Digital Image Processing (CS/ECE 545)
Lecture 6: Morphological Filters

Prof Emmanuel Agu

Computer Science Dept.

Worcester Polytechnic Institute (WPI)

Mathematical Morphology

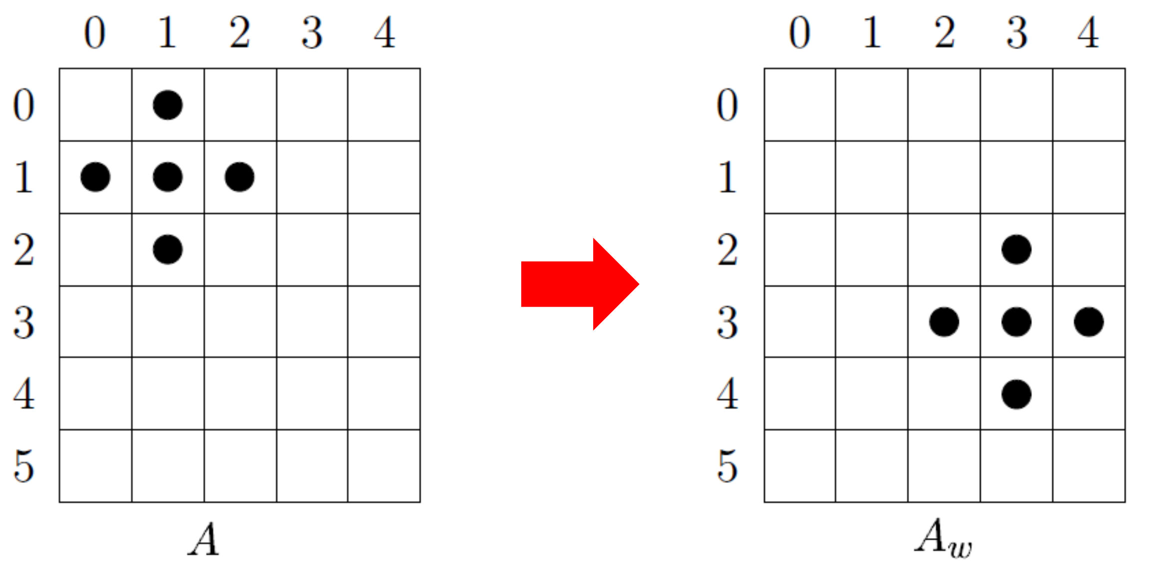
- Originally operated on Binary (black and white) images
- Binary images?
  - Faxes, digitally printed images
  - Obtained from thresholding grayscale images
- Morphological filters alter local structures in an image

Translation

- *A* is set of pixels in binary image
- *w* = (*x*,*y*) is a particular coordinate point
- *A* is set *A* "translated" in direction (*x*,*y*). i.e

$$A_x = \{(a,b) + (x,y) : (a,b) \in A\}.$$

- Example: If *A* is the cross-shaped set, and *w* = (2,2)

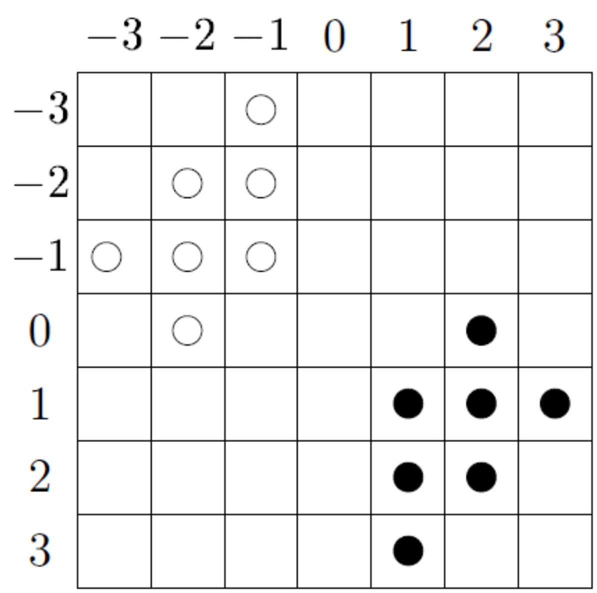


Reflection

- A is set of pixels
- Reflection of A is given by

$$\hat{A} = \{(-x, -y) : (x, y) \in A\}.$$

- An example of a reflection



Mathematical Morphology

- 2 basic mathematical morphology operations, (built from translations and reflections)
  - **Dilation**
  - **Erosion**

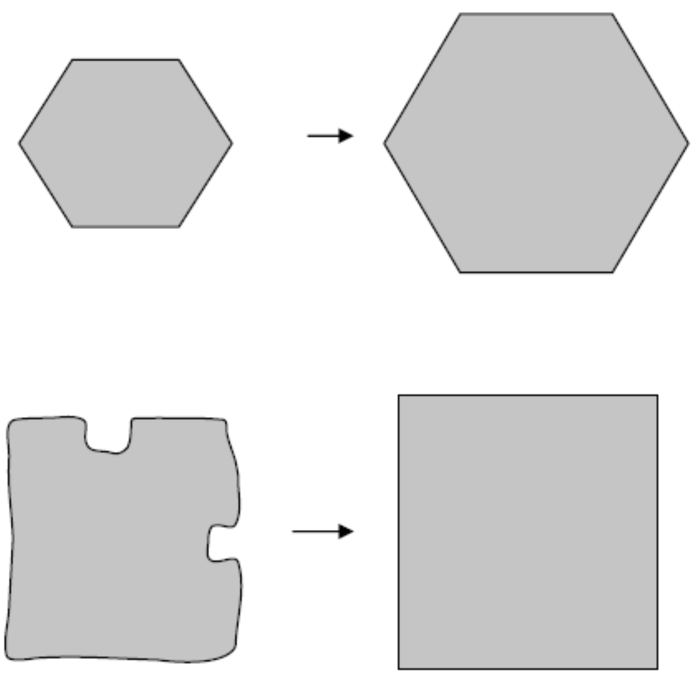
Also several composite relations

- **Closing and Opening**
- **Conditional Dilation**
- **. . .**

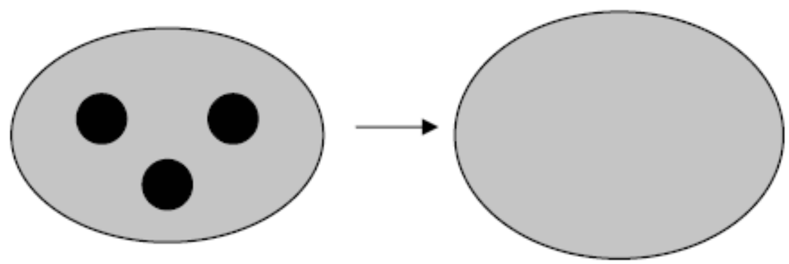
Dilation

- Dilation **expands** connected sets of 1s of a binary image. It can be used for

1. Growing features



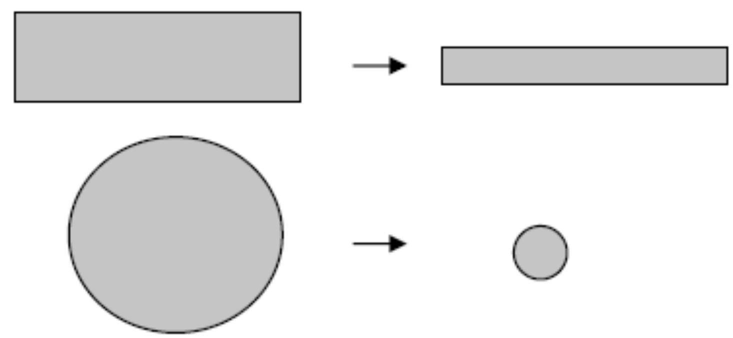
2. Filling holes and gaps



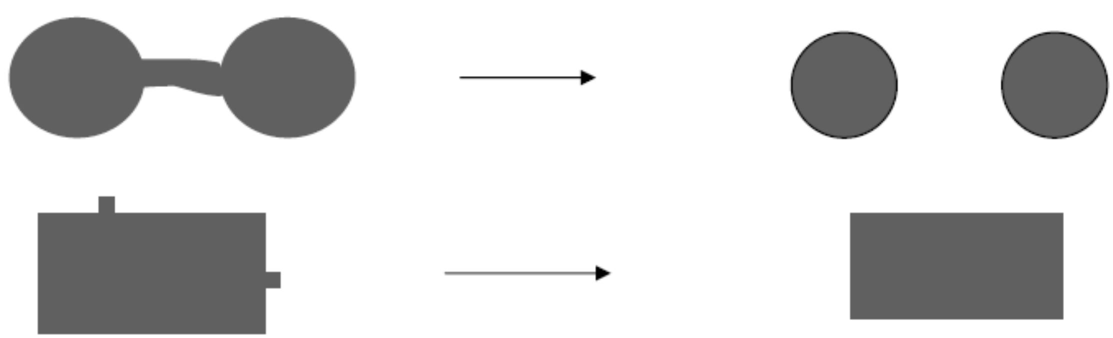
Erosion

- Erosion **shrinks** connected sets of 1s in binary image.
- Can be used for

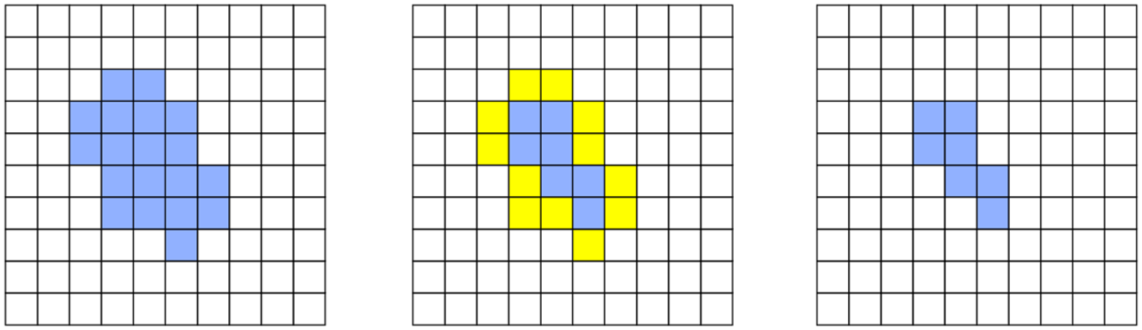
1. shrinking features



2. Removing bridges, branches and small protrusions

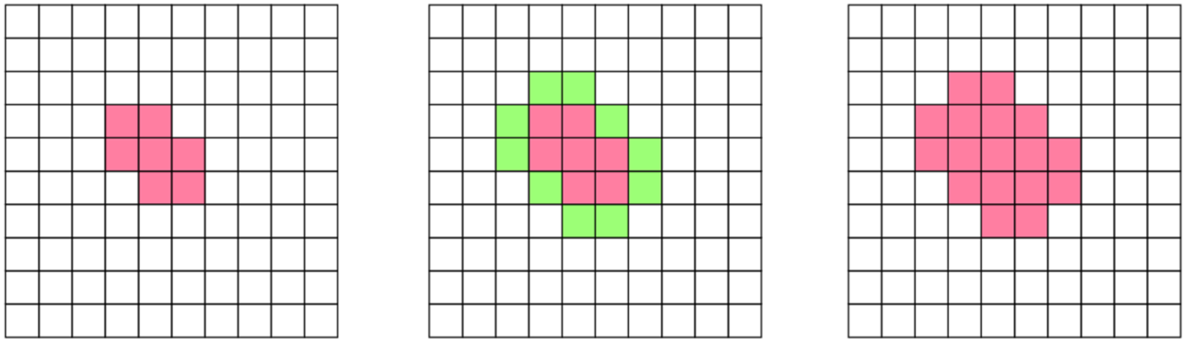


Shrink and Let Grow



(a) (b) (c)

Shrinking: remove border pixels

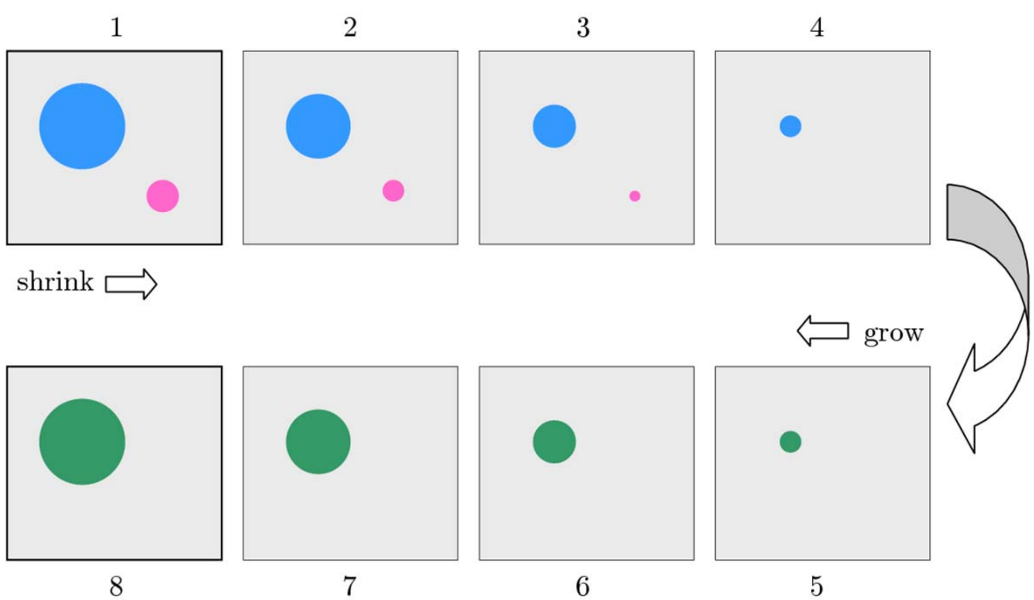


(a) (b) (c)

Growing: add layer of pixels at border

Shrinking and Let Grow

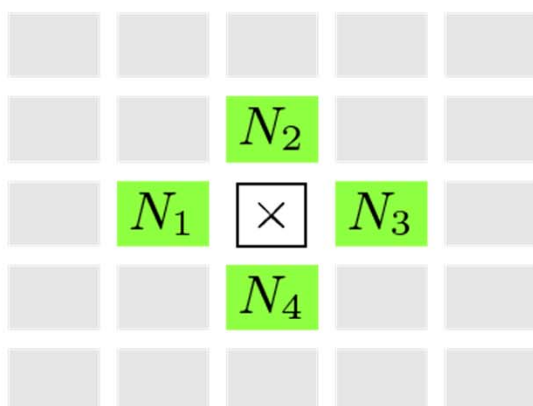
- Image structures are iteratively shrunk by peeling off a layer of thickness (layer of pixel) at boundaries
- Shrinking removes smaller structures, leaving only large structures
- Remaining structures are then grown back by same amount
- Eventually, large structures back to original size while smaller regions have disappeared



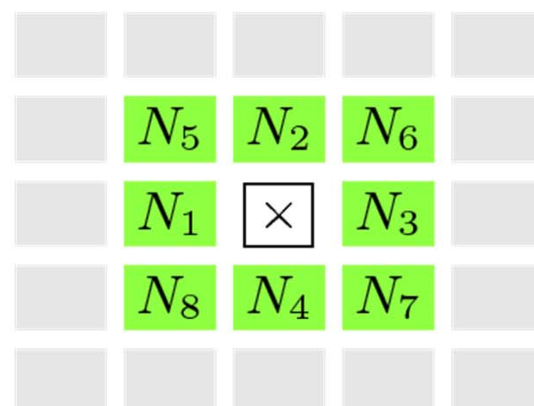
Basic Morphological Operations

- Definitions:
  - **4-Neighborhood (*N_4*):** 4 pixels adjacent to given pixel in horizontal and vertical directions
  - **8-Neighborhood (*N_8*): :** 4 pixels in N_4 + 4 pixels adjacent along diagonals

$\mathcal{N}_4$



$\mathcal{N}_8$



Formal Specification as Point Sets

- Morphological operations can be expressed by describing images as **2D point sets**
- For example, for a binary image ($I(u,v) \in \{0,1\}$)

$$\mathcal{Q}_I = \{\boldsymbol{p} \mid I(\boldsymbol{p}) = 1\}$$

- **Example:** OR operation union of individual sets

$$\mathcal{Q}_{I_1 \vee I_2} = \mathcal{Q}_{I_1} \cup \mathcal{Q}_{I_2}$$

Dilation

- Suppose *A* and *B* are sets of pixels, **dilation of A by B**

$$A \oplus B = \bigcup_{x \in B} A_x.$$

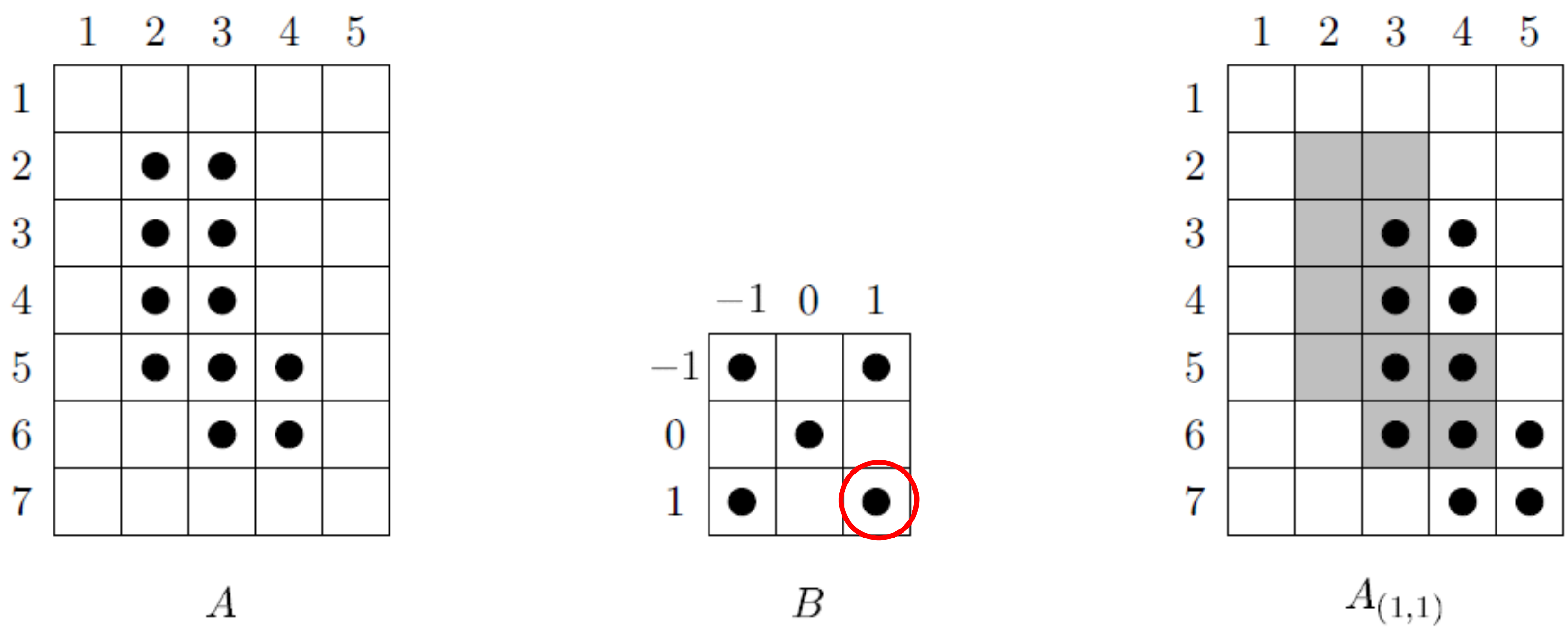
- Also called **Minkowski addition**. **Meaning?**
- Replace every pixel in *A* with copy of *B* (or vice versa)
- For every pixel *x* in *B*,
  - Translate *A* by *x*
  - Take union of all these translations

$$A \oplus B = \{(x, y) + (u, v) : (x, y) \in A, (u, v) \in B\}.$$

Dilation Example

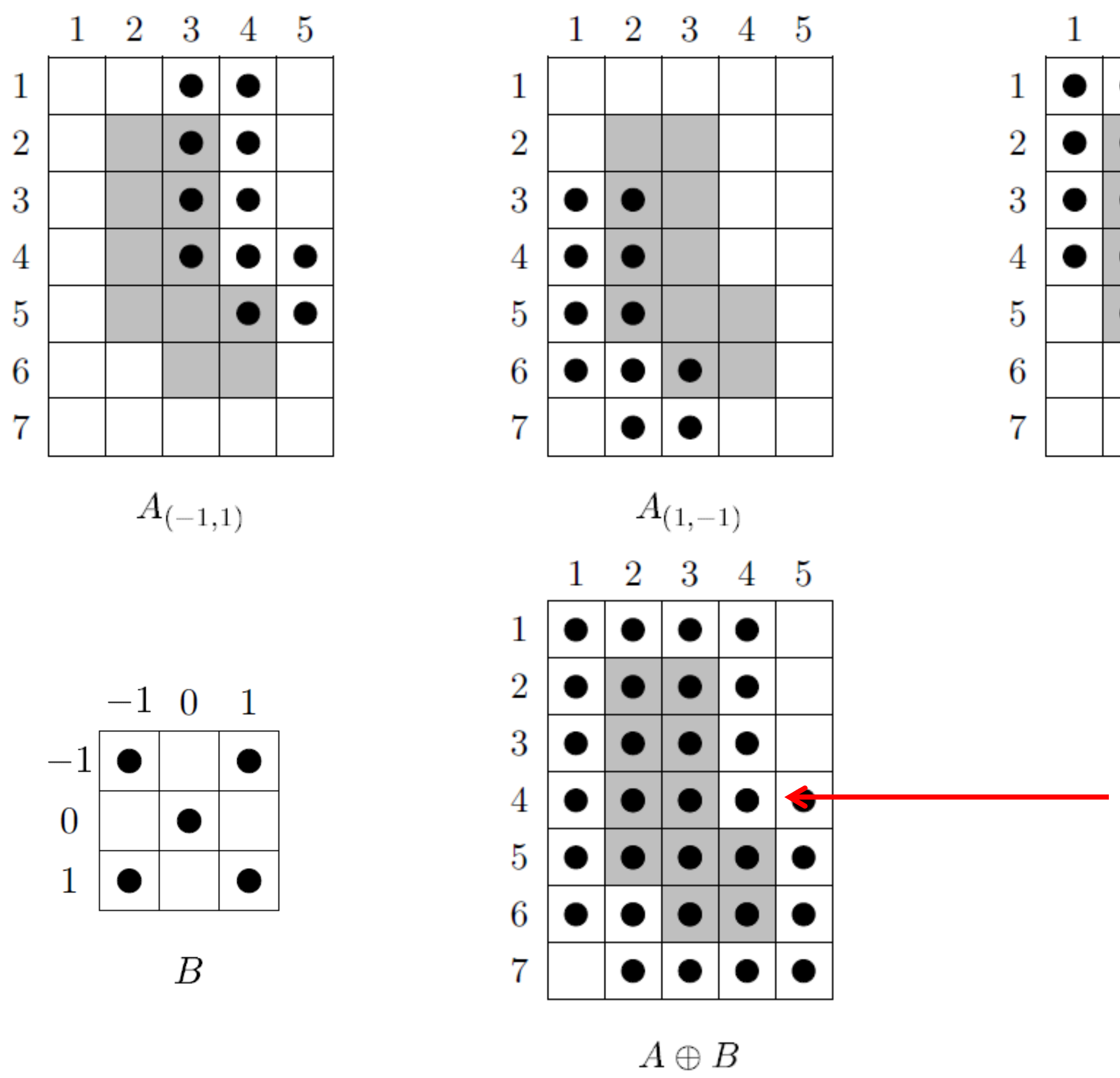
- For *A* and *B* shown below

$$B = \{(0,0),(1,1),(-1,1),(1,-1),(-1,-1)\}$$



Translation of *A* by (1,1)

Dilation Example



Union of all translations

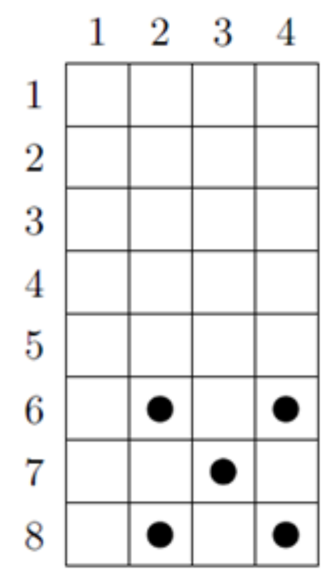
Another Dilation Example

- Dilation increases size of structure
- *A* and *B* do not have to overlap
- **Example:** For the same *A*, if we change *B* to

$$B = \{(7,3), (6,2), (6,4), (8,2), (8,4)\}$$

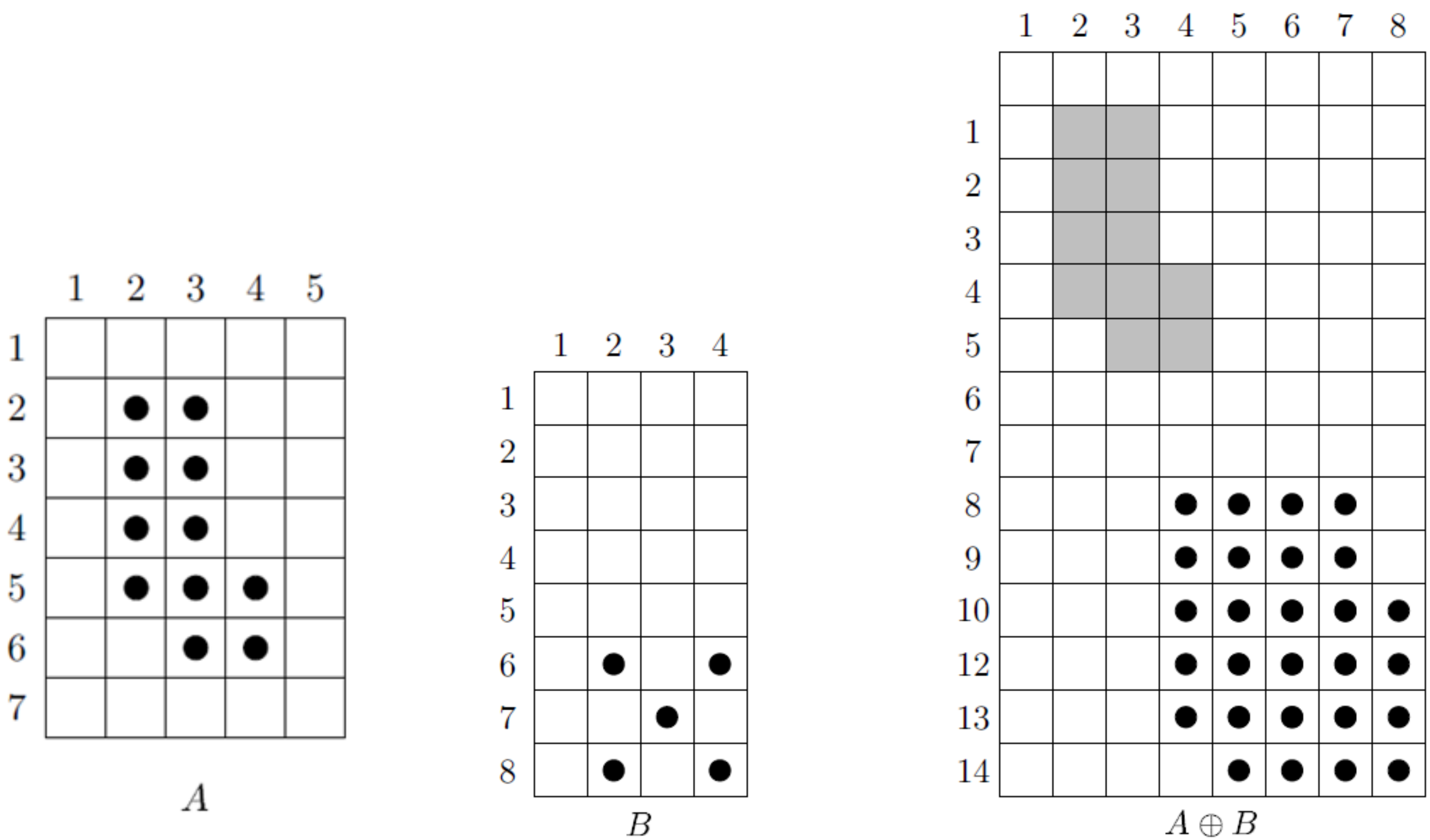
so that

$$A \oplus B = A_{(7,3)} \cup A_{(6,2)} \cup A_{(6,4)} \cup A_{(8,2)} \cup A_{(8,4)}$$

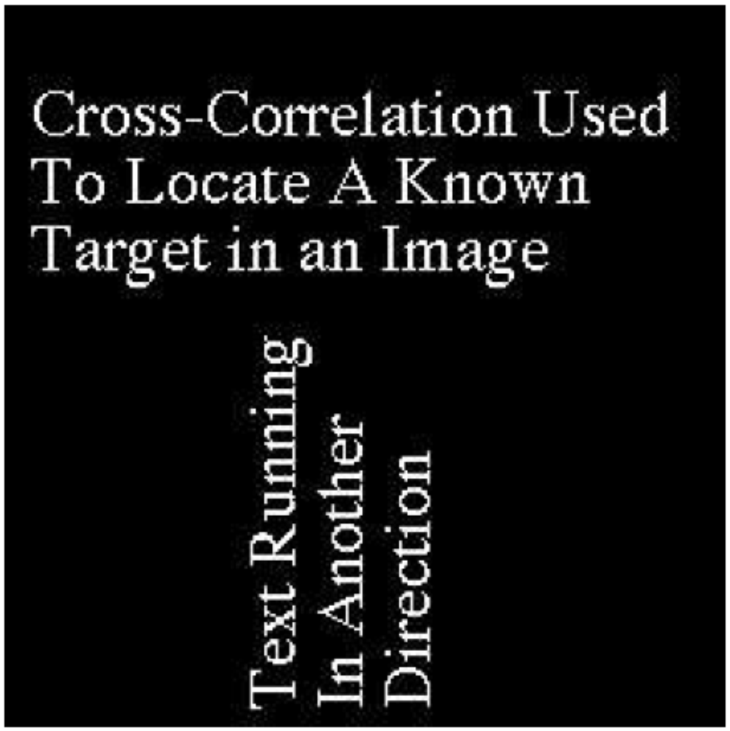


B

Another Dilation Example

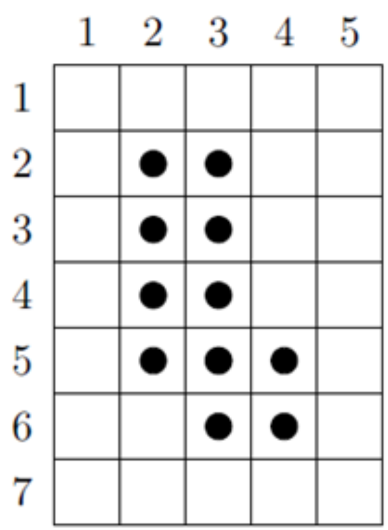


Example: Dilation of a Binary Image

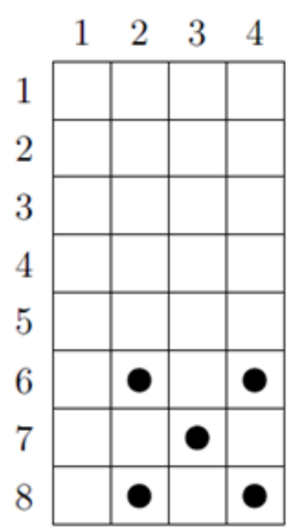


Dilation

- We usually assume
  - *A* is being processed
  - *B* is a smaller set of pixels, called the **structuring element**



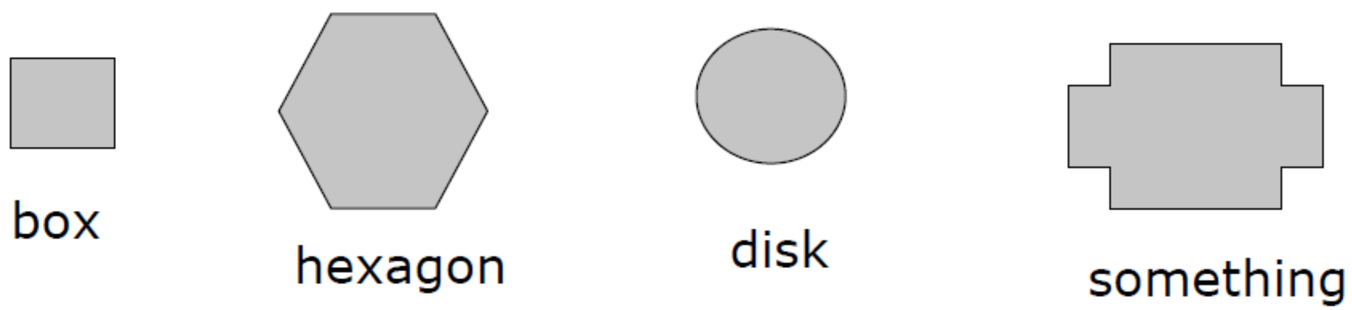
A



B

The Structuring Element

- A structuring element is a shape mask used in the basic morphological operations
- They can be any shape and size that is digitally representable, and each has an origin.



The Structuring Element

- Structuring element somewhat similar to a filter
- Contains only 0 and 1 values
- **Hot spot** marks origin of coordinate system of ***H***
- **Example of structuring element:** 1-elements marked with •, 0-cells are empty

$$H(i,j) \in \{0,1\}$$

$$H = \begin{array}{|c|c|c|} \hline & \bullet & \\ \hline \bullet & \boxed{\bullet} & \bullet \\ \hline & \bullet & \\ \hline \end{array}$$

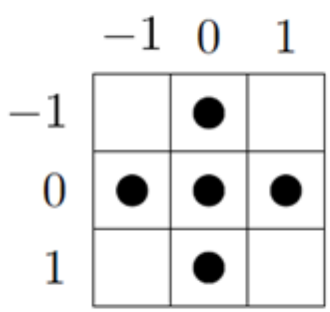
■ origin (hot spot)

Erosion

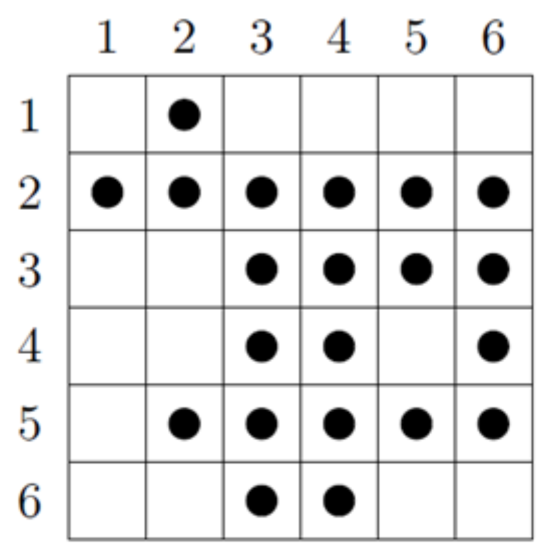
- Given sets A and B, the **erosion of *A* by *B***

$$A \ominus B = \{w : B_w \subseteq A\}.$$

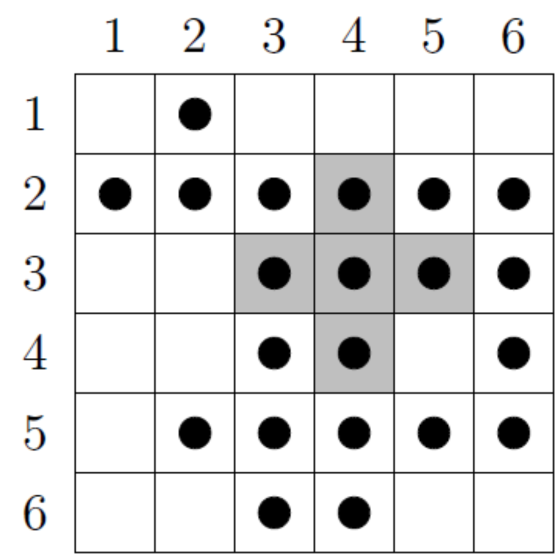
- Find all occurrences of B in A



B



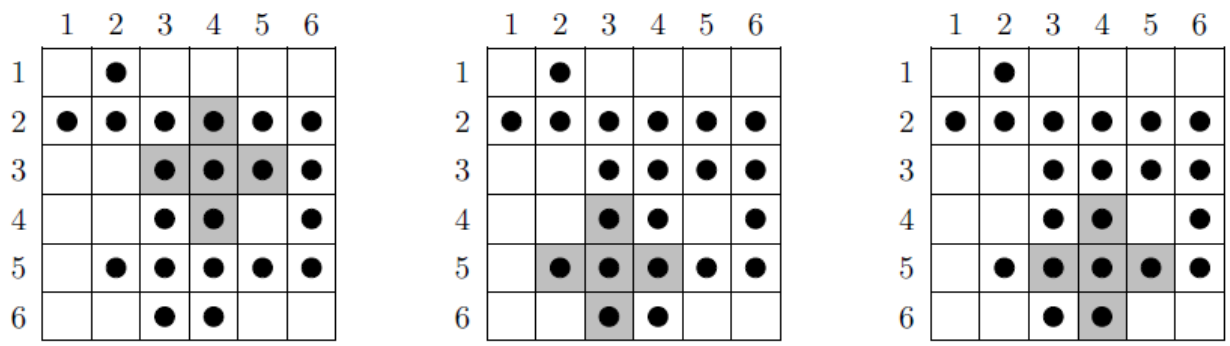
A



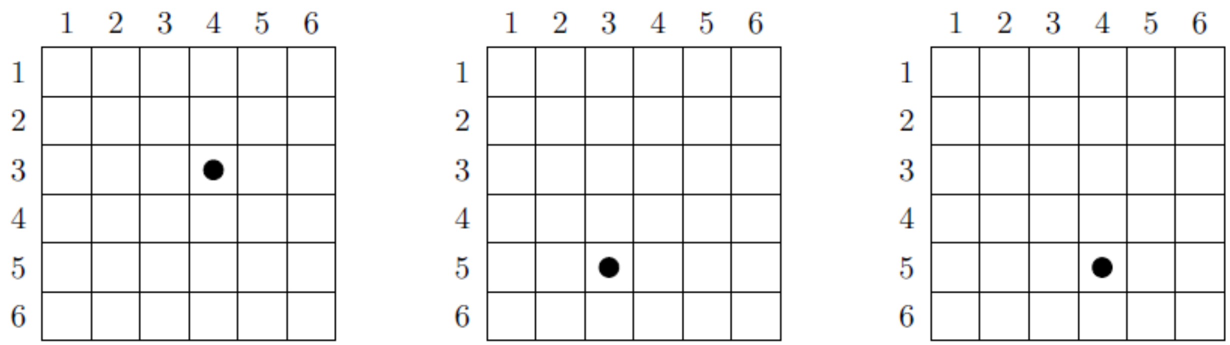
Example: 1 occurrence of *B* in *A*

Erosion

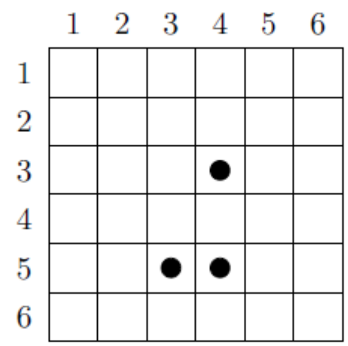
All occurrences of *B* in *A*



For each occurrences Mark center of *B*



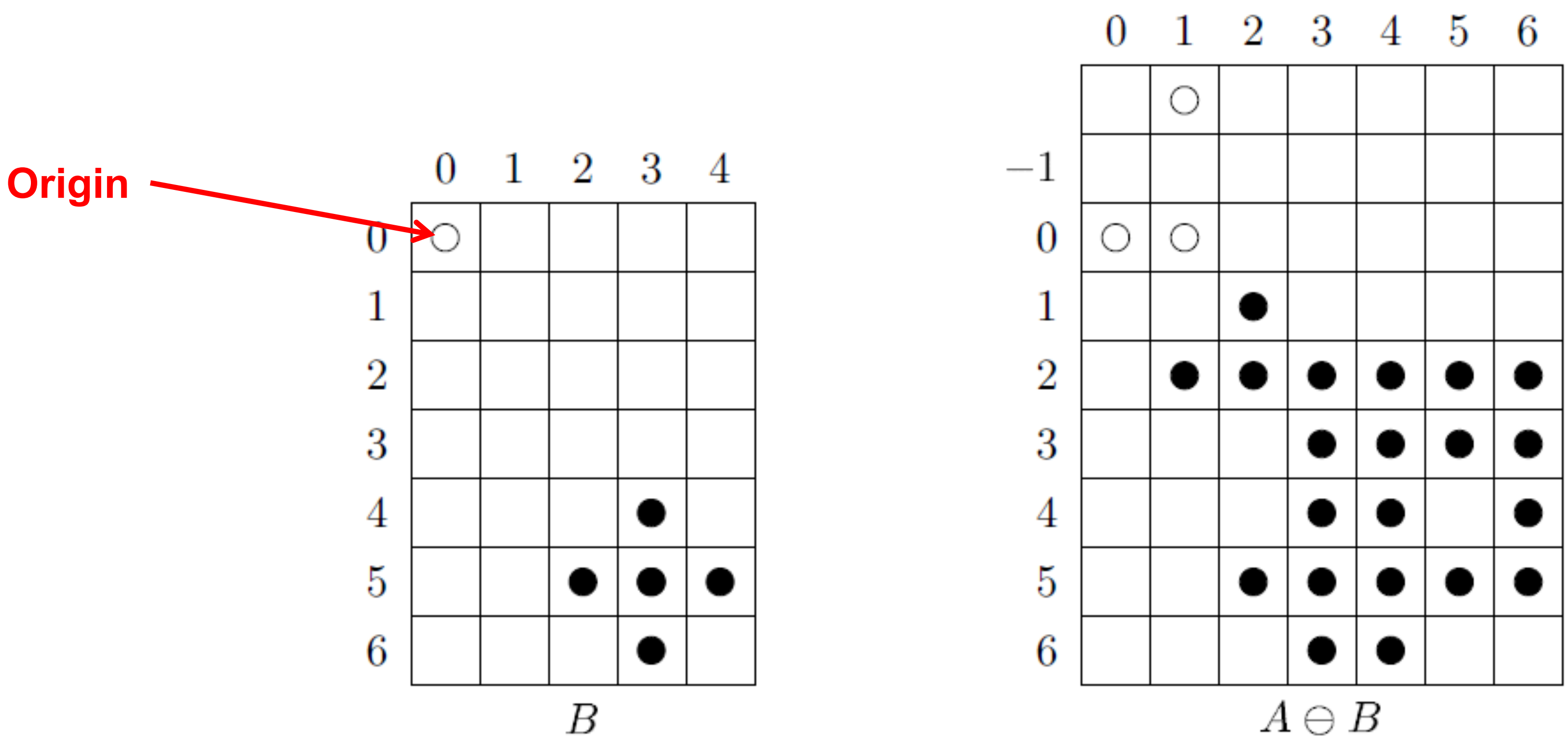
Erosion: union of center of all occurrences of *B* in *A*



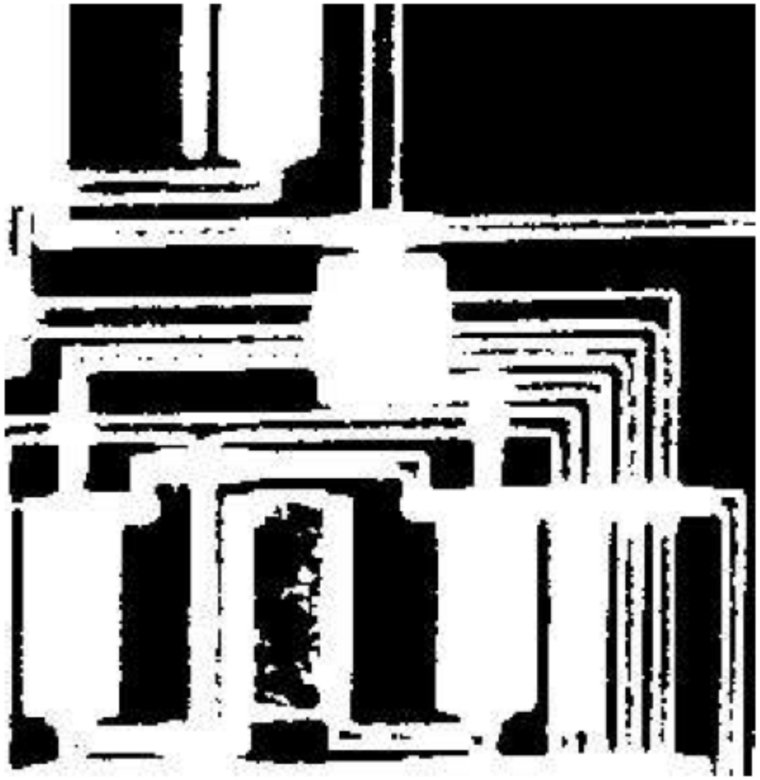
$A \ominus B$

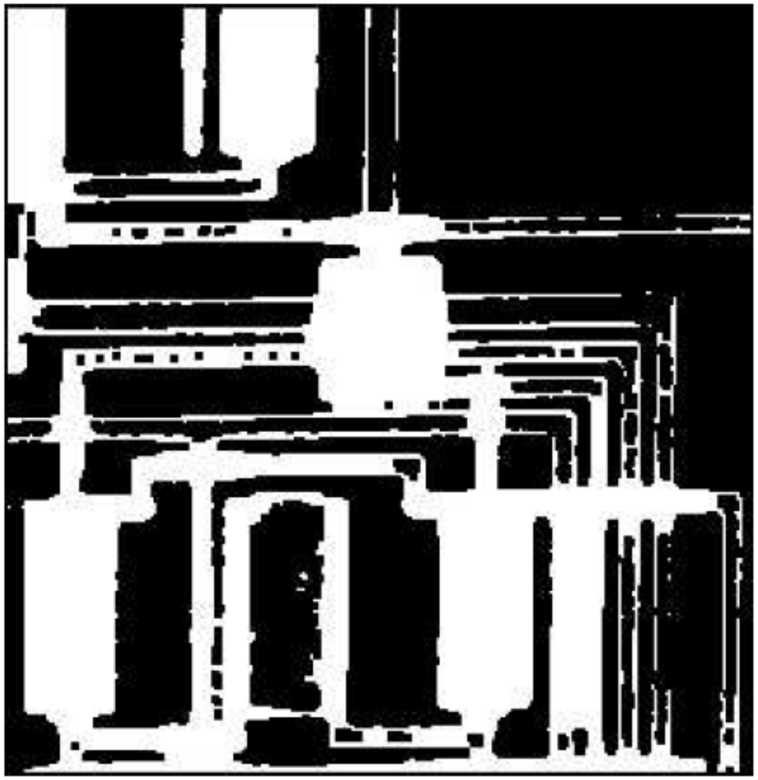
Another Erosion Example

- The structuring element (*B*) does not have to contain the origin
- Another example where ***B*** does not contain the origin



Example: Erosion of Binary Image





Erosion

- Erosion related to **minkowski subtraction**

$$A - B = \bigcap_{b \in B} A_b.$$

- Erosion and dilation are **inverses** of each other
- It can be shown that

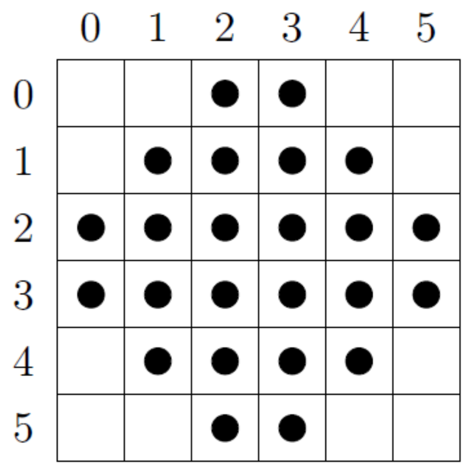
$$\overline{A \ominus B} = \overline{A} \oplus \hat{B}.$$

- And also that

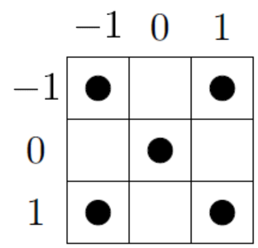
$$\overline{A \oplus B} = \overline{A} \ominus \hat{B}.$$

An Application: Boundary Detection

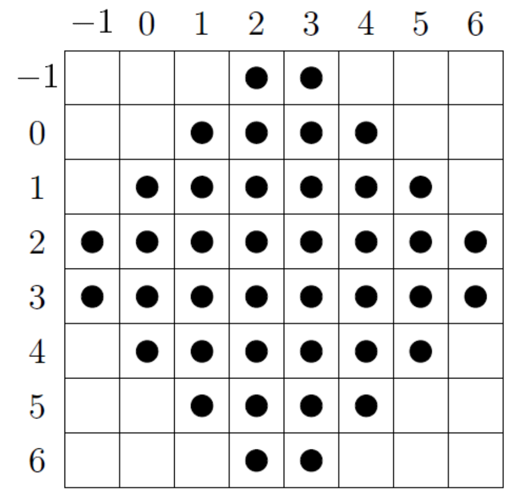
- Given an image *A* and structuring element *B*



A

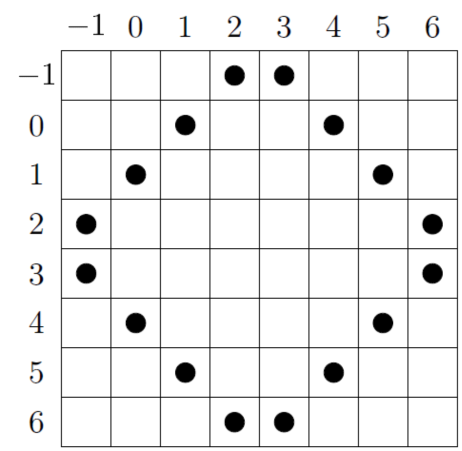


B



$A \oplus B$

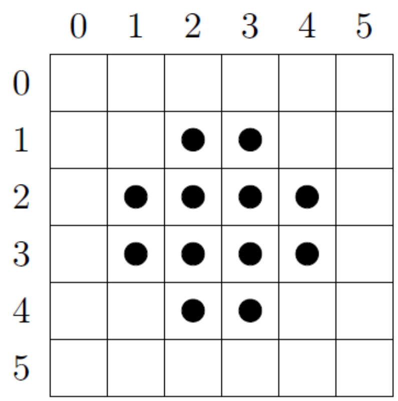
We can define **external boundary**



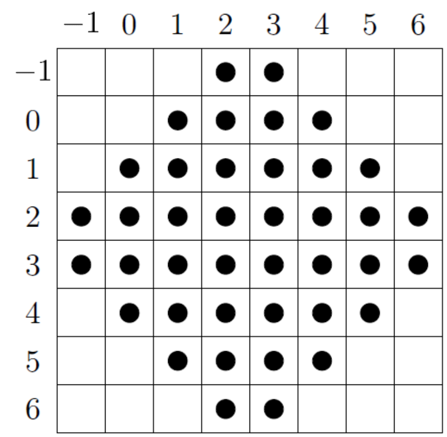
$(A \oplus B) - A)$

An Application: Boundary Detection

- **Dilation of image *A* - erosion image *A*** (by structuring element *B*)



$A \ominus B$

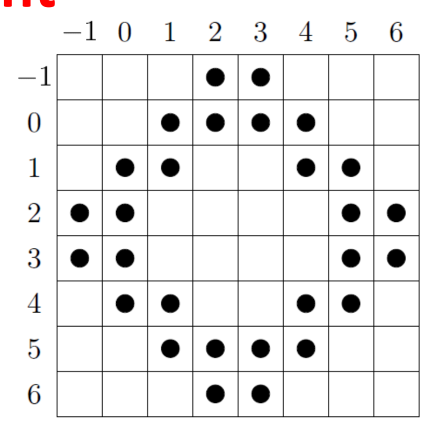


$A \oplus B$

- We can define **morphological gradient**

$$(A \oplus B) - (A \ominus B)$$

Morphological gradient = Dilation - erosion

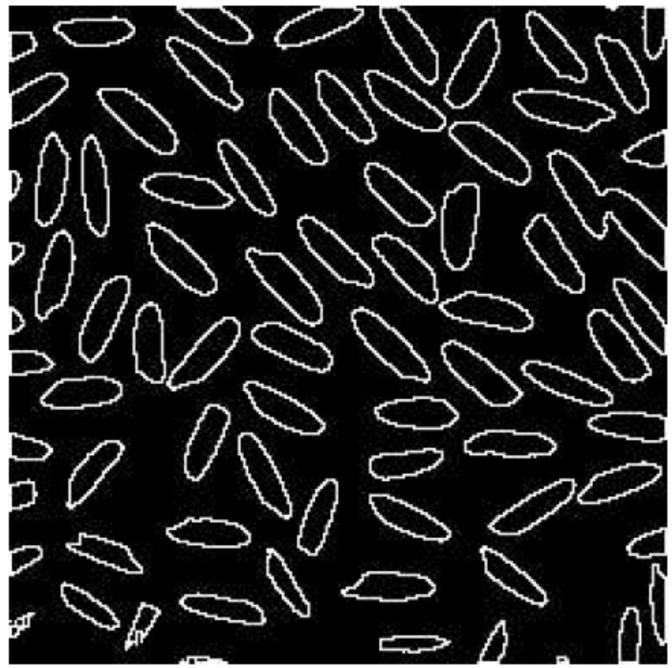


$(A \oplus B) - (A \ominus B)$

Example: Internal Boundary of Binary Image

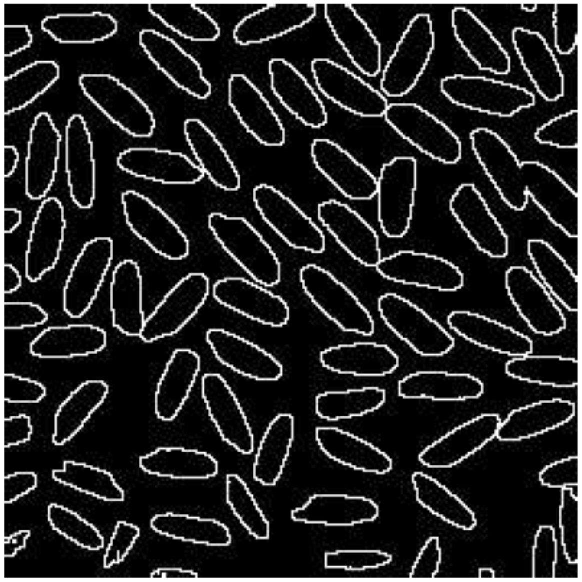
- We can also define **internal boundary** as

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



Example: External Boundary and Morphological Gradient

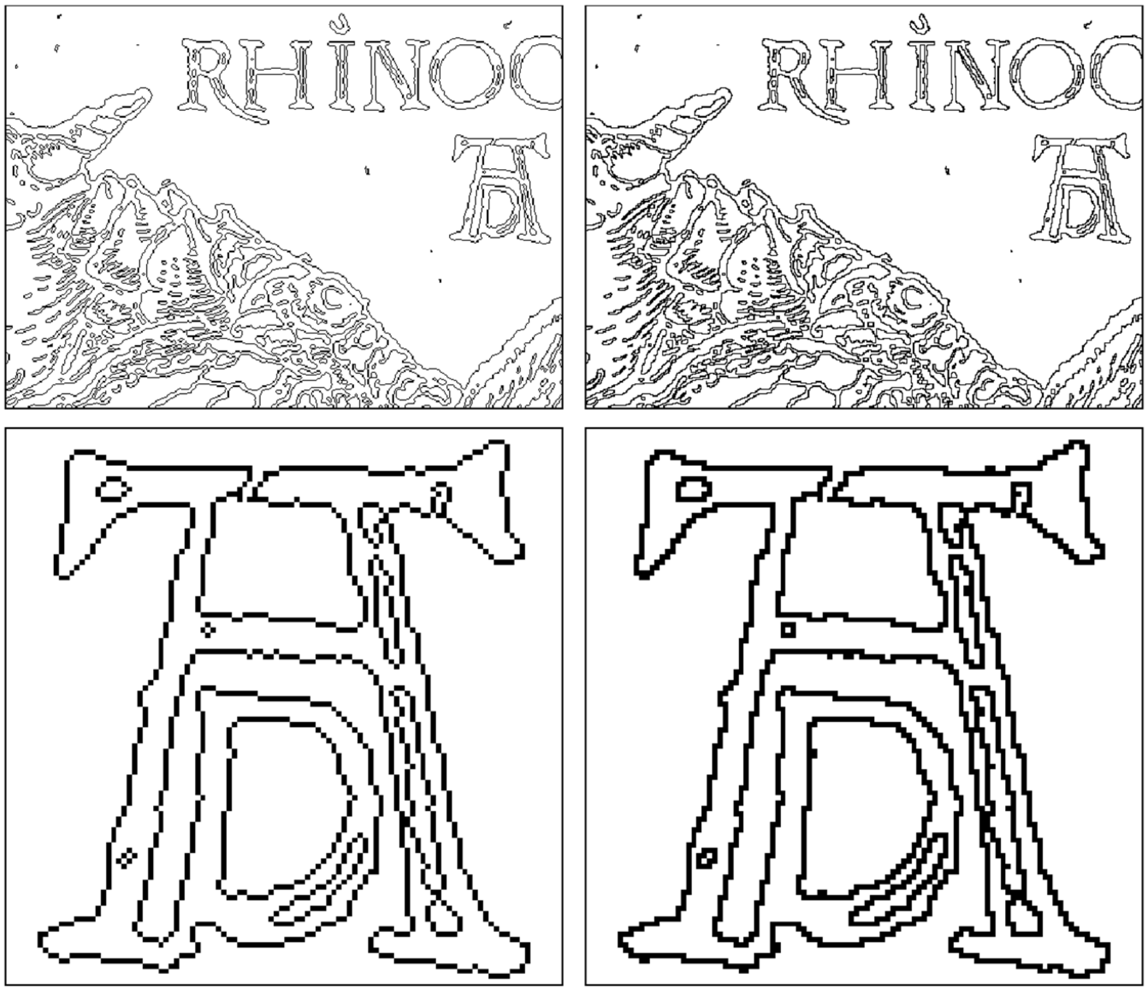
Image



External Boundary

Morphological Gradient

Example: Extraction of Boundary Pixels using Morphological Operations



(a) (b)

Properties of Dilation

- Dilation operation is **commutative**

$$I \oplus H = H \oplus I$$

- Dilation is **associative** (ordering of applying it not important)

$$(I_1 \oplus I_2) \oplus I_3 = I_1 \oplus (I_2 \oplus I_3)$$

- Thus as with separable filters, more efficient to apply large structuring element as sequence of smaller structuring elements

$$I \oplus H_{\text{big}} = (\ldots((I \oplus H_1) \oplus H_2) \oplus \ldots \oplus H_K)$$

Properties of Erosion

- Erosion is **not commutative**

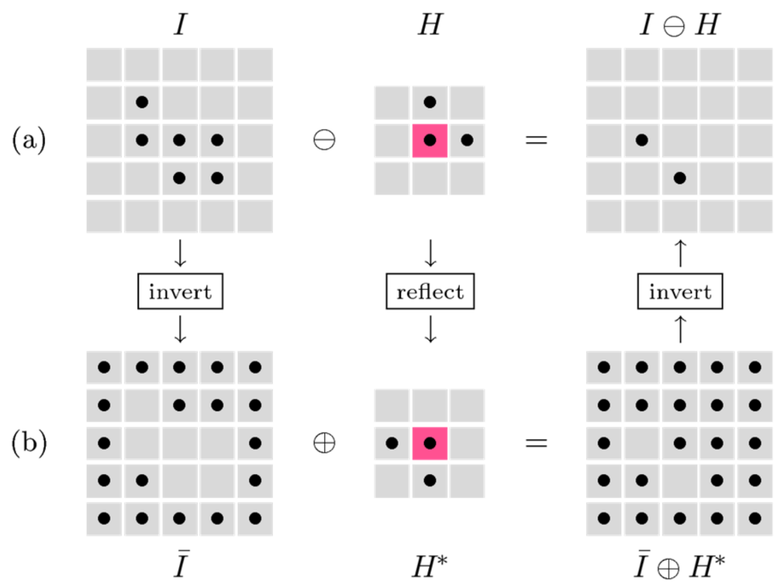
$$I \ominus H \neq H \ominus I$$

- If erosion and dilation are combined, this chain rule holds

$$(I_1 \ominus I_2) \ominus I_3 \;=\; I_1 \ominus (I_2 \oplus I_3)$$

- Dilation of **foreground** = inverting (erosion of **background**)

$$I \oplus H \;\equiv\; \overline{(\bar{I} \ominus H^*)}$$



Dilation and Erosion Algorithm

DILATE (I, H)

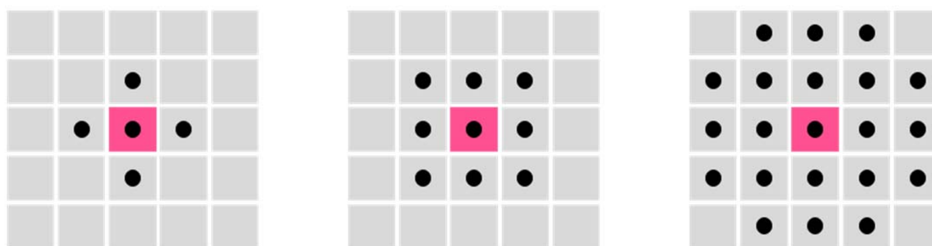
I: binary image of size $w \times h$
H: binary structuring element defined over region $\mathcal{R}_H$
Returns the dilated image $I' = I \oplus H$

$I' \leftarrow$ new binary image of size $w \times h$
$I'(u,v) \leftarrow 0,$ for all (u,v) $\quad \triangleright\ I' \leftarrow \varnothing$
for all $(i,j) \in \mathcal{R}_H$ **do** $\quad \triangleright\ (i,j) = \boldsymbol{q}$
  if $H(i,j) = 1$ **then** $\quad \triangleright\ \boldsymbol{q} \in H$
    MERGE THE SHIFTED $I_{\boldsymbol{q}}$ WITH I': $\quad \triangleright\ I' \leftarrow I' \cup I_{\boldsymbol{q}}$
    for $u \leftarrow 0 \ldots (w-1)$ **do**
      for $v \leftarrow 0 \ldots (h-1)$ **do** $\quad \triangleright\ (u,v) = \boldsymbol{p}$
        if $I(u,v) = 1$ **then** $\quad \triangleright\ \boldsymbol{p} \in I$
          $I'(u+i, v+j) \leftarrow 1$ $\quad \triangleright\ I' \leftarrow I' \cup (\boldsymbol{p}+\boldsymbol{q})$
return I'.

ERODE (I, H)
$\bar{I} \leftarrow$ INVERT(I) $\quad \triangleright\ \bar{I} \leftarrow \neg I$
$H^* \leftarrow$ REFLECT(H)
return INVERT(DILATE$(\bar{I}, H^*)$). $\quad \triangleright\ I \ominus H = \overline{(\bar{I} \oplus H^*)}$

Designing Morphological Filters

- A morphological filter is specified by:
  - Type of operation (e.g. dilation, erosion)
  - Contents of structuring element



(a) 4-neighborhood

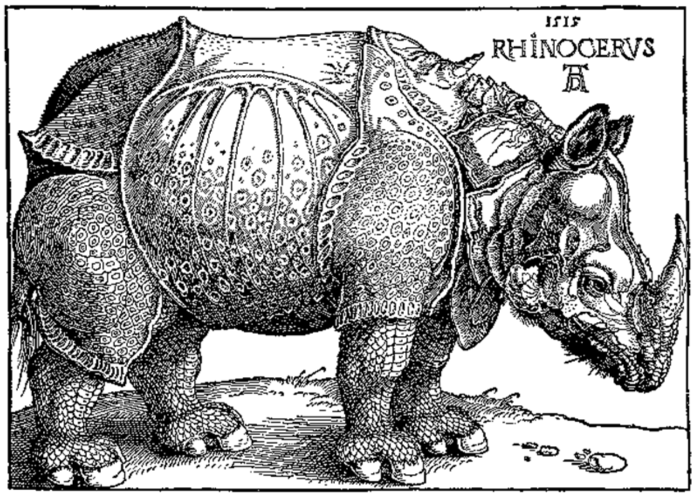
(b) 8-neighborhood

(c) Small Disk (circular)

- In practice, quasi-circular shaped structuring elements used
- Dilation with circular structuring of radius ***r*** adds thickness ***r***
- Erosion with circular structuring of radius ***r*** removes thickness ***r***

Example: Dilation and Erosion

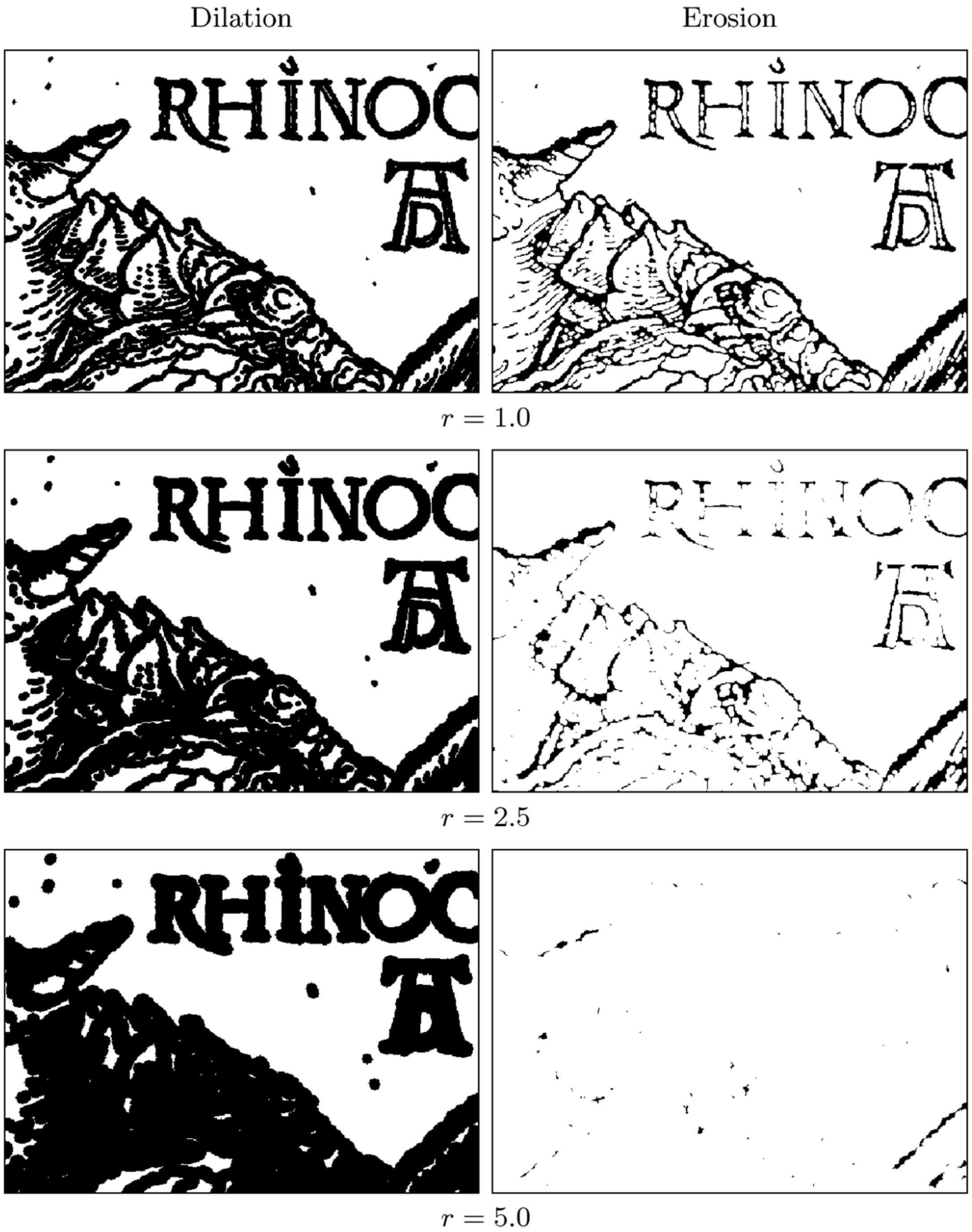
- What if we erode and dilate the following image with disk-shaped structuring element?



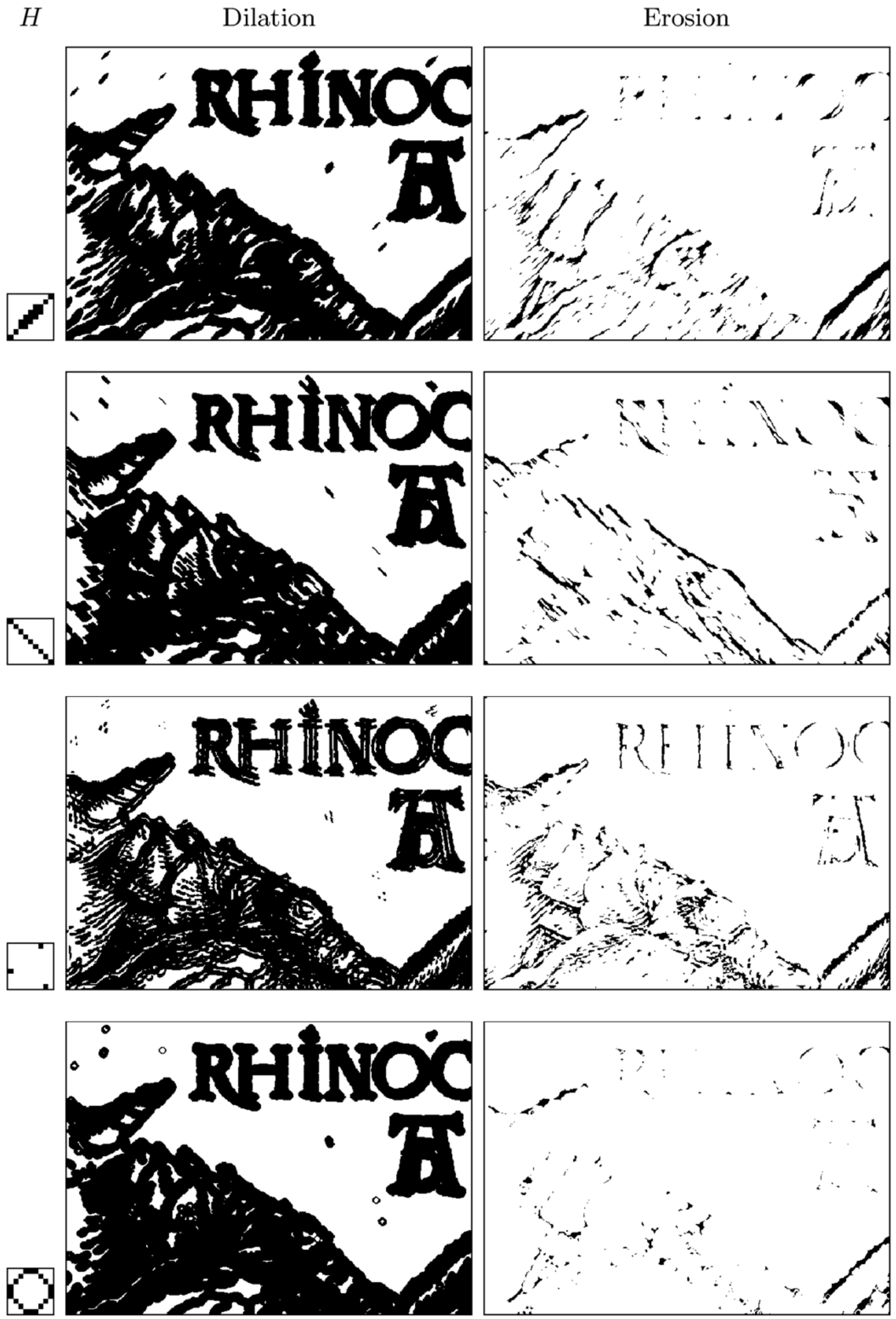
Original image

Apply dilation and erosion to this close up section

Example: Dilation and Erosion



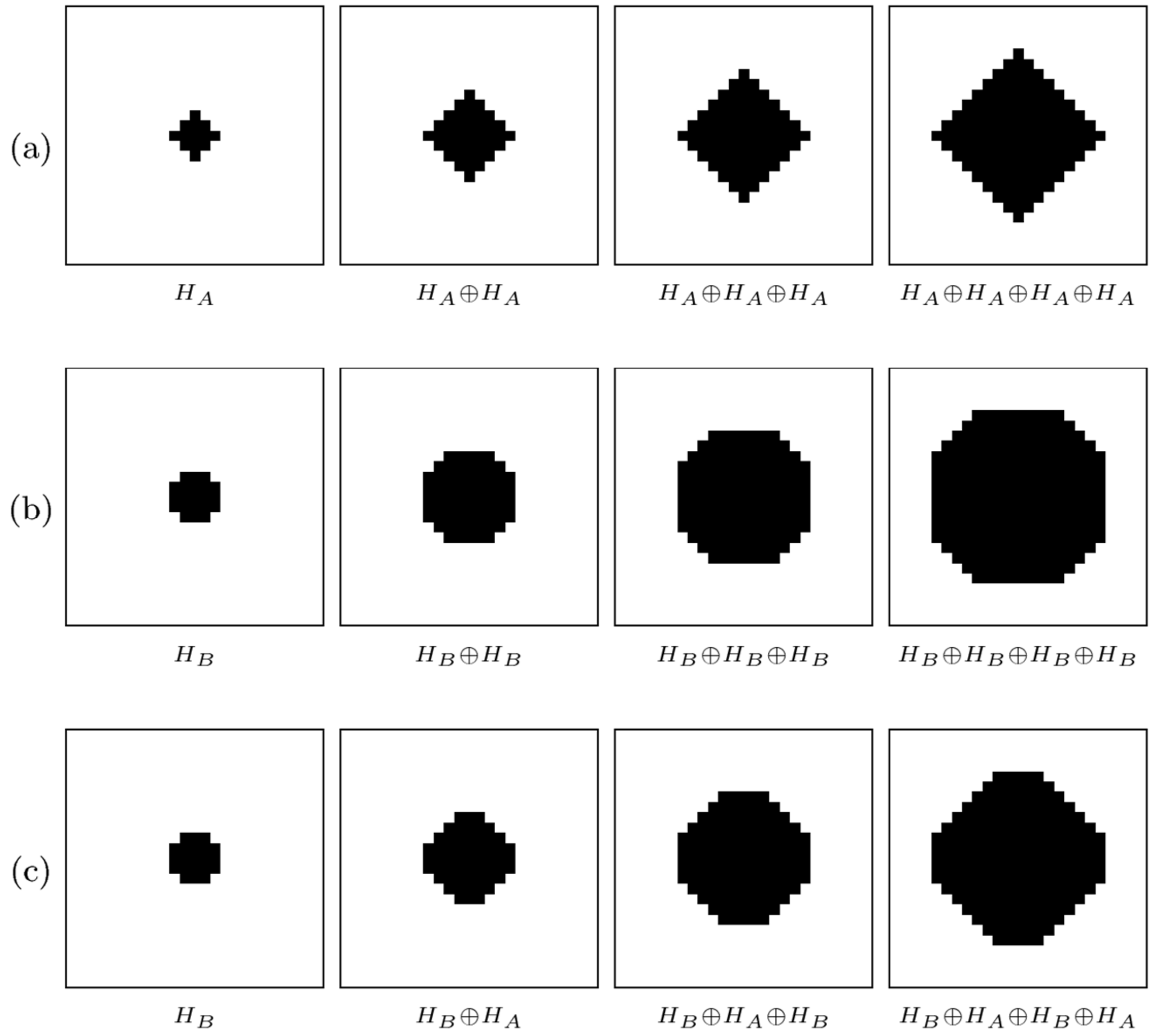
Dilation and Erosion using Different Structuring Elements



Example: Composing Large Filters by Repeatedly Applying Smaller Filters

- More efficient
- E.g. composing

Isotropic filter



References

- Wilhelm Burger and Mark J. Burge, Digital Image Processing, Springer, 2008
- University of Utah, CS 4640: Image Processing Basics, Spring 2012
- Rutgers University, CS 334, Introduction to Imaging and Multimedia, Fall 2012
- Gonzales and Woods, Digital Image Processing (3rd edition), Prentice Hall