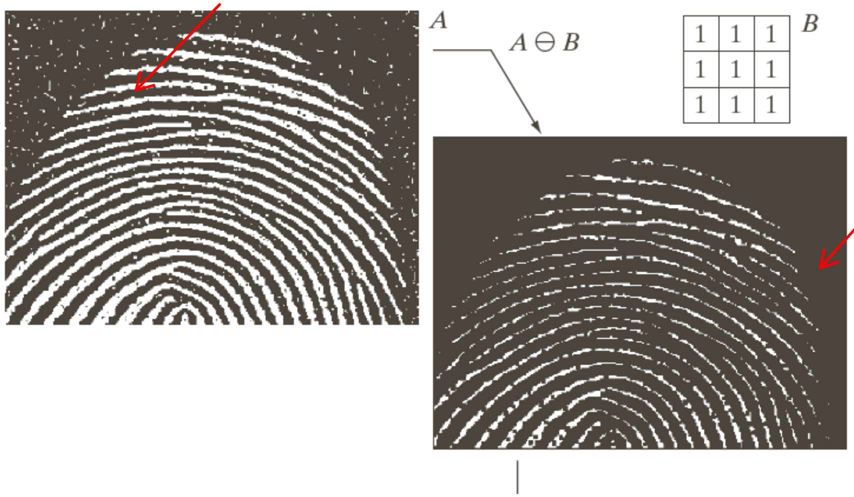
Example: Opening and closing

White and black dots due to noise



Example: Opening and closing

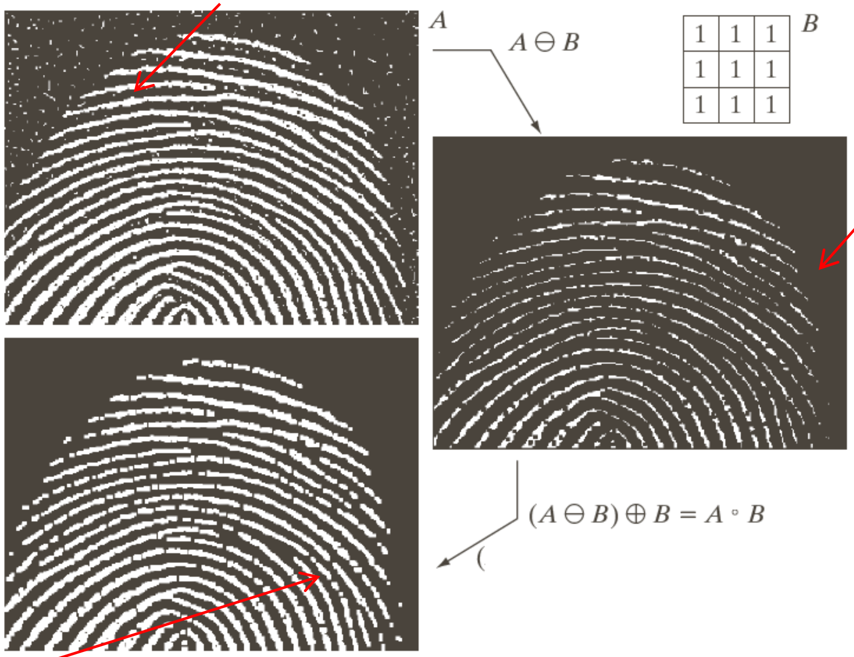
White and black dots due to noise



Background noise (white dots) removed
Black dots enhanced

Example: Opening and closing

White and black dots due to noise



Background noise (white dots) removed
Black dots enhanced

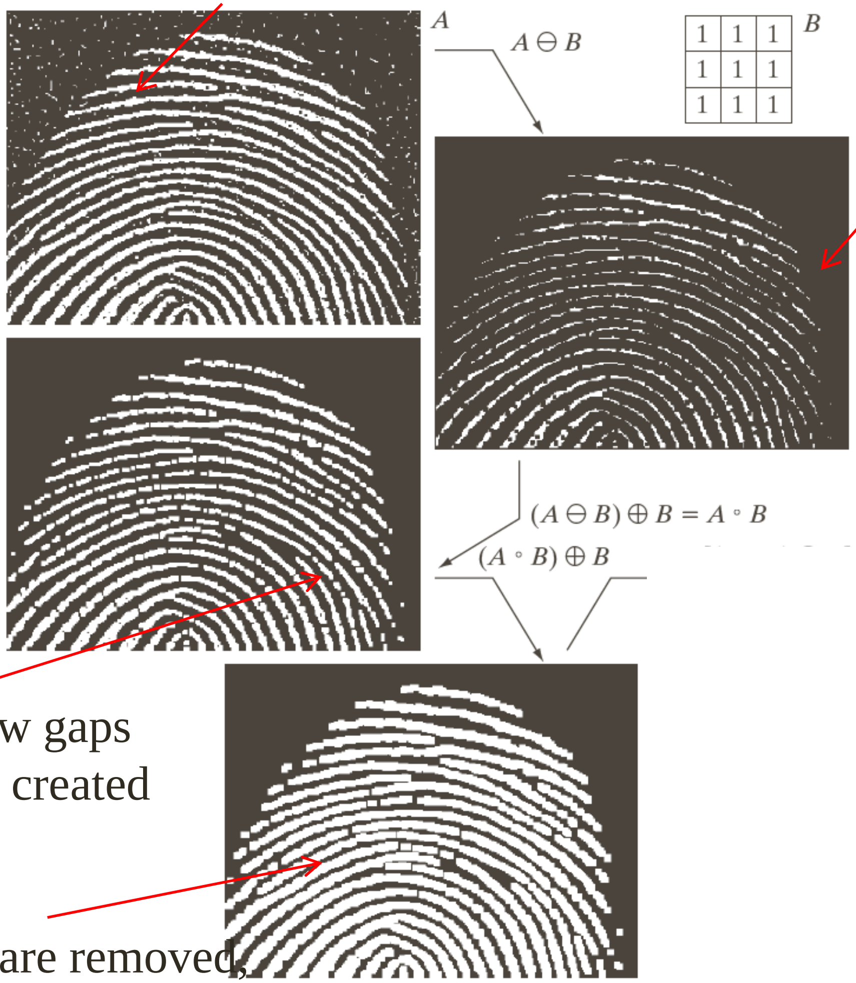
New gaps are created

Example: Opening and closing

White and black dots due to noise

Background noise (white dots) removed

Black dots enhanced

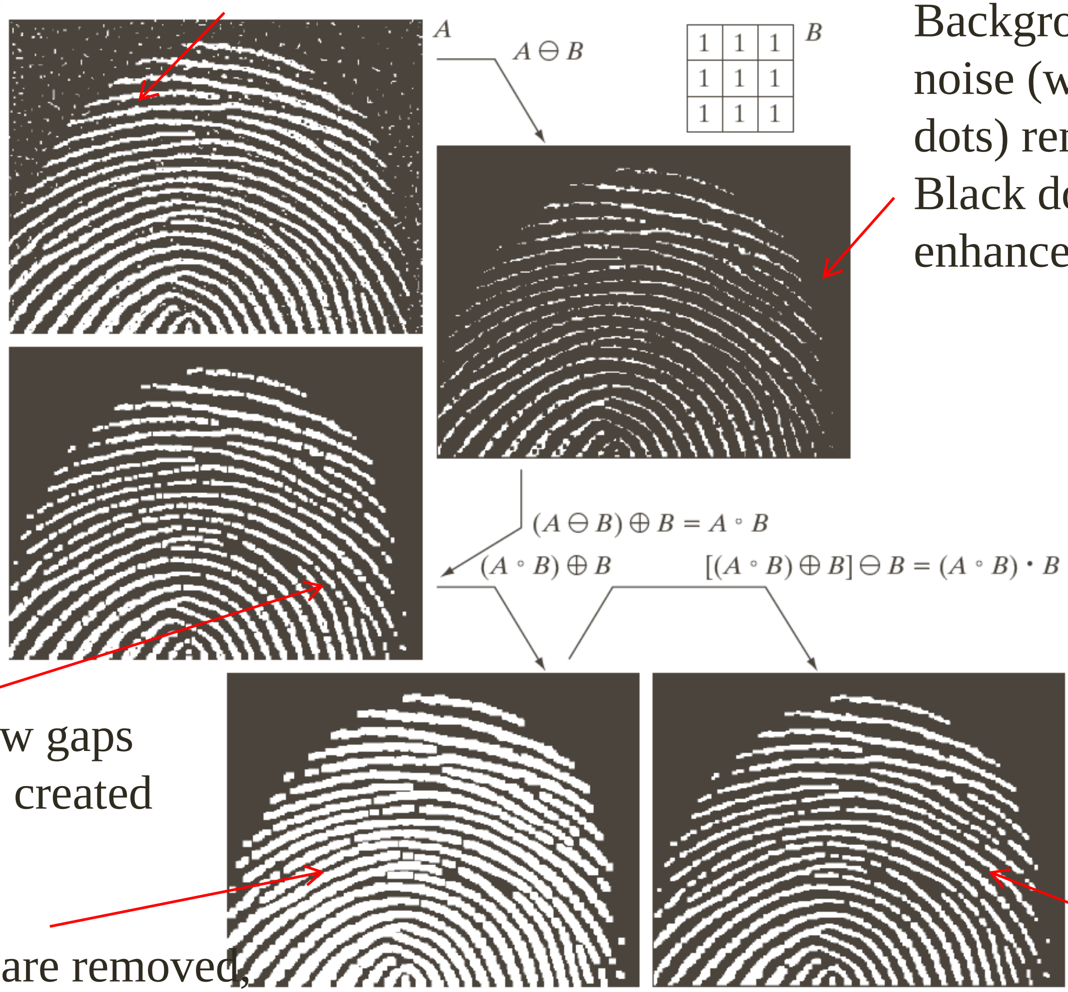


New gaps are created

Gaps are removed, ridges thickened

Example: Opening and closing

White and black dots due to noise



Background noise (white dots) removed

Black dots enhanced

New gaps are created

Gaps are removed, ridges thickened

Ridges thinned

The Hit-or-Miss Transformation

- A basic morphological tool for existence of object
- Finds location of a object, **X** in a larger image, **A**

The Hit-or-Miss Transformation

- Let X be enclosed by a small window, **W**
- The local background of X with respect to W is defined as B_2=(**W - X**)
- Apply erosion operator on **A** by B_1
- Apply erosion operator on the complement of **A** by the local background **(W – X)**
- Find intersection of outputs of the above two operations
- Intersection is precisely the location of object

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

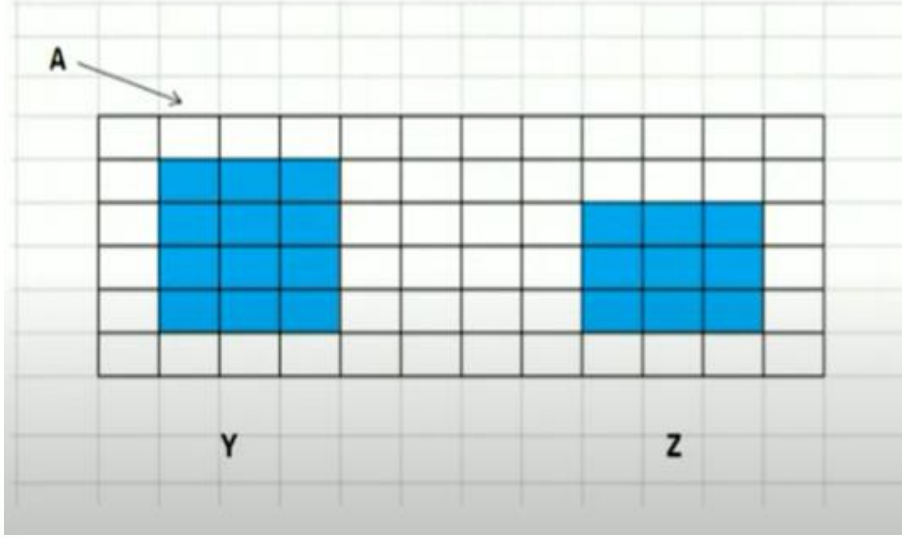
The Hit-or-Miss Transformation

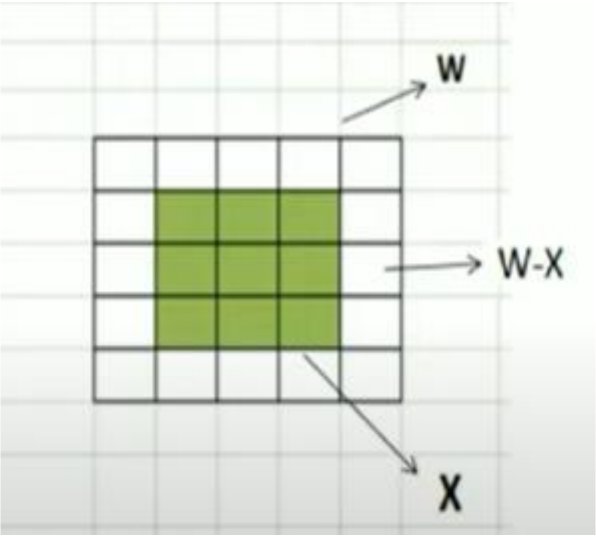
- Uses two structuring elements (B1 and B2)
- Mathematically, the HoM transform of image A by the structuring element set B (B = (B1,B2)),

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

Example: Hit or Miss Transform

- Find whether object of size 3×3 exist (hit) or not (miss)
- Consider SE, X of size, 3×3
- Blue and green are representative colors

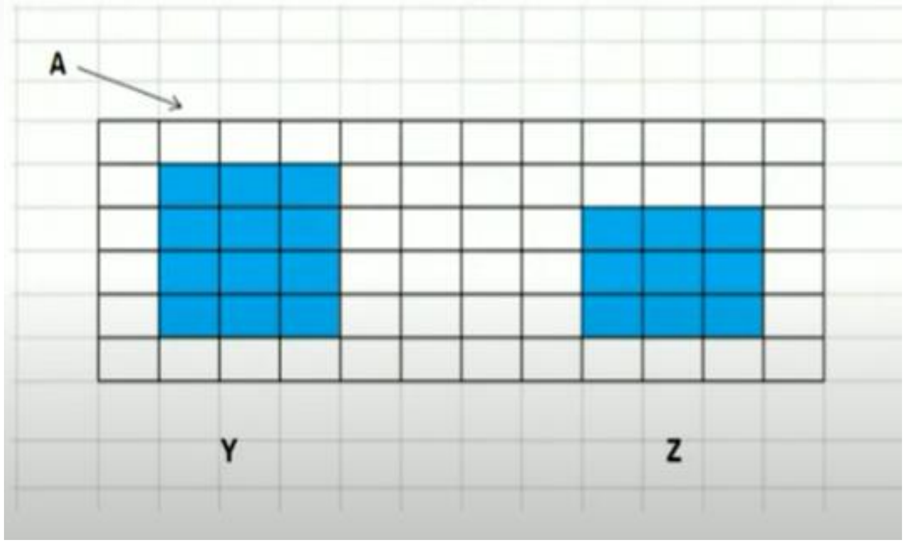


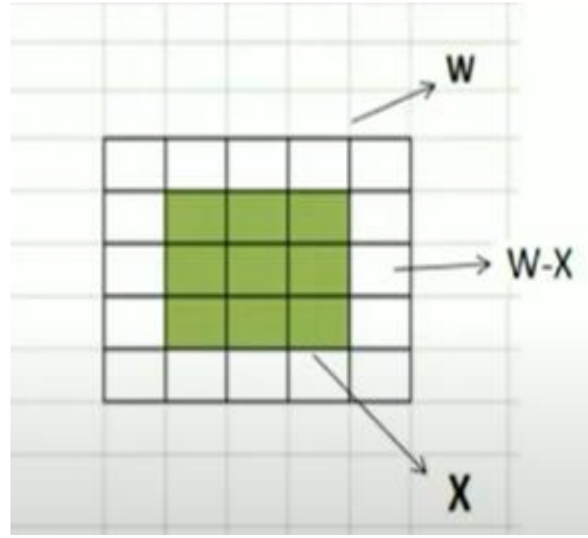


Image, A

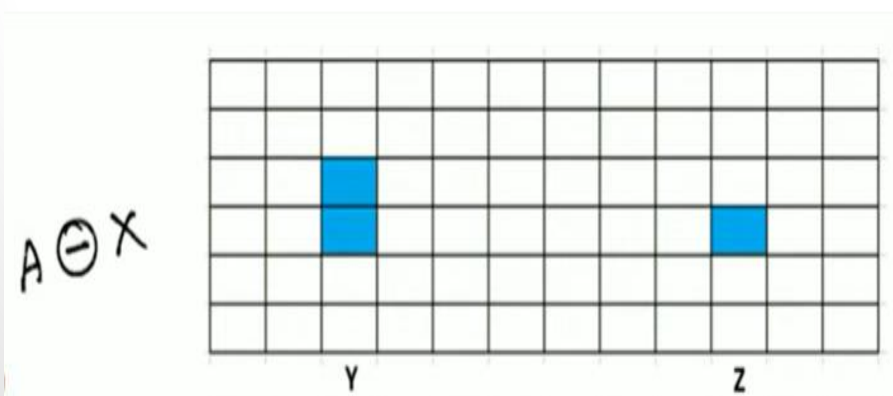
- W is window size larger than X
- Boundary of window is W-X

Example: Hit or Miss Transform

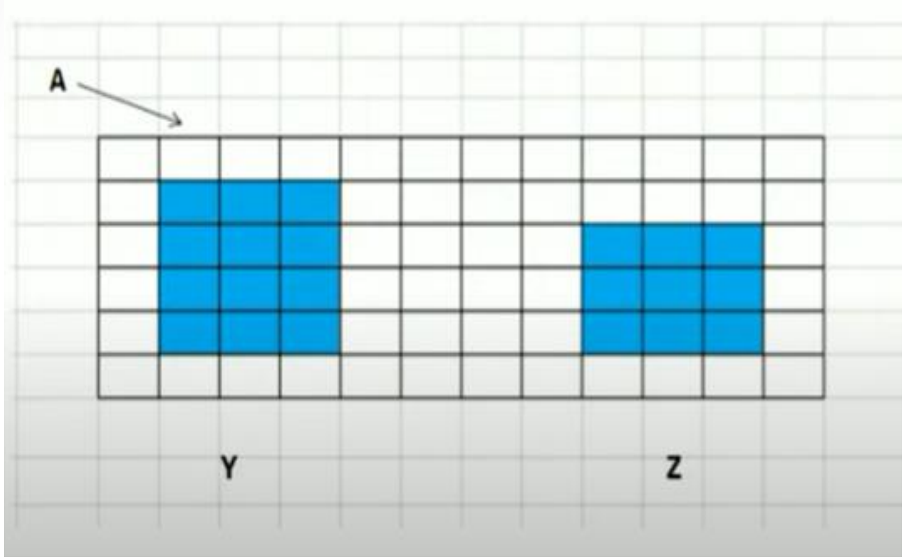


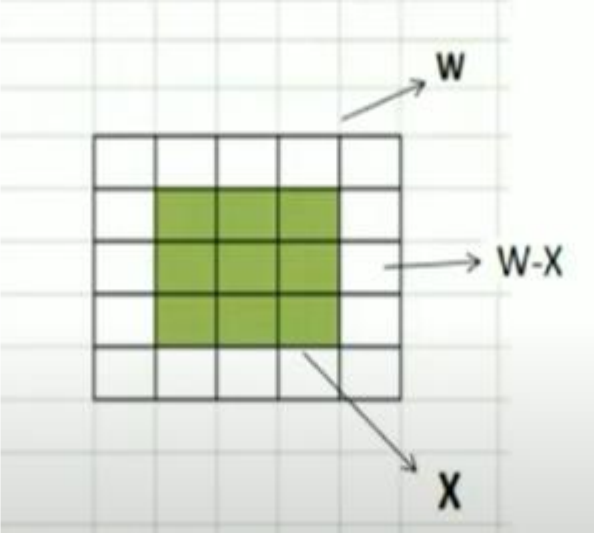


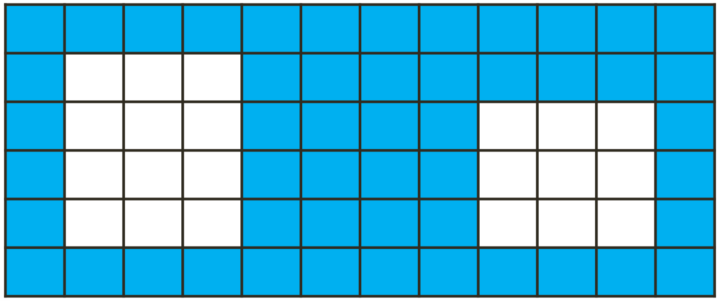
Image, A



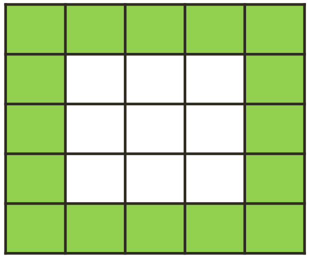
Example: Hit or Miss Transform





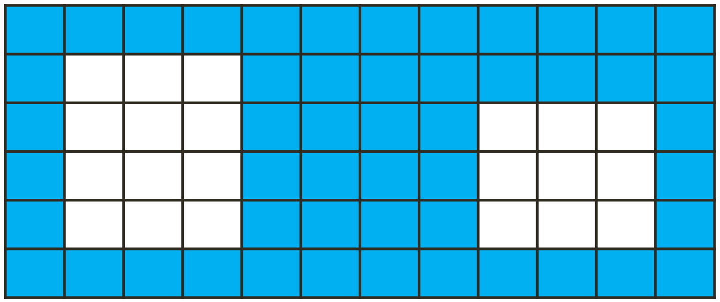


A^c

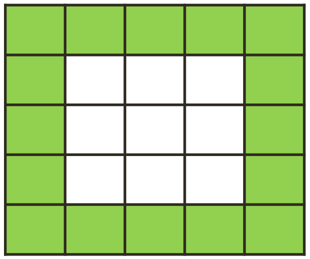


W-X

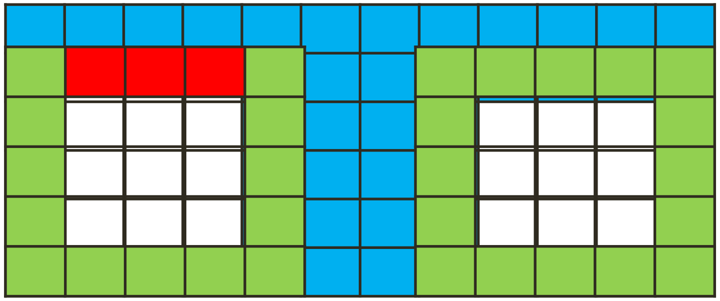
Example: Hit or Miss Transform

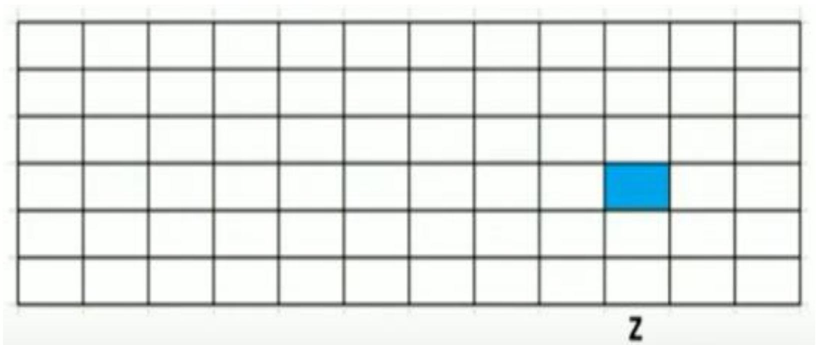


A^c



W-X



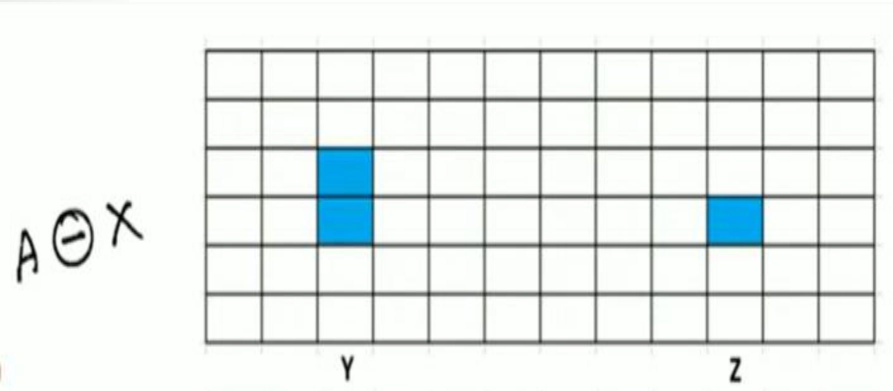


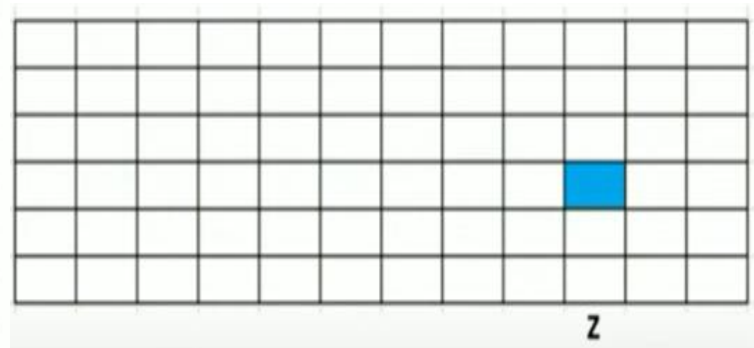
$A^c \ominus (w - x)$

For first region, red cells of A^c and W-X do not match, it is a miss

For second Region, A^c and W-X match, it is a hit

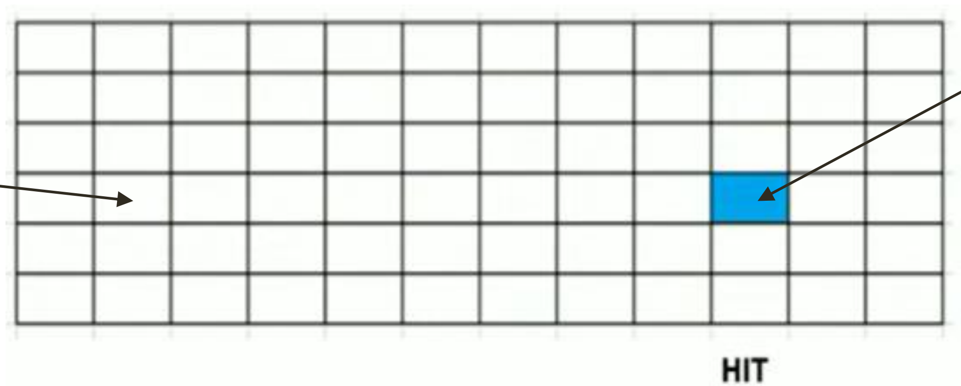
Example: Hit or Miss Transform





$A^c \ominus (W - X)$

Miss



At this location object, X exists

$(A \ominus X) \cap A^c \ominus (W - X)$

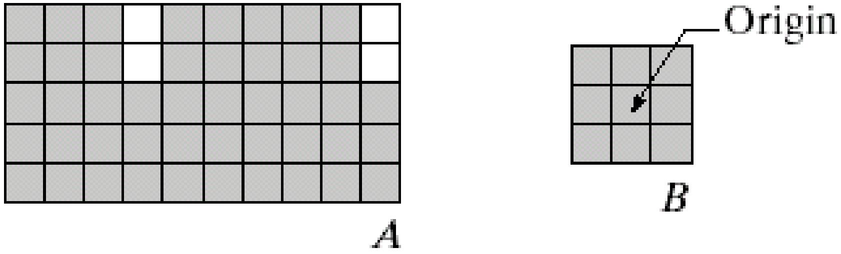
Boundary extraction

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

Boundary extraction

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

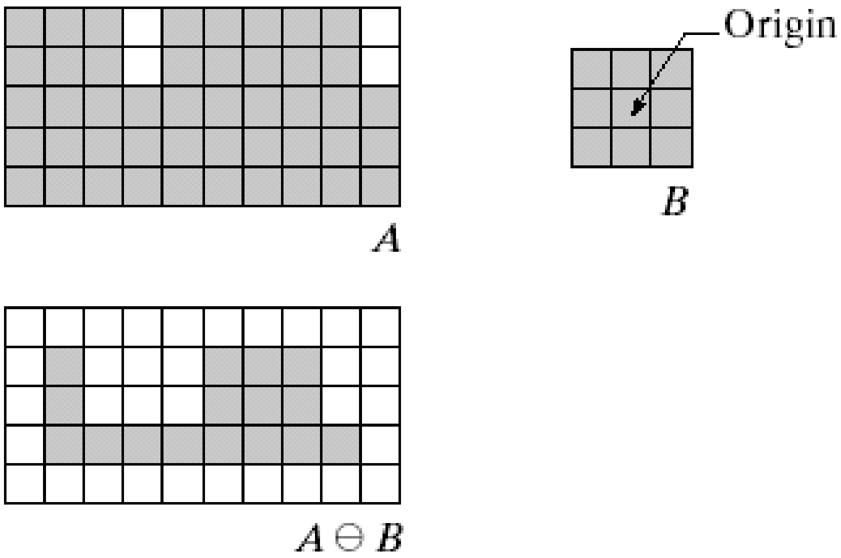
Foreground pixel(dark) is 1 and background (white) is 0



Boundary extraction

$$\boldsymbol{\beta}(\boldsymbol{A}) = \boldsymbol{A} - (\boldsymbol{A} \ominus \boldsymbol{B})$$

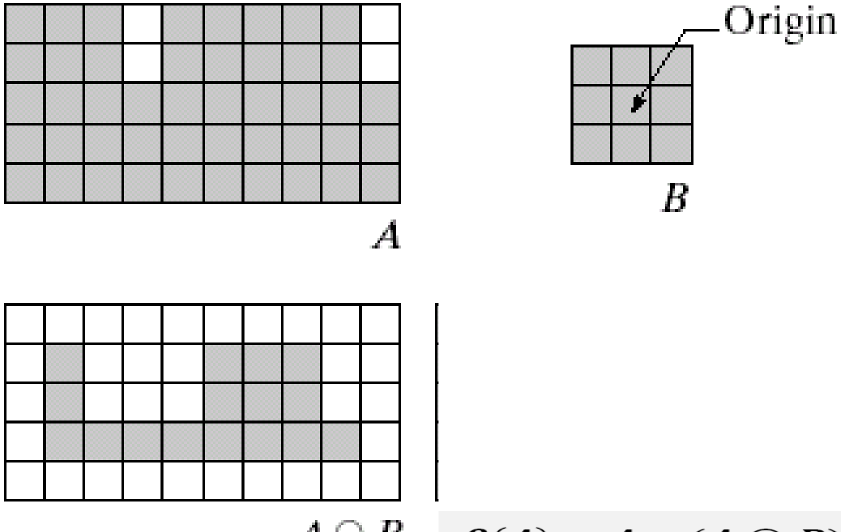
Foreground pixel(dark) is 1 and background (white) is 0



Boundary extraction

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

Foreground pixel(dark) is 1 and background (white) is 0

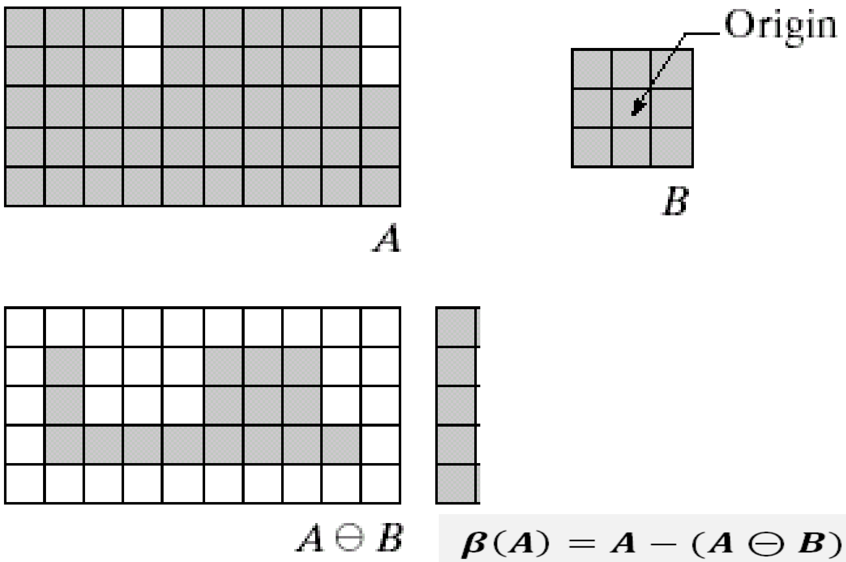


$$\beta(A) = A - (A \ominus B)$$

Boundary extraction

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

Foreground pixel(dark) is 1 and background (white) is 0

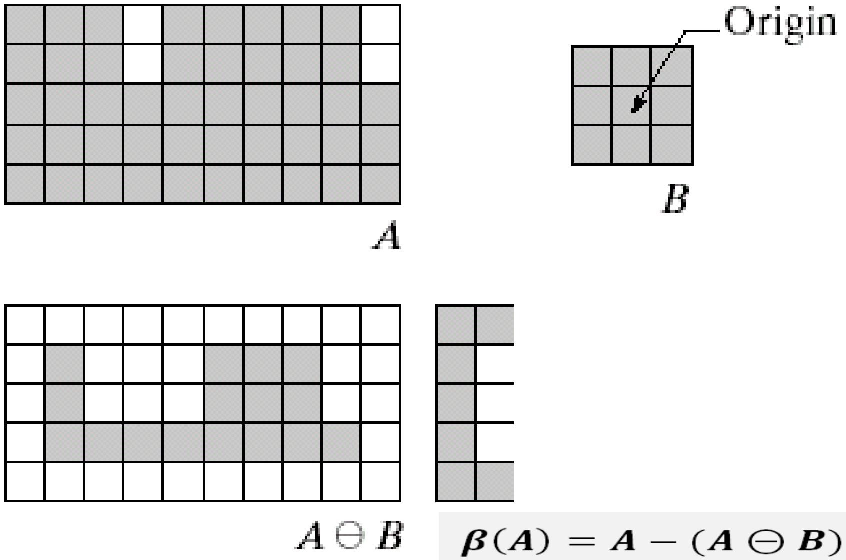


$$\beta(A) = A - (A \ominus B)$$

Boundary extraction

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

Foreground pixel(dark) is 1 and background (white) is 0

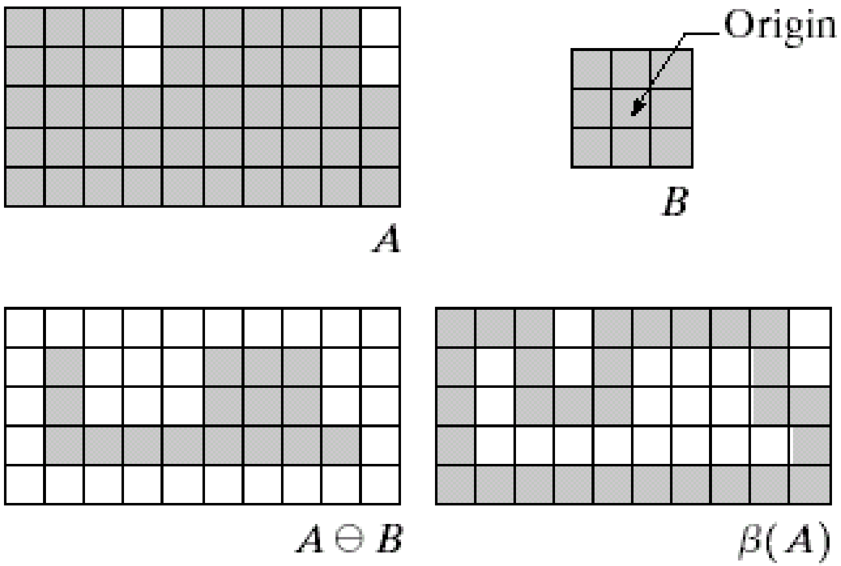


$$\beta(A) = A - (A \ominus B)$$

Boundary extraction

$$\boldsymbol{\beta}(\boldsymbol{A}) = \boldsymbol{A} - (\boldsymbol{A} \ominus \boldsymbol{B})$$

Foreground pixel(dark) is 1 and background (white) is 0



Boundary extraction

Foreground pixel is 1 (white)
background pixel is 0 (black)

Original image

3x3 structuring element is used

Boundary extraction

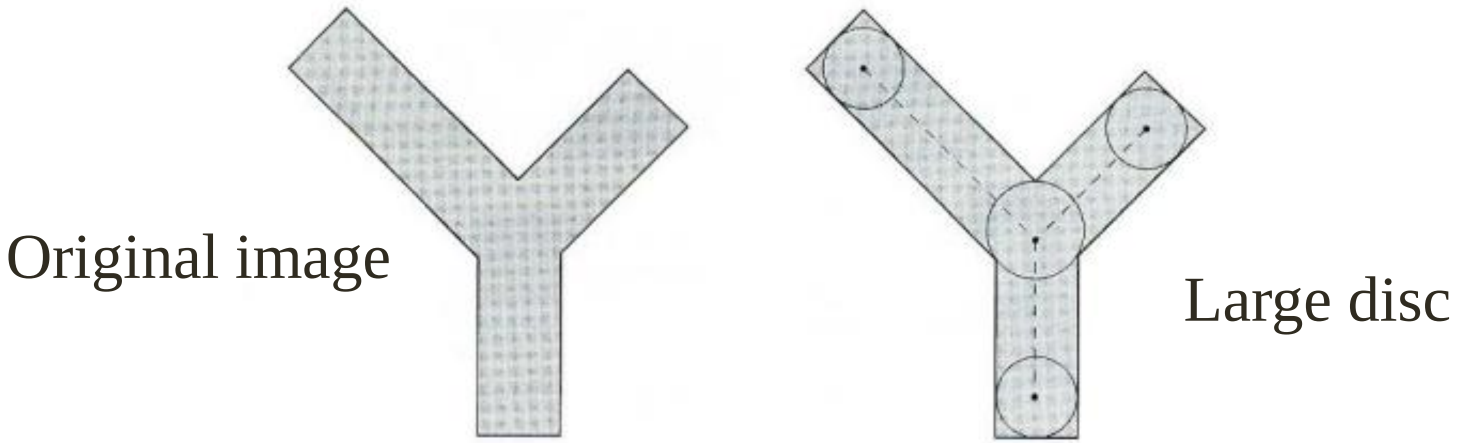
Foreground pixel is 1 (white)
background pixel is 0 (black)

Original image

Boundary in image

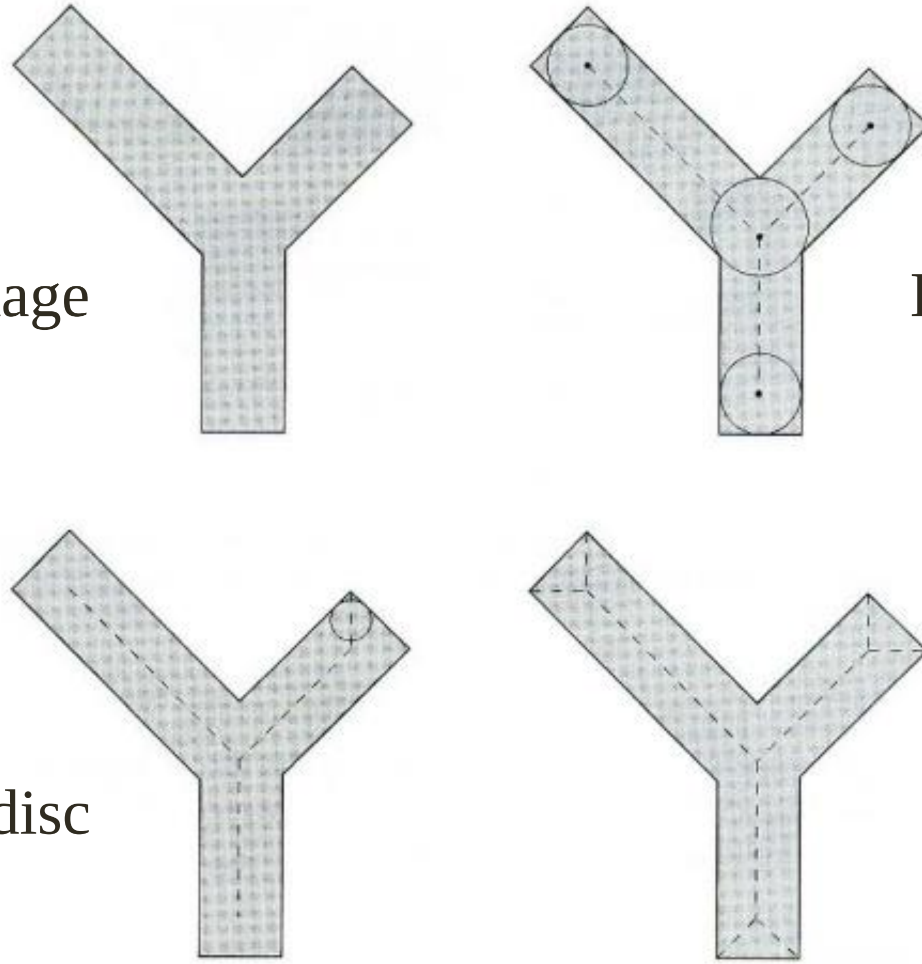
3x3 structuring element is used

Skeleton (intuition)



- Largest disk centered in the image and is contained in A
- Dotted line is skeleton of image

Skeleton



Original image

Large disc

Smaller disc

Skeleton

- Smaller disk can be used provided it touches the boundary of **A** at two or more different places

Skeleton

- The skeleton of A is

$$S(A) = \bigcup_{k=0}^{K} S_k(A)$$

$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

- Where B is structuring element
- And $S_k(A)$ is skeleton subset
- And kB indicates k successive erosions of A

$$(A \ominus kB) = (\ldots((A \ominus B) \ominus B) \ominus \ldots) \ominus B$$

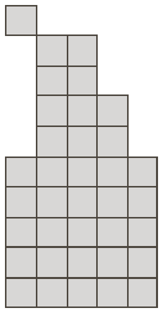
Skeleton

- K is the last iterative step before A erodes to an empty set

$$K = \max\{k|(A \ominus kB) \neq \emptyset\}$$

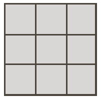
$$S(A) = \bigcup_{k=0}^{K} S_k(A)$$

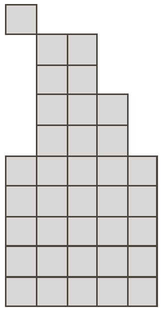
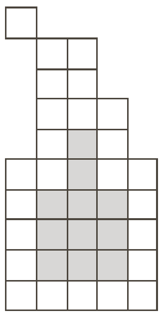
- S(A) can be obtained as the union of skeleton subsets $S_k(A)$

k	$\boxed{A \ominus kB}$
0	

$$S_k(A) = \boxed{(A \ominus kB)} - (A \ominus kB) \circ B$$

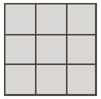
B

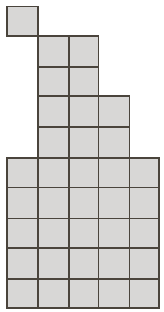
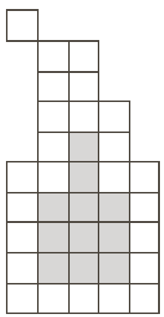
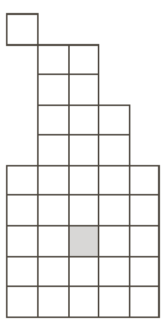


k	$A \ominus kB$
0	
1	

$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

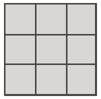
B

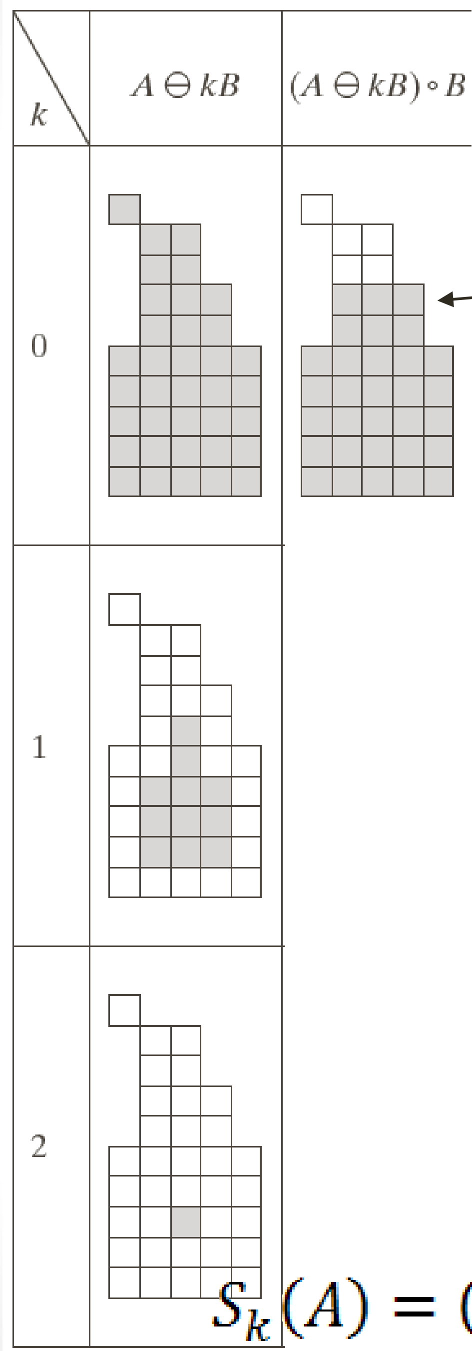


k	$A \ominus kB$
0	
1	
2	

$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

B

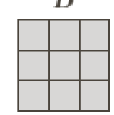


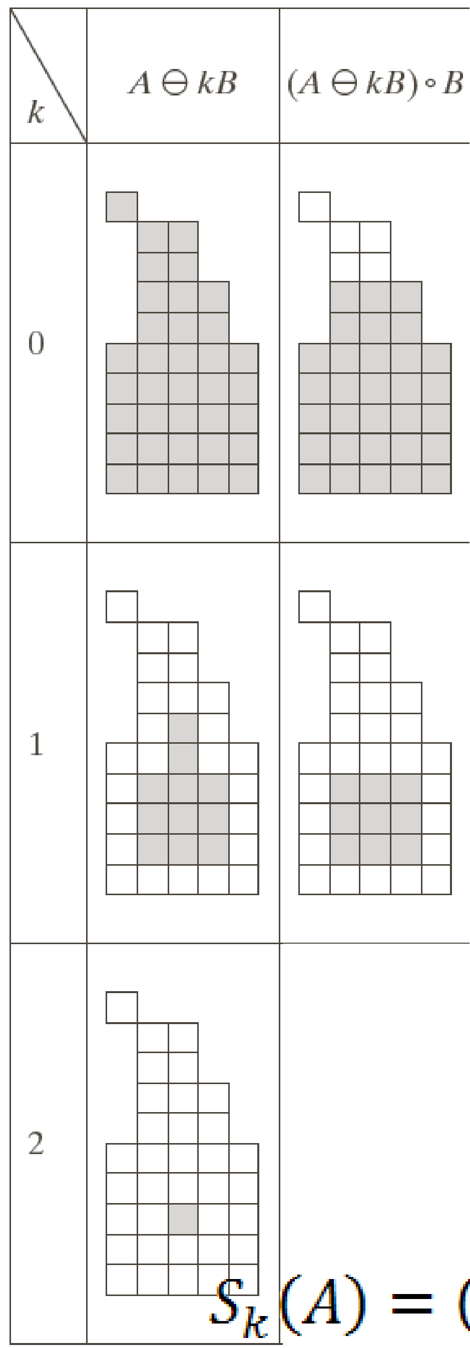


Opening removes parts of image which can not contain B

$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

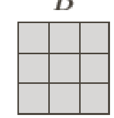
B

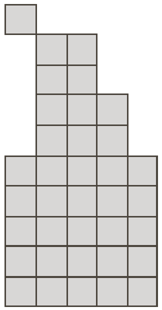
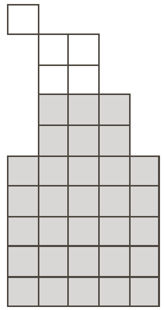
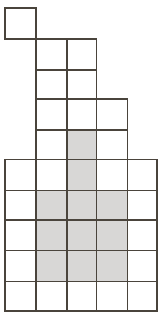
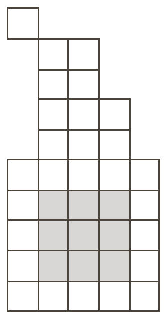
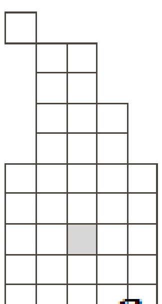
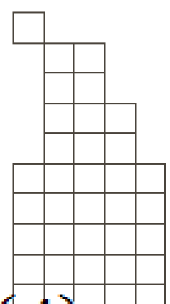




$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

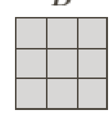
B

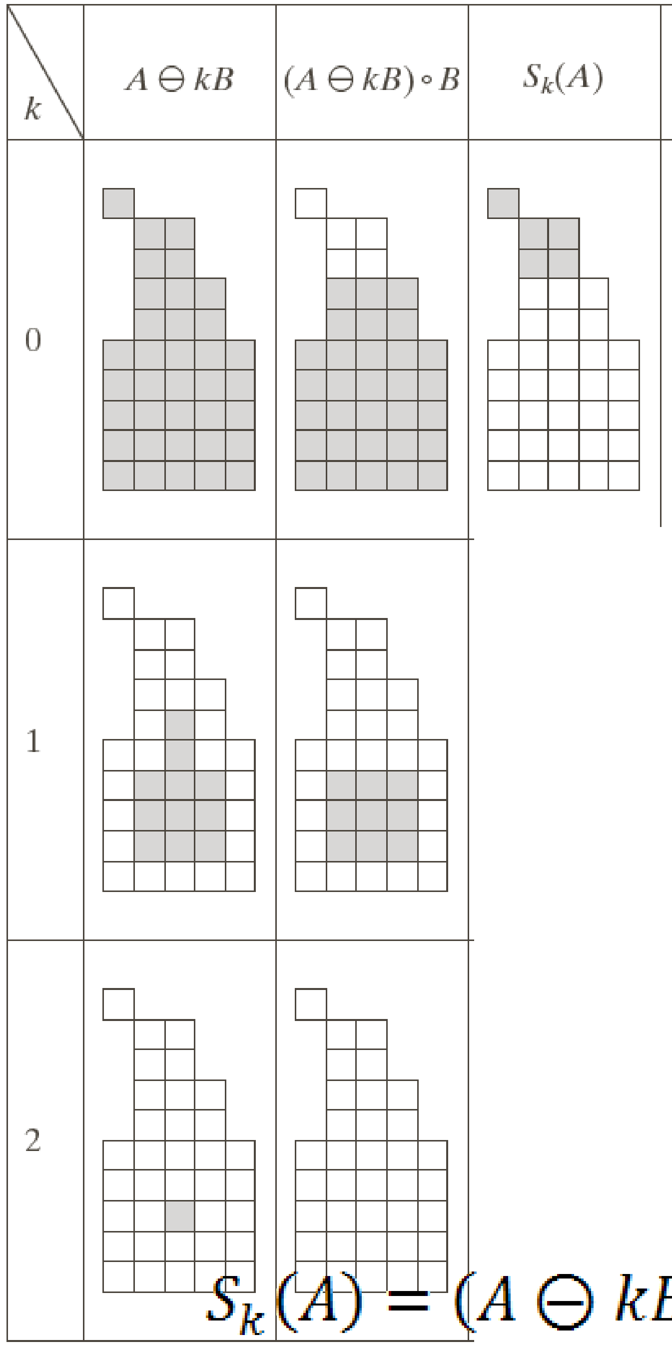


k	$A \ominus kB$	$(A \ominus kB) \circ B$
0		
1		
2		

$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

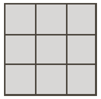
B

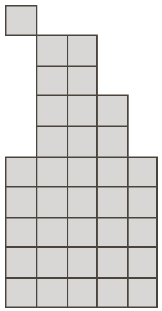
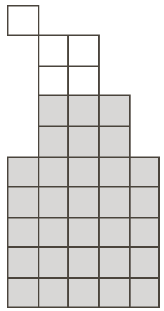
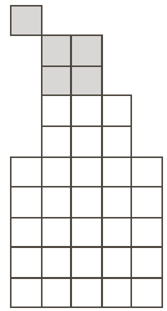
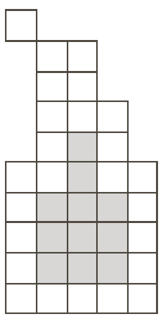
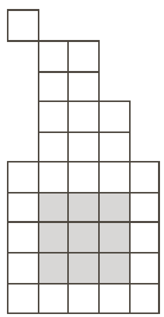
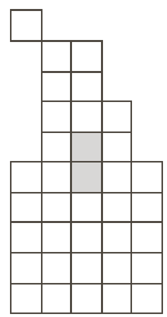
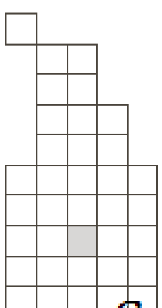
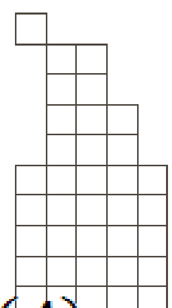




$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

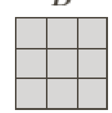
B

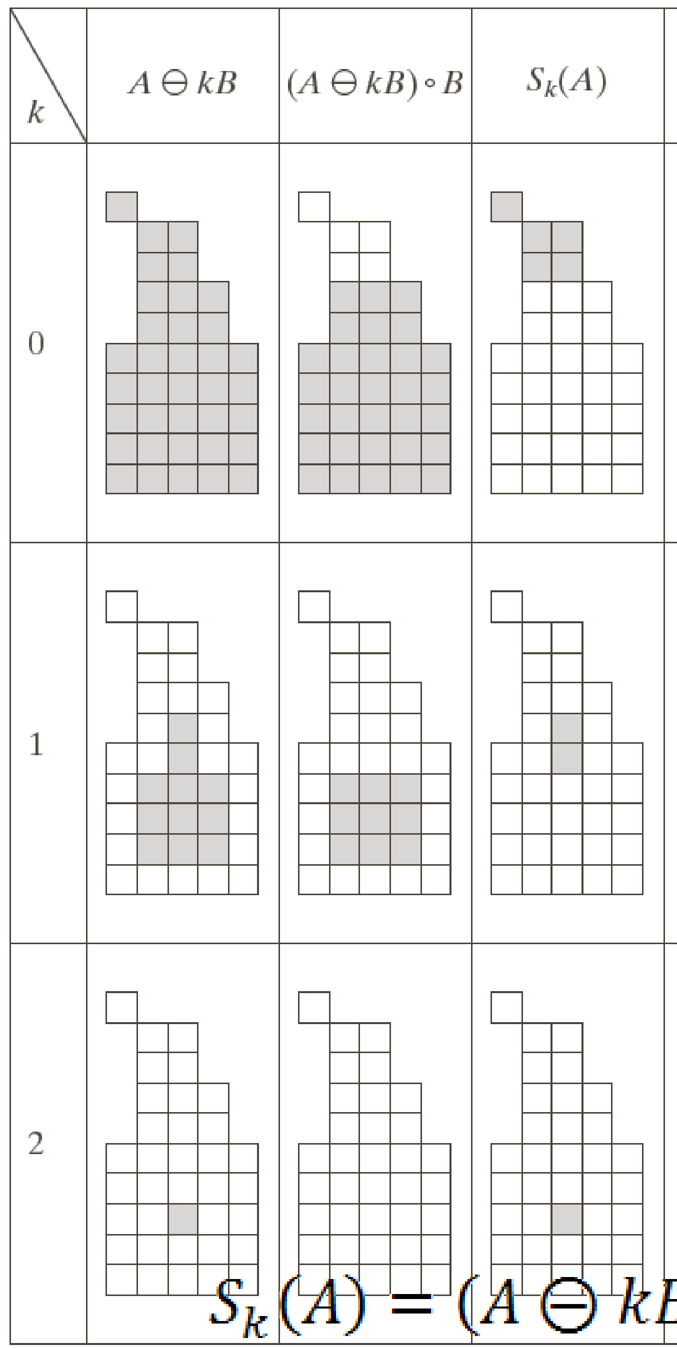


k	$A \ominus kB$	$(A \ominus kB) \circ B$	$S_k(A)$
0			
1			
2			

$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

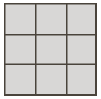
B

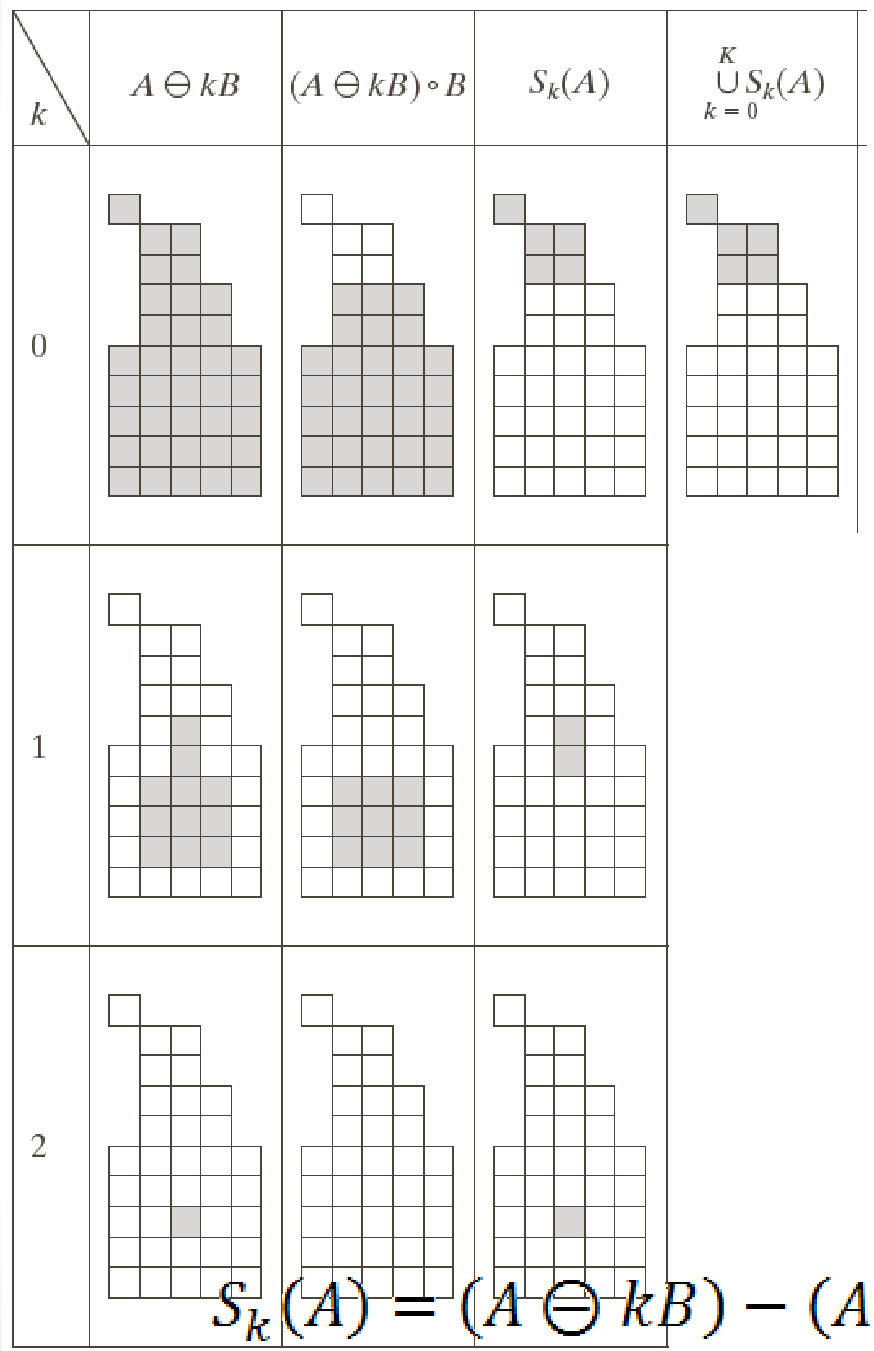




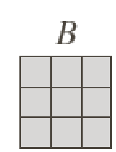
$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

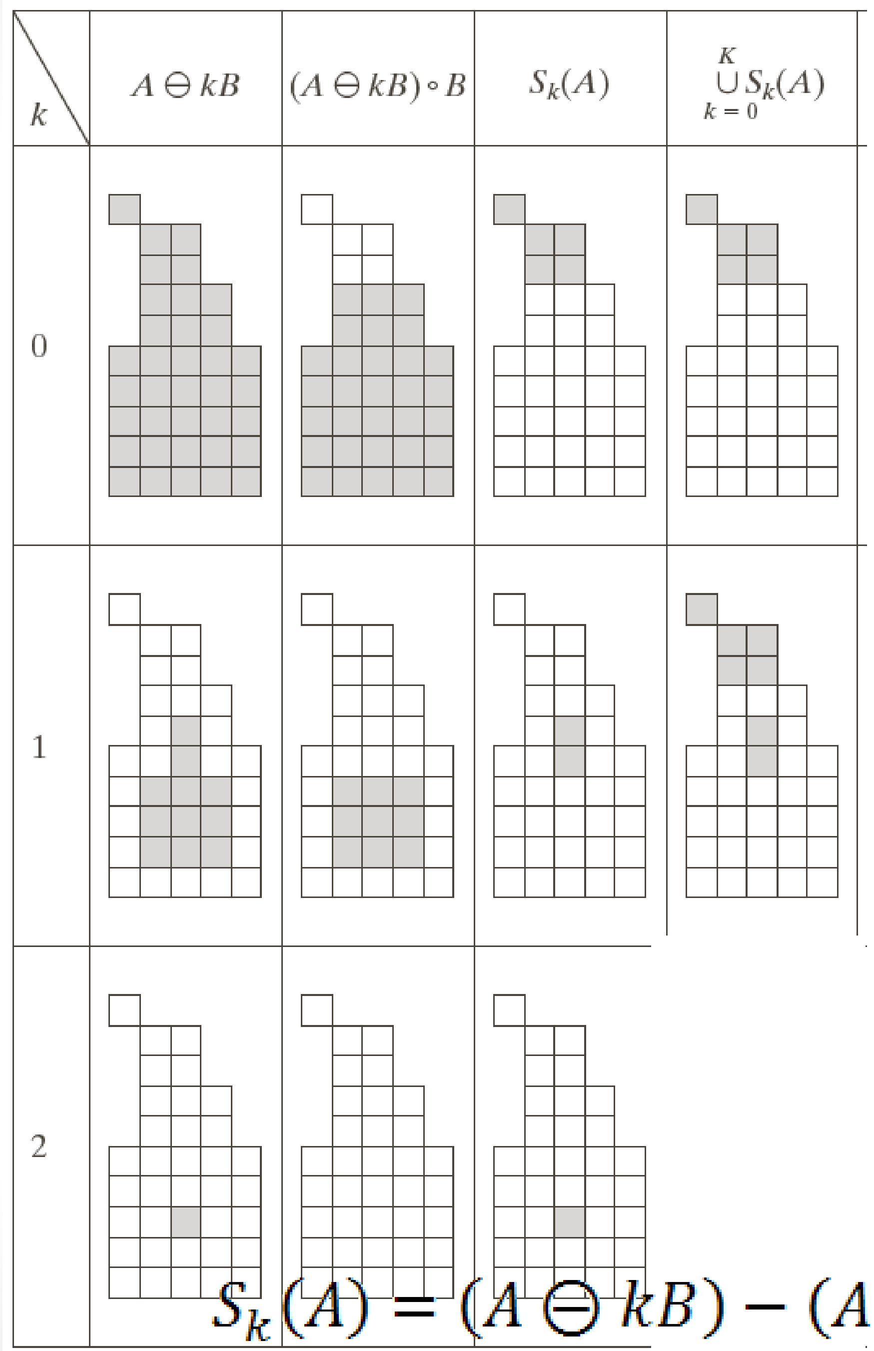
B





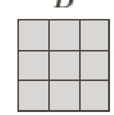
$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

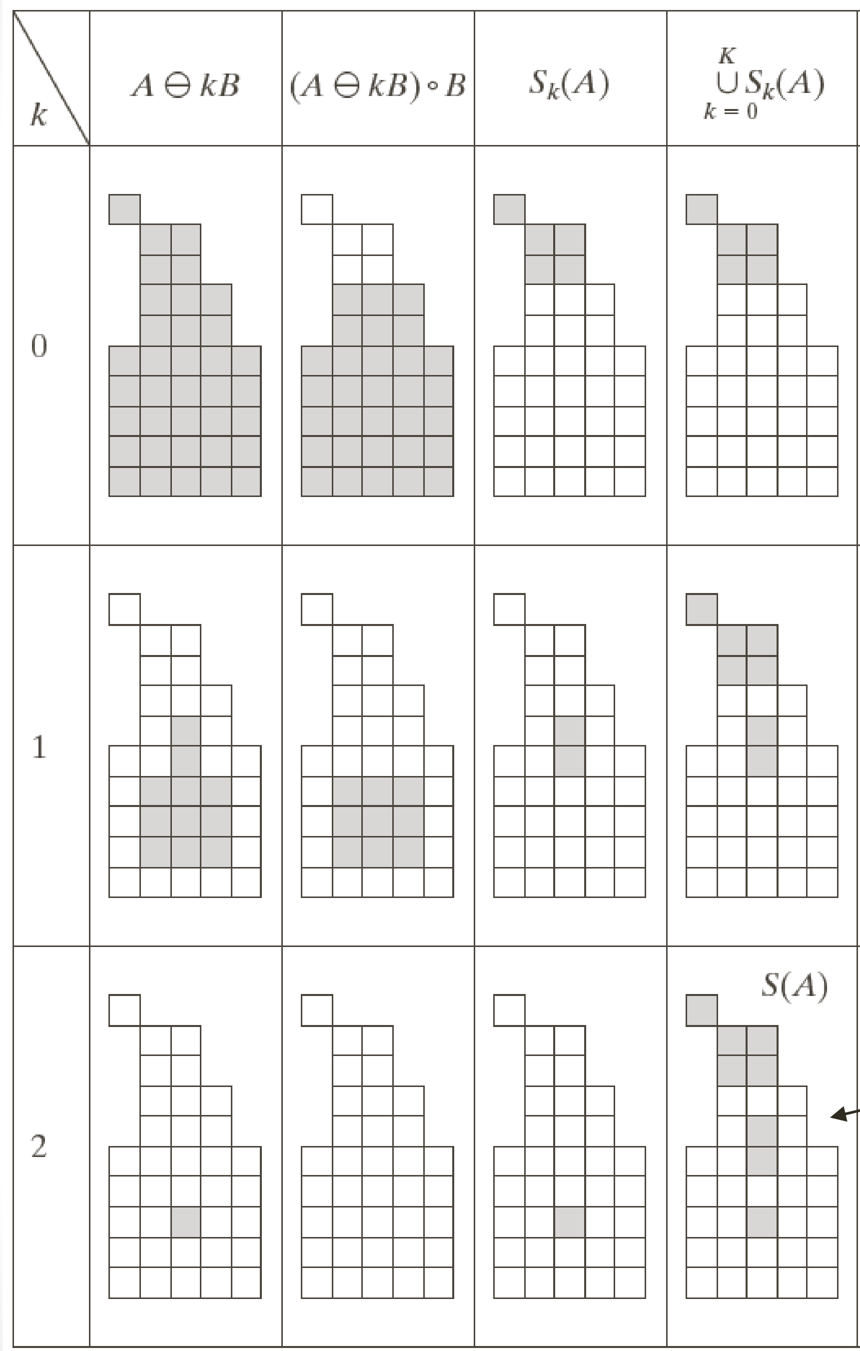




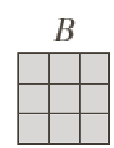
$$S_k(A) = (A \ominus kB) - (A \ominus kB) \circ B$$

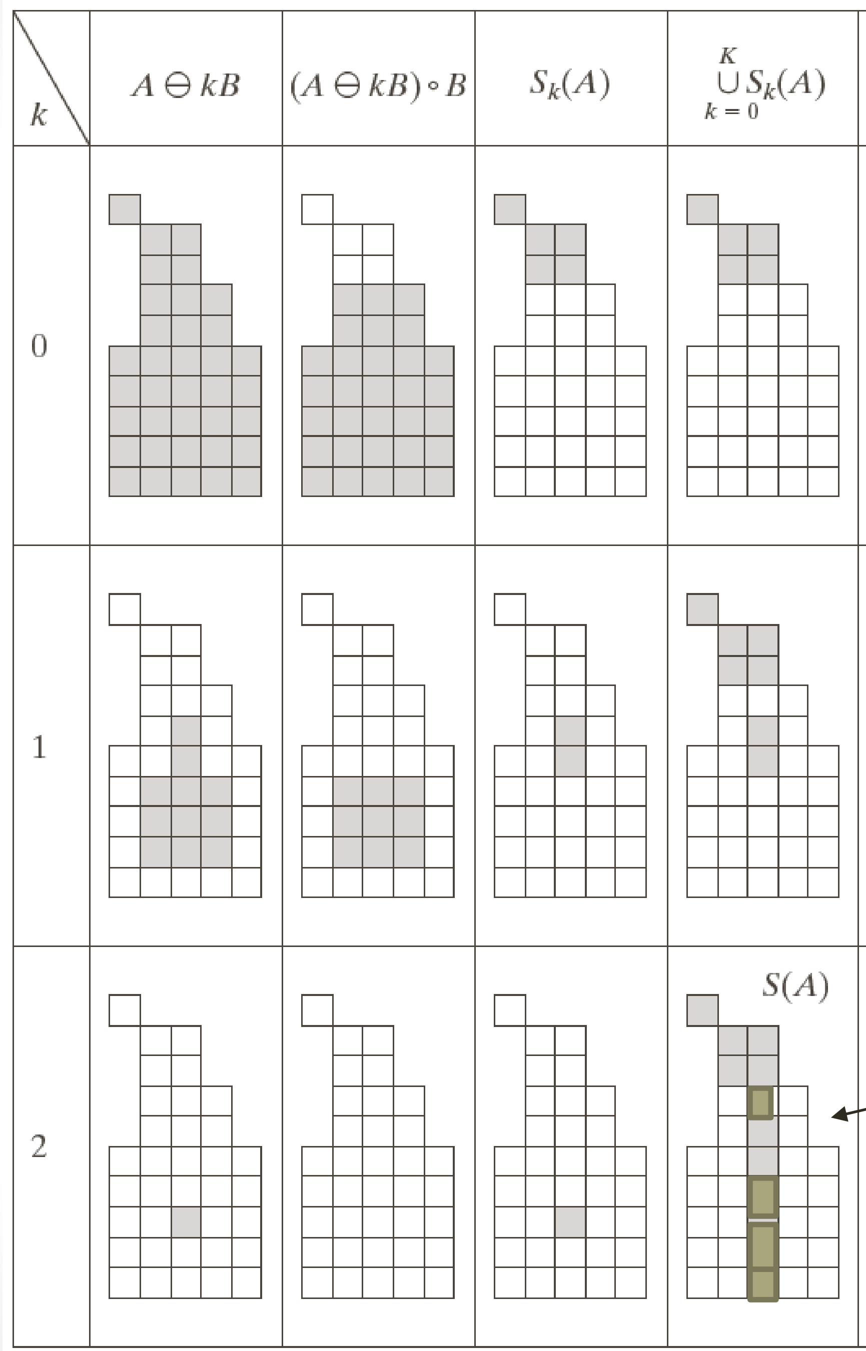
B





Skeleton





Ideal Skeleton

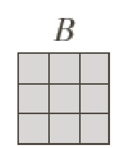


Image Thinning

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

Image, A

- Foreground is 1 and background is 0
- After thinning the number of zeroes increase

Image Thinning

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

Image, A

Structuring Elements (SE)

0	0	0
X	1	X
1	1	1

B_1

X	0	0
1	1	0
1	1	X

B_2

1	X	0
1	1	0
1	X	0

B_3

- X is don’t care
- Eight SEs are used
- B_1 is rotated clockwise to generate B_2
- Same process is repeated for remaining structuring elements

Image Thinning

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

Image, A

0	0	0
X	1	X
1	1	1

B_1

X	0	0
1	1	0
1	1	X

B_2

1	X	0
1	1	0
1	X	0

B_3

- Thin(A,B_1) = A-(A$\circledast B_1$), where $\circledast$ is Hit or Miss Transform
- Overlap B_1 on each part of image
- If overlapped portion and B match then replace center of overlapped portion by 0 else don’t change the center

Image Thinning

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

Image, A

0	0	0
X	1	X
1	1	1

B_1

- Apply A- (A $\circledast B_1$) multiple times to generate A_{B1}

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

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

Image Thinning

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

Image, A

0	0	0
X	1	X
1	1	1

B_1

- Apply A- ($A \circledast B_1$) multiple times to generate A_{B1}

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

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

Image Thinning

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

Image, A_{B1}

X	0	0
1	1	0
1	1	X

B_2

- Apply A_{B1}- ($A_{B1} \circledast B_2$) multiple times to generate A_{B2}

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

Image, A_{B2}

Image Thinning

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

Image, A_{B2}

1	X	0
1	1	0
1	X	0

B_3

- Apply A_{B2}- ($A_{B2} \circledast B_3$) multiple times

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

Image, A_{B3}

Image Thinning

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

Image, A_{B3}

1	X	0
1	1	0
1	X	0

B_3

1	1	X
1	1	0
X	0	0

B_4

- Apply A_{B3}- ($A_{B3} \circledast B_4$) multiple times

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

Image, $A_{B4} = A_{B3}$as B4 does not match with any part of image

Image Thinning

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

Image, A_{B4}

1	1	X
1	1	0
X	0	0

B_4

1	1	1
X	1	X
0	0	0

B_5

- Apply $A_{B4} - (A_{B4} \circledast B_5)$ multiple times

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

Image, $A_{B5} = A_{B4}$ as B5 does not match with any part of image

Image Thinning

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

Image, A_{B5}

1	1	1
X	1	X
0	0	0

B_5

X	1	1
0	1	1
0	0	X

B_6

0	X	1
0	1	1
0	X	1

B_7

0	0	X
0	1	1
X	1	1

B_8

- Apply A_{B5}- ($A_{B5} \circledast B_6$) multiple times

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

Image, $A_{B6} = A_{B5}$ as B6 does not match with any part of image

- Apply A_{B6}- ($A_{B6} \circledast B_6$) multiple times

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

Image, $A_{B7} = A_{B6}$ as B7 does not match with any part of image

Image Thinning

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

Image, A_{B7}

1	1	1
X	1	X
0	0	0

B_5

X	1	1
0	1	1
0	0	X

B_6

0	X	1
0	1	1
0	X	1

B_7

0	0	X
0	1	1
X	1	1

B_8

- Apply A_{B7}- ($A_{B7} \circledast B_8$) multiple times

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

Image, $A_{B8} = A_{B7}$ as B8 does not match with any part of image

Image Thinning

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

Image, A

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

Thinned Image, A_{B8}

Image Thickening

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

Image, A

- Foreground is 1
- Background is 0
- After thickening number of ones increase

Image Thickening

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

Image, A

0	0	0
X	0	X
1	1	1

B_1

X	0	0
1	0	0
1	1	X

B_2

1	X	0
1	0	0
1	X	0

B_3

- X is don’t care
- B_1 is rotated clockwise to generate B_2
- Same process repeated for remaining 8 structuring elements

Image Thickening

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

Image, A

0	0	0
X	0	X
1	1	1

B_1

X	0	0
1	0	0
1	1	X

B_2

1	X	0
1	0	0
1	X	0

B_3

- Thin(A,B_1) = $A \cup (A \circledast B_1)$, where $\circledast$ is Hit or Miss Transform
- Overlap B_1 on each part of image
- If overlapped portion and B match then replace center of overlapped portion by 1 else don't change the center

Image Thickening

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

Image, A

0	0	0
X	0	X
1	1	1

B_1

X	0	0
1	0	0
1	1	X

B_2

Apply A $U(A \circledast B_1)$ multiple times

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

No change in A, A_{B1} =A

Apply $A_{B1}U(A \circledast B_2)$ multiple times

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

No change $A_{B2} = A_{B1}$

Image Thickening

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

Image, A_{B2}

1	X	0
1	0	0
1	X	0

B_3

1	1	X
1	0	0
X	0	0

B_4

Apply $A_{B2} \cup (A_{B2} \circledast B_3)$ multiple times

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

Image, A_{B3}

Apply $A_{B3} \cup (A_{B3} \circledast B_4)$ multiple times

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

No change, $A_{B4} = A_{B3}$

Image Thickening

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

Image, A_{B4}

1	1	1
X	0	X
0	0	0

B_5

X	1	1
0	0	1
0	0	X

B_6

Apply $A_{B4} U (A_{B4} \circledast B_5)$ multiple times

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

No change, $A_{B5} = A_{B4}$

Apply $A_{B5} U (A_{B5} \circledast B_6)$ multiple times

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

No change, $A_{B6} = A_{B5}$

Image Thickening

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

Image, A_{B6}

0	X	1
0	0	1
0	X	1

B_7

0	0	X
0	0	1
X	1	1

B_8

Apply $A_{B6} U (A_{B6} \circledast B_7)$ multiple times

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

No change, $A_{B7} = A_{B6}$

Apply $A_{B7} U (A_{B7} \circledast B_8)$ multiple times

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

No change, $A_{B8} = A_{B7}$

Image Thickening

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

Image, A

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

Thickened Image, A_{B8}