Imperial College London

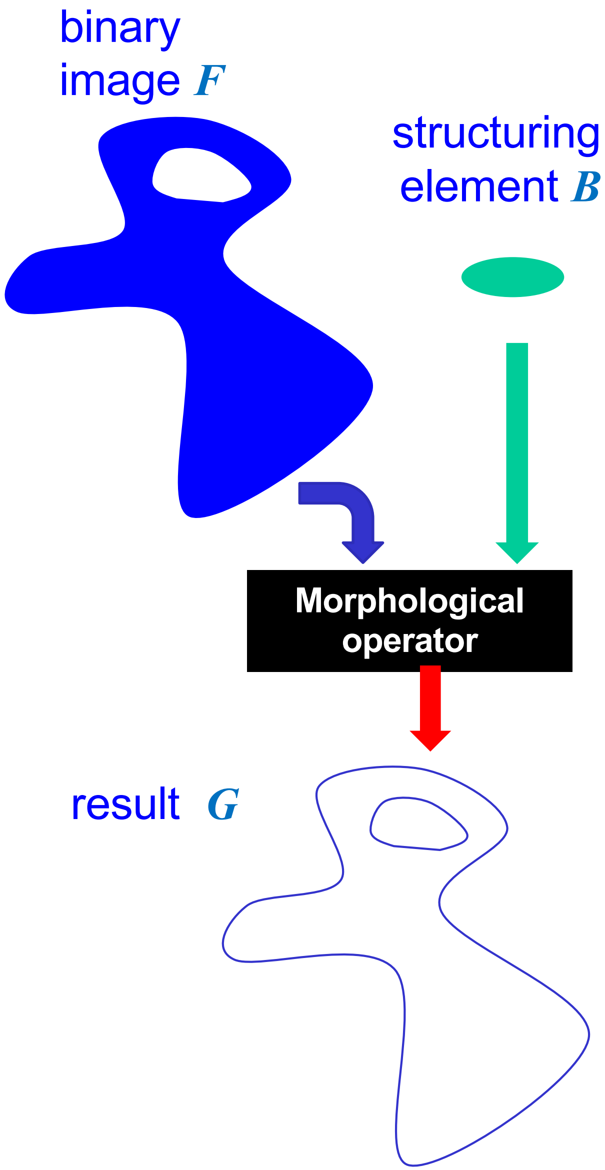
6 – Morphological Operations Part 1

Prof Peter YK Cheung

Dyson School of Design Engineering

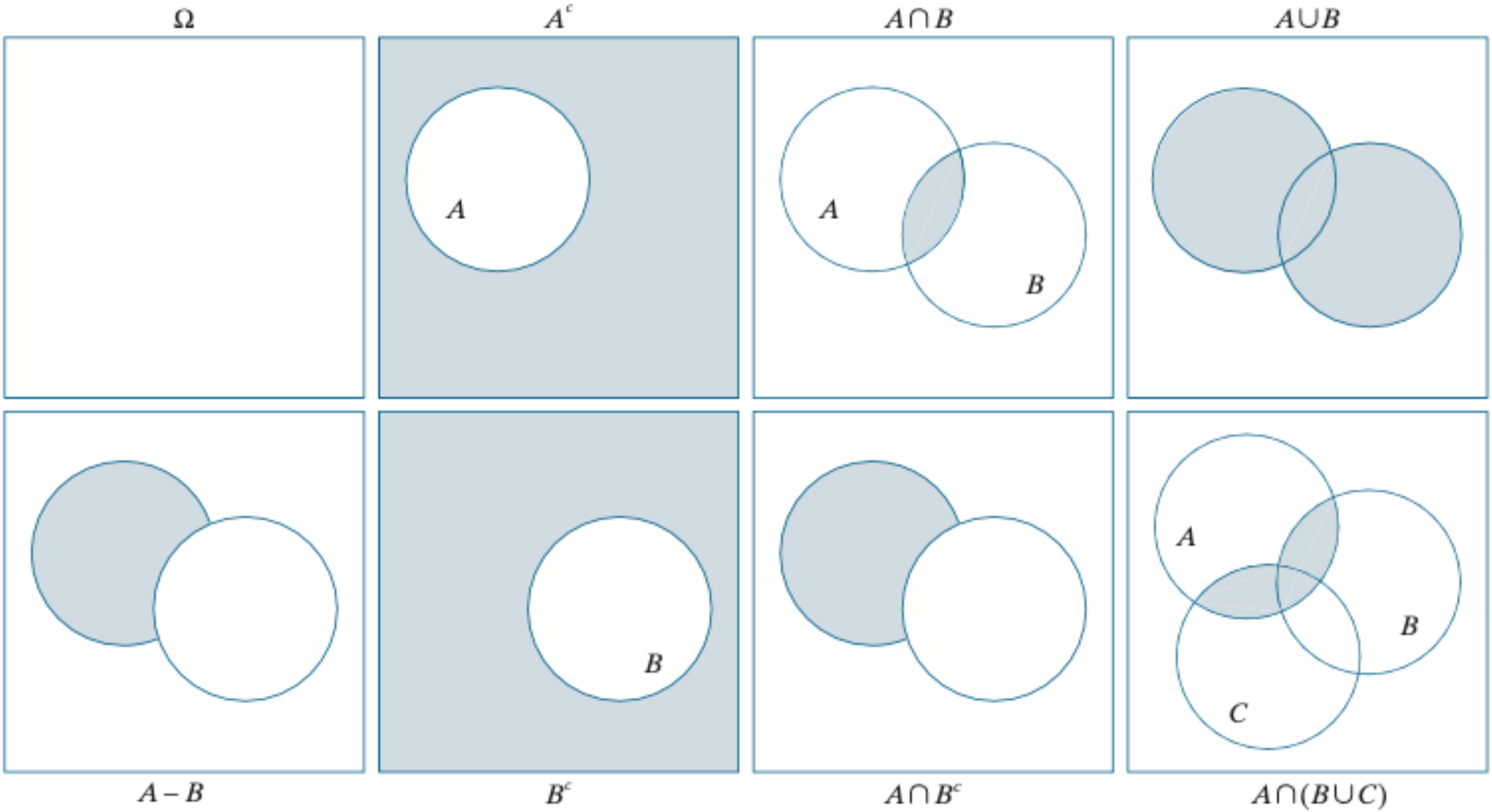
URL: www.ee.ic.ac.uk/pcheung/teaching/DE4_DVS/
E-mail: p.cheung@imperial.ac.uk

Morphological Operation



- Take binary image ***F***
- Take structuring element ***B*** (similar to filter kernel)
- Perform **morphological operation** (similar to, but different from, convolution in spatial filter)
- Produce a new image ***G***
- Useful for:
  - Image **pre-processing** (noise filtering, shape simplification)
  - **Enhancing object structure** (skeletonizing, thinning, thickening, convex hull,)
  - **Segmenting objects** from the background
  - **Quantitative description** of objects (area, perimeter, projections,)

Logic Operators



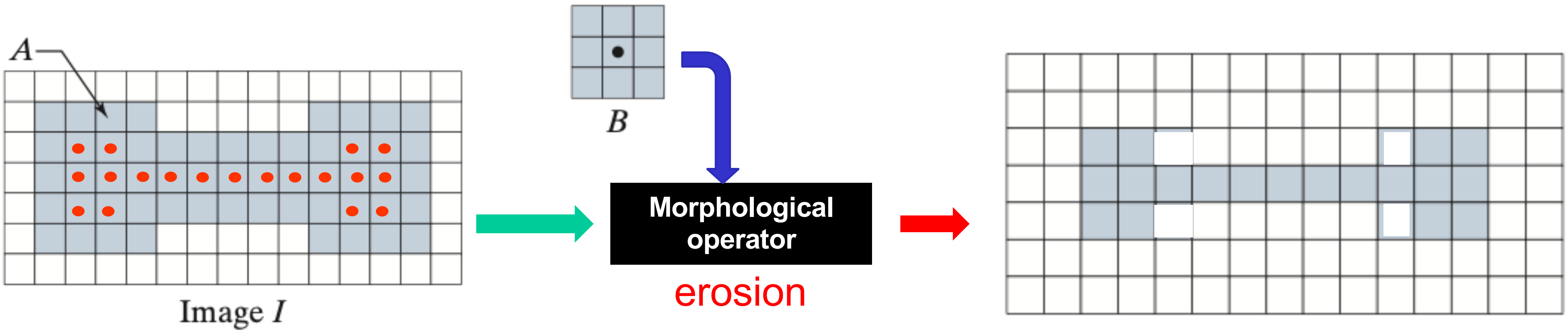
Notations for Set and Logical Operations

- If *a* is an *element* of set *A*, then $a \in A$
- If *a* is NOT an *element* of set *A*, then $a \notin A$
- A set is denoted by the contents of two braces: { .}.
  - For example: $$C = \left\{ c \mid c = -d,\ d \in D \right\}$$

  means that *C* is the set of elements, *c*, such that *c* is formed by multiplying each of the elements of set *D* by −1.
- If every element of a set A is also an element of a set B, then A is said to be a **subset** of B, denoted as: $A \subseteq B$
- **Union**: $C = A \cup B$ **Intersection**: $D = A \cap B$ **disjoint** $A \cap B = \varnothing$
- Complement: $$A^c = \left\{ w \mid w \notin A \right\}$$
- Difference of two sets A and B: $A - B = \left\{ w \mid w \in A, w \notin B \right\} = A \cap B^c$

Example of a Morphological Operation

- Binary image, I, consisting of an object (set) A shown shaded.
- 3 × 3 **structuring element** (SE) whose elements are all 1's (foreground pixels). The background pixels are (0's).
- Morphological operations:
  1. Form an image containing only the object A as image I
  2. Move B over image I, pixel-by-pixel
  3. At each pixel, if B is **completely contained** in A, mark the location of the origin of B as a **foreground** pixel (i.e. 1) in the new image, else leave as **background**

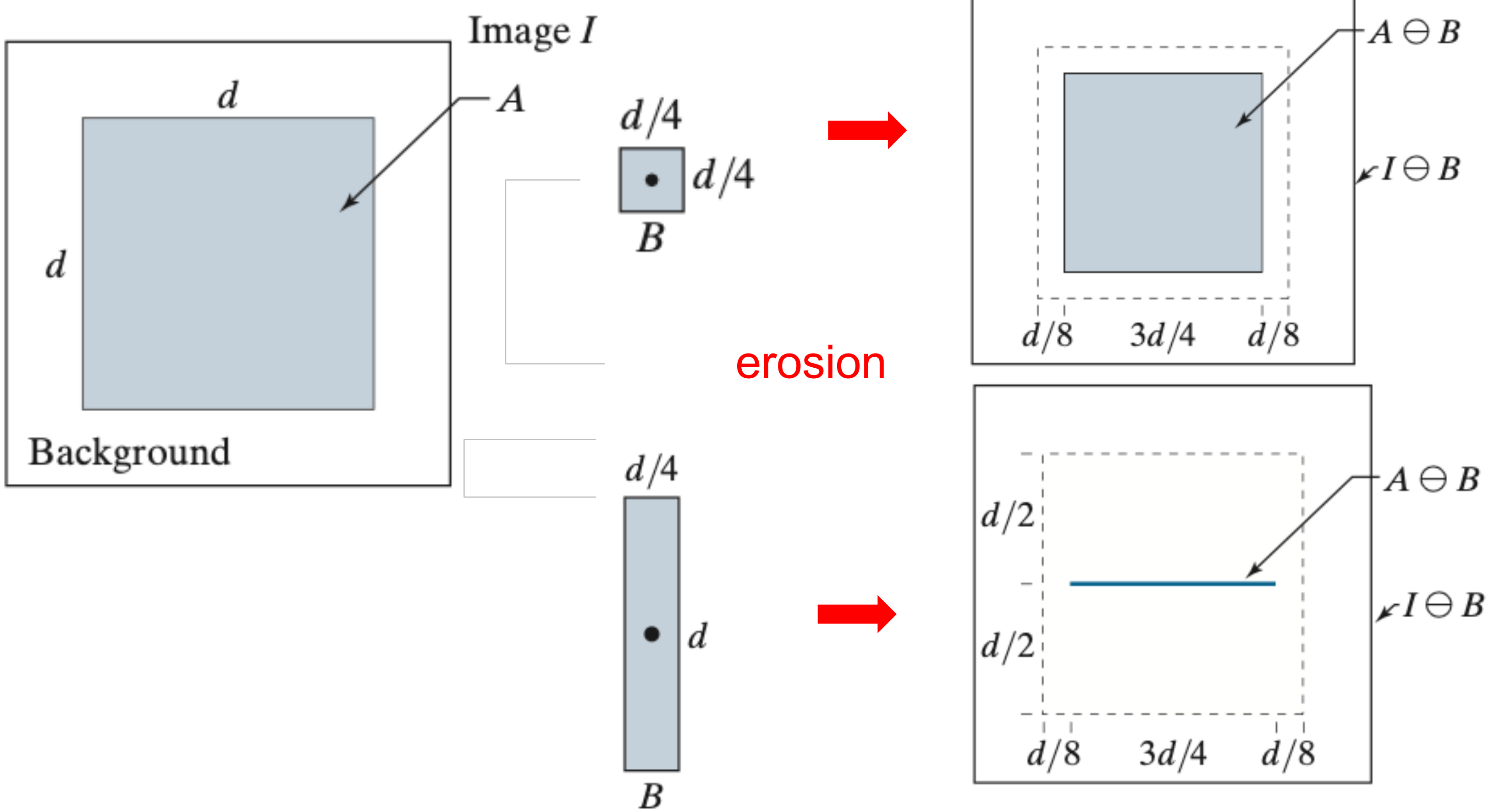


Erosion Operator $\ominus$

- Formal definition of **erosion**: $A \ominus B = \{z | (B)_z \subseteq A\}$

  such that — contained in

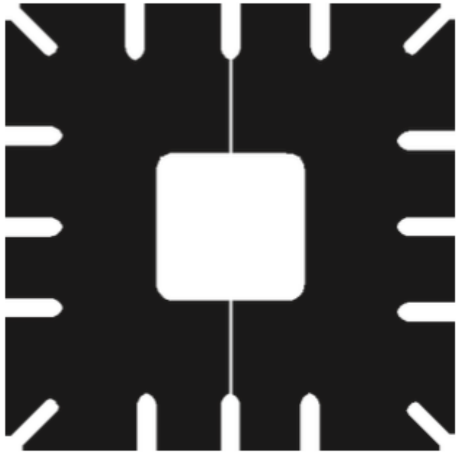
- In plain words, out pixel is '1' if **ALL** the pixels in image under B are '1'.



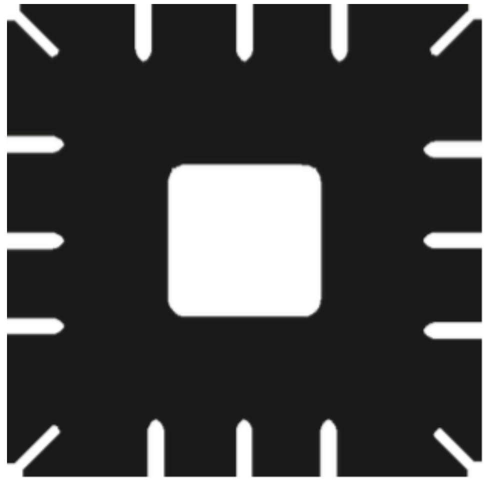
Application of Erosion Operator ⊖

486 × 486 binary image
of a wire-bond mask
(foreground is white)

B = 11x11 ones



B = 15x15 ones

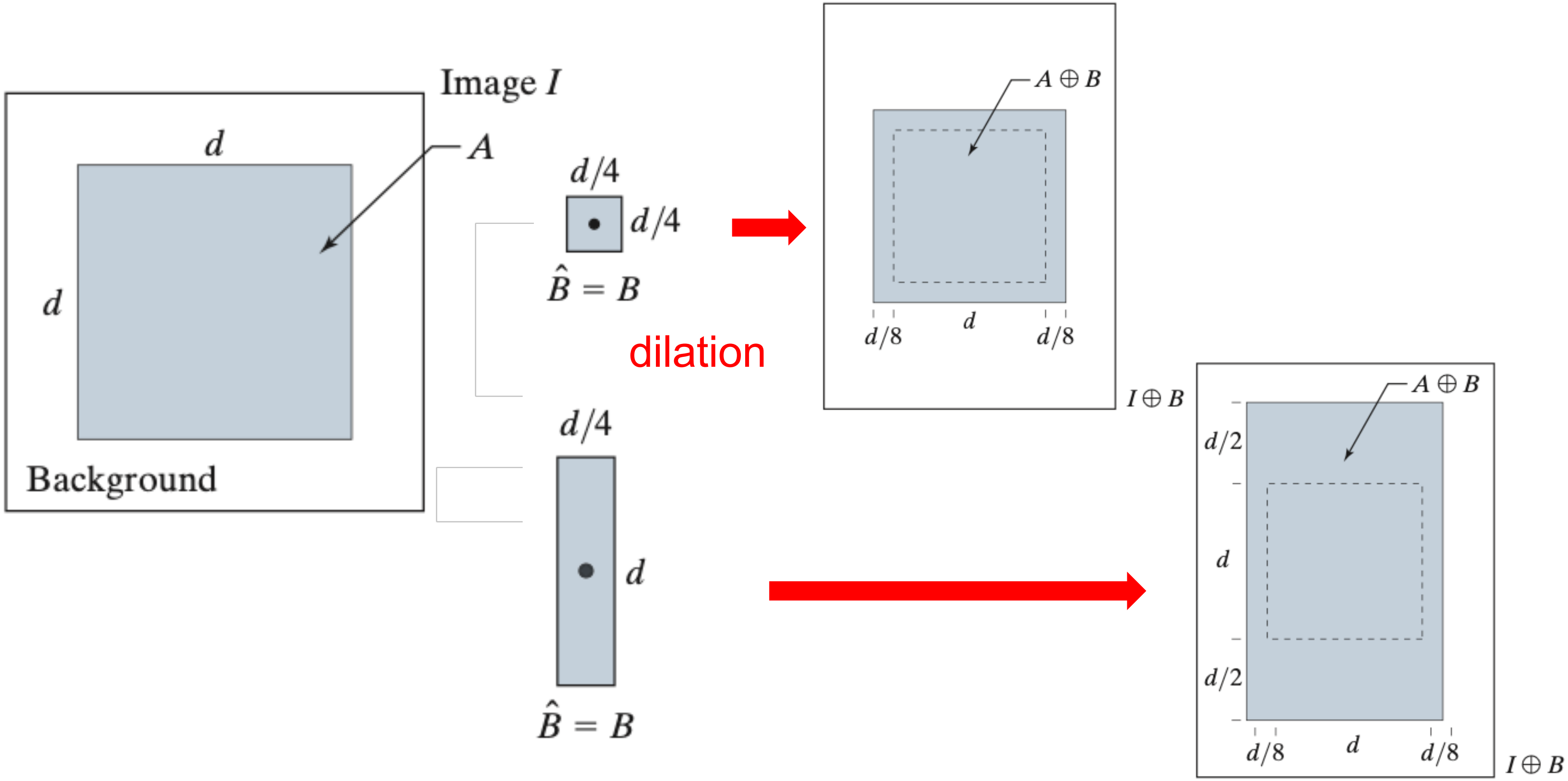


B = 45x45 ones

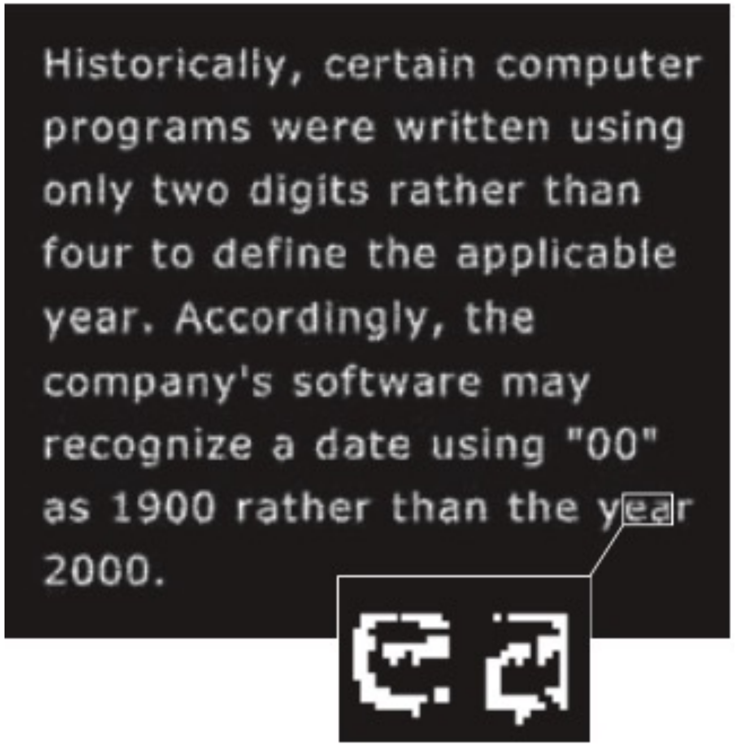
Dilation Operator $\oplus$

- Formal definition of **dilation** is: $A \oplus B = \left\{ z \mid \left(\hat{B}\right)_z \cap A \neq \emptyset \right\}$

- In plain words, pixel is '1' if **ANY** of the pixels in image under B is '1'.



Application of Dilation Operator



$$\oplus \quad \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad =$$

$$\hat{B} = B$$

Duality of Erosion and Dilation

- Erosion of A by B is the complement of the dilation of A^c by $\hat{B}$

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

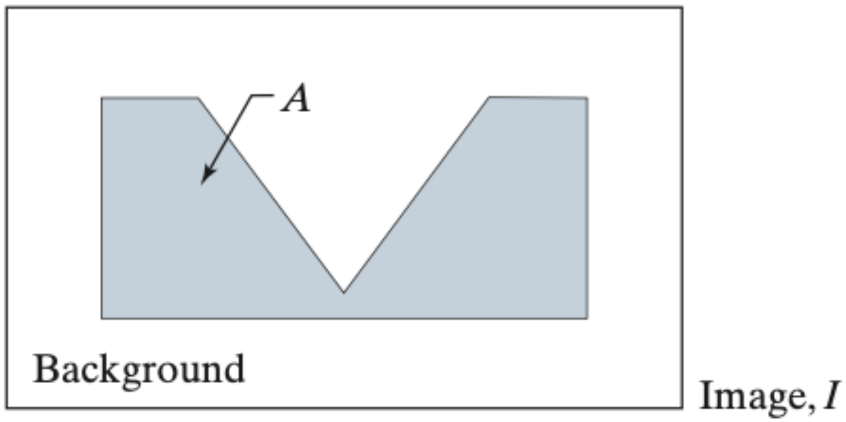
- If B is symmetrical with respect to its origin, then $\hat{B} = B$. Then erosion of A is the same as dilating its background and complementing the result.

- Dilation of A by B is the complement of the erosion of A^c by $\hat{B}$

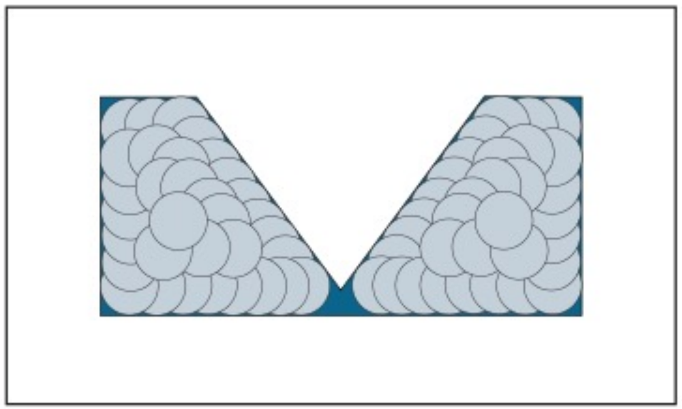
$$(A \oplus B)^c = A^c \ominus \hat{B}$$

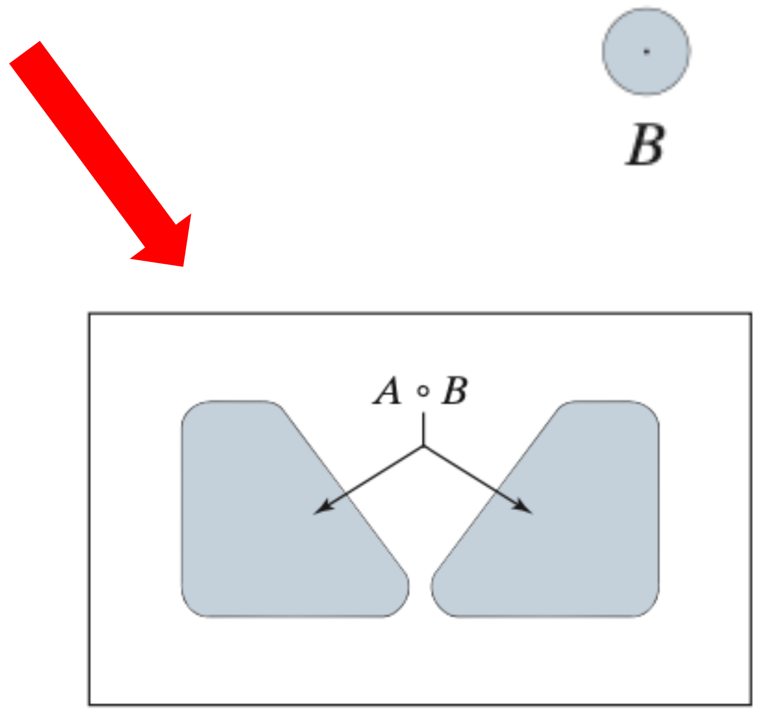
Opening Operator - ∘

- Combine erosion and dilation operators results in other operators.
- **Opening operator**: smoothes contour, breaks narrow passages, and eliminates thin protrusions.
- Opening A by B is **erosion** of A by B, **followed by dilation** of the result by B:

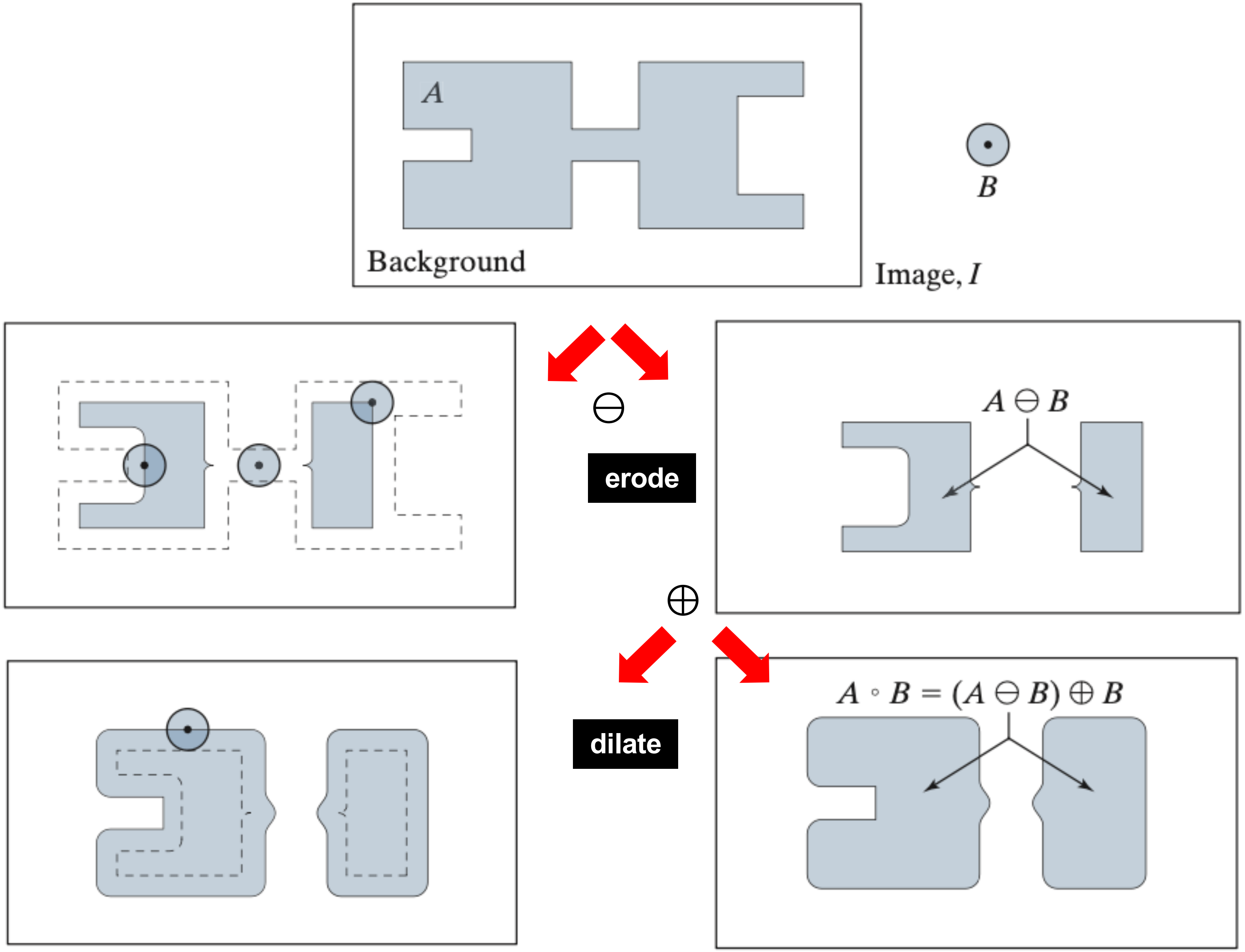


$$A \circ B = (A \ominus B) \oplus B$$



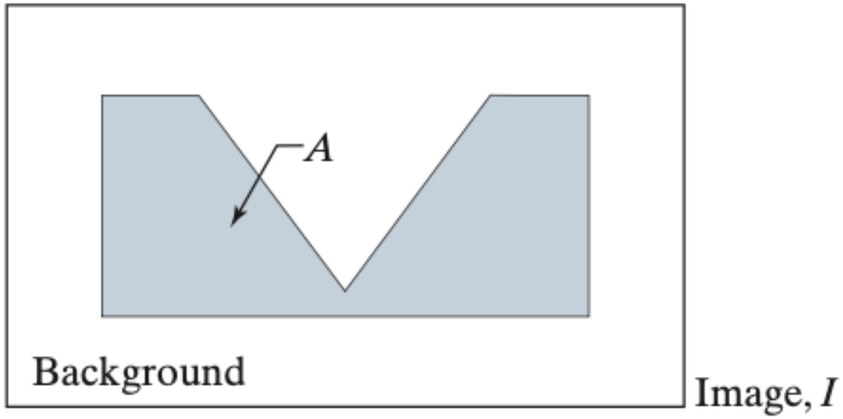


Opening in Action

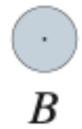


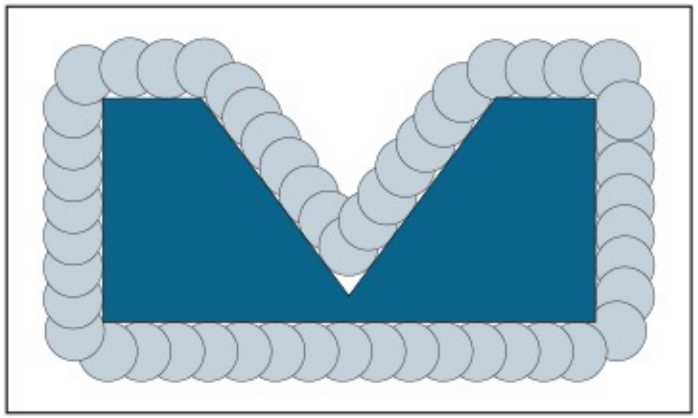
Closing Operator - •

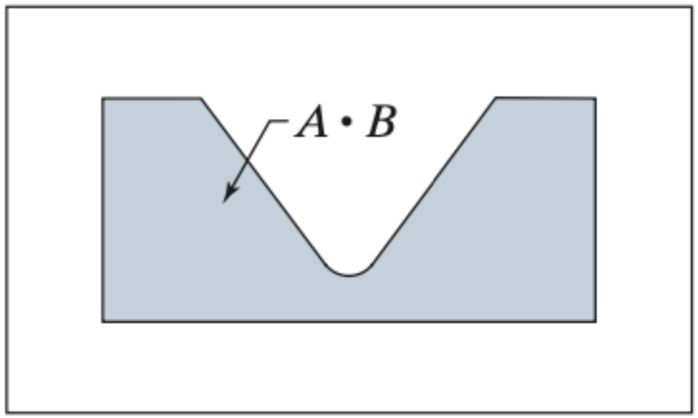
- **Closing operator**: smooth sections of contours, fuses narrow breaks and long thin gulfs, eliminates small holes, and fills gaps in the contour.
- Closing of A by B is dilation of A by B, followed by erosion of result by B
- Closing of set A by structuring element B, denoted as $A \cdot B$, is defined as:



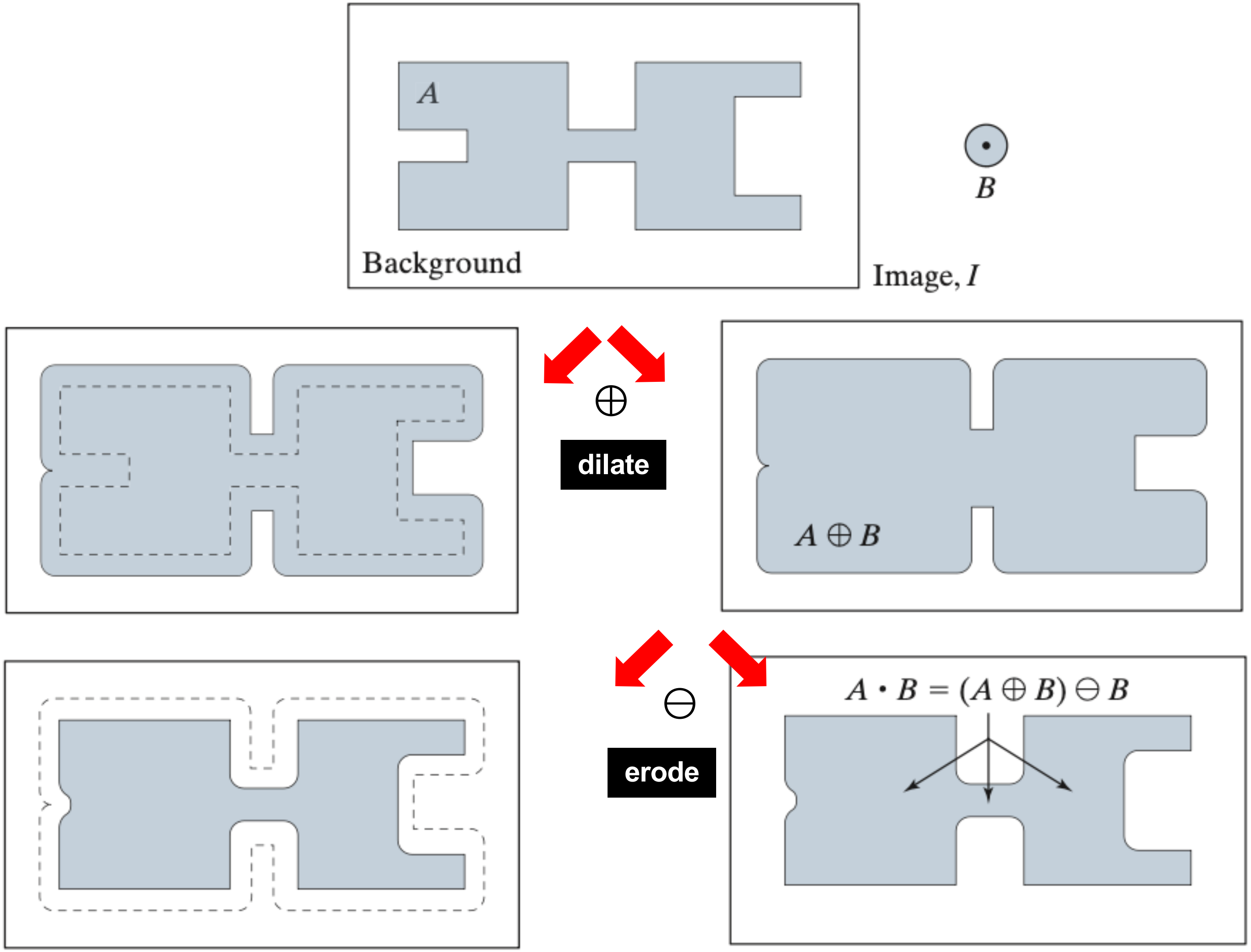
$$A \bullet B = (A \oplus B) \ominus B$$



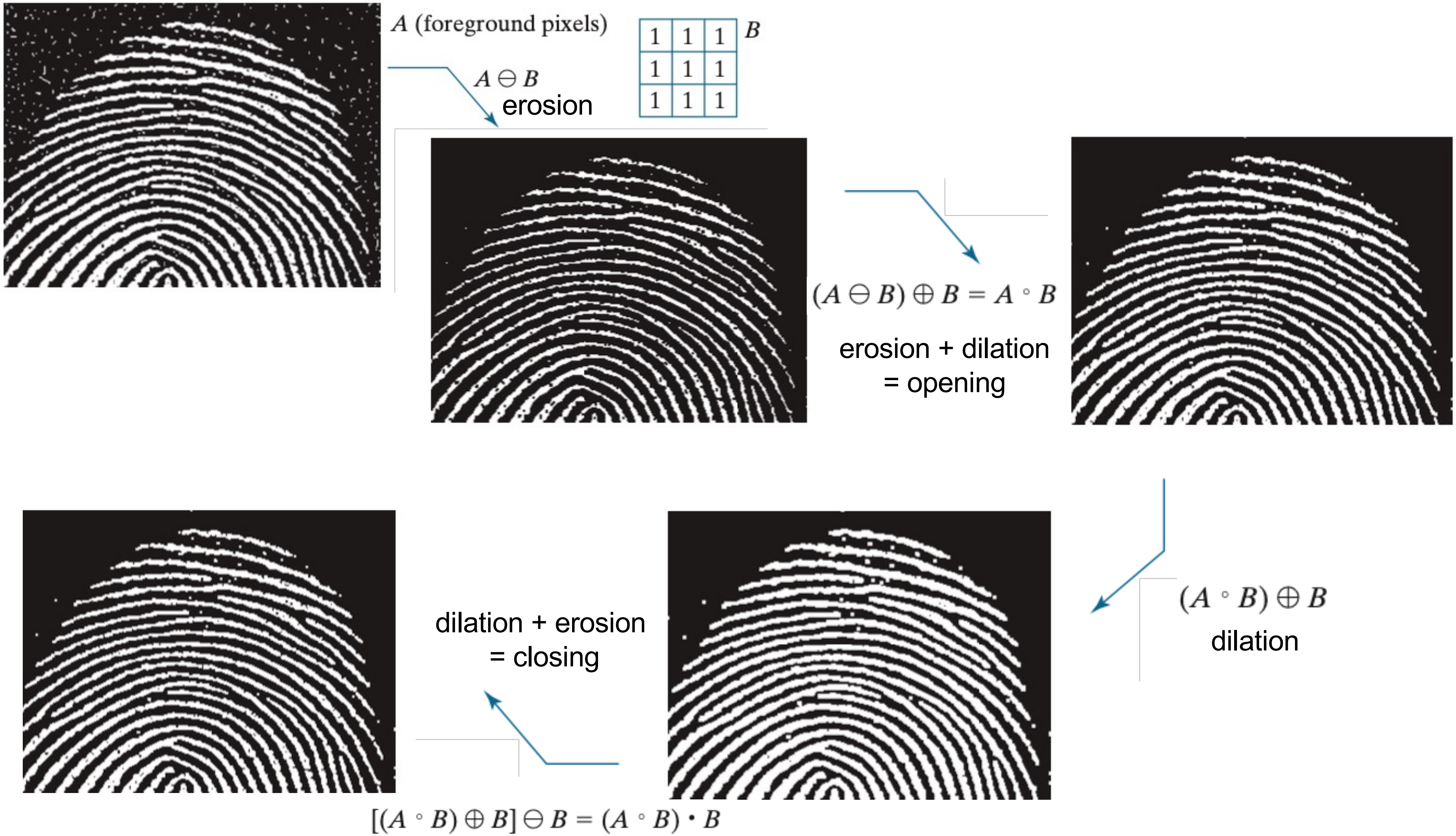




Closing in Action



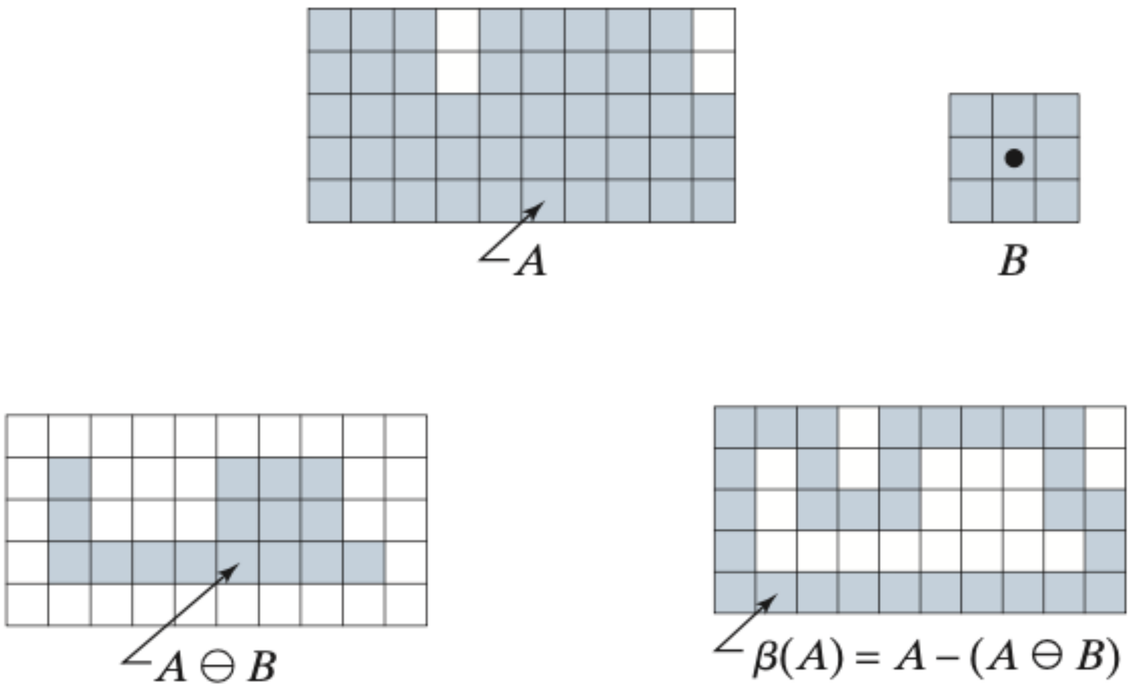
Morphological Filtering



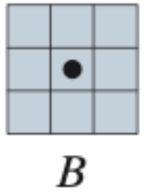
Boundary Extraction

- Boundary of a set A of foreground pixels denoted by $\beta(A)$, can be obtained by first eroding A by a suitable structuring element B, and then performing the set difference between A and its erosion.

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



Boundary Extraction Example



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

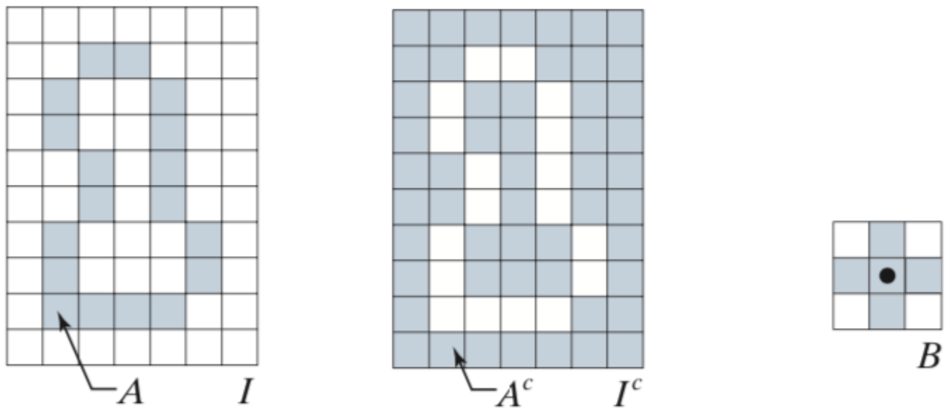
Hole filling

- A *hole* is defined as a background region surrounded by a **connected border** of foreground pixels.
- The formal definition of the hole filling algorithm is:

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

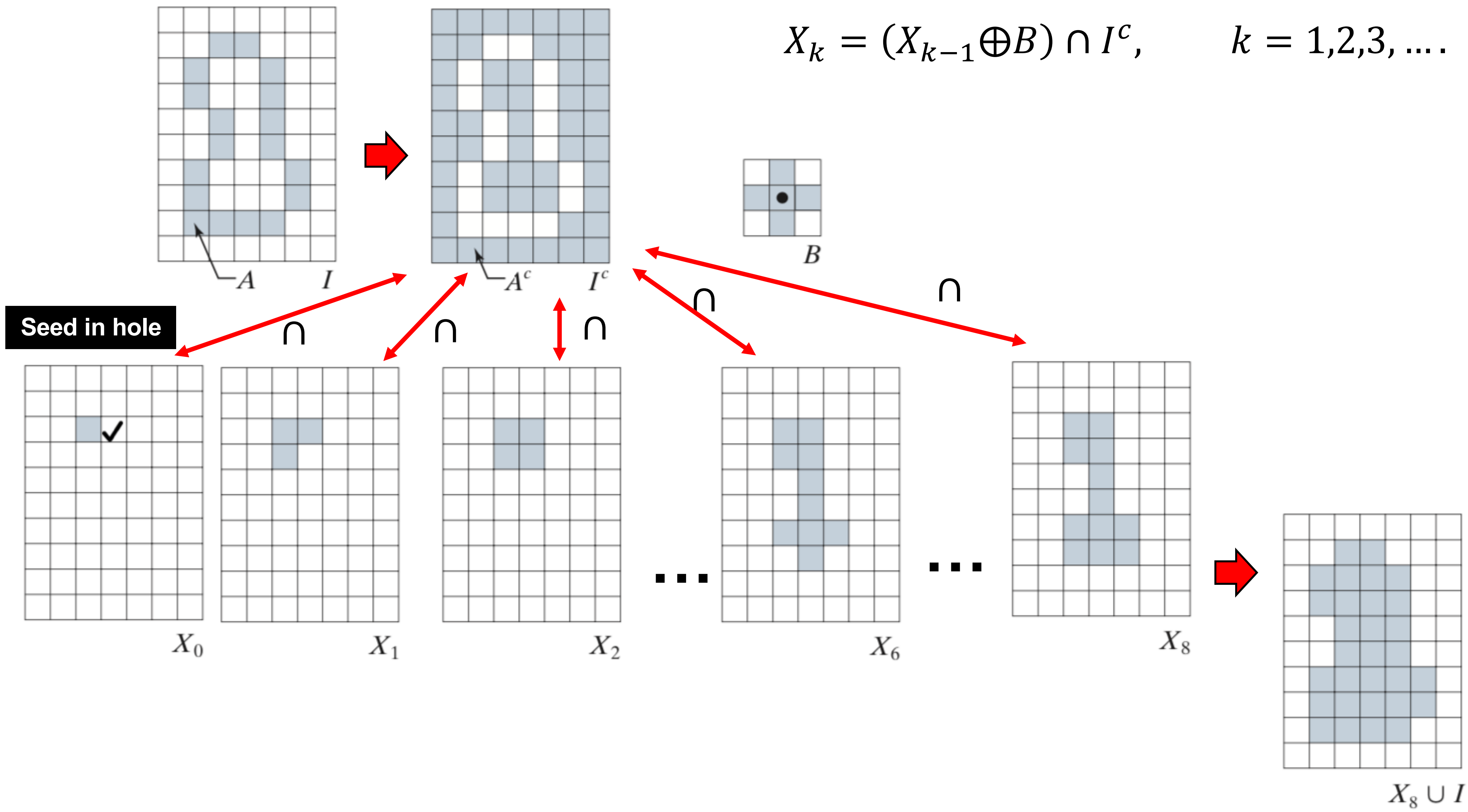
where X_0 is the first pixel known to be in the hole and X_k are the other hole pixels.

- Consider this example:



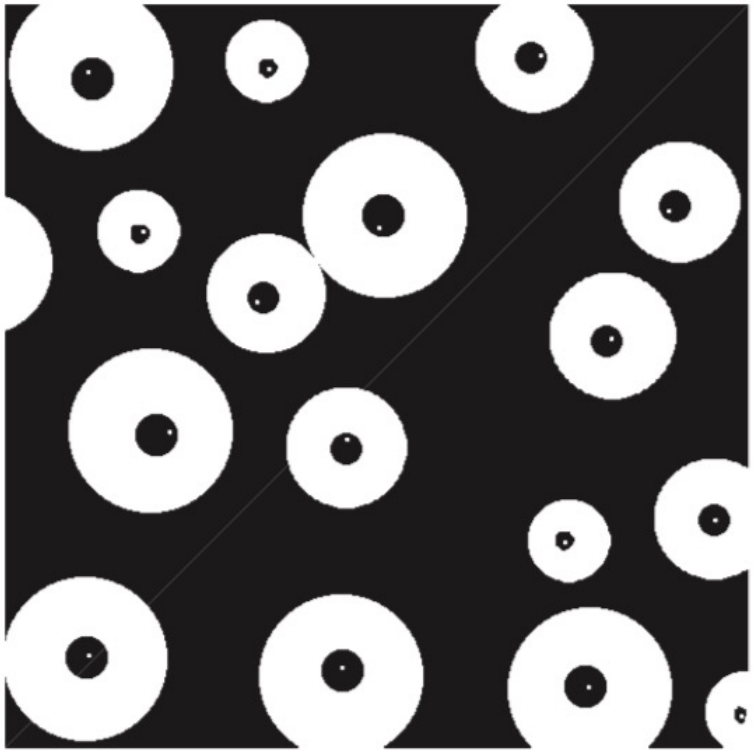
Hole filling

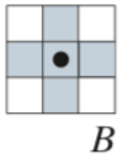
$$X_k = (X_{k-1} \oplus B) \cap I^c, \qquad k = 1,2,3,\ldots.$$

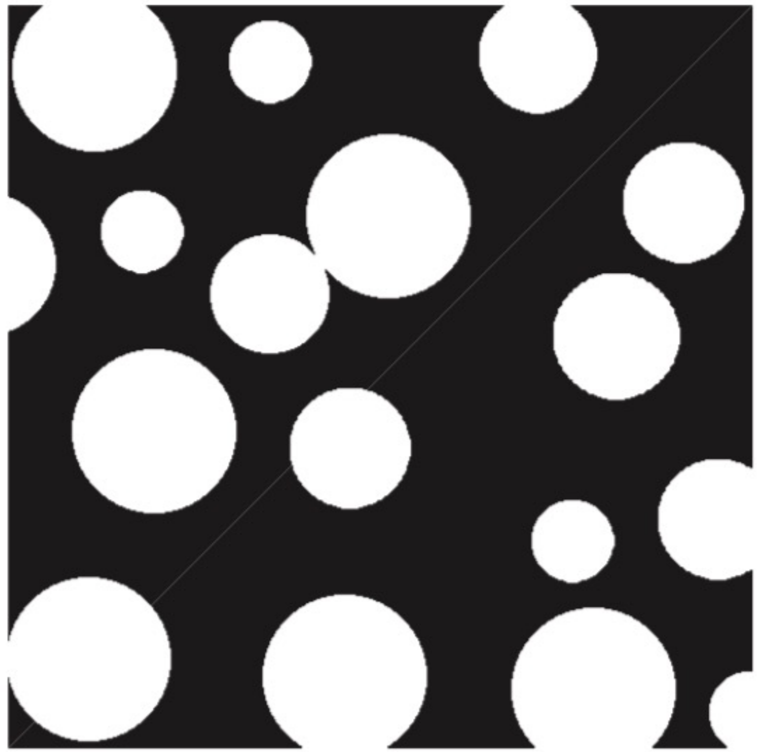


Example of Hole Filling

Thresholded image of ball bearings



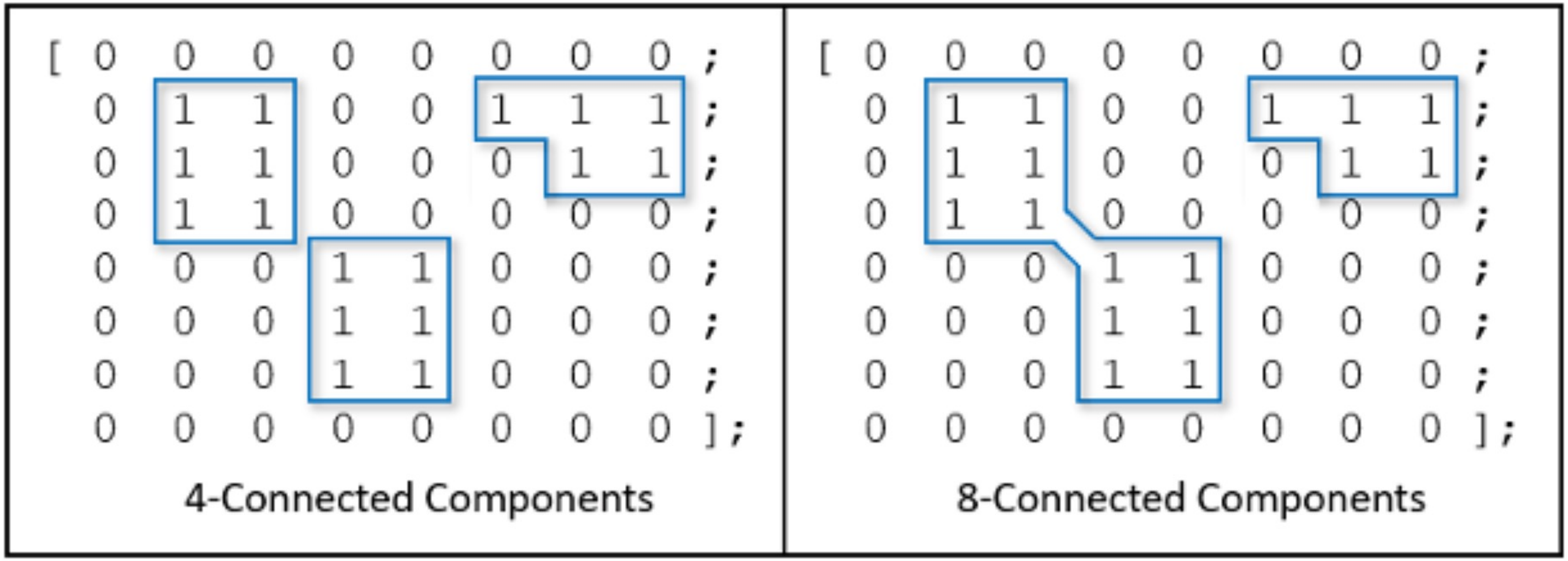




$$X_k = (X_{k-1} \oplus B) \cap I^c, \qquad k = 1,2,3, \dots .$$

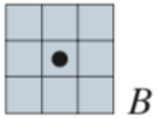
What are Connected Components?

- A **connected component** is a set of adjacent pixels in a binary image.
- Two possible definitions of what is “connected”:
  - 4-connectivity — Pixels are connected if their edges touch.
  - 8-connectivity — Pixels are connected if their edges or corners touch.

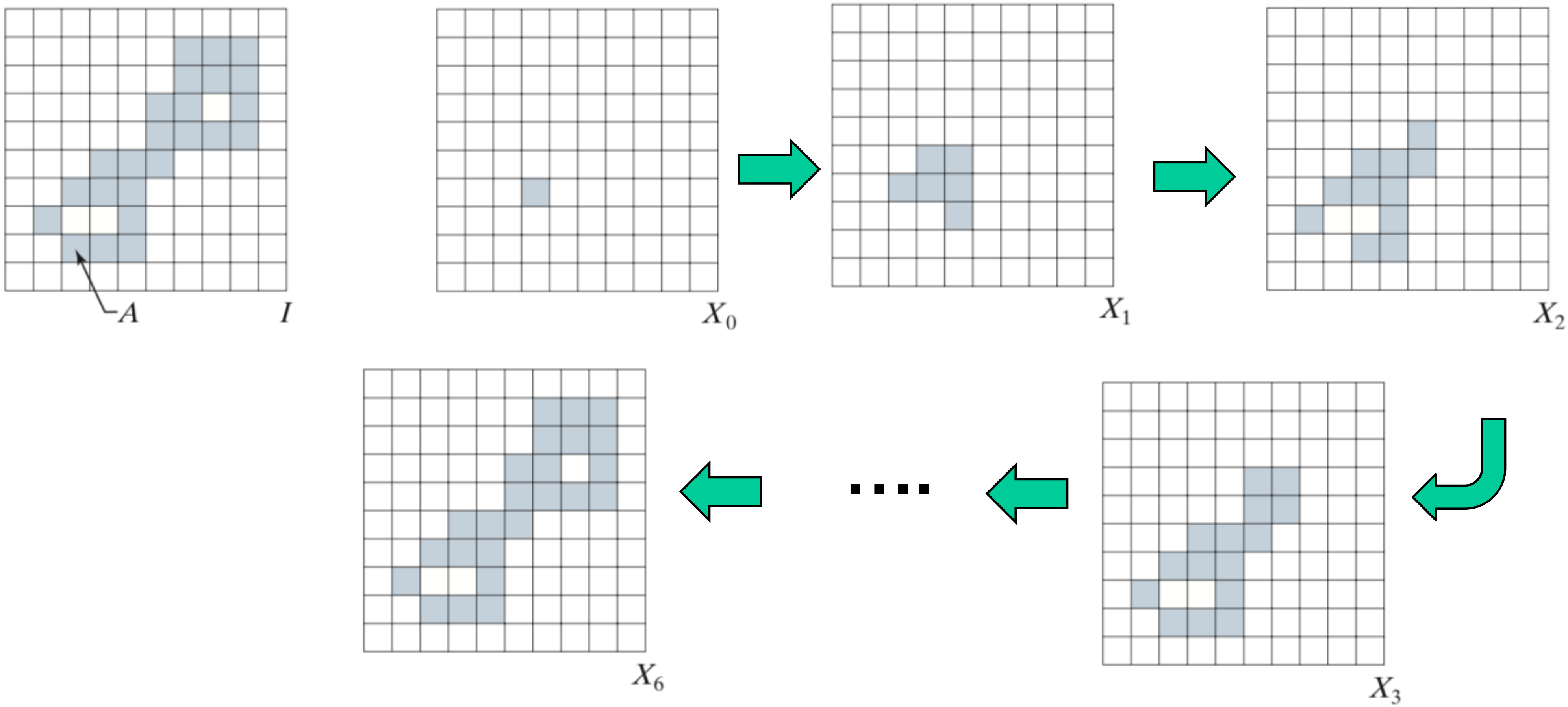


Extraction of Connected Components

- A connected component is a set of adjacent pixels in a binary image.

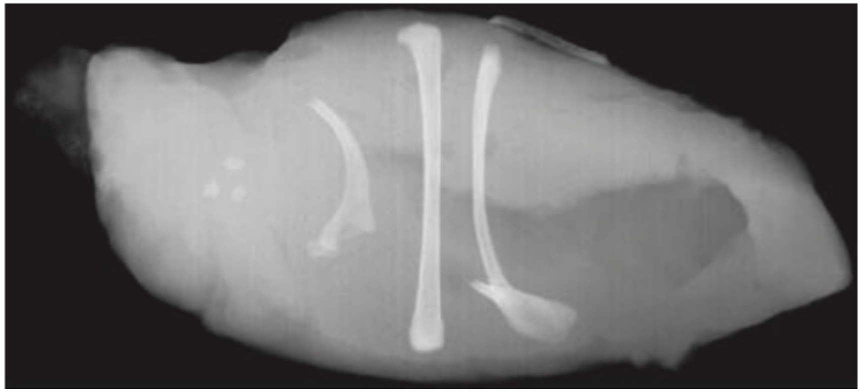


$$X_k = (X_{k-1} \oplus B) \cap I, \qquad k = 1,2,3,\dots.$$



Example application of Connected Components

- X-ray image of a chicken fillet with bone fragments embedded.



thresholding

Erosion with 5x5 SE of 1's

$$X_k = (X_{k-1} \oplus B) \cap I$$

15 connected components found with sizes in pixels

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