Digital Image Processing

Morphological Image Processing

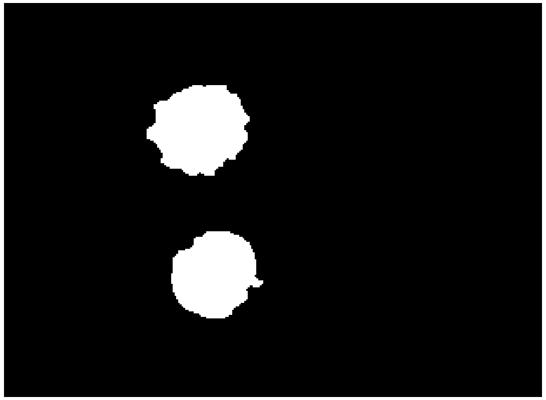
Presented By:
Diwakar Yagyasen
Sr. Lecturer
CS&E, BBDNITM, Lucknow

Preview

- Morphology
  - About the **form** and **structure** of animals and plants
- Mathematical morphology
  - Using set theory
  - Extract image component
  - Representation and description of region shape

Preview (cont.)

- Sets in mathematical morphology represent objects in an image



- Example
  - Binary image: the elements of a set is the coordinate (x,y) of the pixels, in $\mathbf{Z}^2$
  - Gray-level image: the element of a set is the triple, (x, y, gray-value), in $\mathbf{Z}^3$

Outline

Binary images

- Preliminaries – set theory
- Dilation and erosion
- Opening and closing
- Hit-or-miss transformation
- Some basic morphological algorithms
- Extensions to gray-scale images

Preliminaries – set theory

- A be a set in $\mathbf{Z}^2$.
- a = (a_1, a_2) is an element of A. $a \in A$
- a is not an element of A $a \notin A$
- Null (empty) set: $\varnothing$

Set theory (cont.)

- Explicit expression of a set

① $A = \{a_1, a_2, \ldots, a_n\}$

② $A = \{\, element \mid \text{condition for set elements} \,\}$

  - Example:

$$C = \{w \mid w = -d,\ \text{for } d \in D\}$$

Set operations

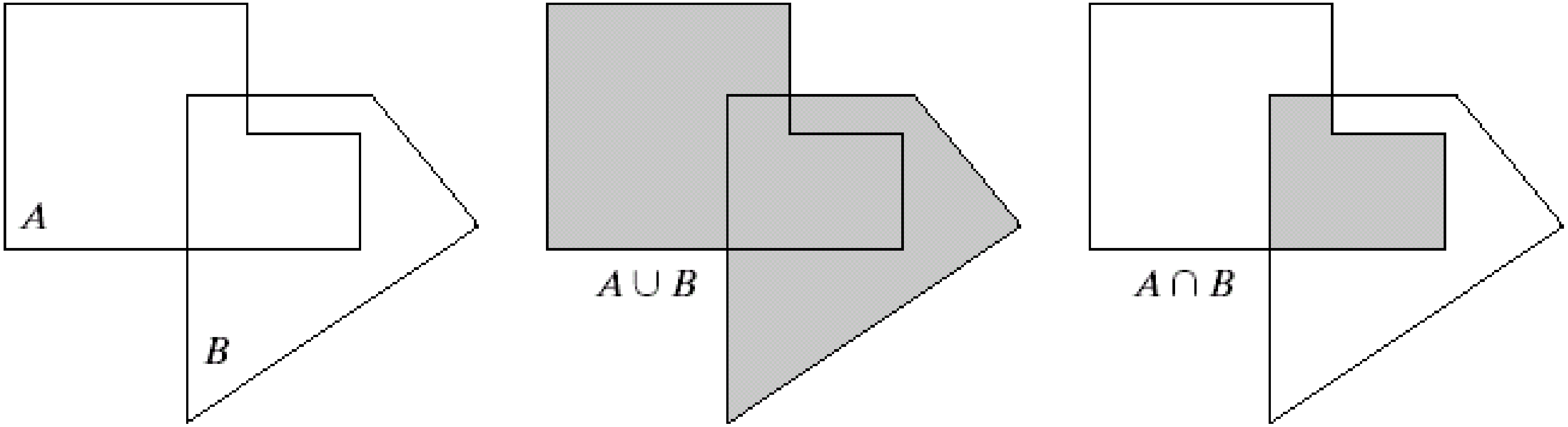
- A is a subset of B: every element of A is an element of another set B $A \subseteq B$

- Union $C = A \cup B$

- Intersection $C = A \cap B$

- Mutually exclusive $A \cap B = \varnothing$

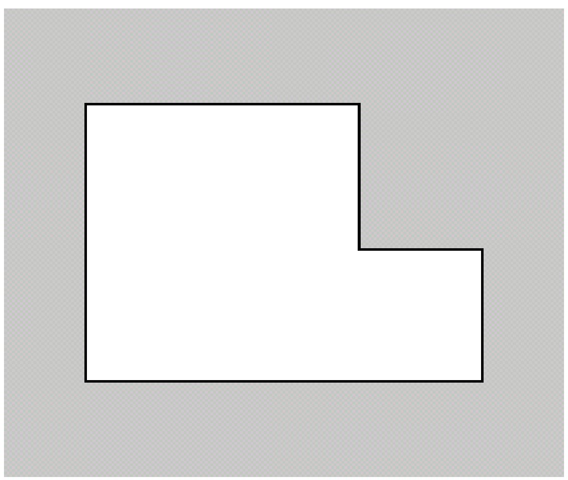
Graphical examples



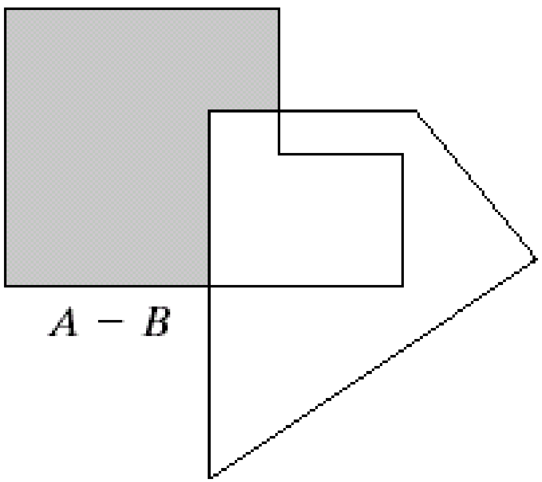
Graphical examples (cont.)

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

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



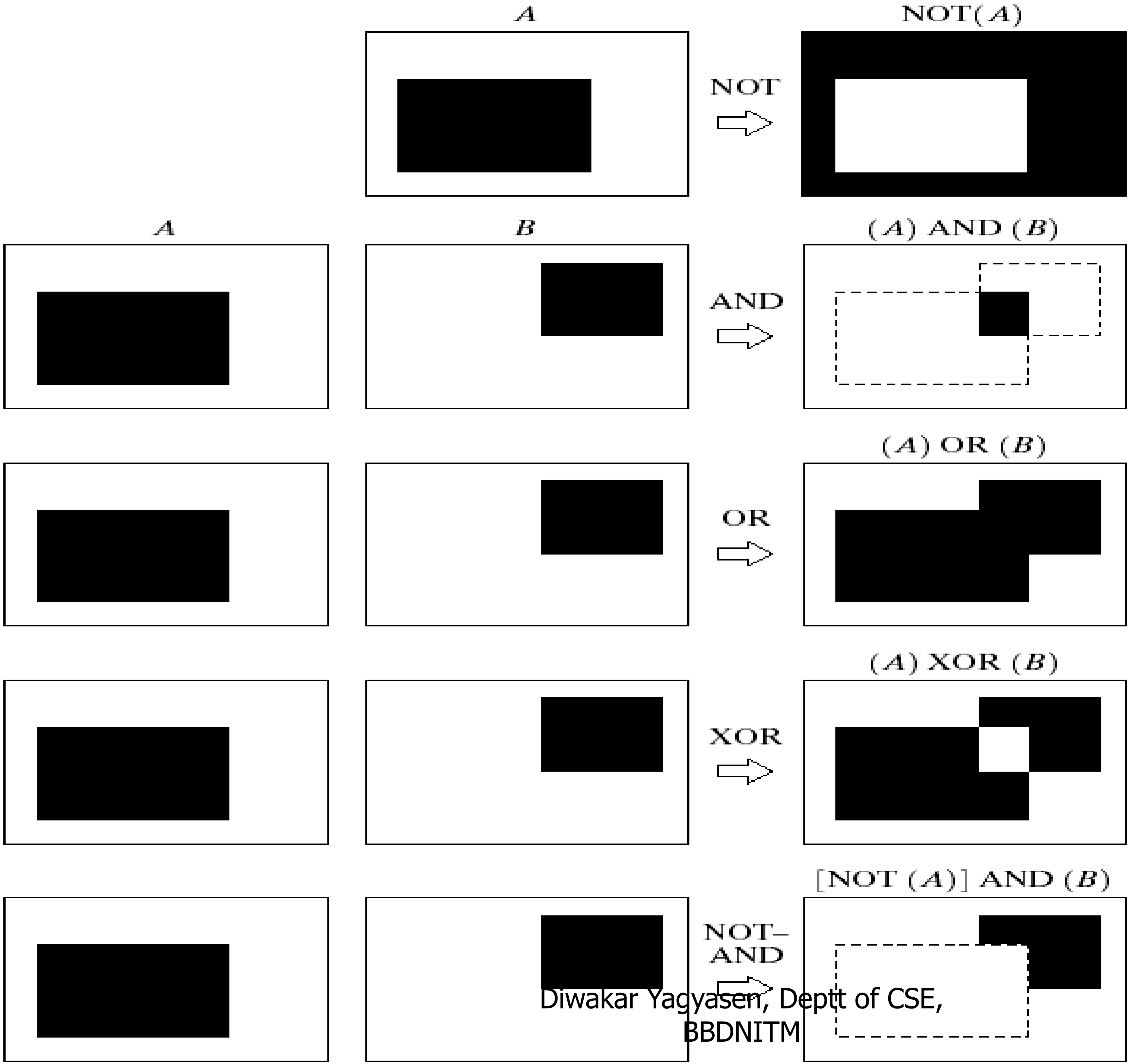
(A)^c



Logic operations on binary images

- Functionally complete operations
  - AND, OR, NOT

p	q	p AND q (also $p \cdot q$)	p OR q (also $p + q$)	NOT (p) (also $\bar{p}$)
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

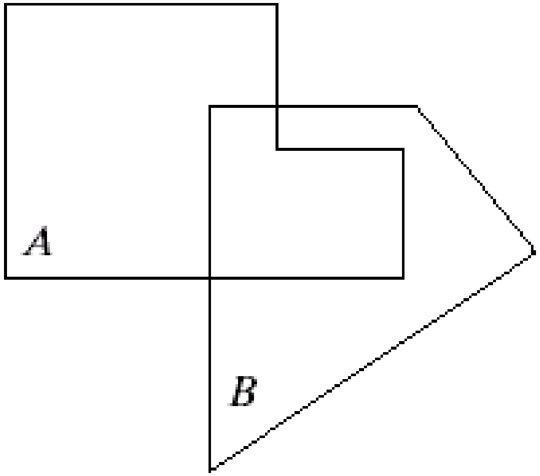


$$A \cap B$$

$$A \cup B$$

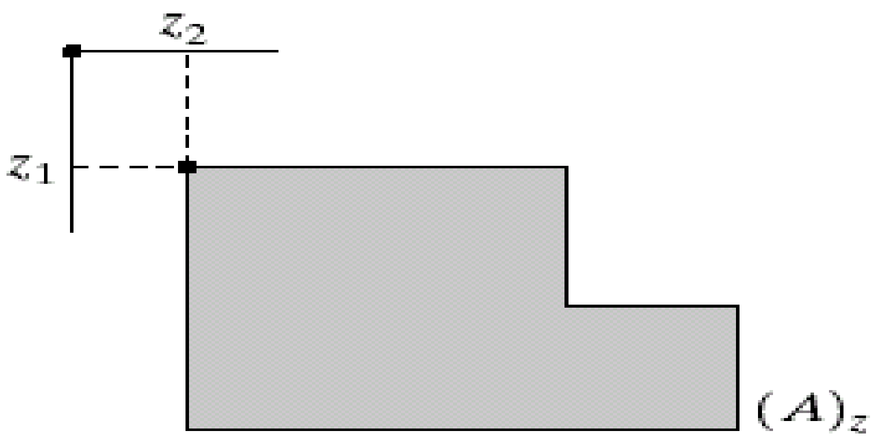
$$B - A$$

Special set operations for morphology



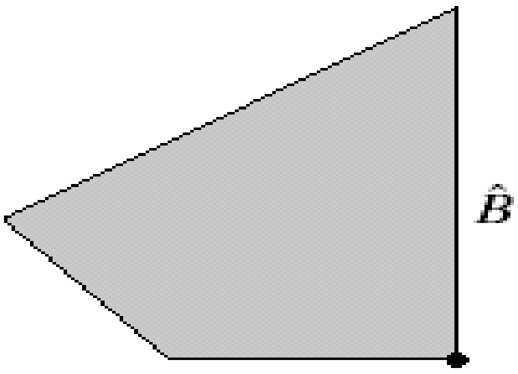
translation

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



reflection

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



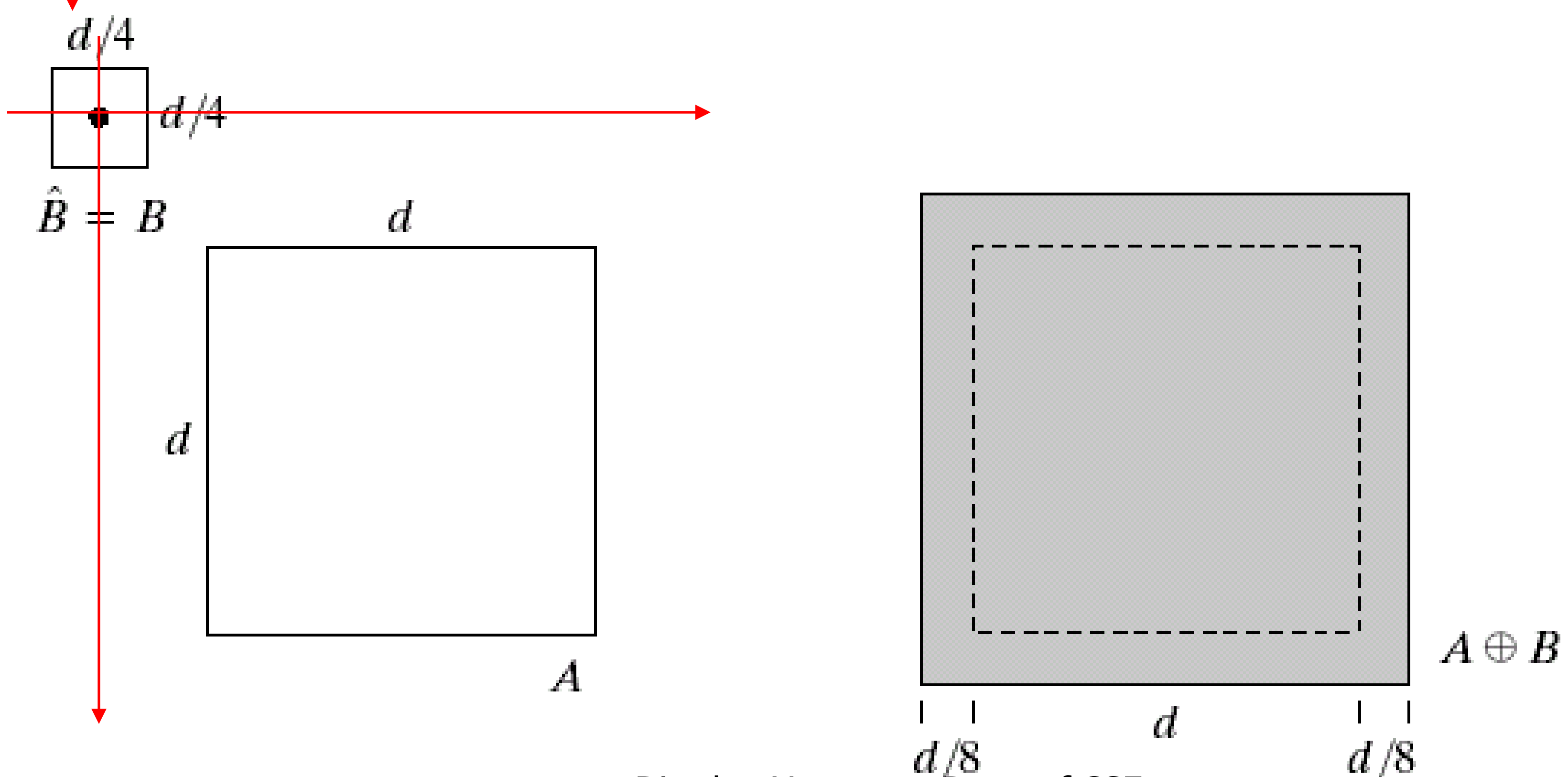
Outline

- Preliminaries
- Dilation and erosion
- Opening and closing
- Hit-or-miss transformation
- Some basic morphological algorithms
- Extensions to gray-scale images

Dilation

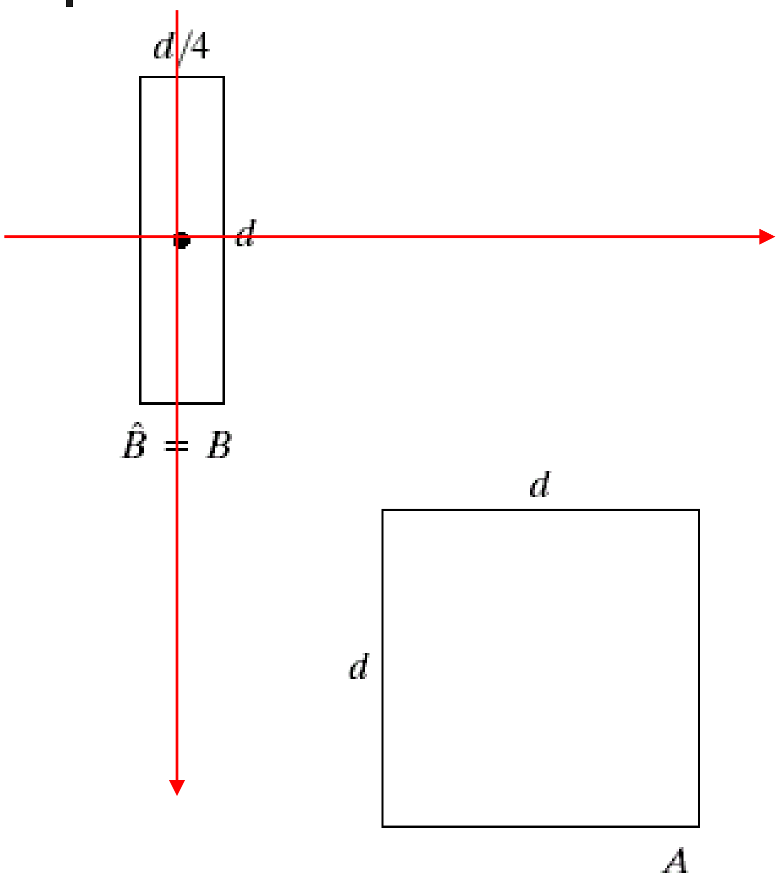
$$A \oplus B = \left\{ z \,\middle|\, (\hat{B})_z \cap A \neq \varnothing \right\}$$

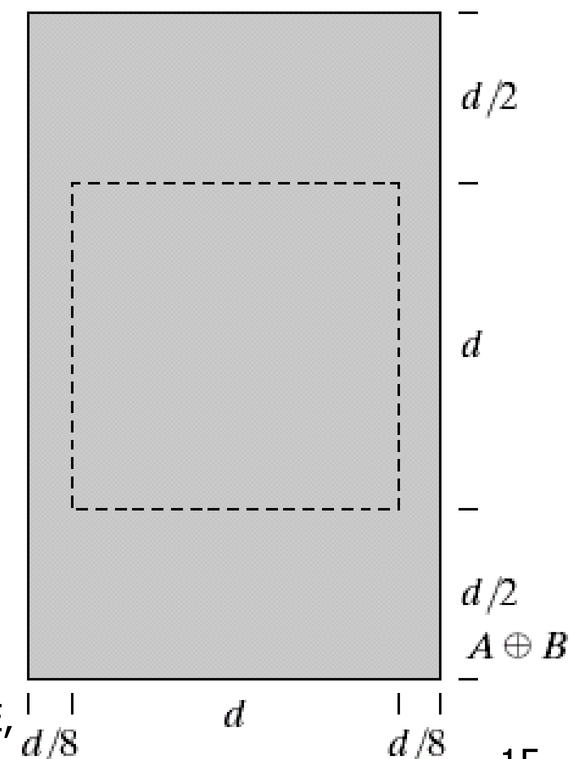
B:structuring element



Dilation: another formulation

$$A \oplus B = \left\{ z \,\middle|\, \left[(\hat{B})_z \cap A \right] \subseteq A \right\}$$



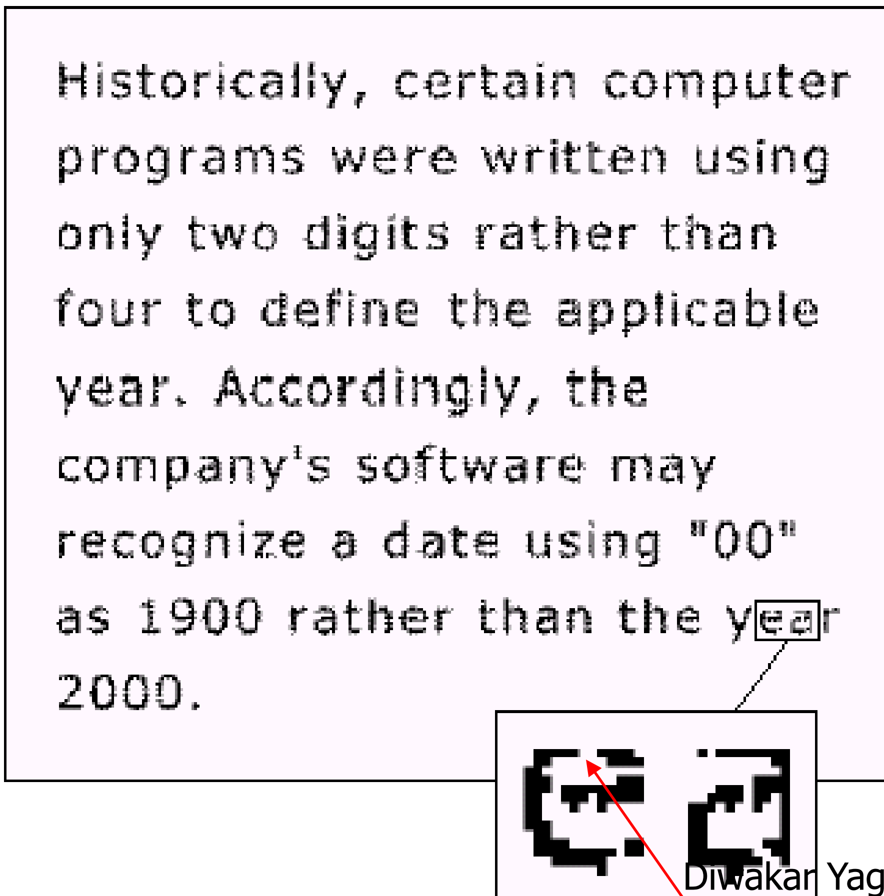


Application of dilation: bridging gaps in images

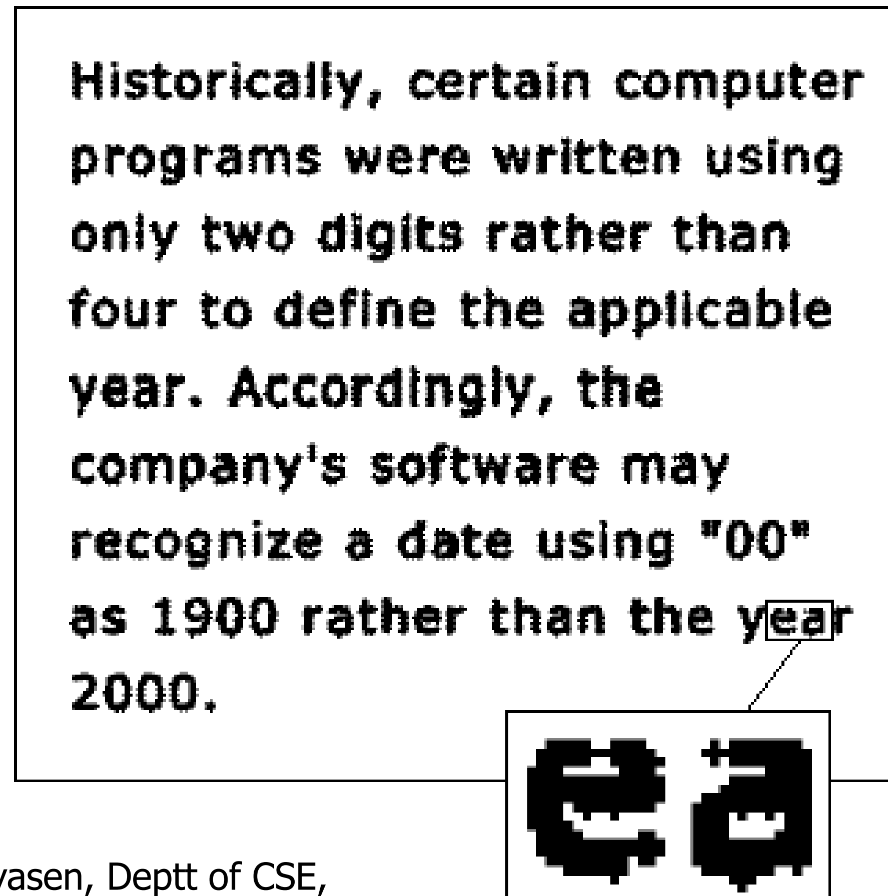
0	1	0
1	1	1
0	1	0

Structuring element

Effects: increase size, fill gap



Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

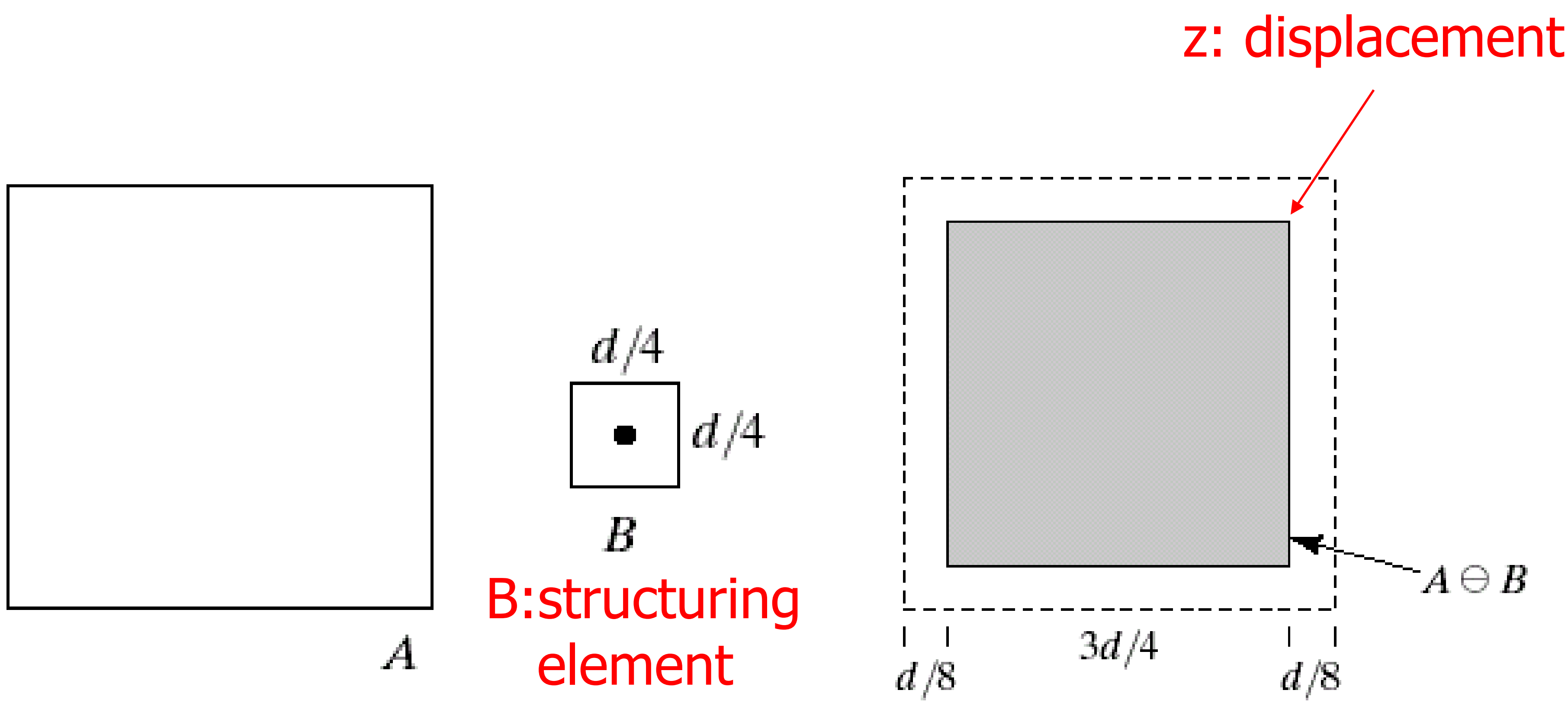


Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

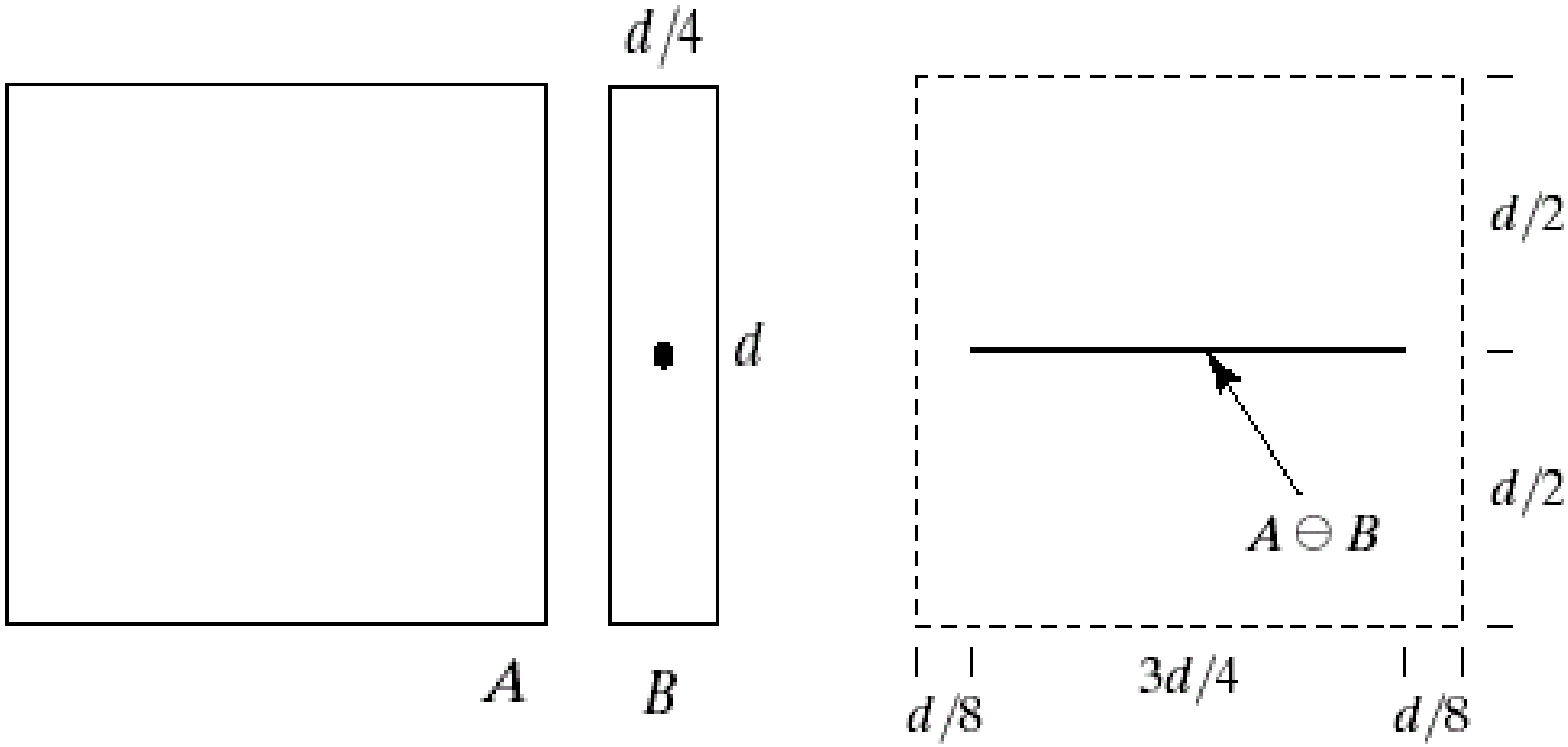
max. gap=2 pixels

Erosion

$$A \ominus B = \{ z \mid (B)_z \subseteq A \}$$



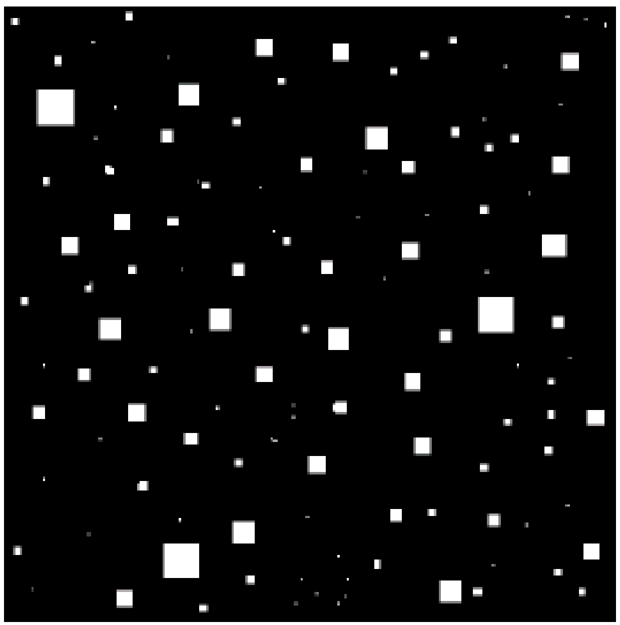
Erosion (cont.)



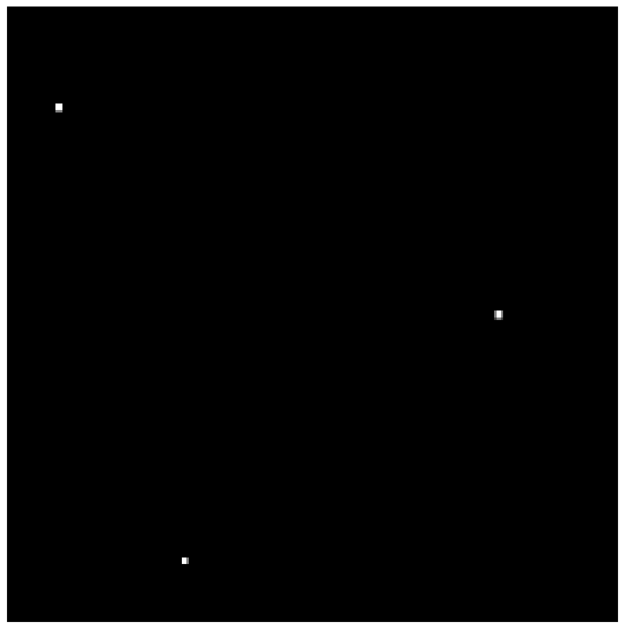
Application of erosion: eliminate irrelevant detail

Squares of size 1,3,5,7,9,15 pels

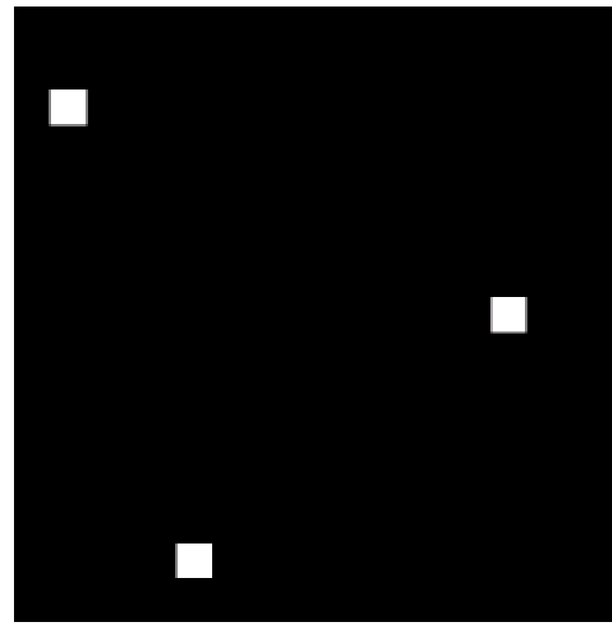
Erode with 13x13 square



original image



erosion



dilation

Dilation and erosion are duals

$$(A \ominus B)^c = \left\{ z \middle| (B)_z \subseteq A \right\}^c$$

$$= \left\{ z \middle| (B)_z \cap A^c = \varnothing \right\}^c$$

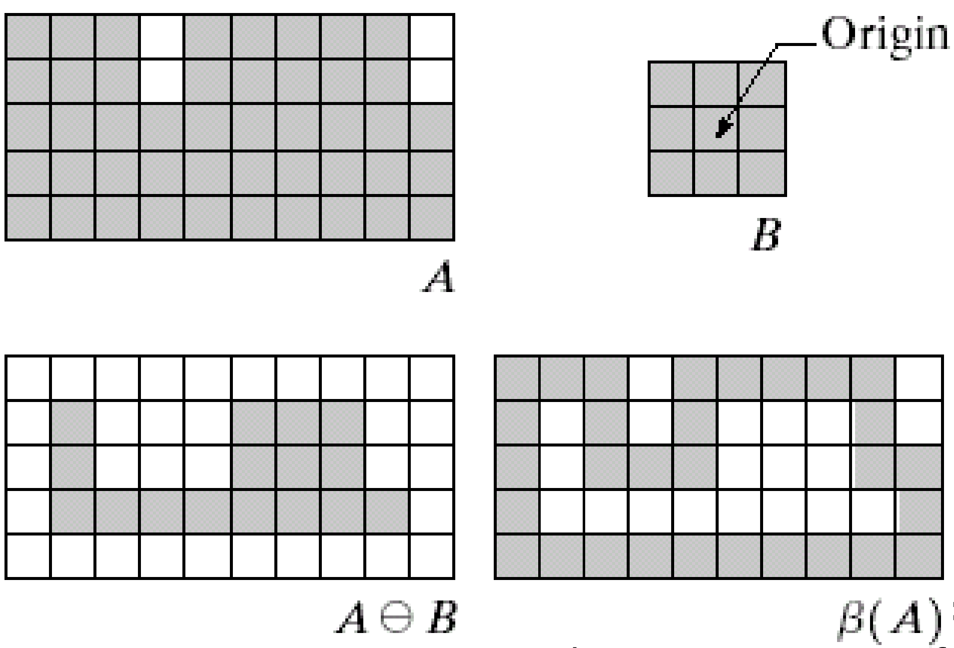
$$= \left\{ z \middle| (B)_z \cap A^c \neq \varnothing \right\}$$

$$= A^c \oplus \hat{B}$$

$$A \oplus B = \left\{ z \middle| (\hat{B})_z \cap A \neq \varnothing \right\}$$

Application: Boundary extraction

- Extract boundary of a set A:
  - First erode A (make A smaller)
  - A – erode(A)



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

Application: boundary extraction

original image

Using 5x5 structuring element

Outline

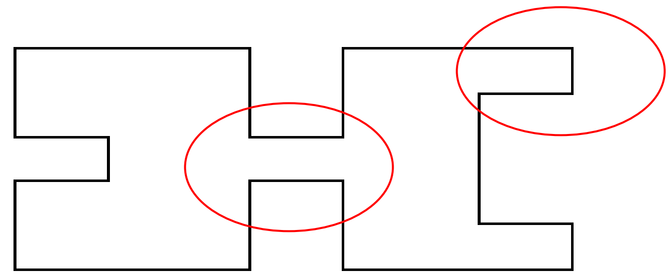
- Preliminaries
- Dilation and erosion
- Opening and closing
- Hit-or-miss transformation
- Some basic morphological algorithms
- Extensions to gray-scale images

Opening

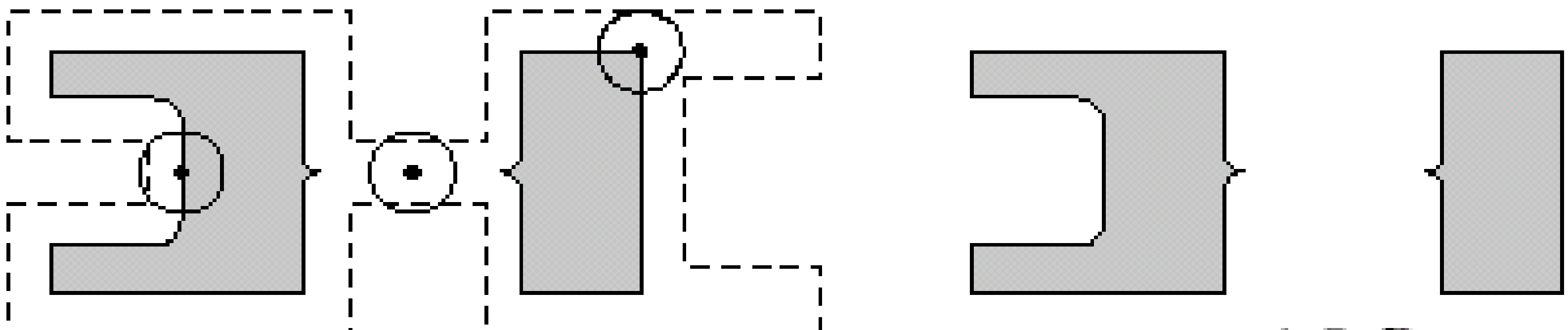
- Dilation: expands image w.r.t structuring elements
- Erosion: shrink image
- erosion+dilation = original image ?
- Opening= erosion + dilation

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

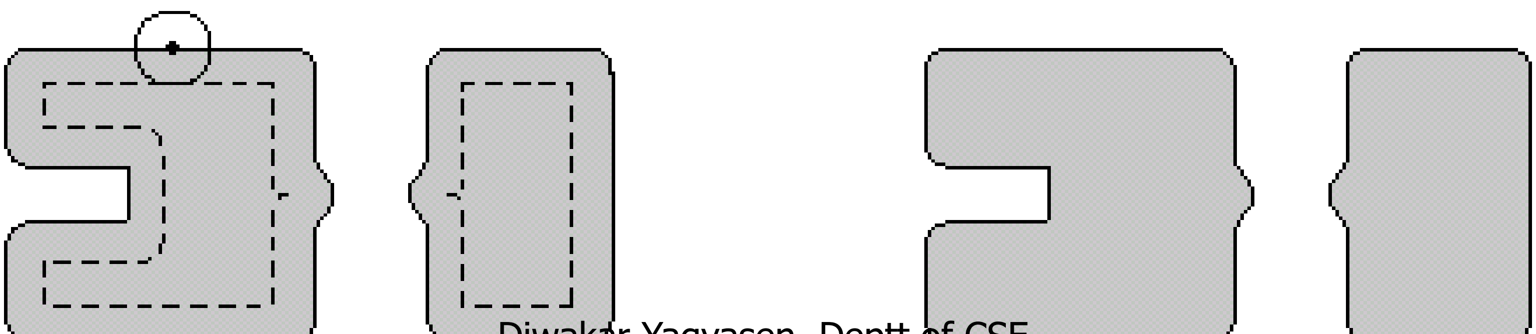
Opening (cont.)



A

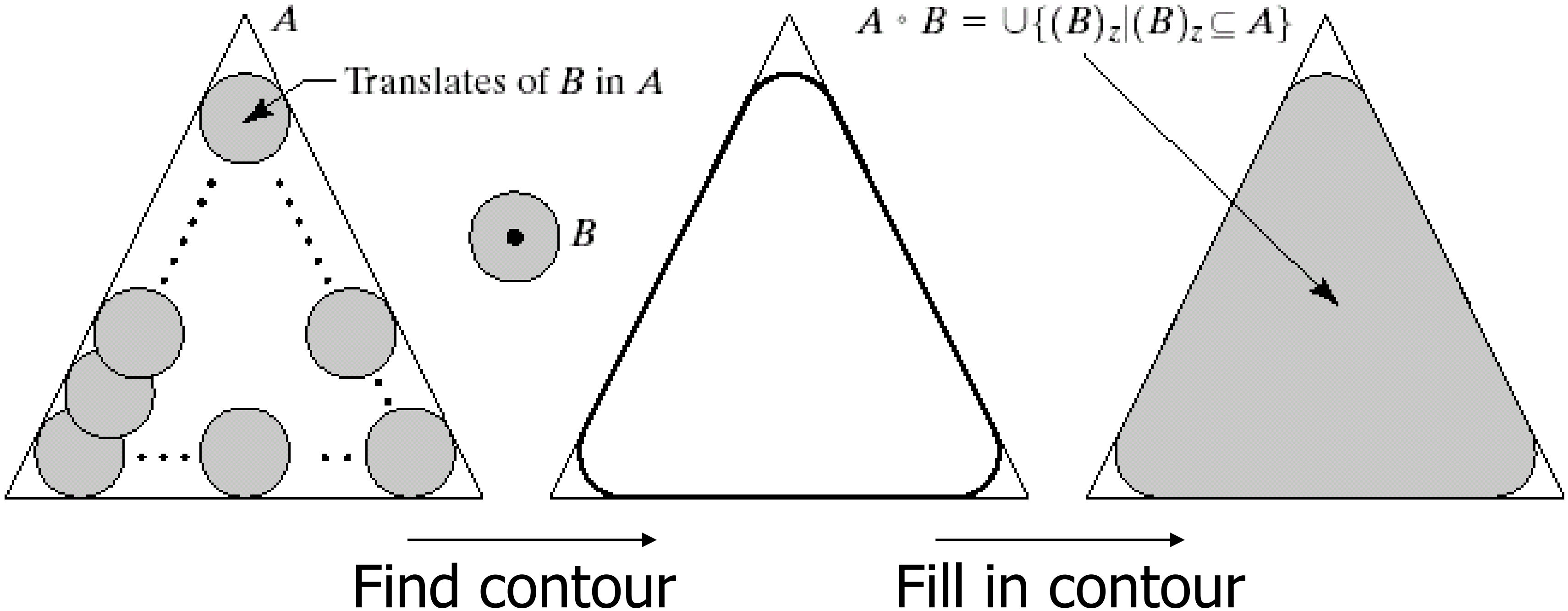


$A \ominus B$



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

Opening (cont.)

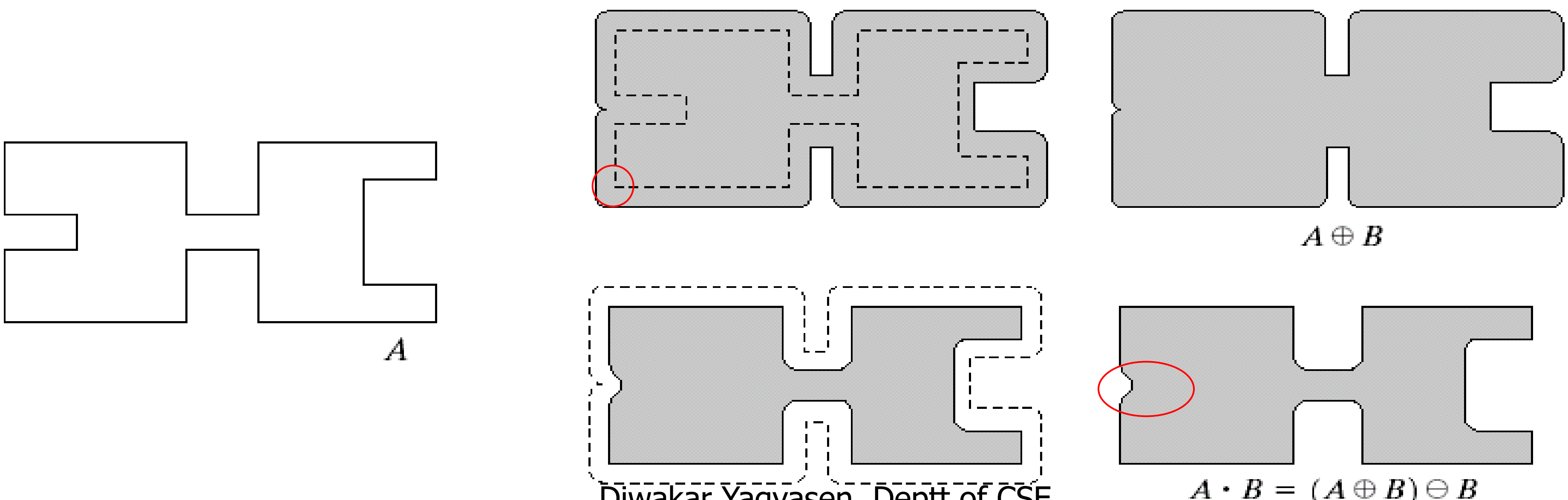


Smooth the contour of an image, breaks narrow isthmuses, eliminates thin protrusions

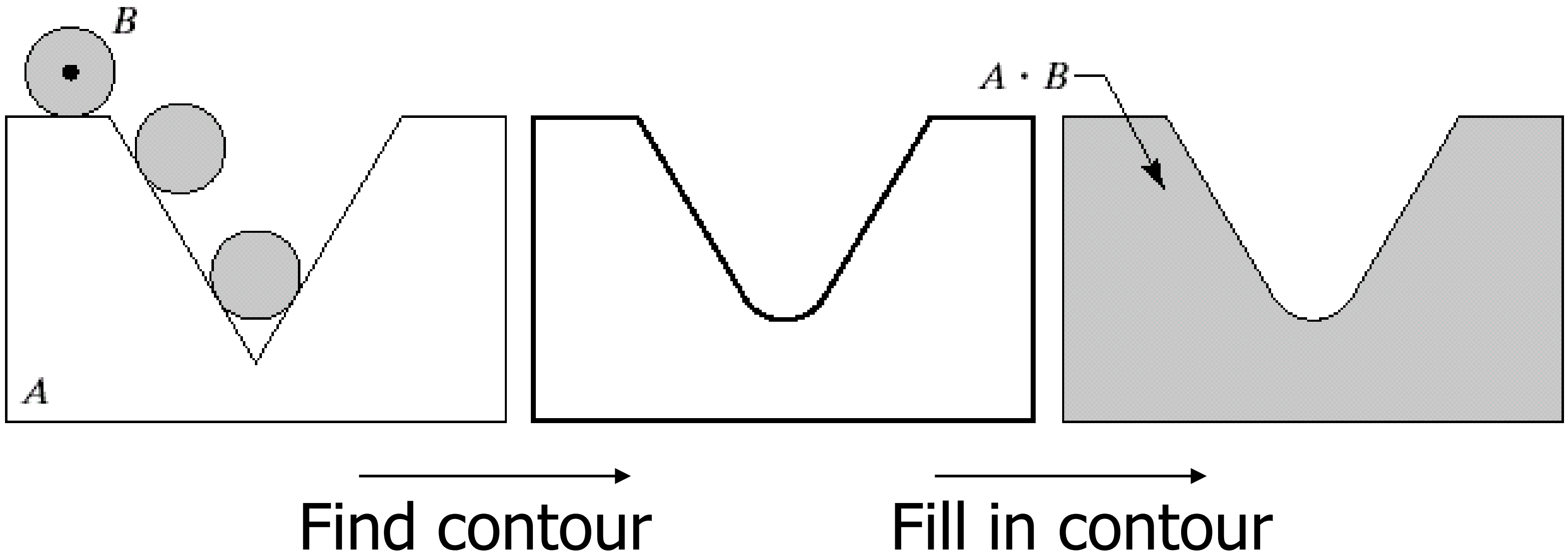
Closing

- Dilation+erosion = erosion + dilation ?
- Closing = dilation + erosion

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



Closing (cont.)



Smooth the object contour, fuse narrow breaks and long thin gulfs, eliminate small holes, and fill in gaps

Properties of opening and closing

- Opening

  (i) $A \circ B$ is a subset (subimage) of A

  (ii) If C is a subset of D, then $C \circ B$ is a subset of $D \circ B$

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

- Closing

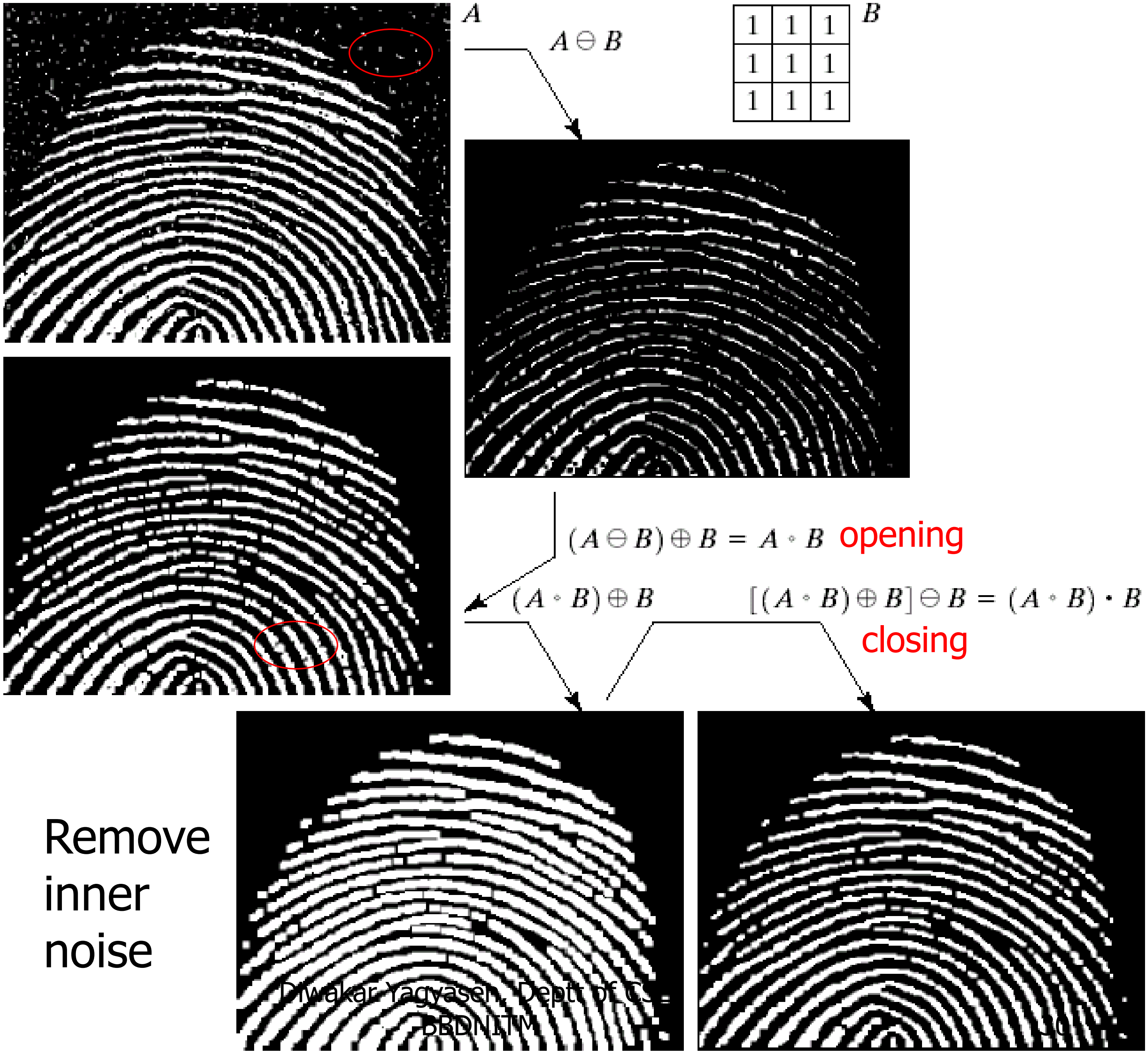
  (i) A is a subset (subimage) of $A \bullet B$

  (ii) If C is a subset of D, then $C \bullet B$ is a subset of $D \bullet B$

  (iii) $(A \bullet B) \bullet B = A \bullet B$

Noisy image

Remove outer noise

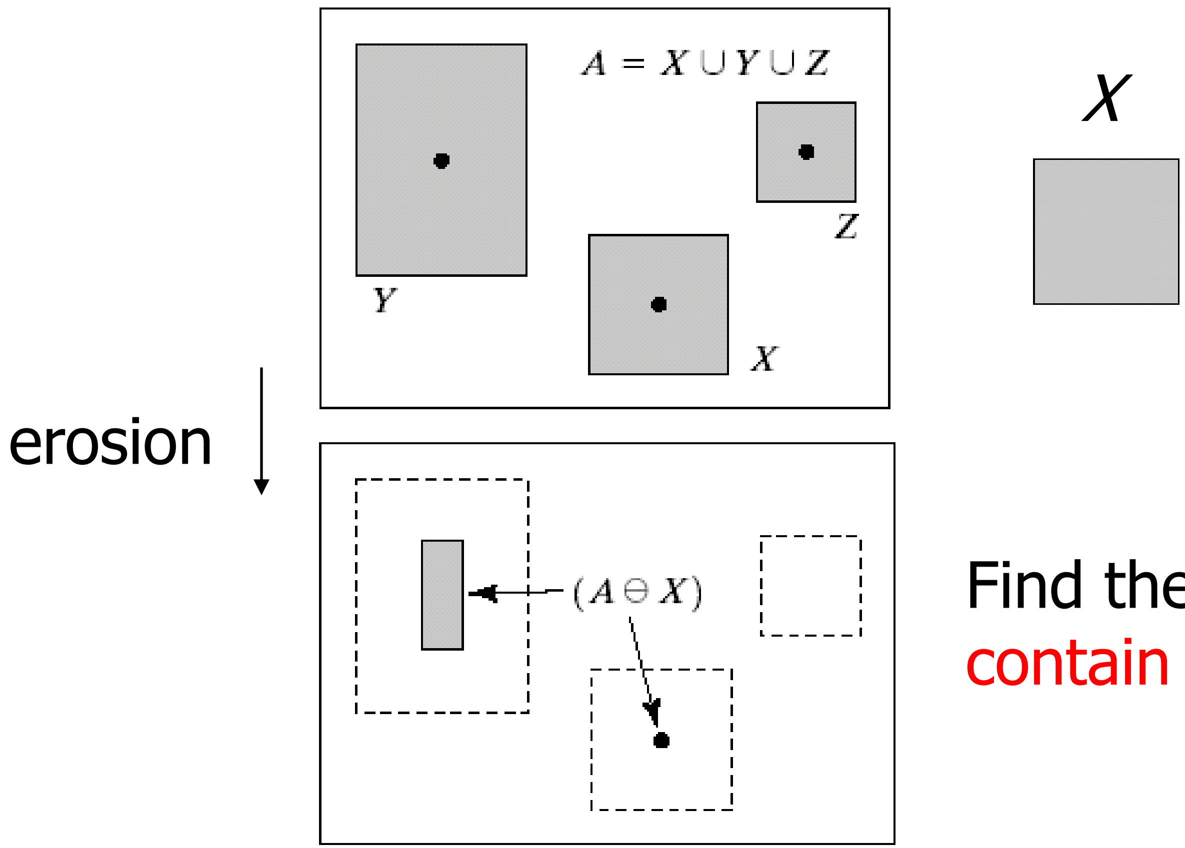


Outline

- Preliminaries
- Dilation and erosion
- Opening and closing
- Hit-or-miss transformation
- Some basic morphological algorithms
- Extensions to gray-scale images

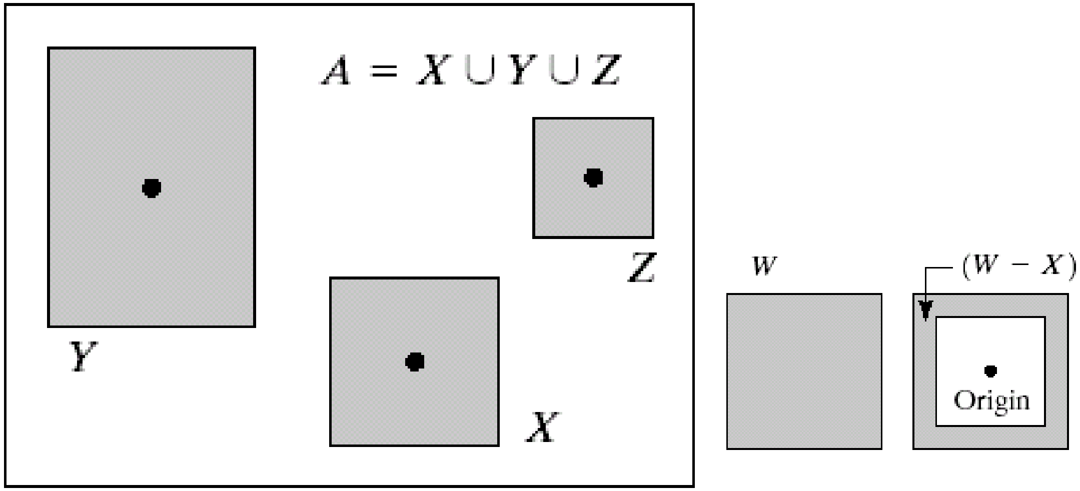
Hit-or-miss transformation

- Find the location of certain shape

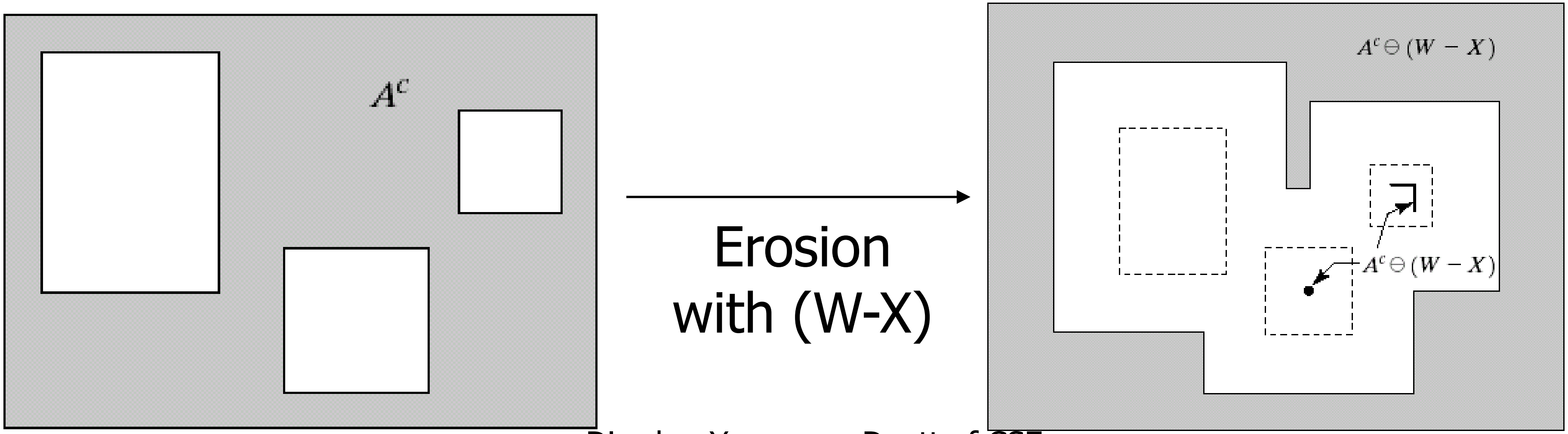


Find the set of pixels that contain shape *X*

Hit-or-miss transformation

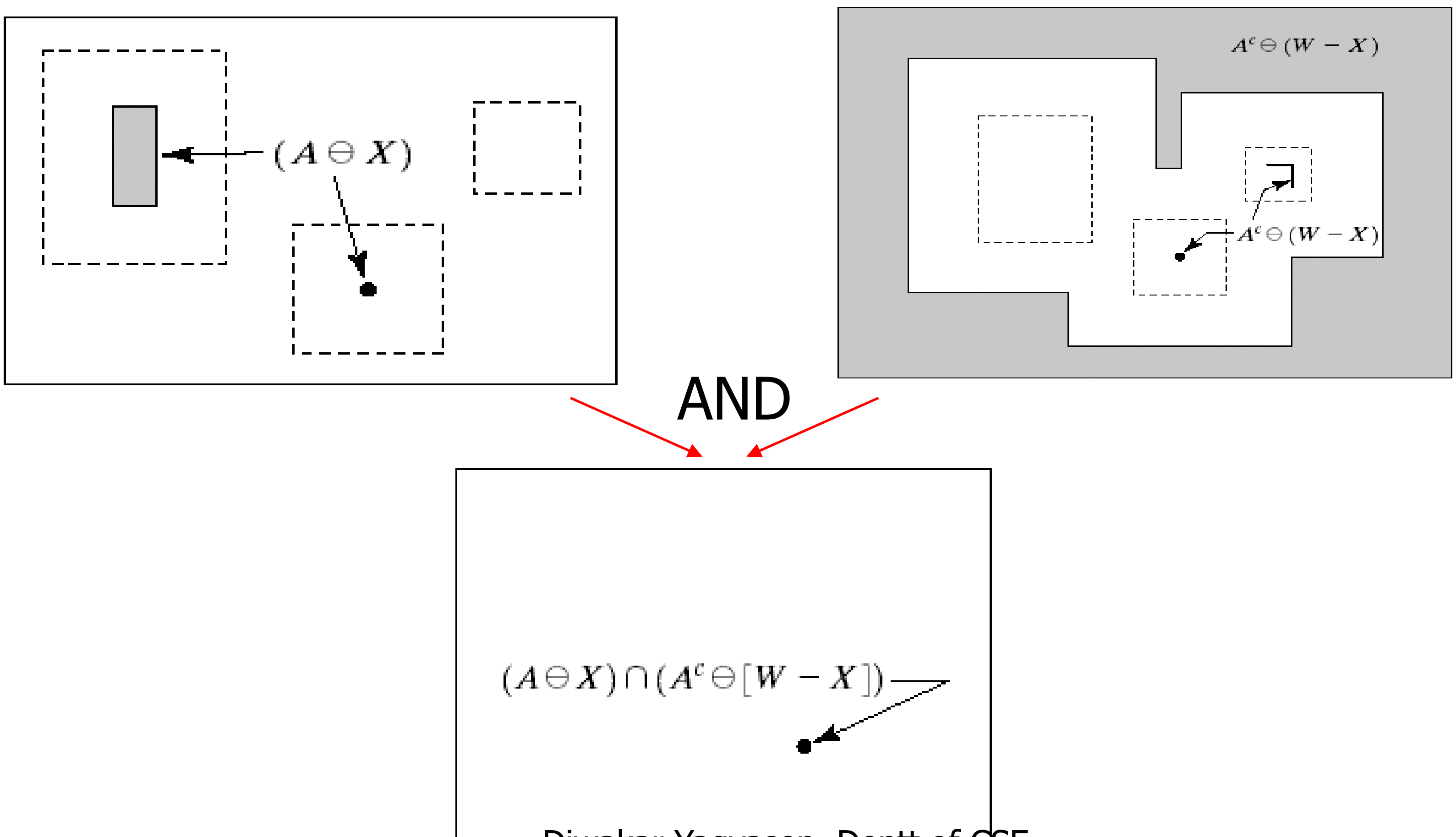


Detect object via background



Hit-or-miss transformation

- Eliminate un-necessary parts



Outline

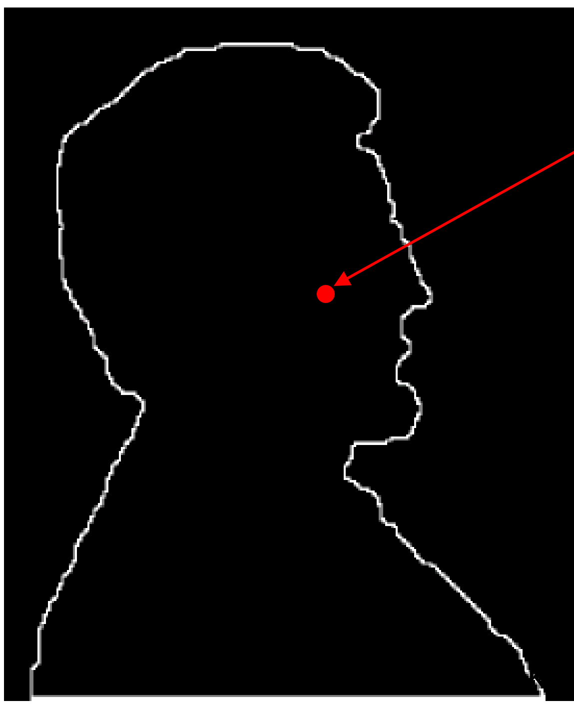
- Preliminaries
- Dilation and erosion
- Opening and closing
- Hit-or-miss transformation
- Some basic morphological algorithms
- Extensions to gray-scale images

Basic morphological algorithms

- Extract image components that are useful in the representation and description of shape
- Boundary extraction
- Region filling
- Extract of connected components
- Convex hull
- Thinning
- Thickening
- Skeleton
- Pruning

Region filling

- How?
- Idea: place a point inside the region, then dilate that point iteratively



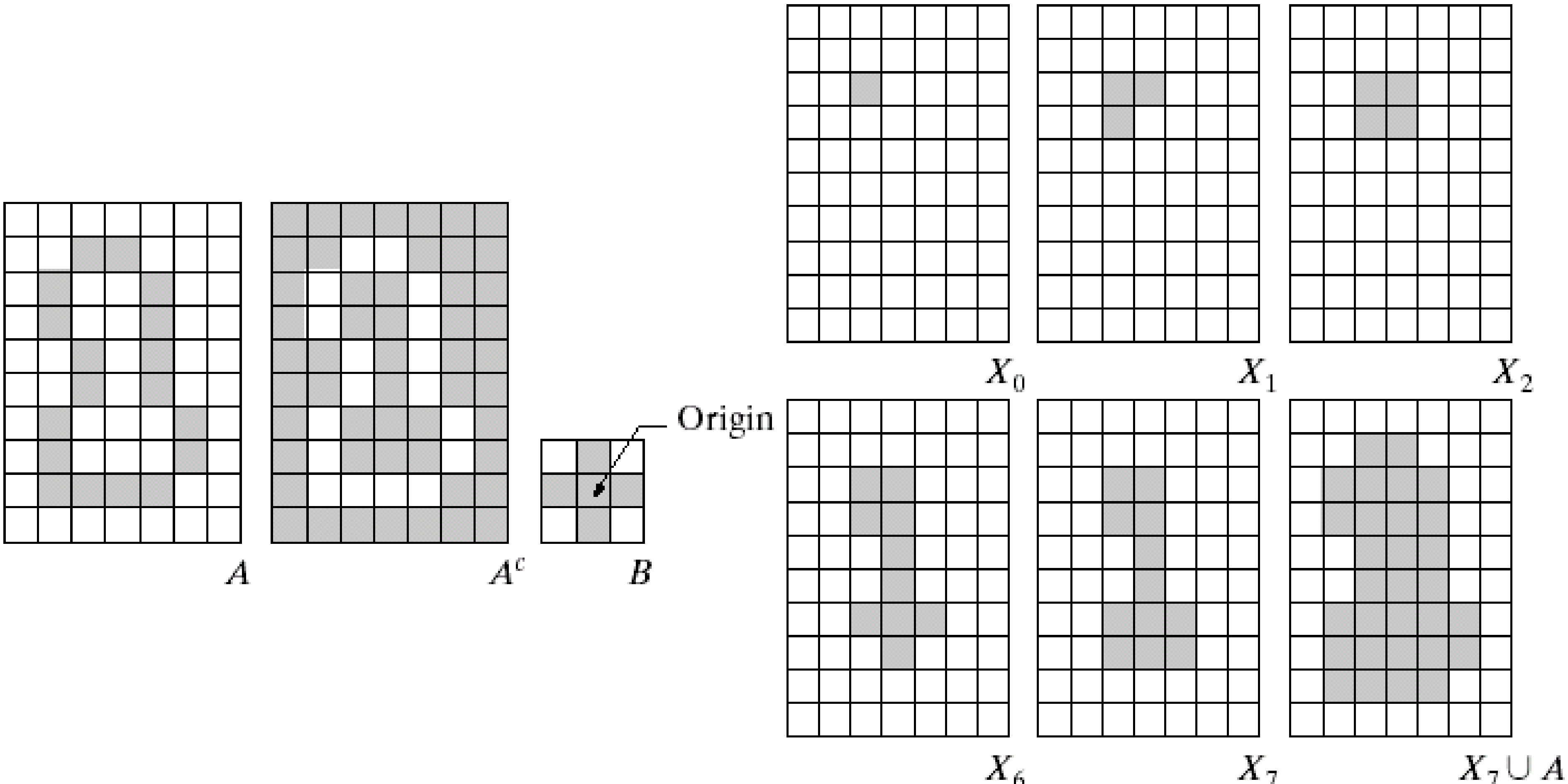
$$X_0 = p$$

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

Until $X_k = X_{k-1}$

Bound the growth

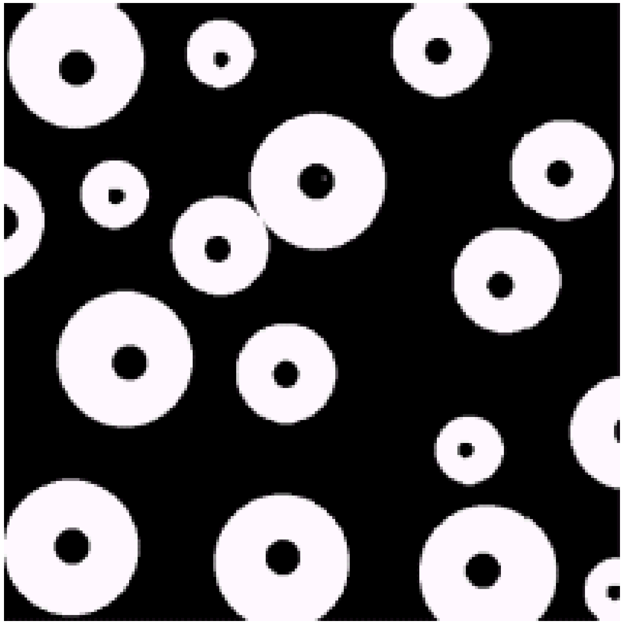
Region filling (cont.)



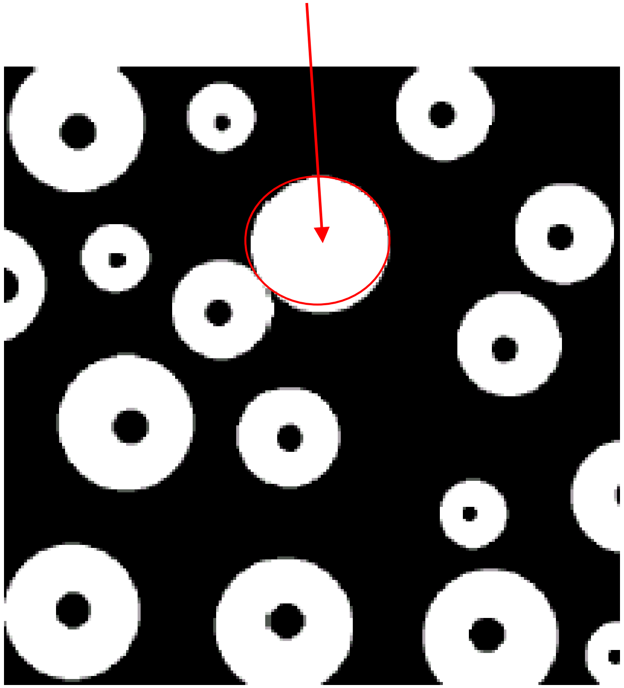
stop

Application: region filling

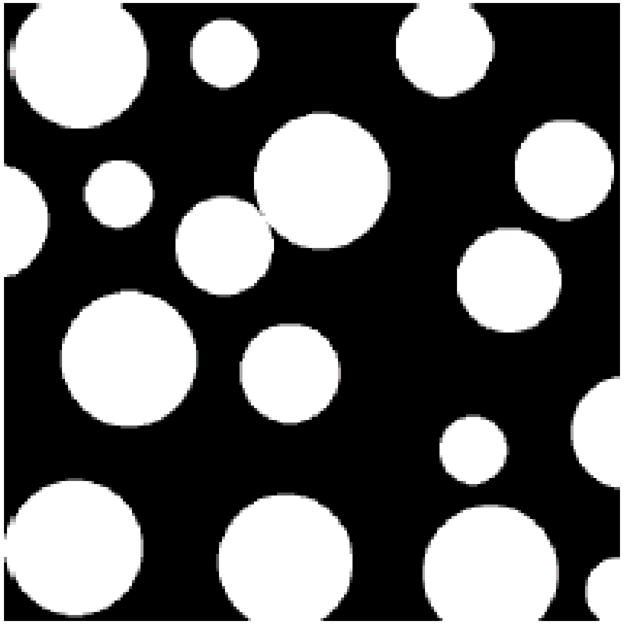
Original image



The first filled
region



Fill all regions



Extraction of connected components

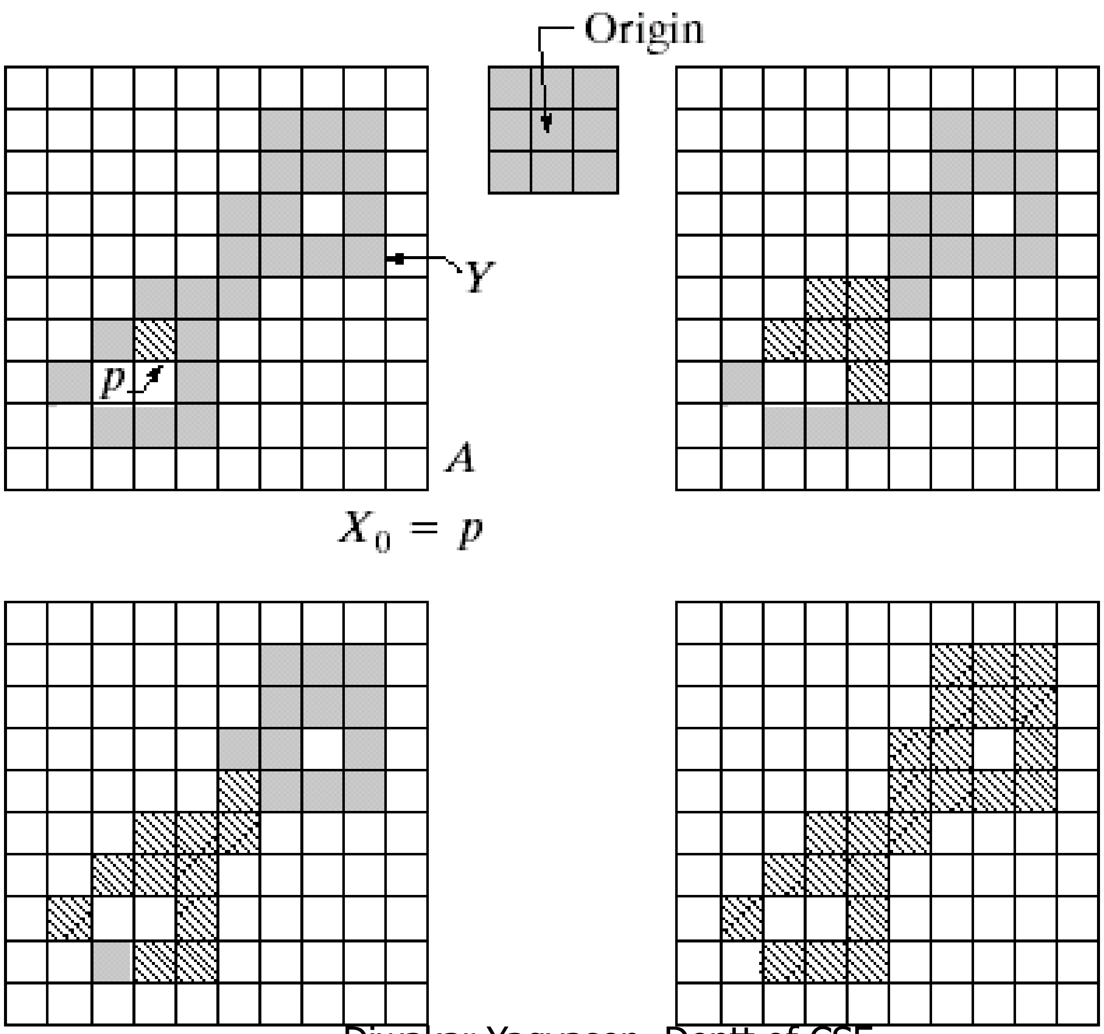
- Idea: start from a point in the connected component, and dilate it iteratively

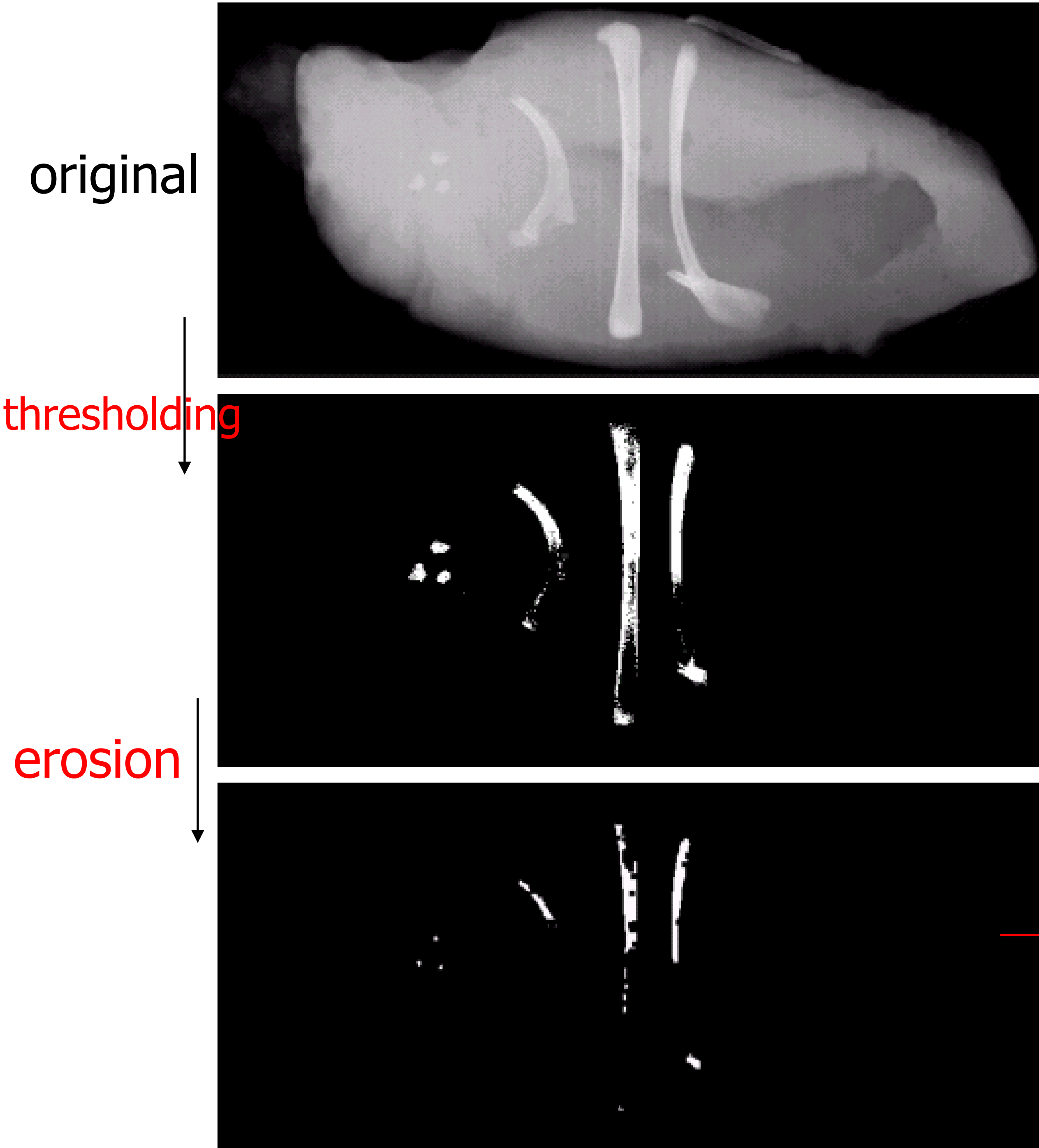
$$X_0 = p$$

$$X_k = (X_{k-1} \oplus B) \cap A, \; k = 1,2,3,\ldots$$

Until $X_k = X_{k-1}$

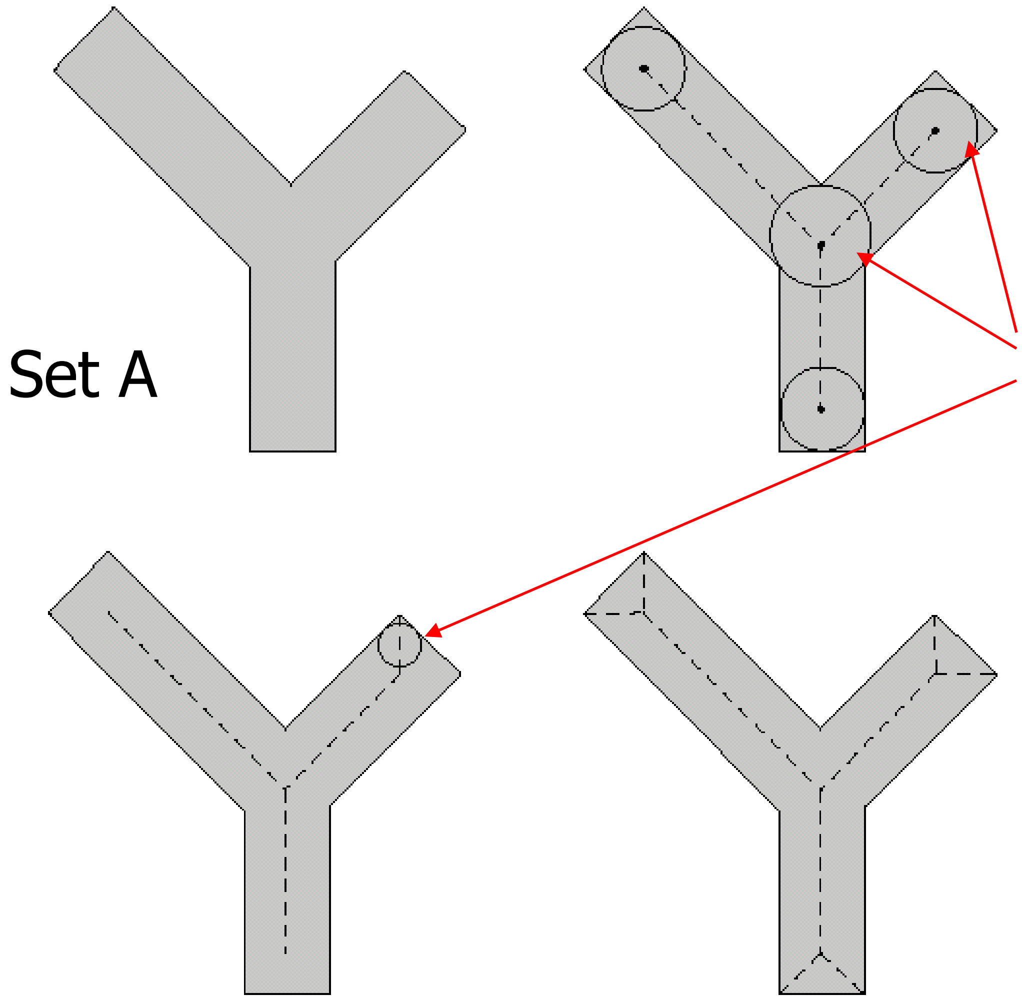
Extraction of connected components (cont.)





Connected component	No. of pixels in connected comp
01	11
02	9
03	9
04	39
05	133
06	1
07	1
08	743
09	7
10	11
11	11
12	9
13	9
14	674
15	85

Skeletons



How to define a Skeletons?

Maximum disk

1. The largest disk Centered at a pixel
2. Touch the boundary of A at two or more places

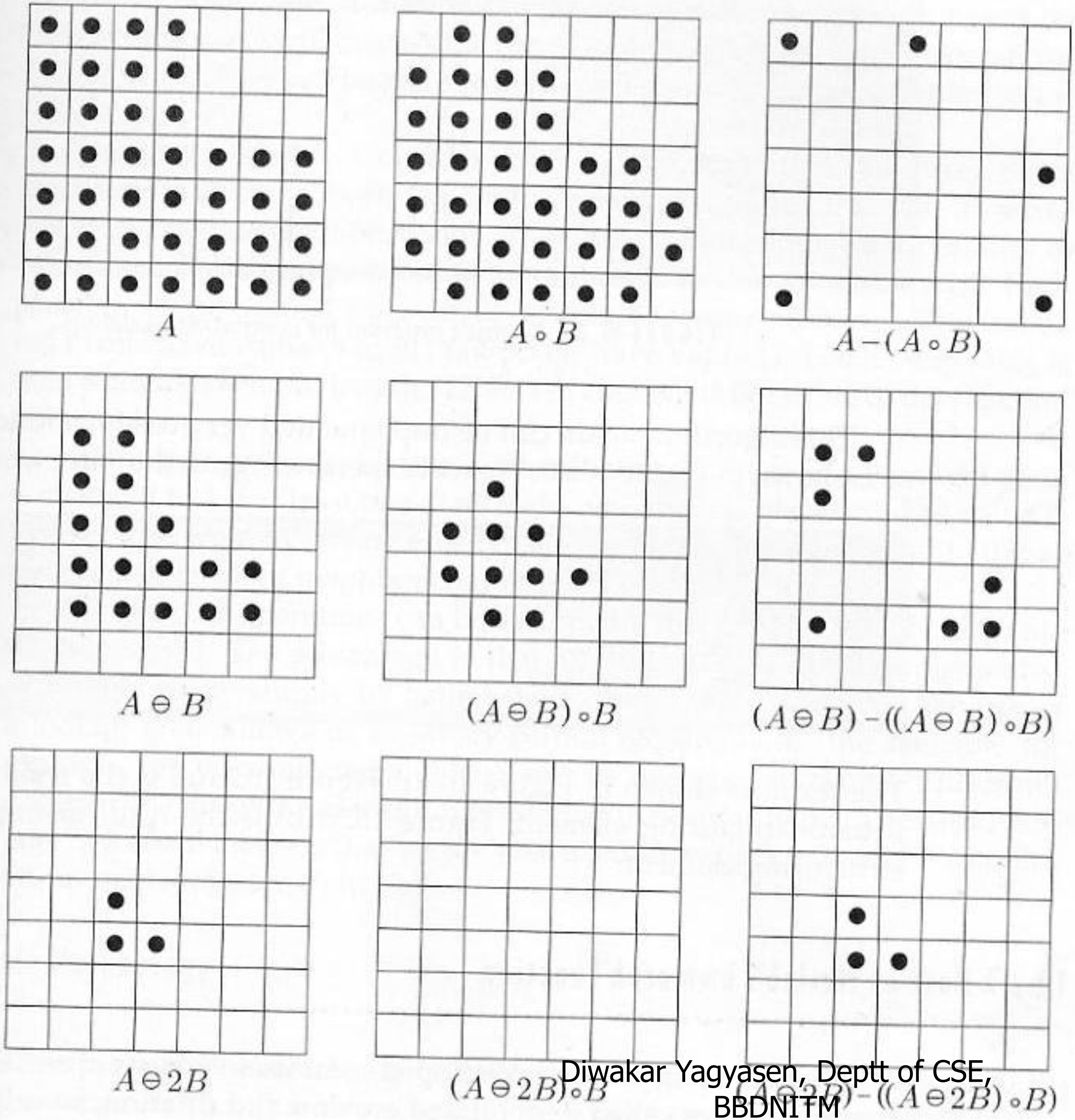
Recall: Balls of erosion!

Skeleton

- Idea: erosion

Erosions	Openings	Set differences
A	$A \circ B$	$A - (A \circ B)$
$A \ominus B$	$(A \ominus B) \circ B$	$(A \ominus B) - ((A \ominus B) \circ B)$
$A \ominus 2B$	$(A \ominus 2B) \circ B$	$(A \ominus 2B) - ((A \ominus 2B) \circ B)$
$A \ominus 3B$	$(A \ominus 3B) \circ B$	$(A \ominus 3B) - ((A \ominus 3B) \circ B)$
$\vdots$	$\vdots$	$\vdots$
$A \ominus kB$	$(A \ominus kB) \circ B$	$(A \ominus kB) - ((A \ominus kB) \circ B)$

Erosion k 次
直到空集合



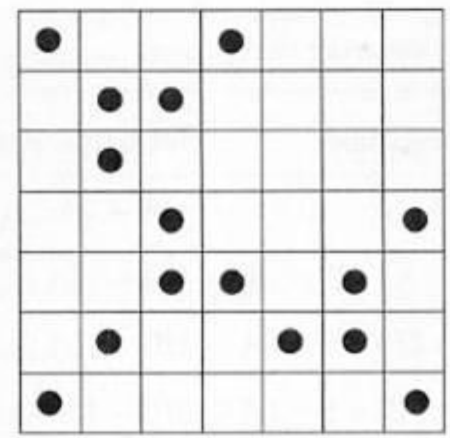
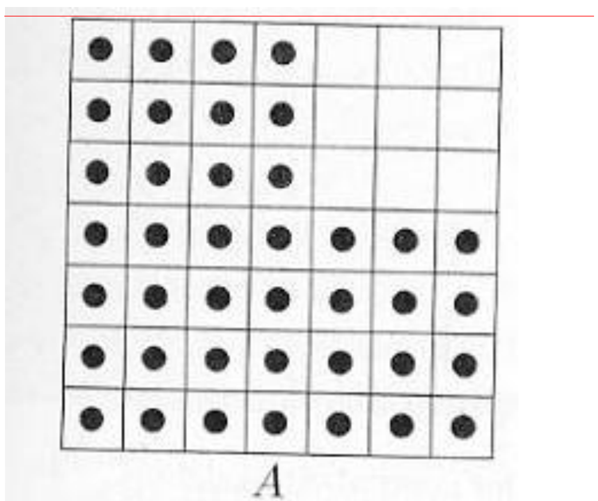


FIGURE 10.29 *The final skeleton.*

Problem

- The scanned image is not adjusted well



- How to detection the direction of lines?
- How to rotate?