Image Processing

Lecture 11: Morphological Image Processing – II

(Ch9 Morphological Image Processing)

Zhiguo Zhang

zhiguozhang@hit.edu.cn

Review of Last Lecture

- In the last lecture we learnt:
  - What is morphology?
  - Basic concepts of set theory
  - Dilation and erosion
  - Opening and closing
  - Hit-or-miss transform

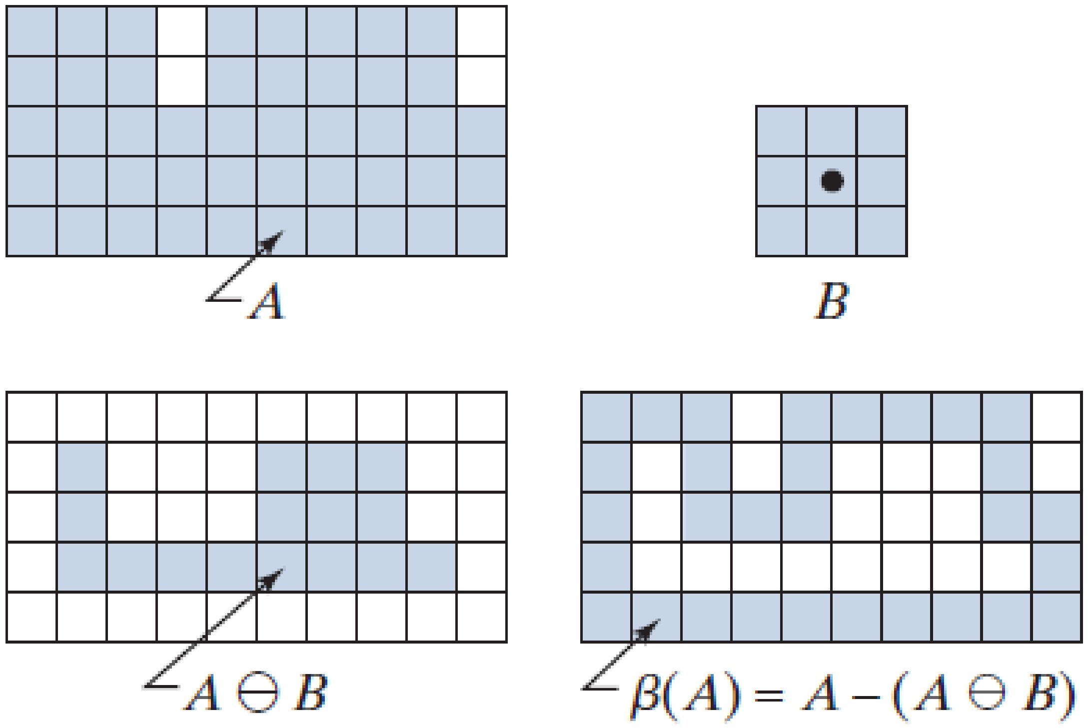
Contents of This Lecture

- In this lecture we will learn:
  - Boundary extraction
  - Region filling
  - Extraction of connected components
  - Convex hull
  - Thinning/thickening
  - Skeletons
  - Grayscale morphological processing

Boundary Extraction

- Extracting the boundary (or outline) of an object is often extremely useful.
- The boundary can be simply given as

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



Boundary Extraction Example

- A simple image and the result of performing boundary extraction using a square 3×3 structuring element.

Original Image

Extracted Boundary

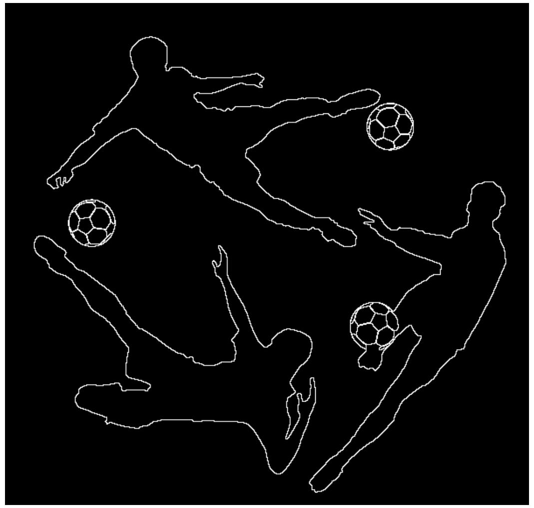
Note: large size of structuring element leads to thick boundary

Boundary Extraction Example

Original

Erosion of original

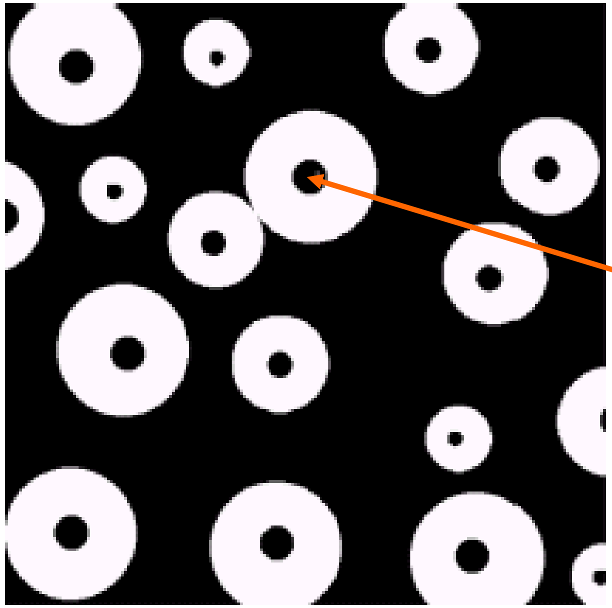
Boundary



Boundary overlaid on original

Region Filling

- Given a pixel inside a boundary, **region filling** attempts to fill that boundary with object pixels.



Given a point inside here, can we fill the whole circle?

Region Filling (cont…)

- Given an image I to be filled, the key equation for region filling is:

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

where X_0 is the starting point inside the boundary, B is a simple structuring element and I^c is the complement of I.

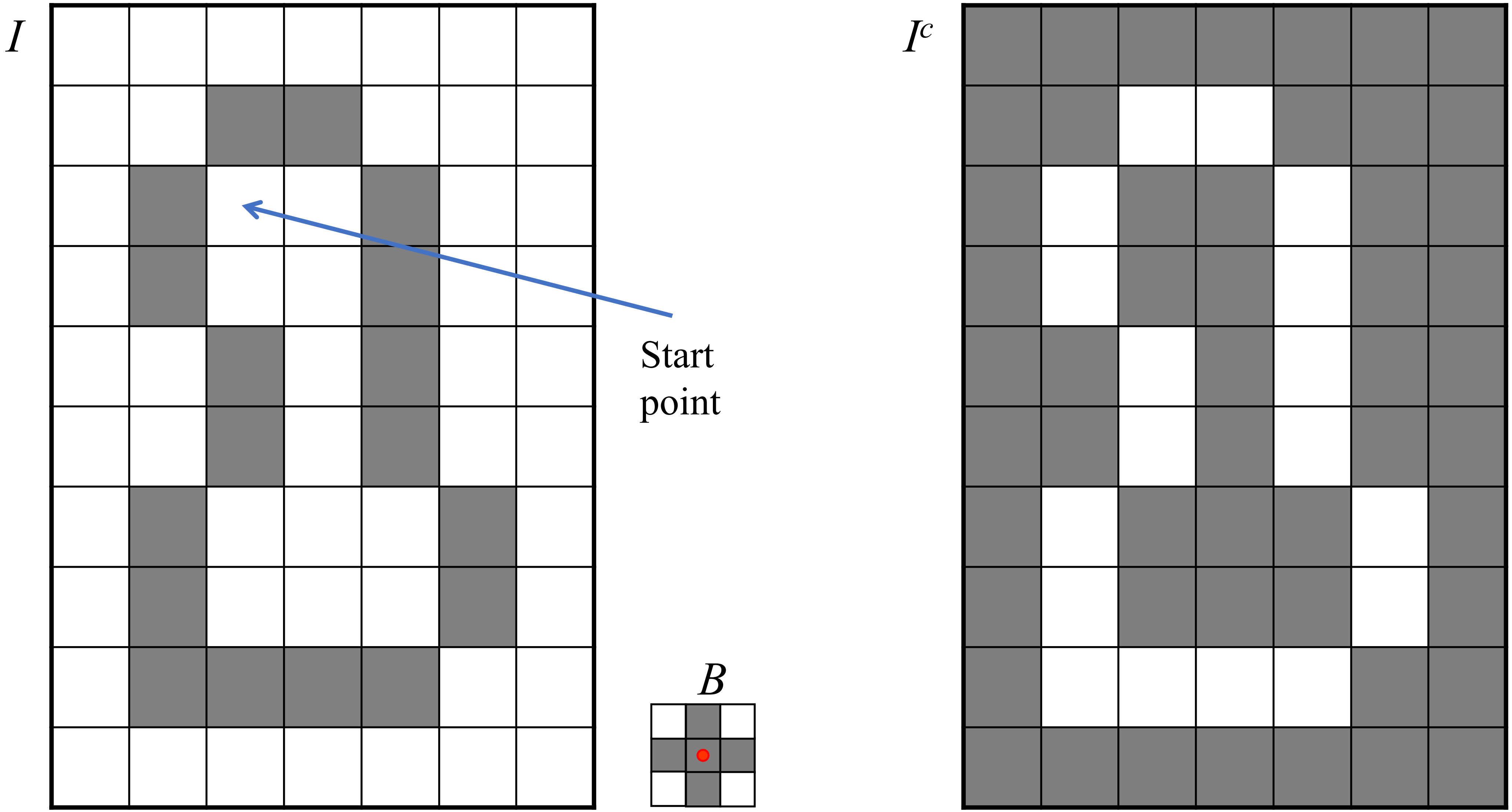
- This equation is applied repeatedly until:

$$X_k = X_{k-1}$$

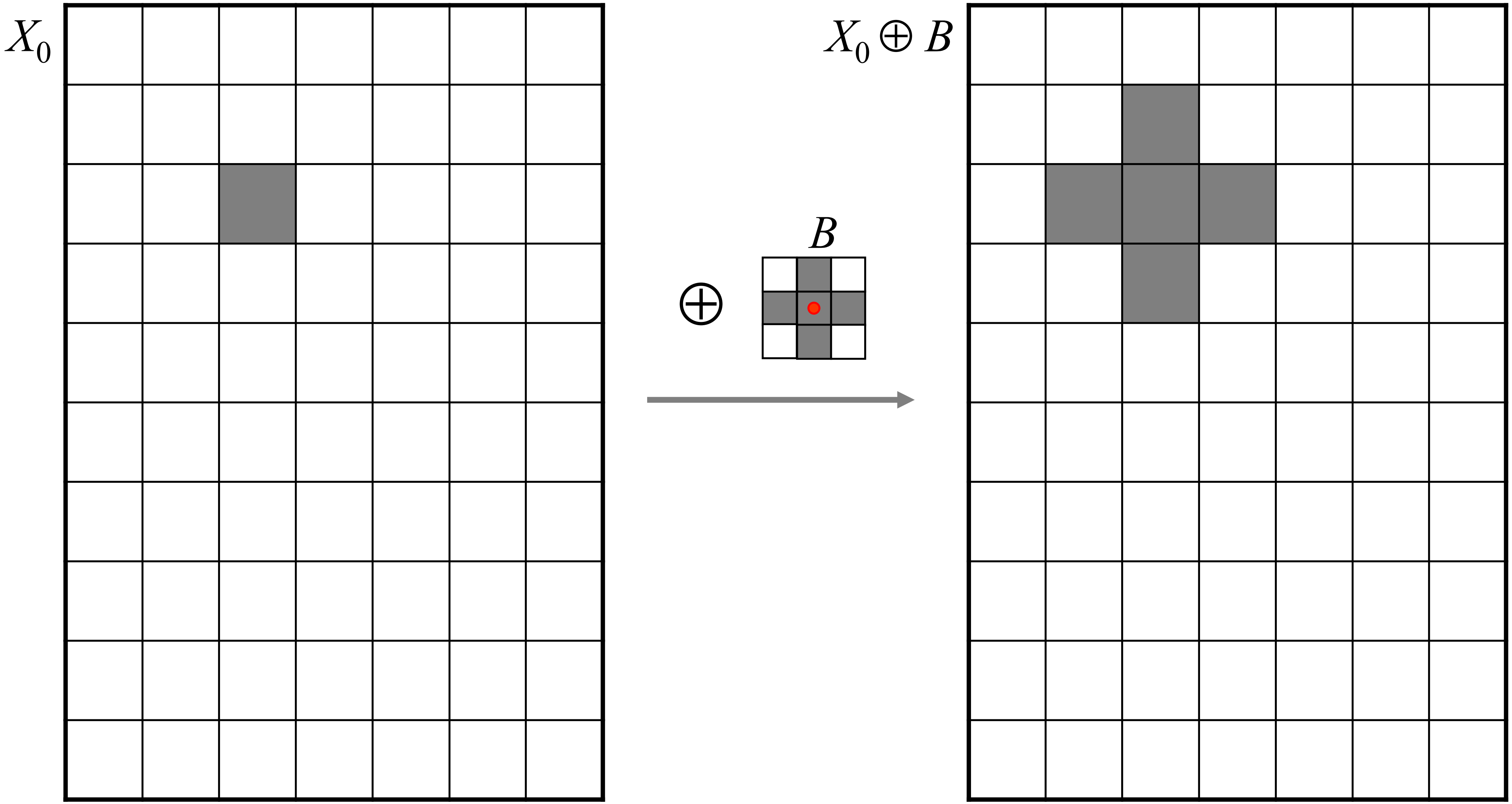
- Finally the result is the union between X_k and the original I:

$$X_k \cup I$$

Region Filling Step By Step

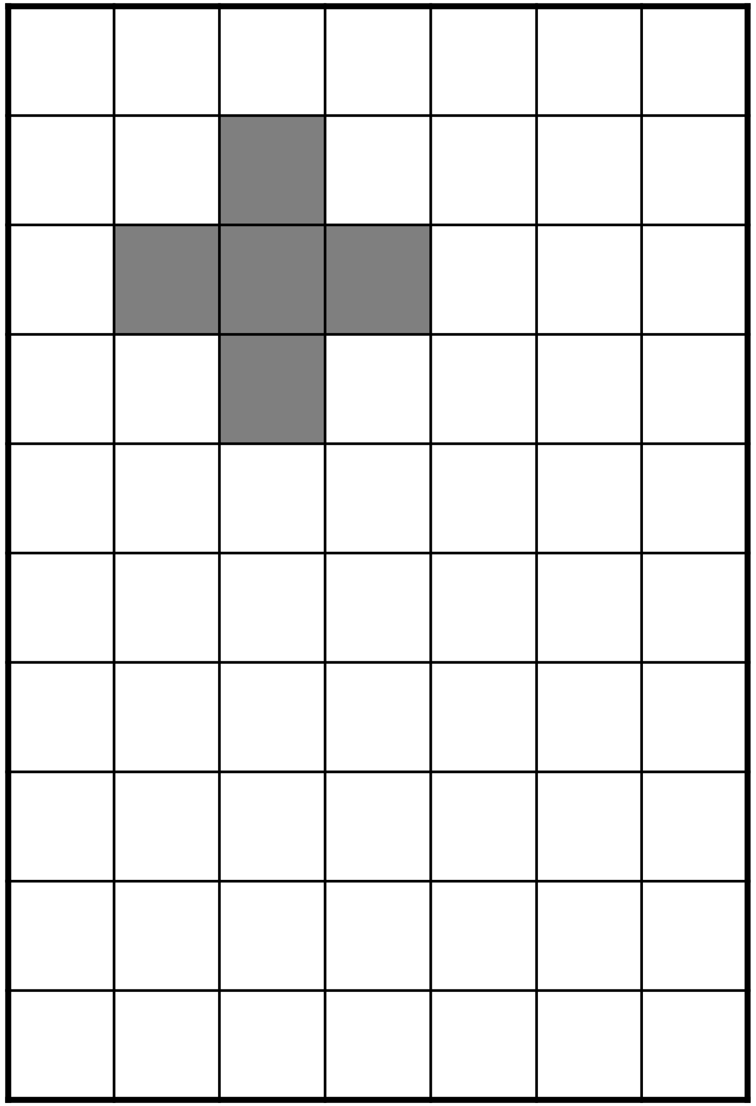


Region Filling Step By Step

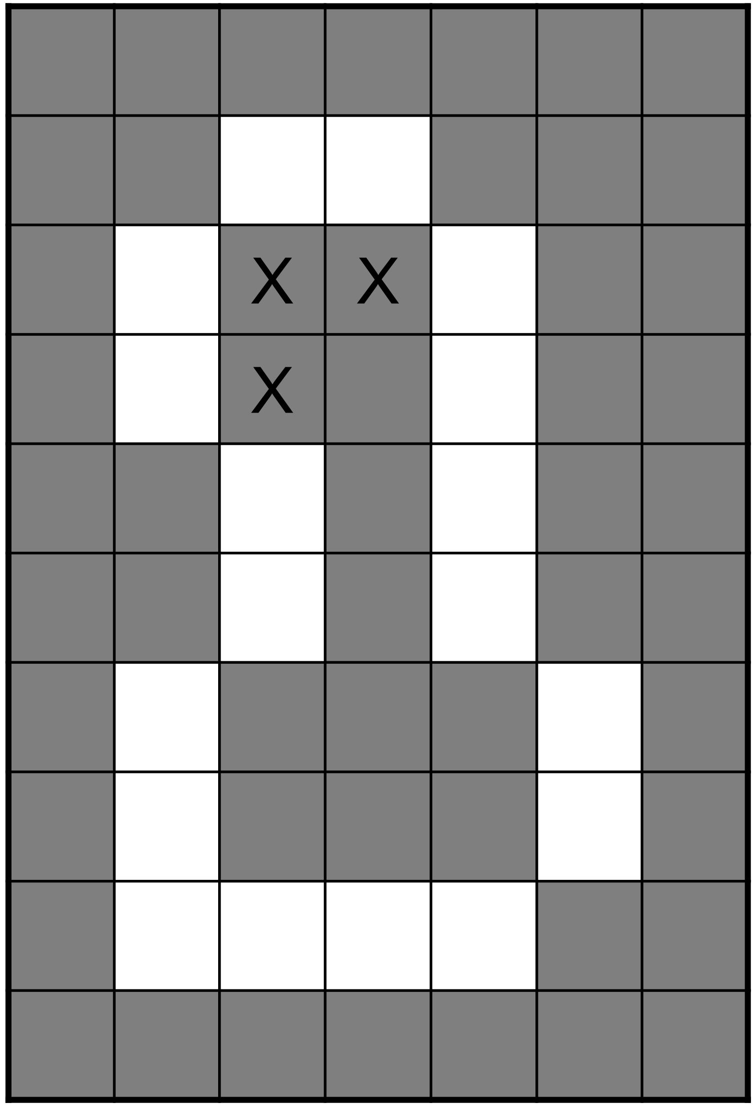


Region Filling Step By Step

$X_0 \oplus B$

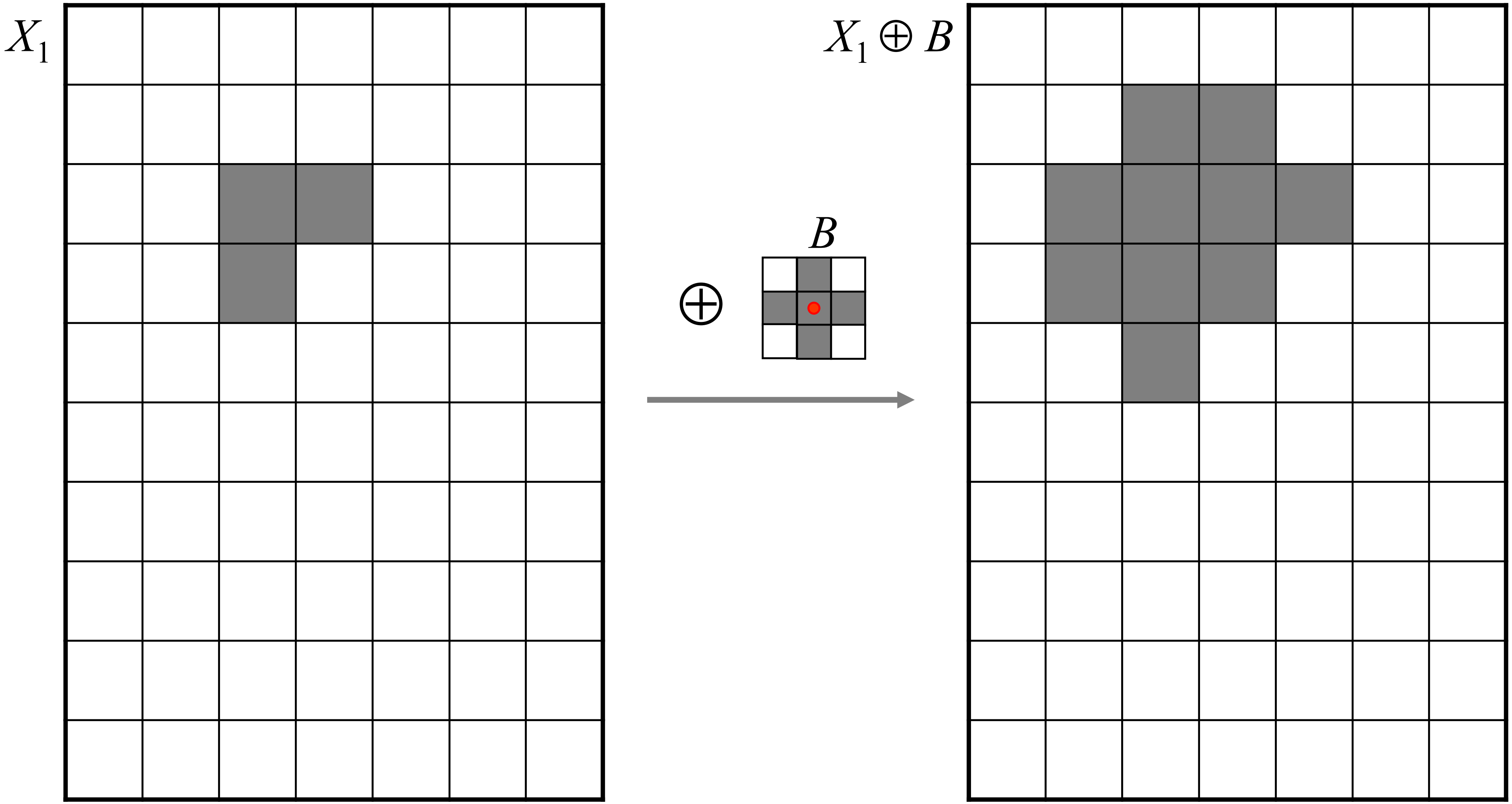


I^c



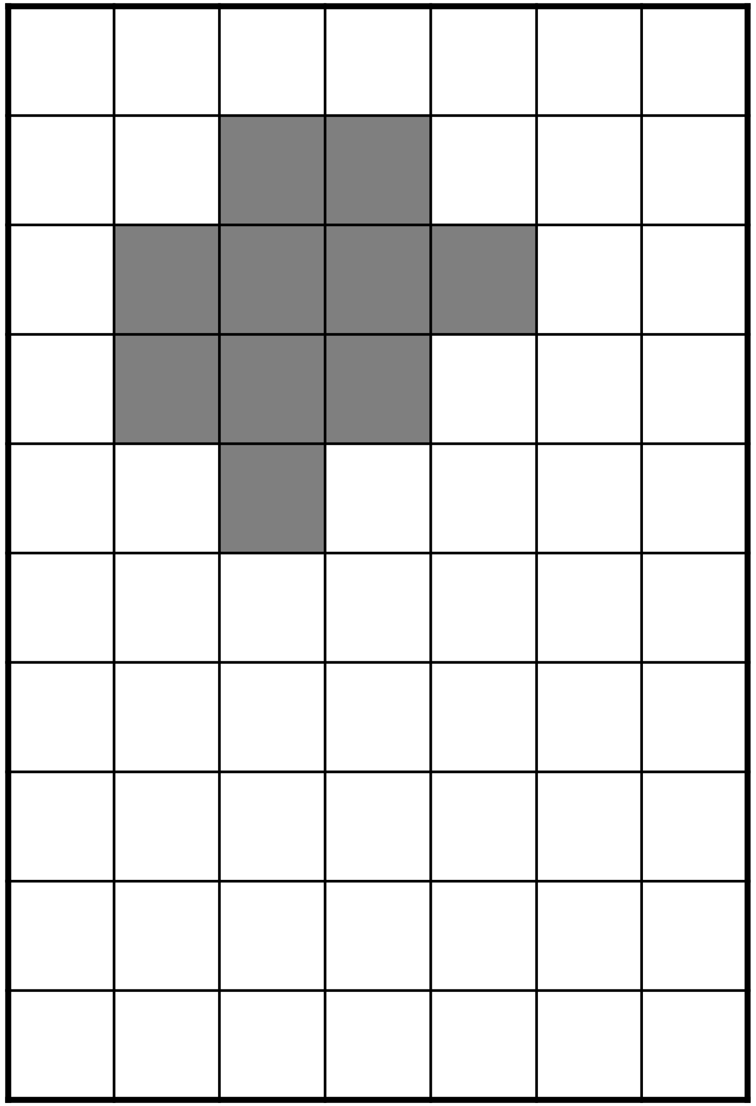
$X_1 = (X_0 \oplus B) \cap I^c$

Region Filling Step By Step

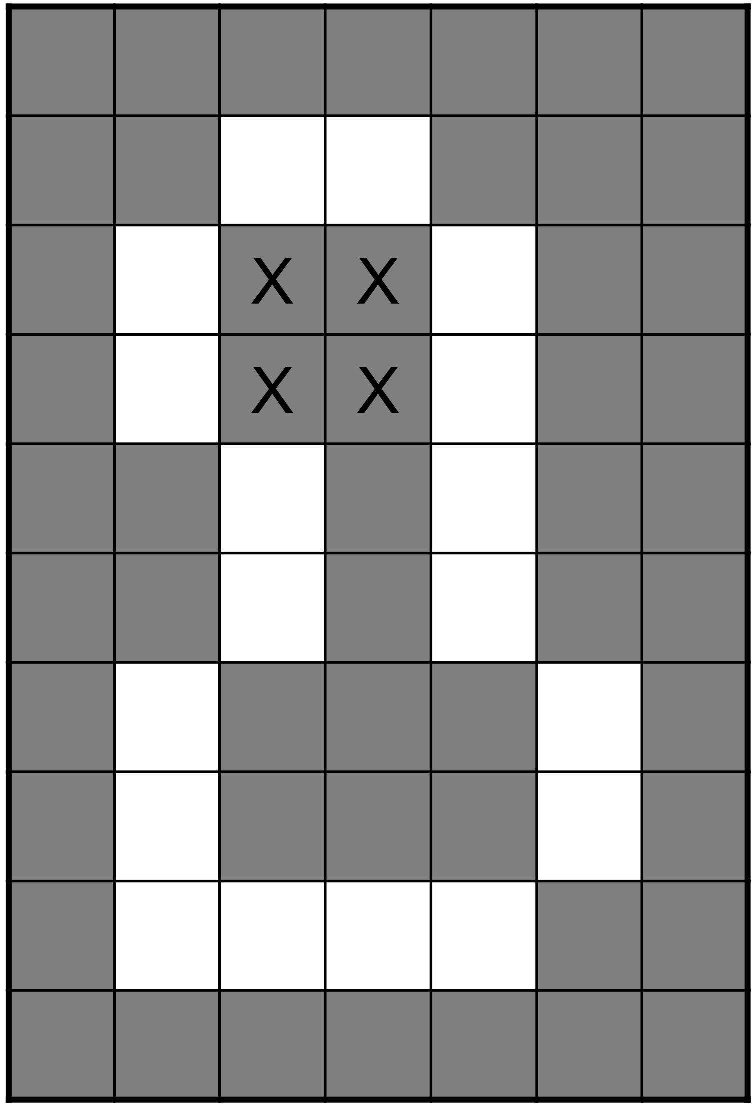


Region Filling Step By Step

$X_1 \oplus B$



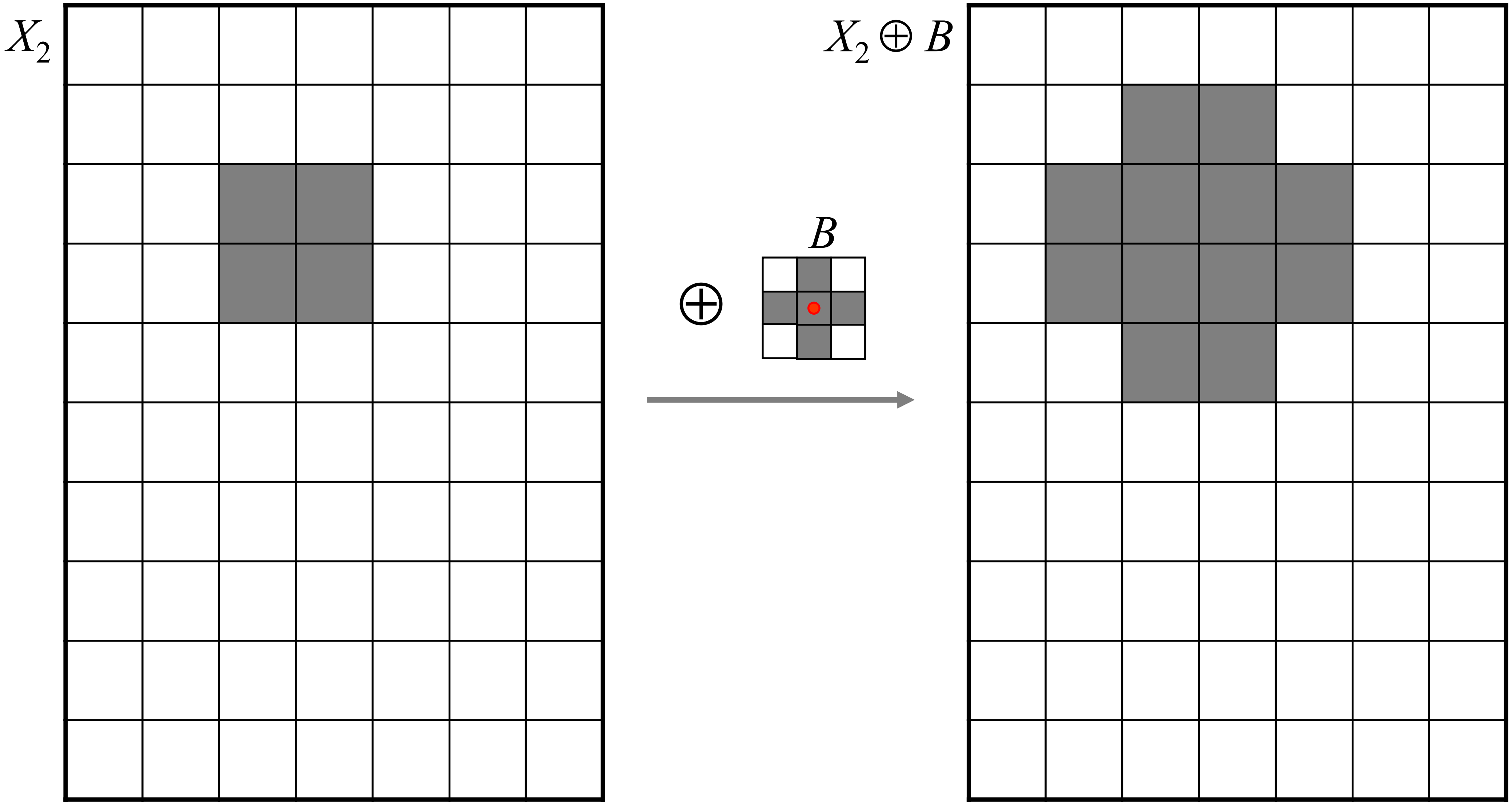
I^c



$\cap$

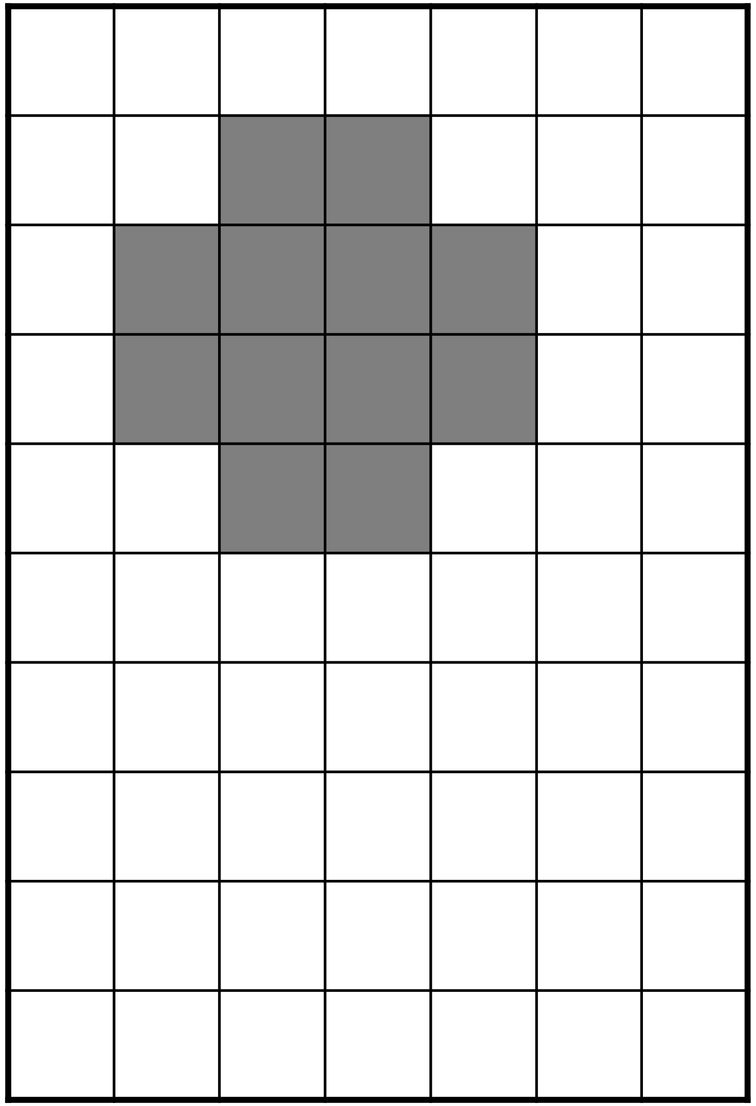
$X_2 = (X_1 \oplus B) \cap I^c$

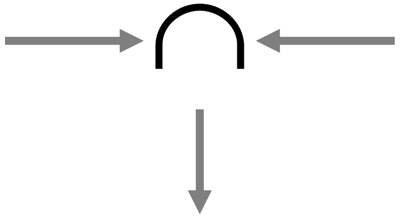
Region Filling Step By Step



Region Filling Step By Step

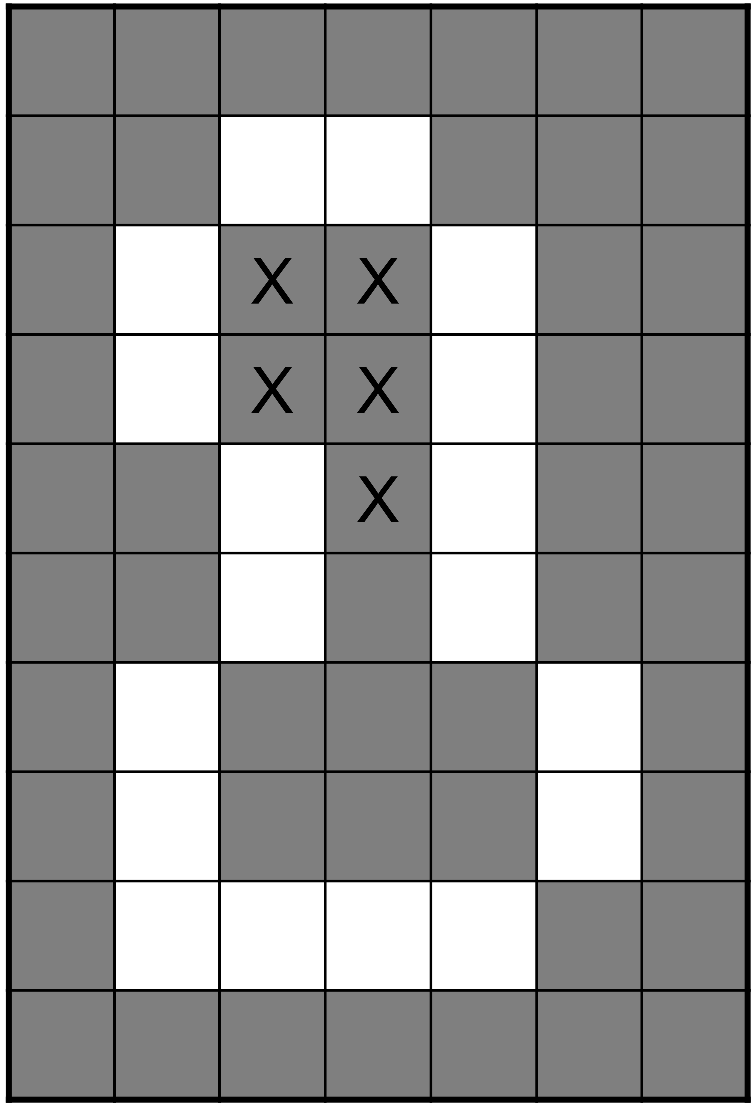
$X_2 \oplus B$



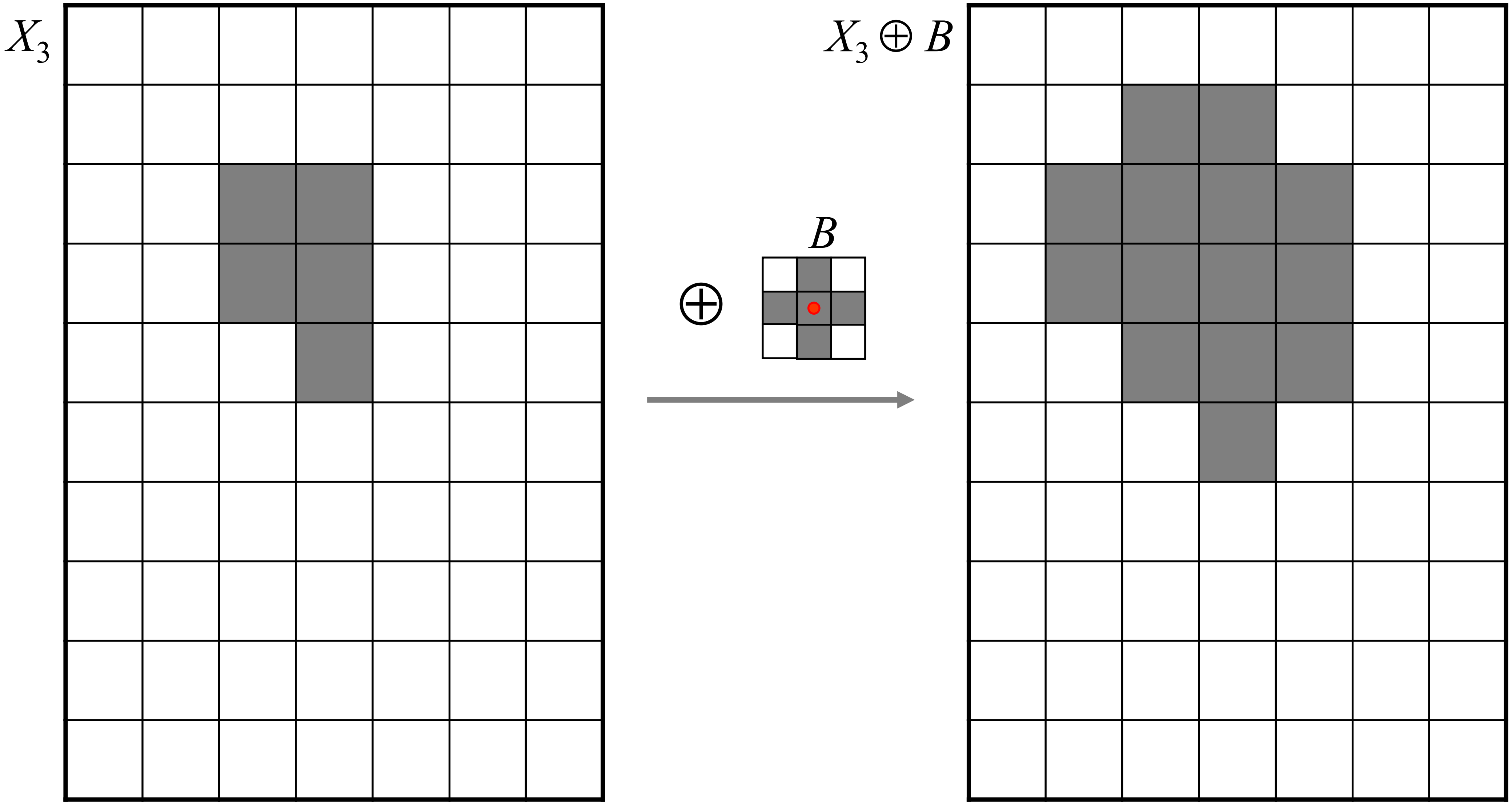


$X_3 = (X_2 \oplus B) \cap I^c$

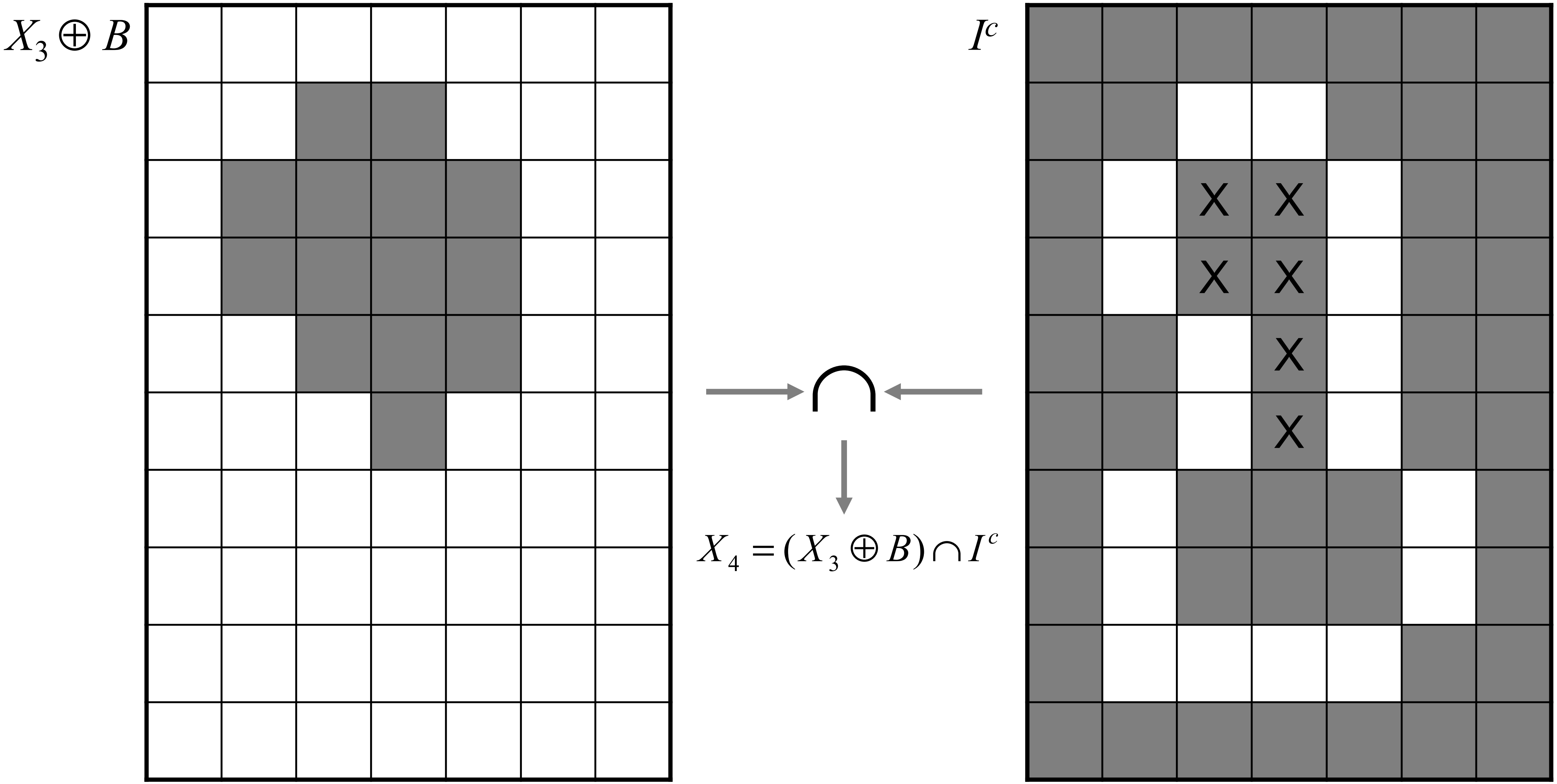
I^c



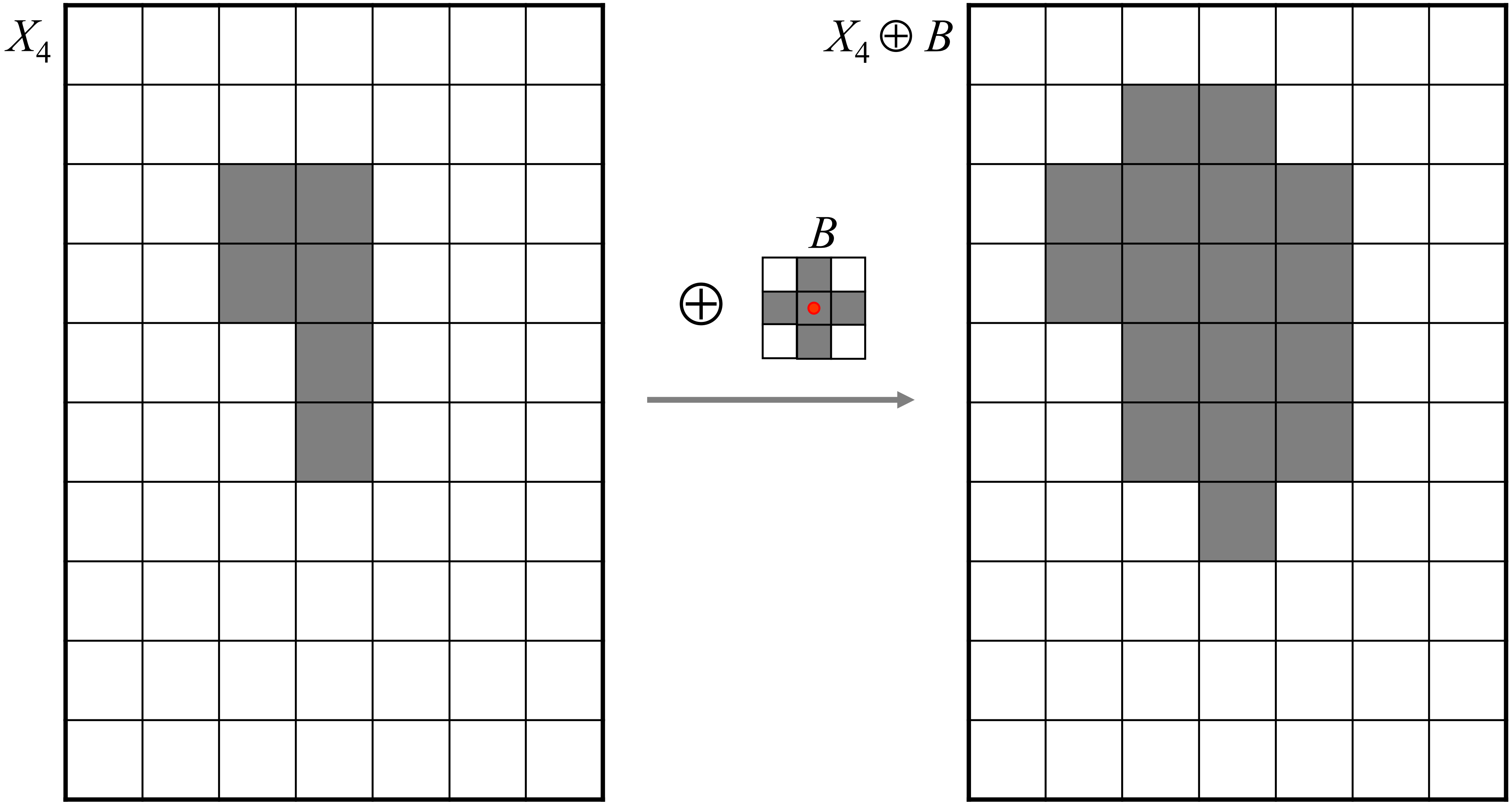
Region Filling Step By Step



Region Filling Step By Step

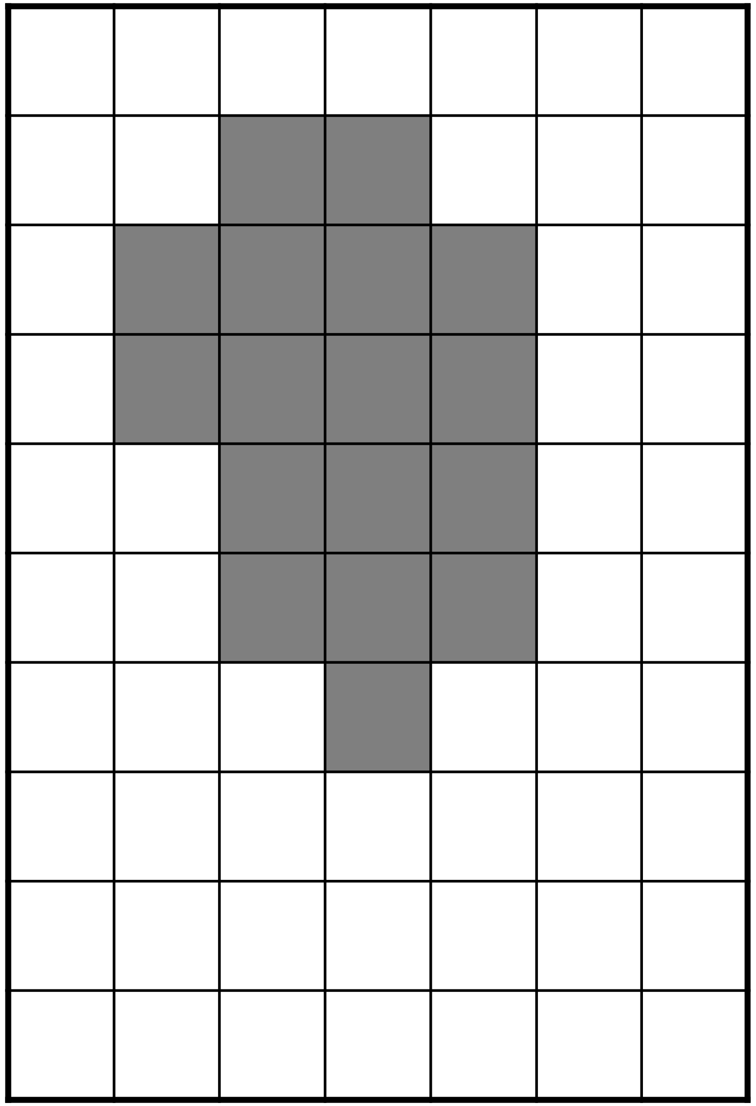


Region Filling Step By Step

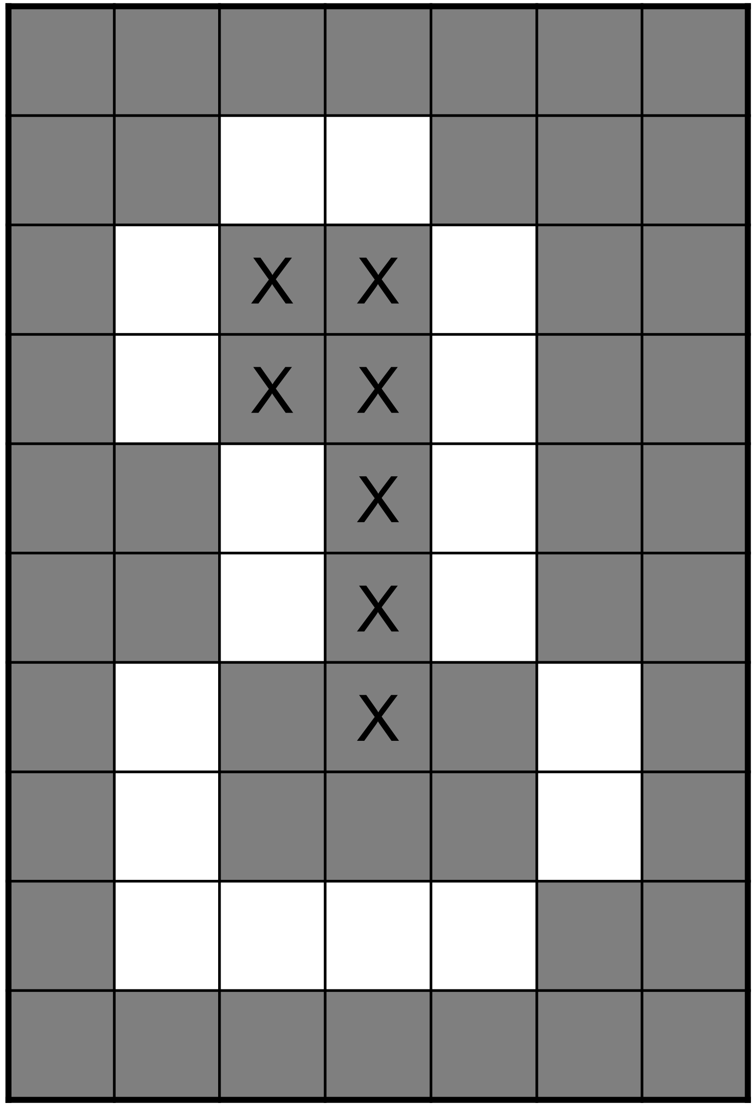


Region Filling Step By Step

$X_4 \oplus B$



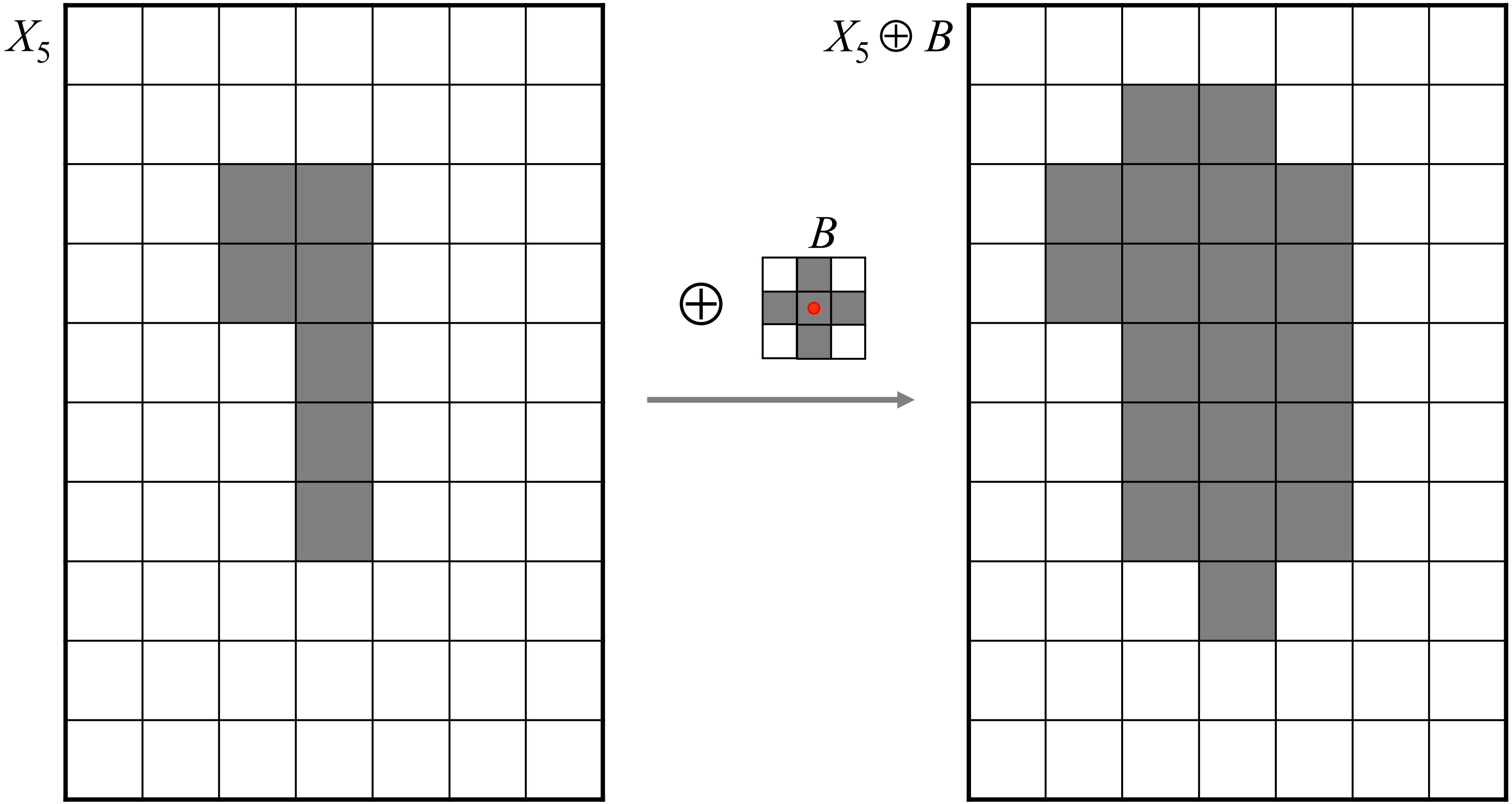
I^c



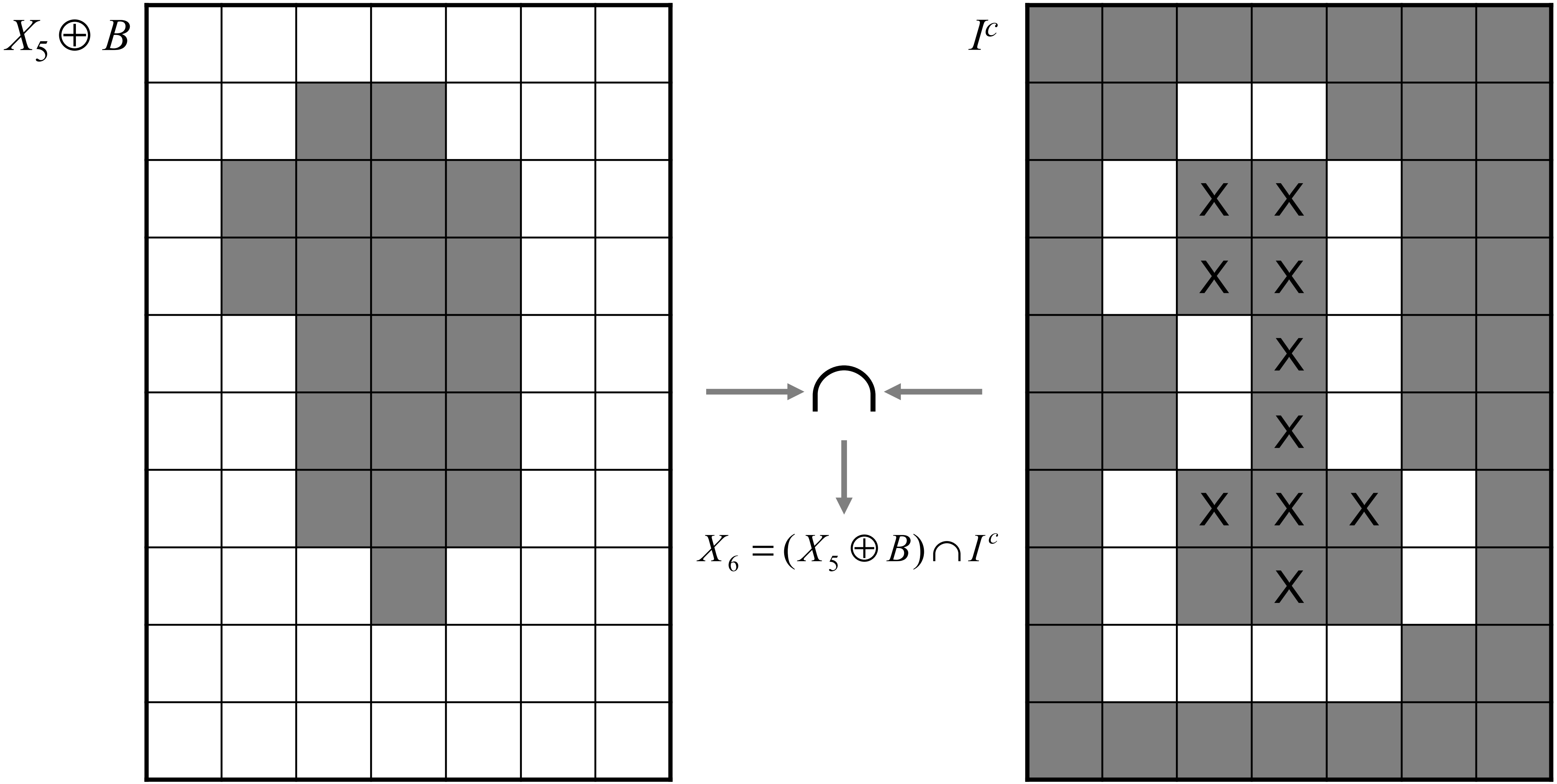
$\cap$

$X_5 = (X_4 \oplus B) \cap I^c$

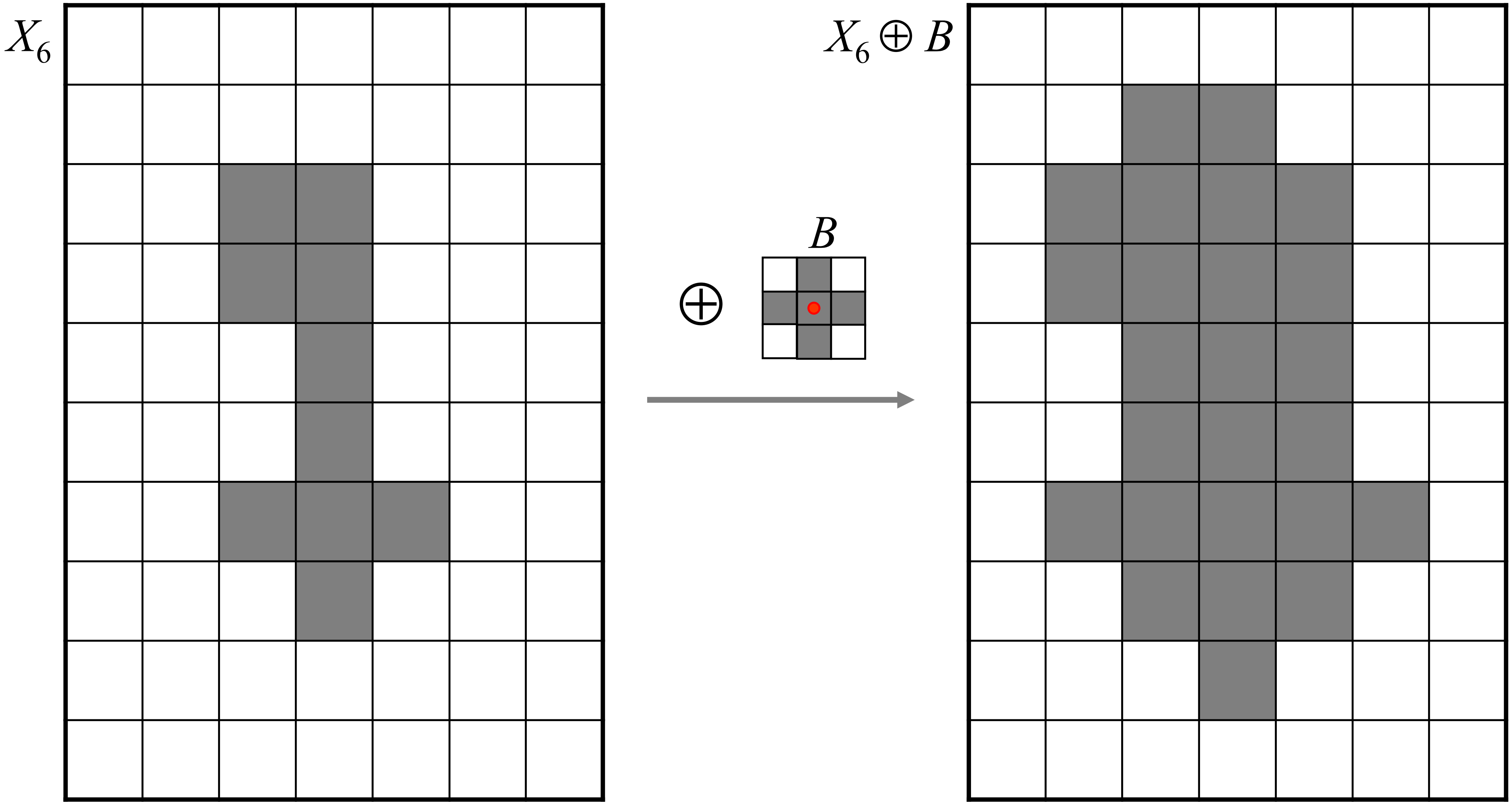
Region Filling Step By Step



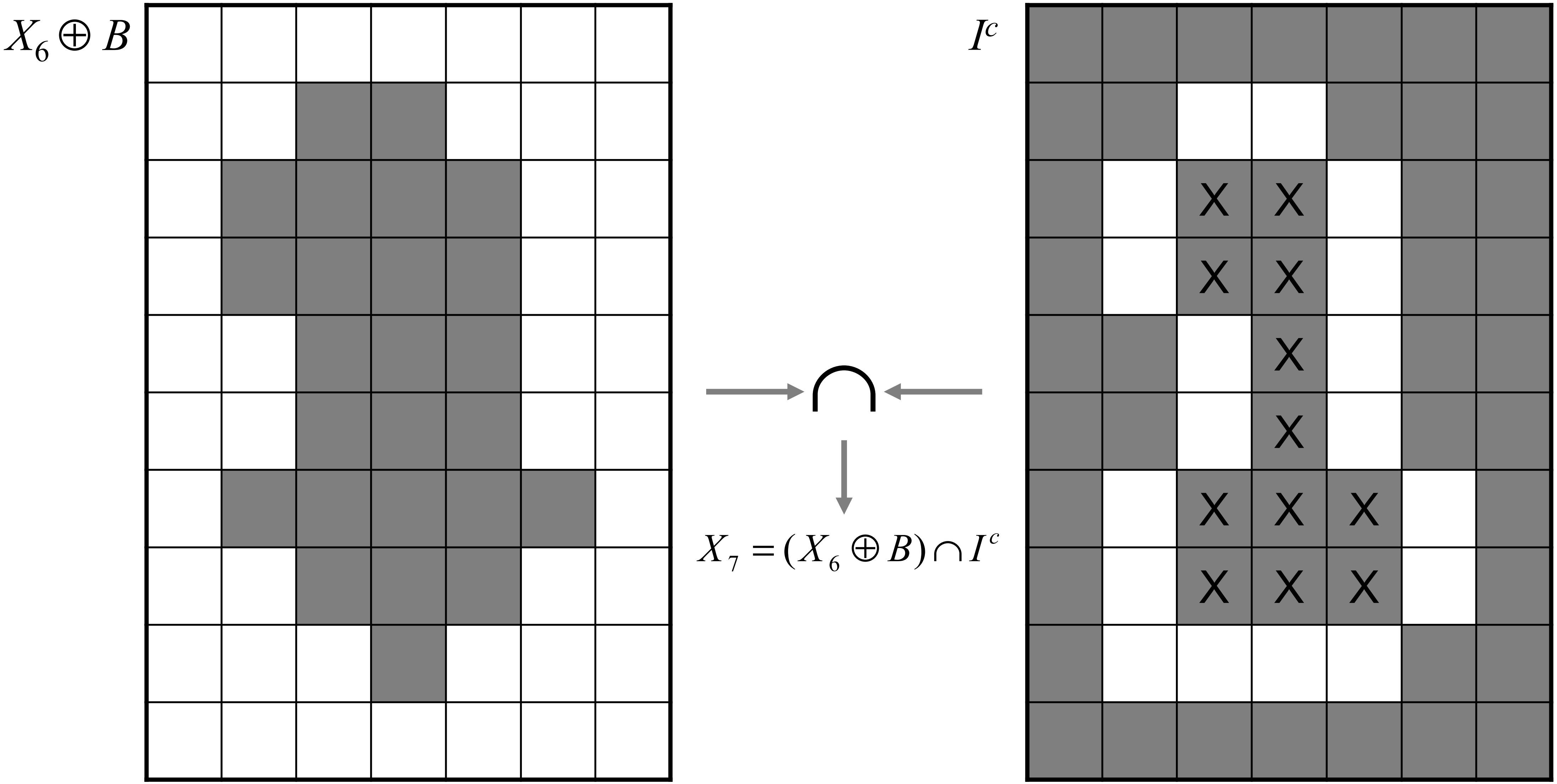
Region Filling Step By Step



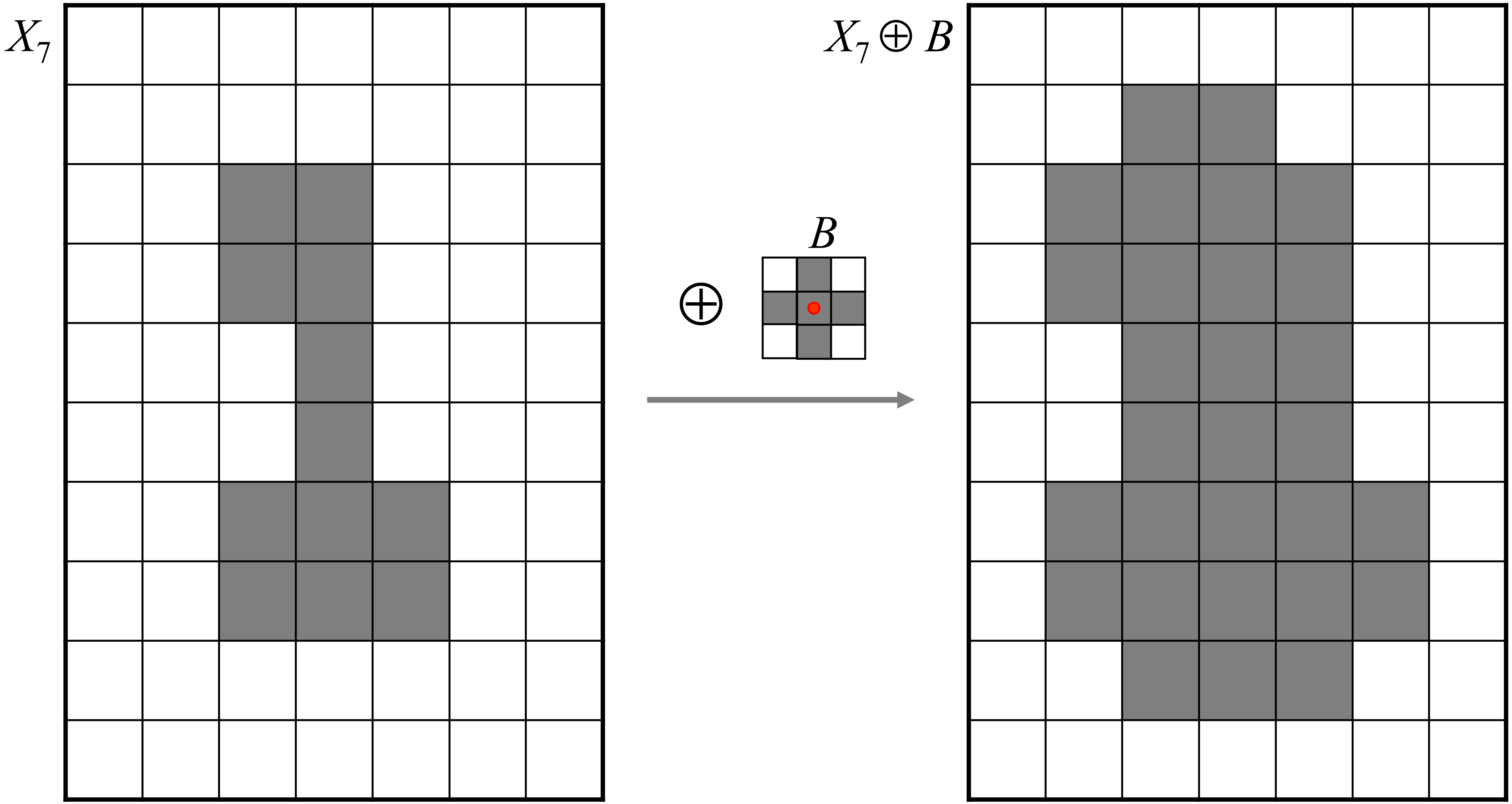
Region Filling Step By Step



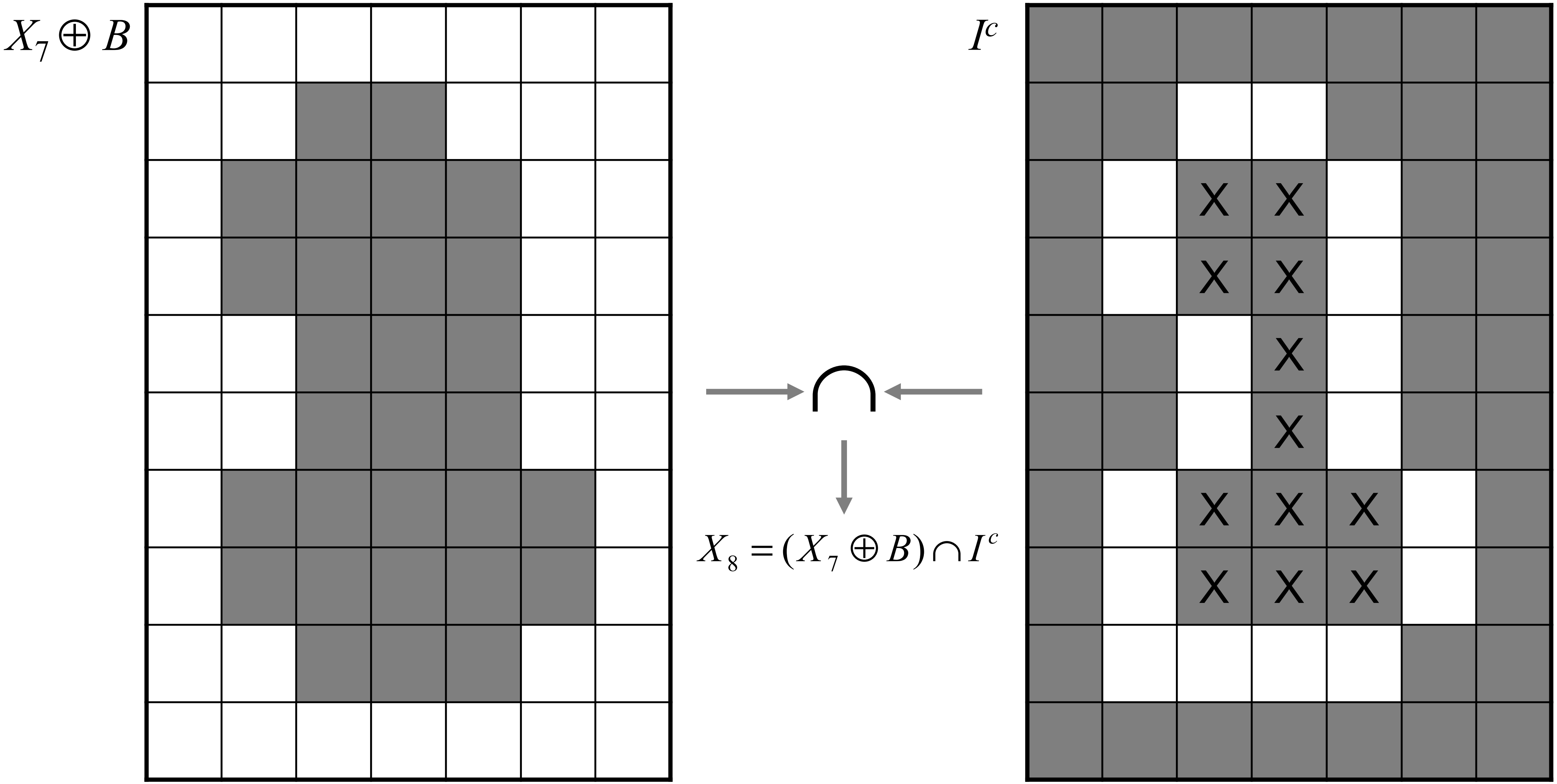
Region Filling Step By Step



Region Filling Step By Step

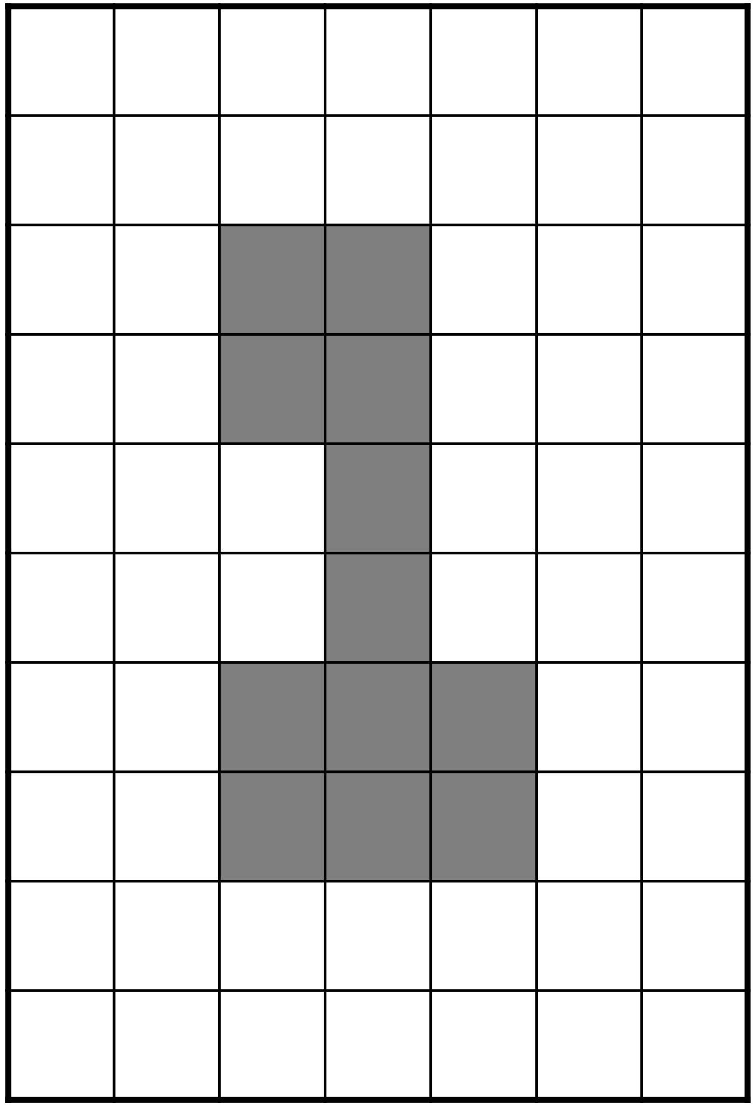


Region Filling Step By Step



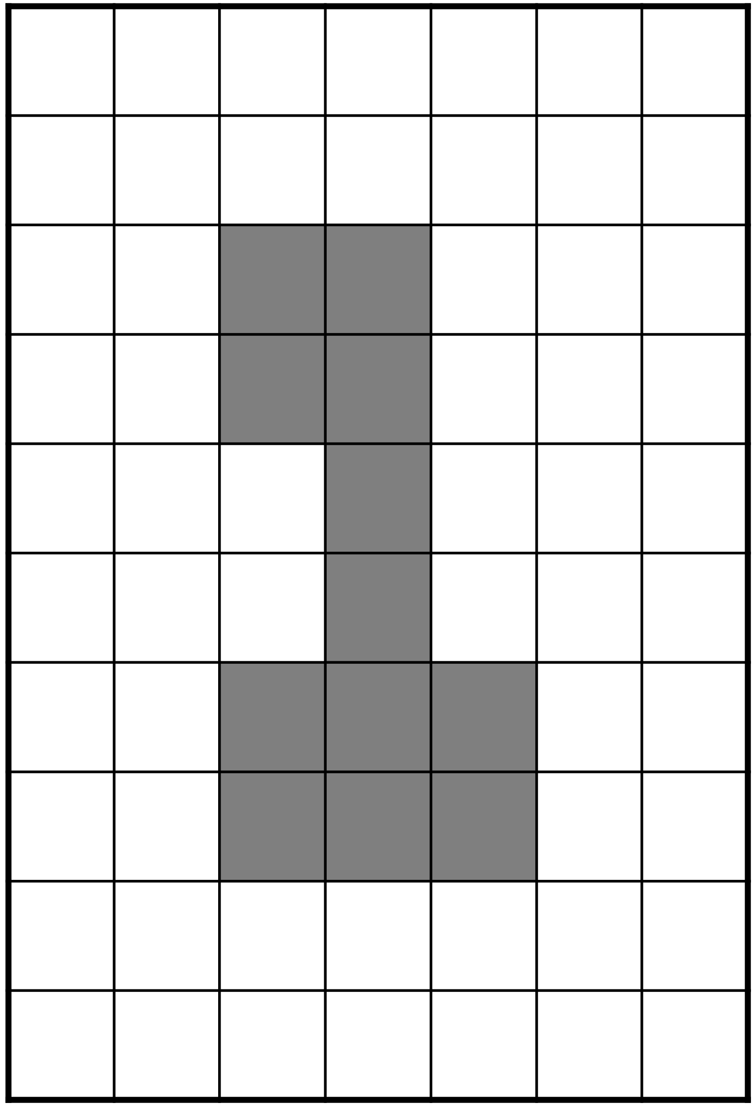
Region Filling Step By Step

X_8

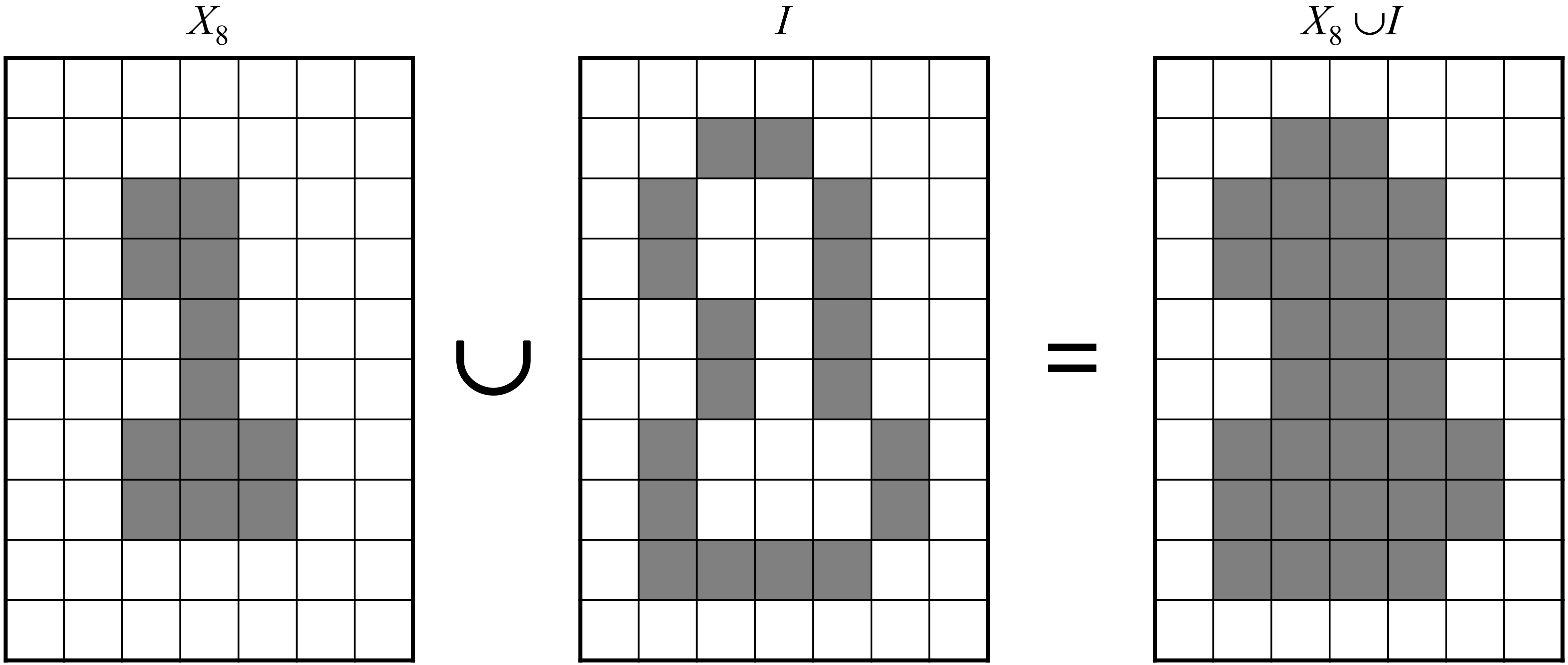


$=$

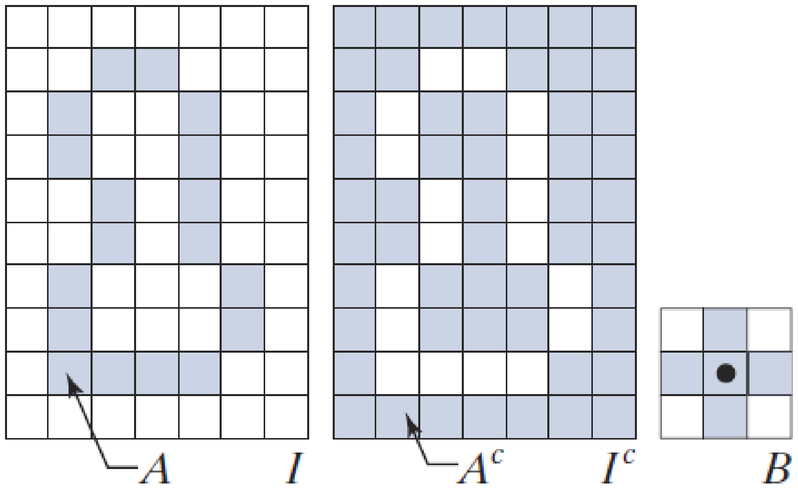
X_7

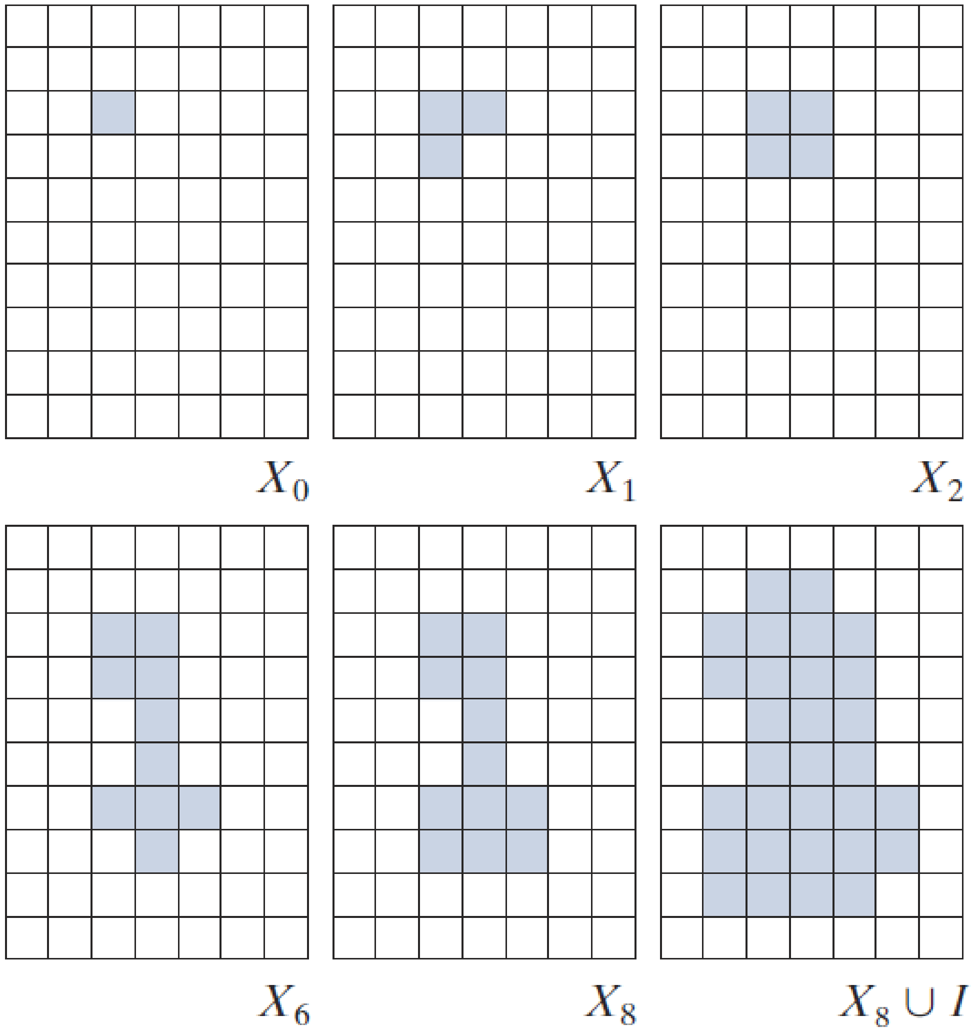


Region Filling Step By Step

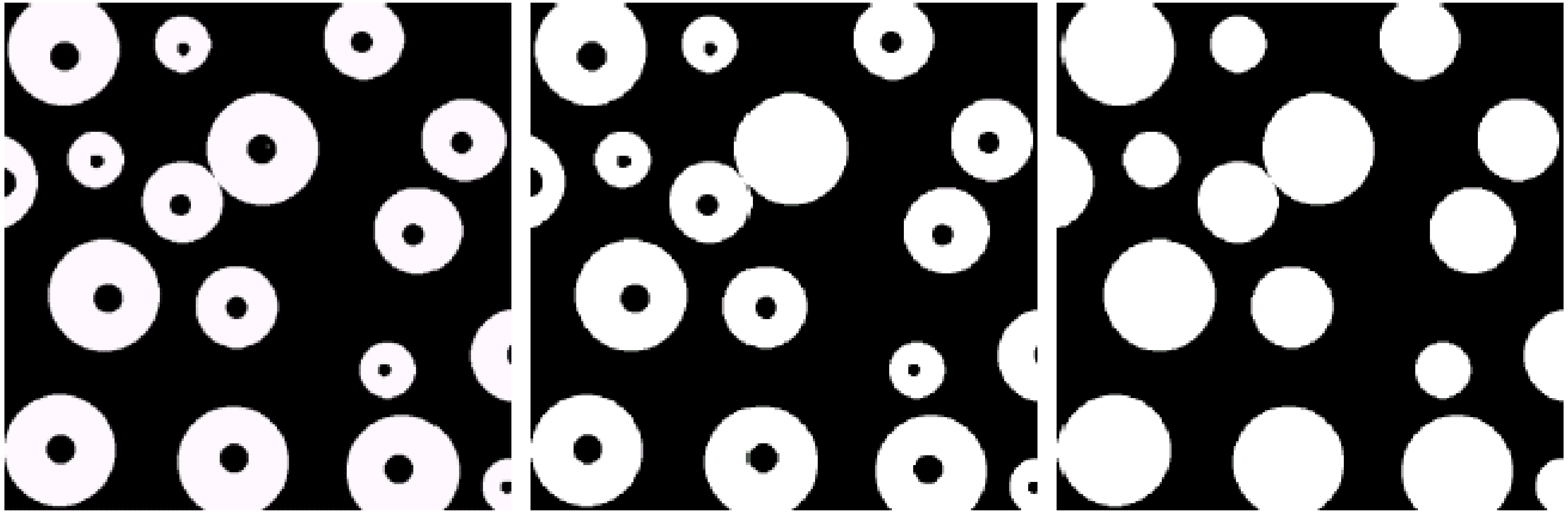


Region Filling Step By Step





Region Filling Example



Original Image One Region Filled All Regions Filled

Note: It is must be known whether black points are background points or sphere inner point. Fully automating this procedure requires additional intelligence be built into the algorithm.

Extraction of Connected Components

- Being able to extract connected components from a binary image is central to many automated image analysis applications.
- Let A be a set of foreground pixels consisting of one or more connected components, and form an image X_0 (of the same size as I, the image containing A) whose elements are 0's (background values), except at each location known to correspond to a point in each connected component in A, which we set to 1 (foreground value).
- The objective is to start with X_0 and find all the connected components in I.

Extraction of Connected Components

- Find a starting point X_0 in connected components

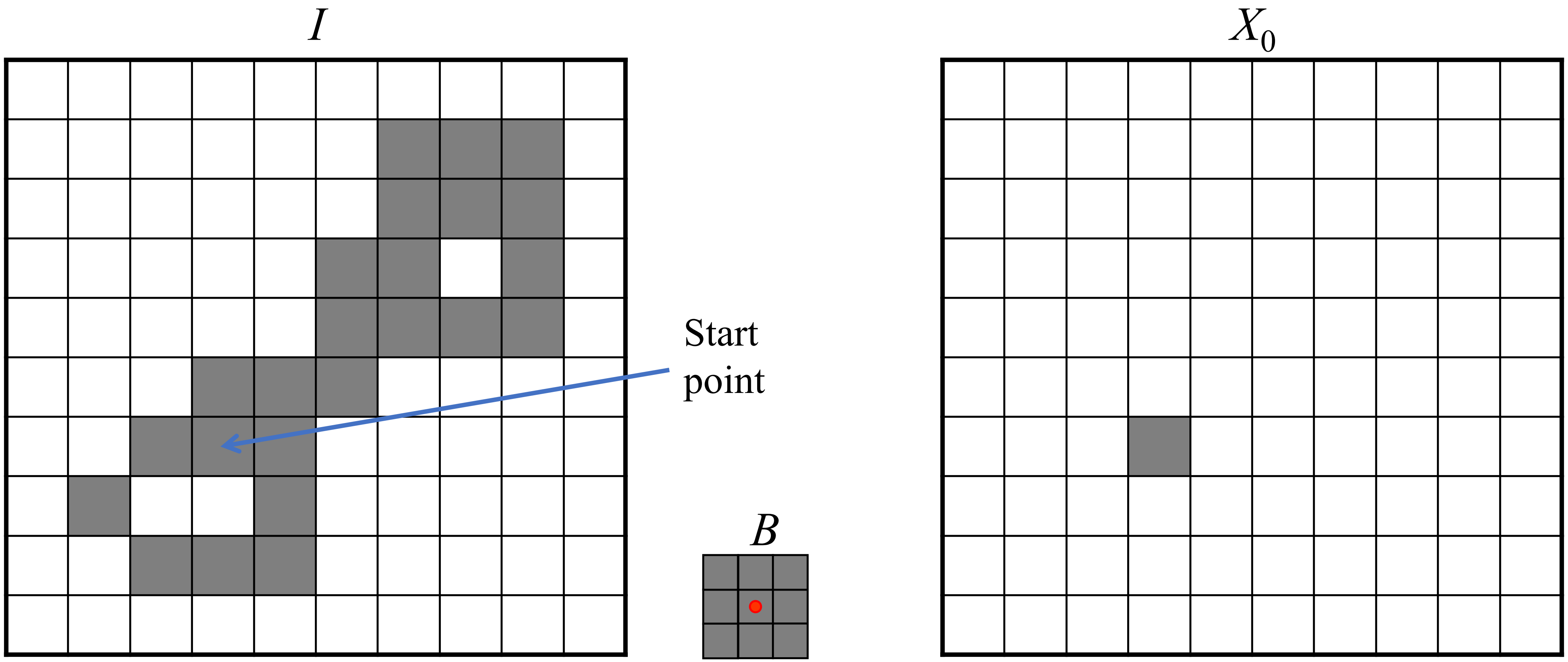
- The following iterative procedure accomplishes this

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

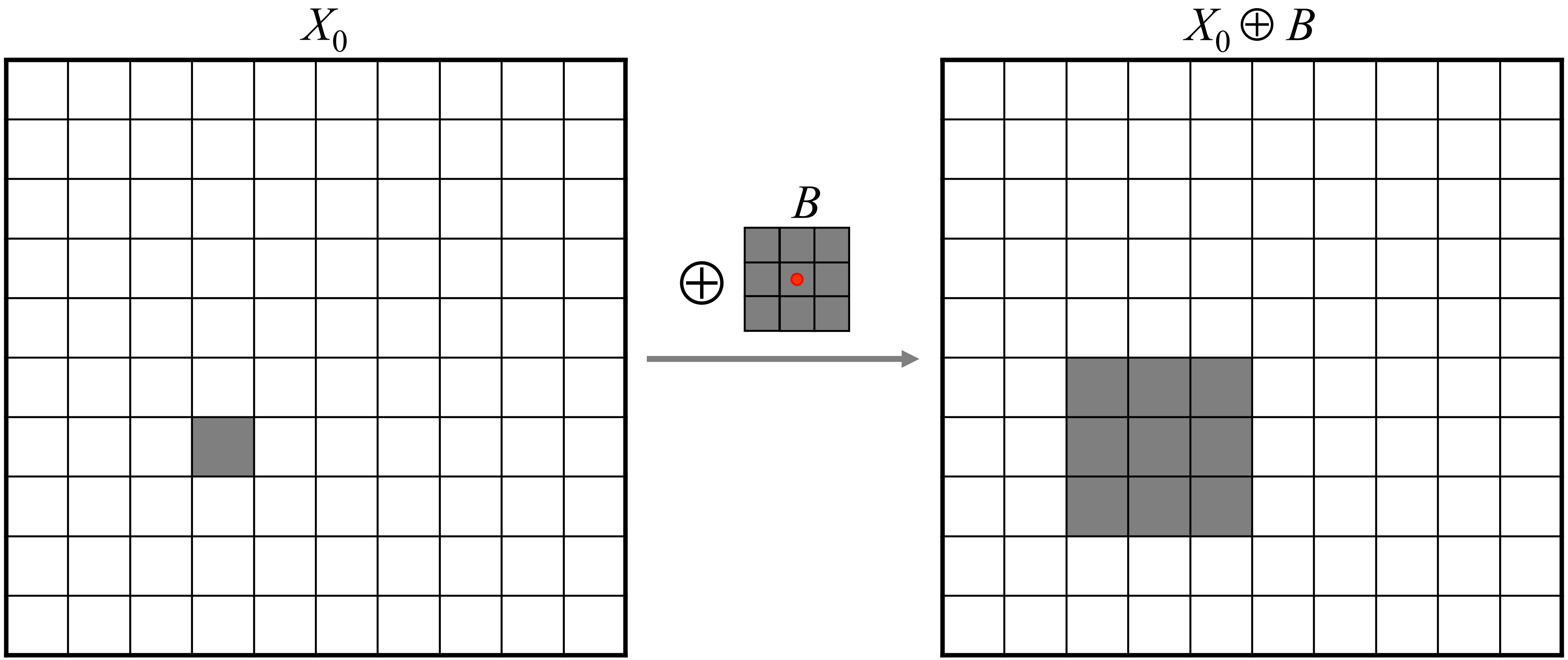
- When $X_k = X_{k-1}$, the algorithm converges.

$$Y = X_k$$

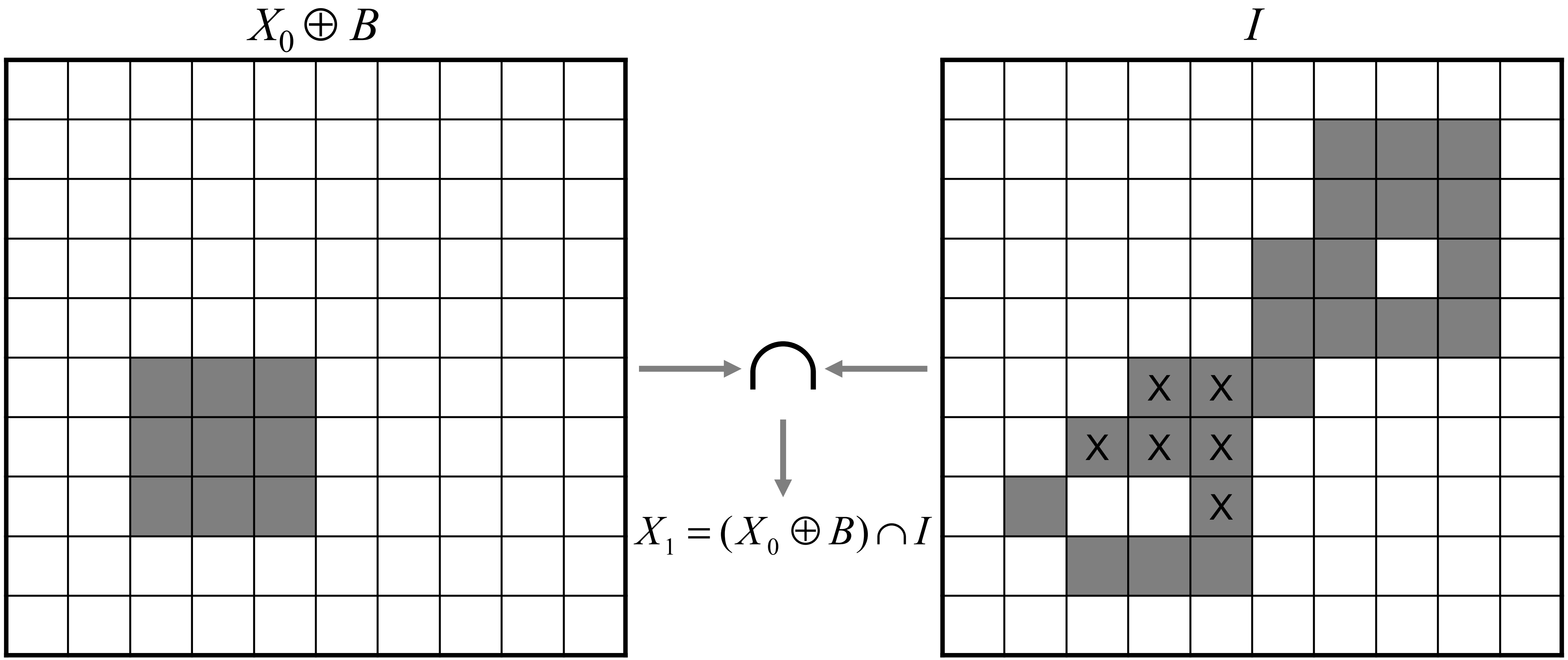
Extraction of Connected Components



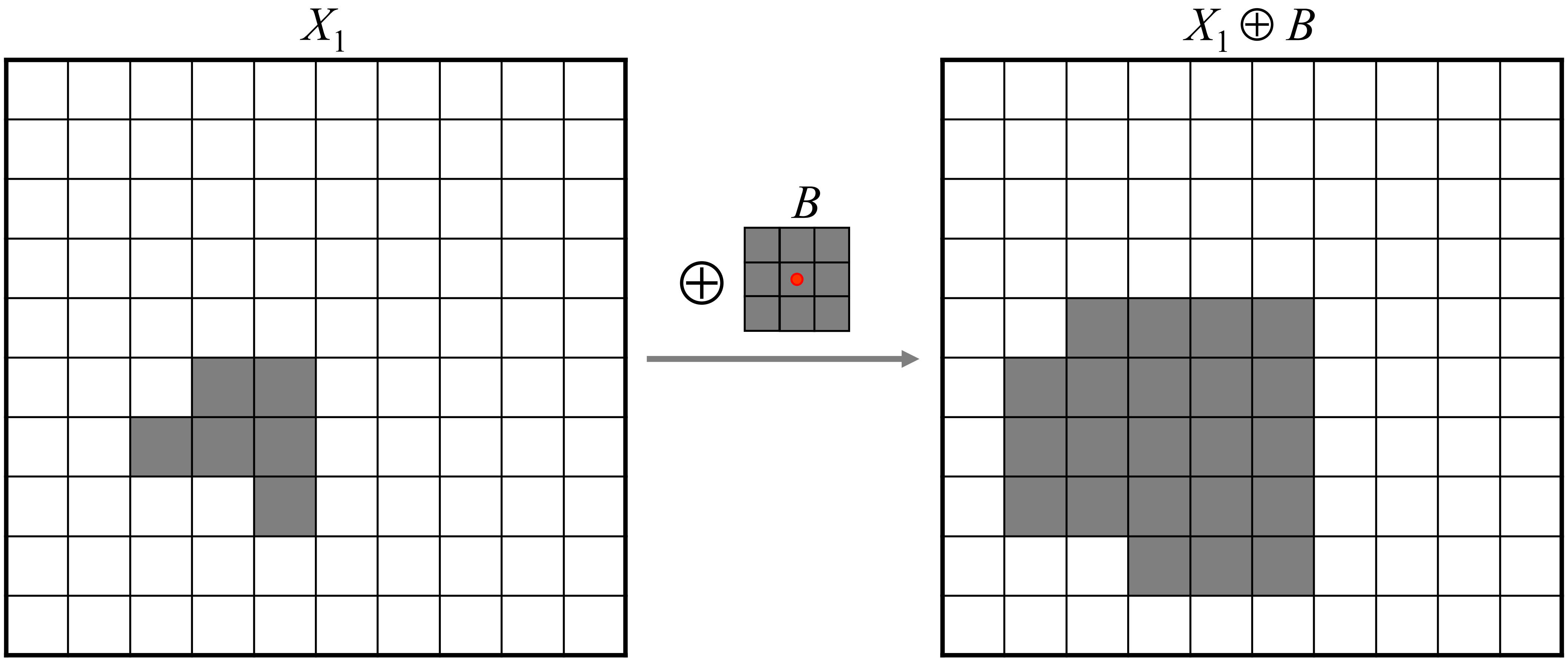
Extraction of Connected Components



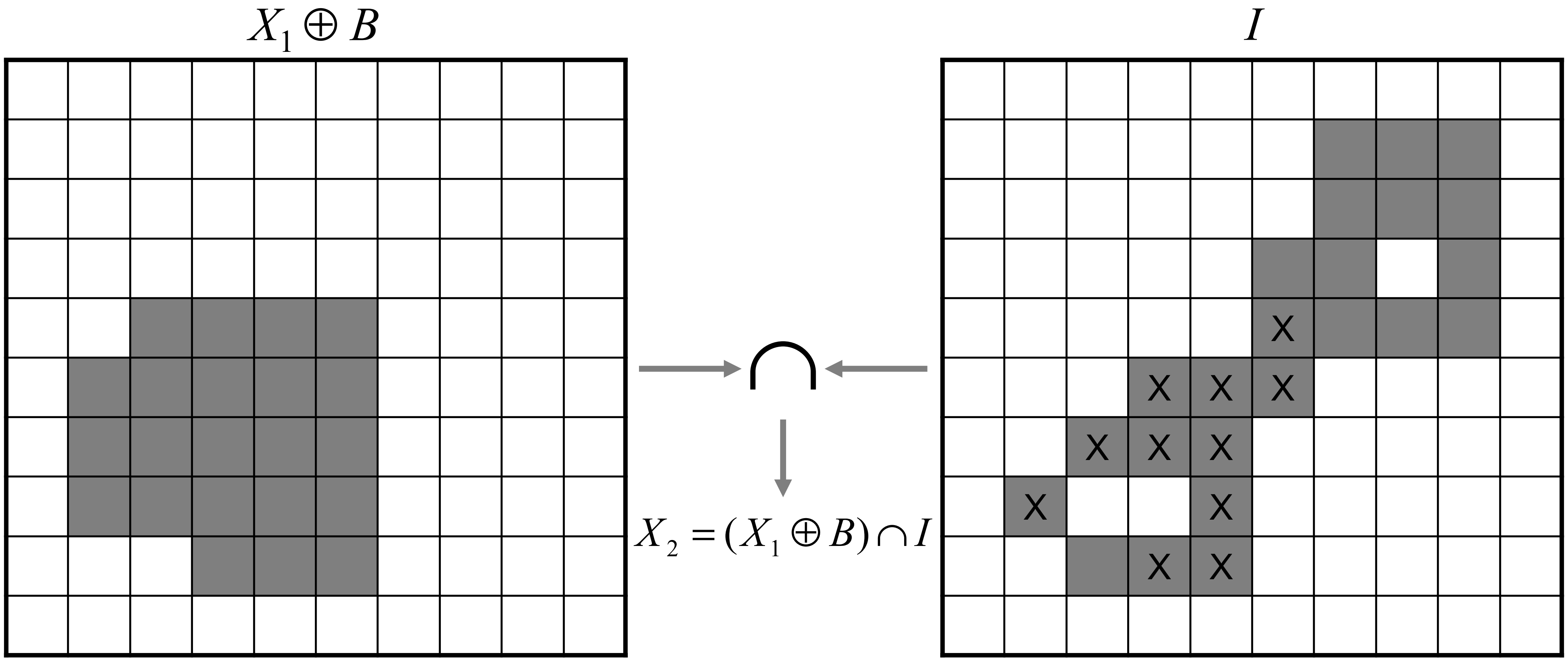
Extraction of Connected Components



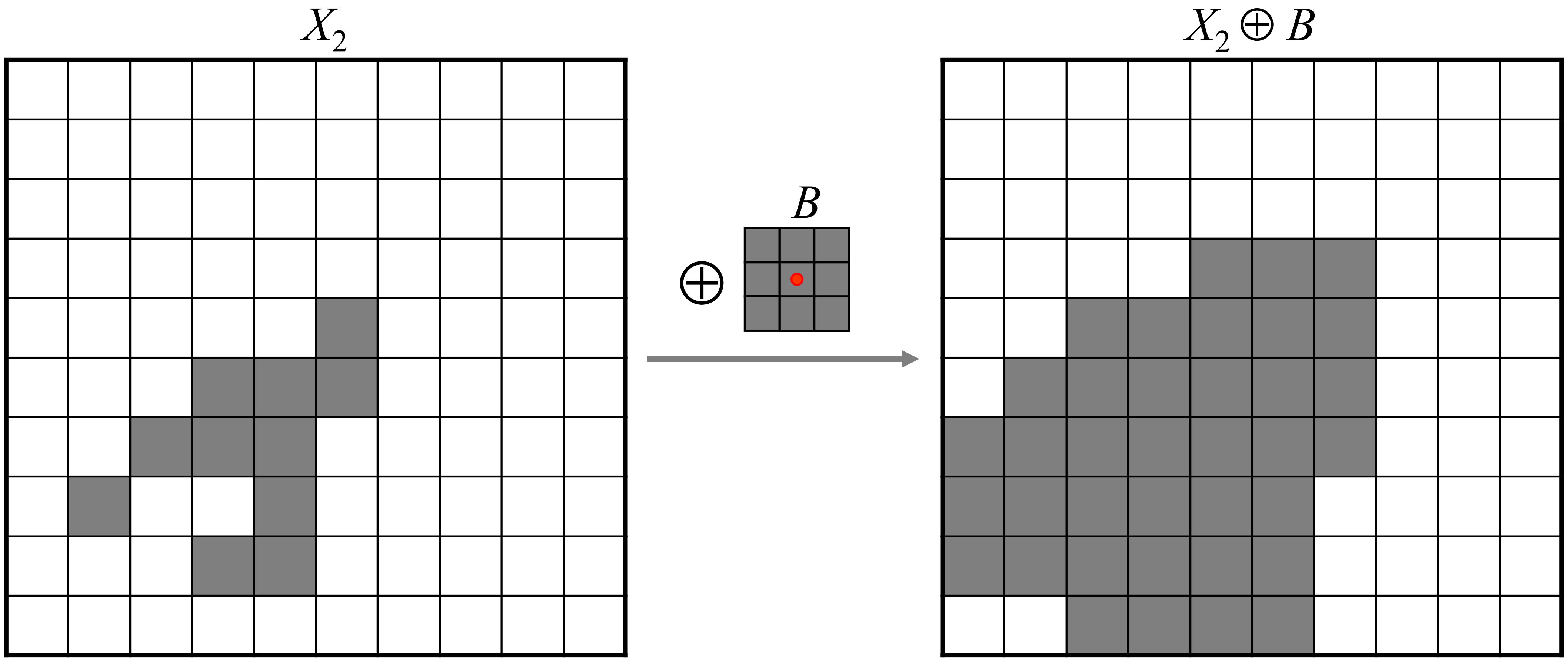
Extraction of Connected Components



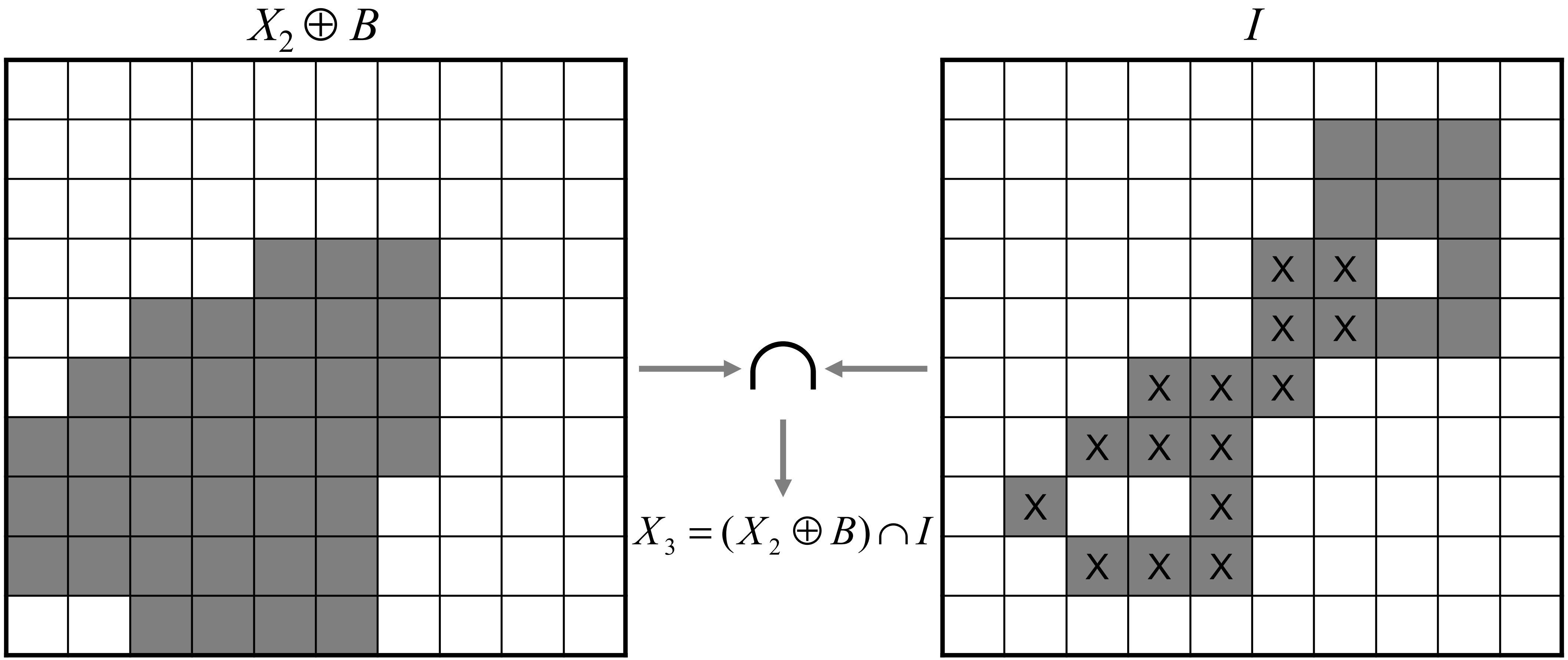
Extraction of Connected Components



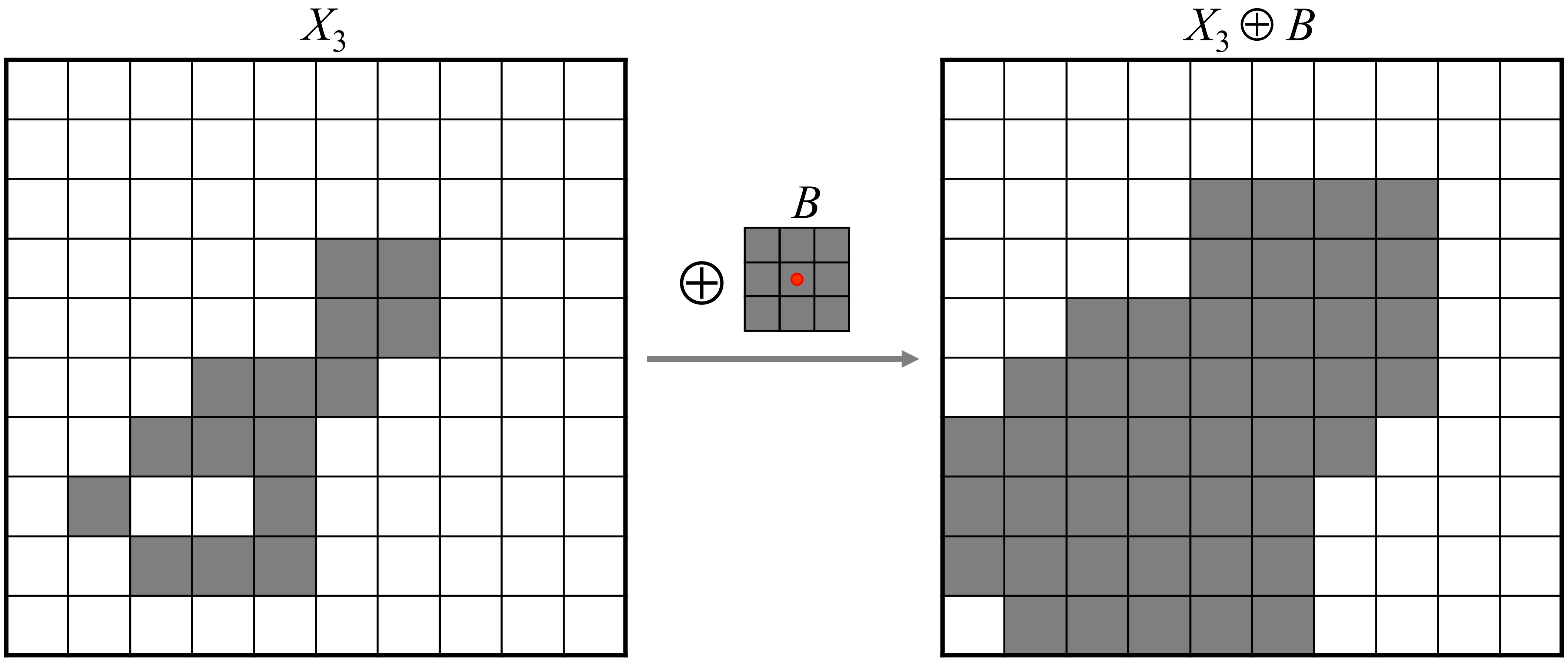
Extraction of Connected Components



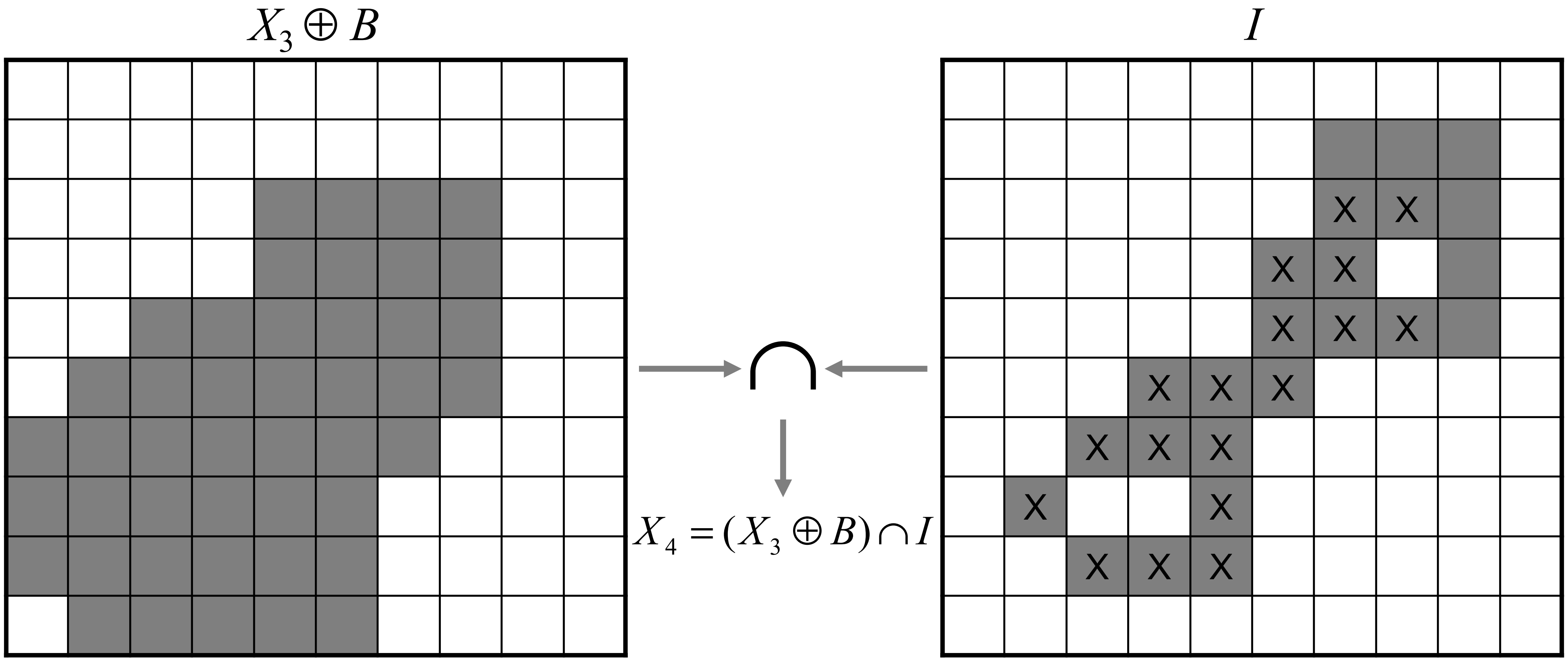
Extraction of Connected Components



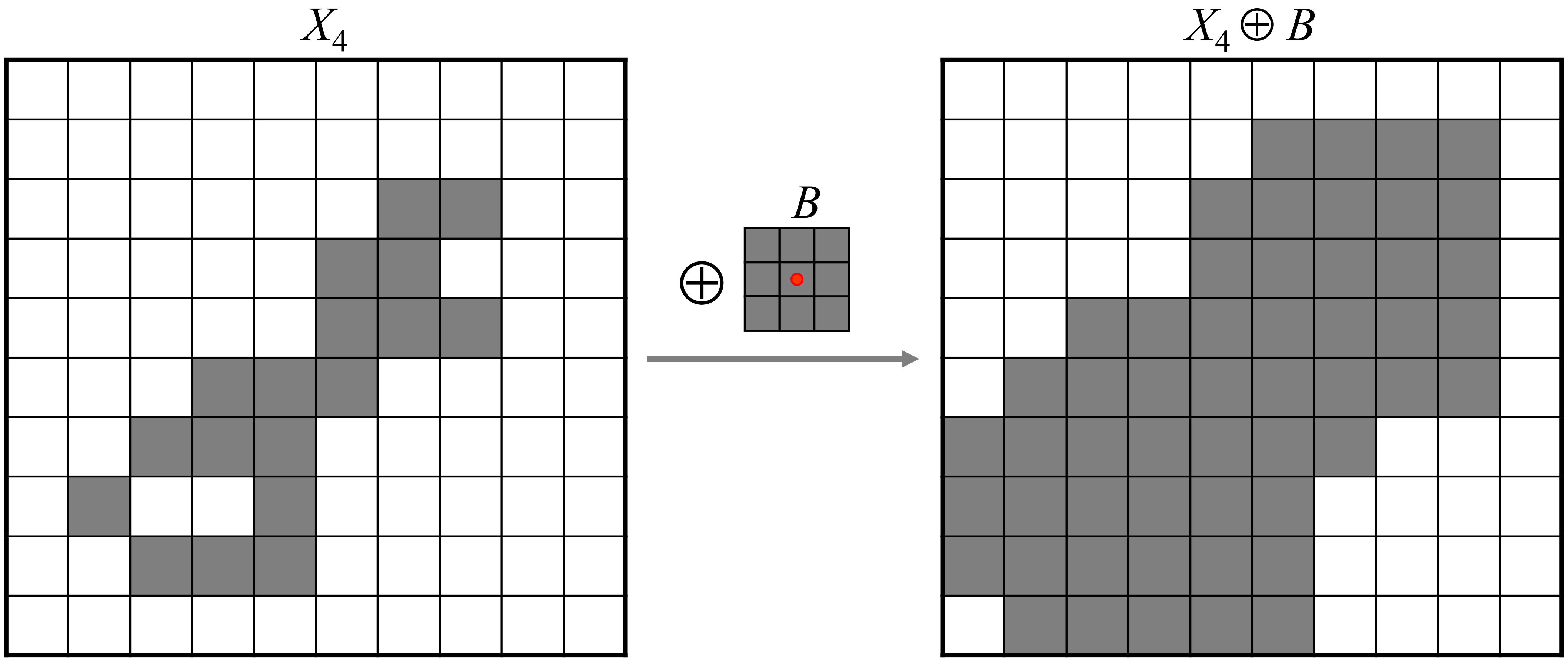
Extraction of Connected Components



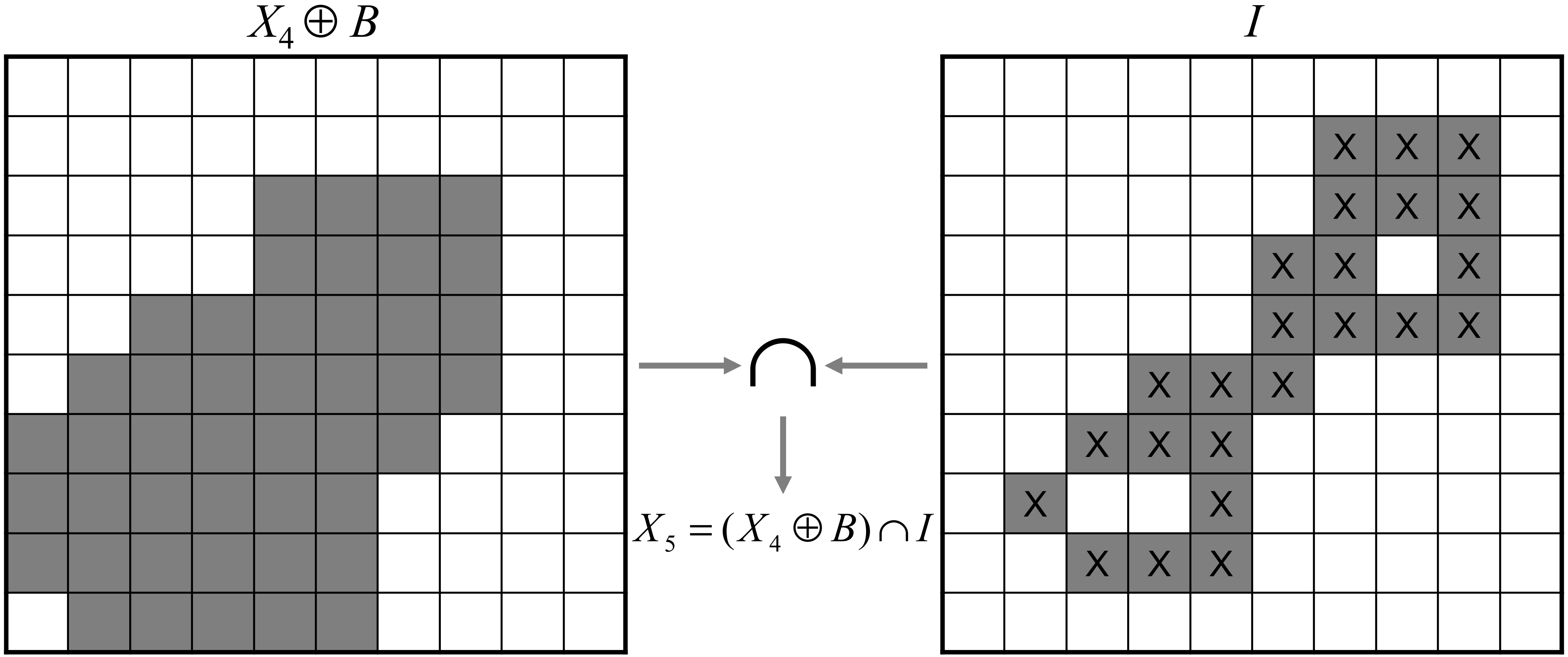
Extraction of Connected Components



Extraction of Connected Components



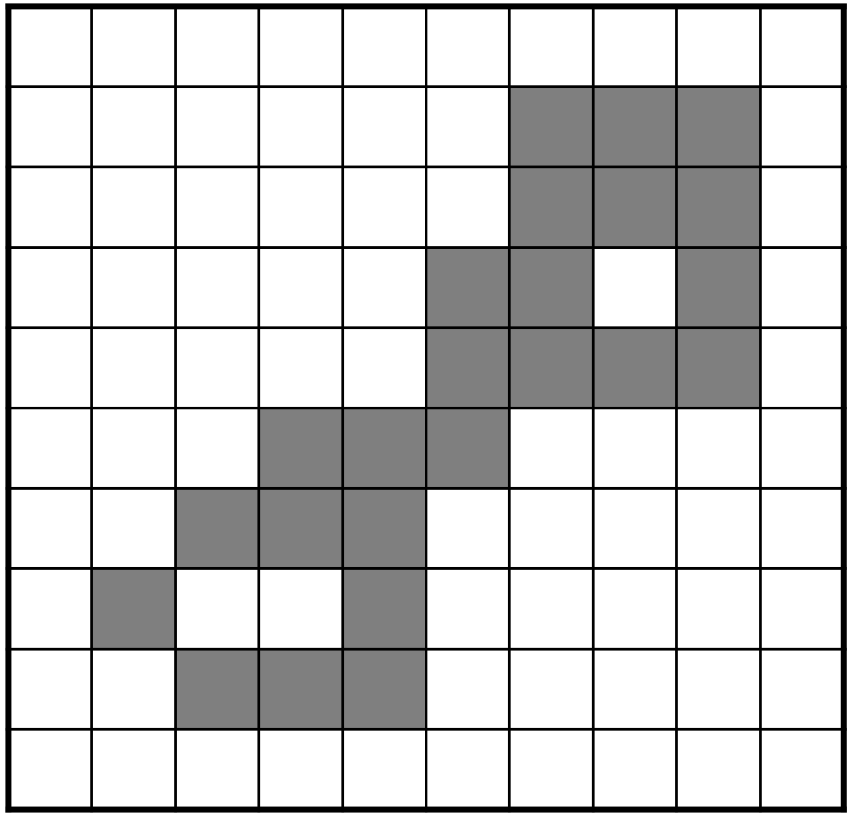
Extraction of Connected Components



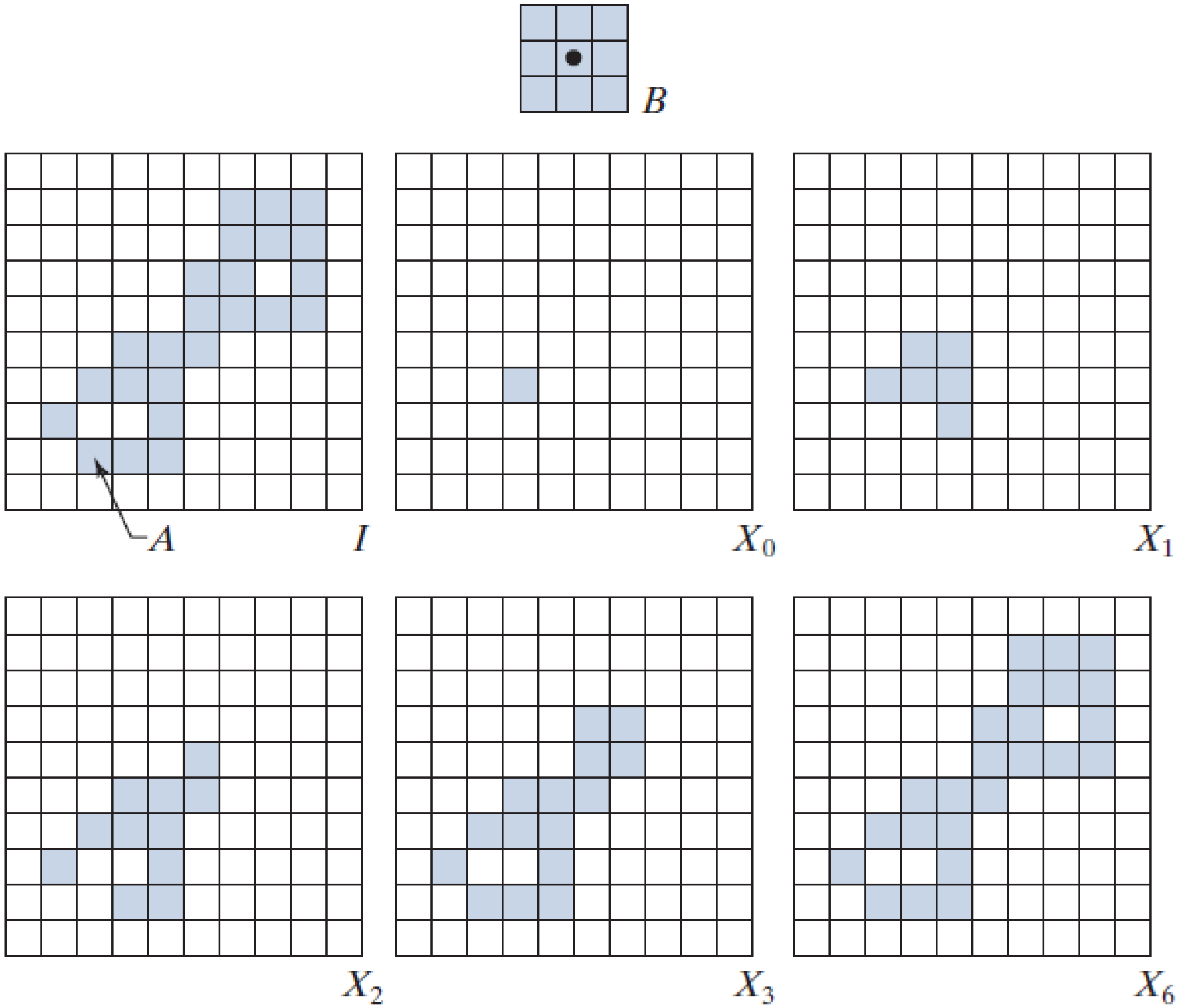
$$X_5 = (X_4 \oplus B) \cap I$$

Extraction of Connected Components

$$X_5 = X_6 = I$$

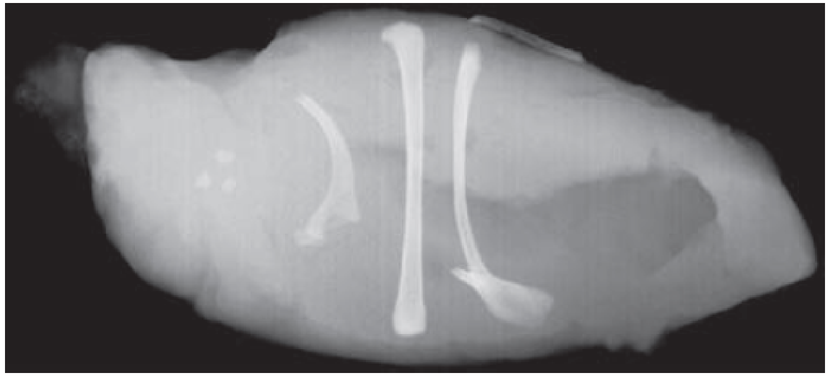


Extraction of Connected Components



Extraction of Connected Components

(a) X-ray image of a chicken filet with bone fragments.



(b) Thresholded image (shown as the negative for clarity).

(c) Image eroded with a 5×5 SE of 1's.

(d)
Number of pixels in the connected components of (c).

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

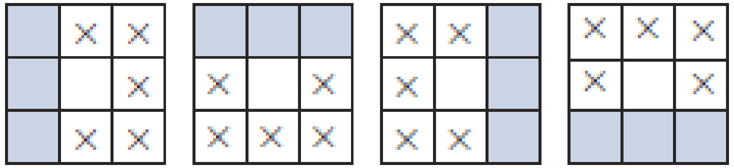
Convex Hull

- A set, *S*, of points in the Euclidean plane is said to be *convex* if and only if a straight line segment joining any two points in *S* lies entirely within *S*.
- The *convex hull*, *H*, of *S* is the smallest convex set containing *S*.
- Let B^i, $i = 1, 2, 3, 4$, denote the four structuring elements below. The procedure consists of implementing the morphological equation

$$X_k^i = (X_{k-1}^i \circledast B^i) \cup X_{k-1}^i \qquad i = 1,2,3,4 \text{ and } k = 1,2,3,\ldots$$

with $X_0^i = I$.

Hit-or-Miss Transform



B^1 B^2 B^3 B^4

Convex Hull

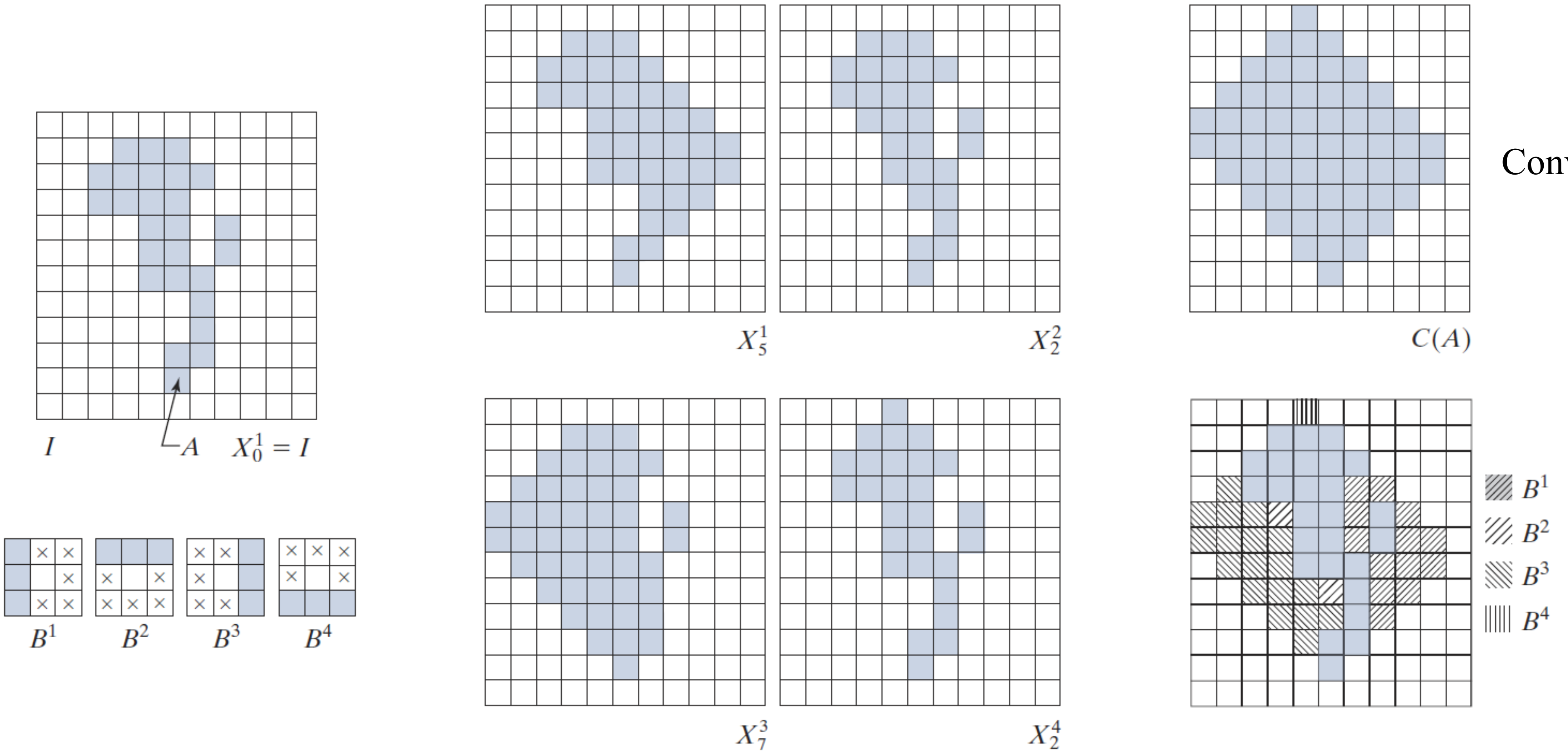
- When the procedure converges using the *i*-th structuring element (i.e., when $X_k^i = X_k^{i-1}$, we let $D^i = X_k^i$.

- Then, the convex hull of A is the union of the four results:

$$C(A) = \bigcup_{i=1}^{4} D^i$$

- Thus, the method consists of iteratively applying the hit-or-miss transform to I with B_1 until convergence, and so on. The union of the four resulting D^i constitutes the convex hull of A.

Convex Hull



Convex hull

Convex hull showing the contribution of each structuring element.

Results of convergence with the structuring elements B^i

Thinning

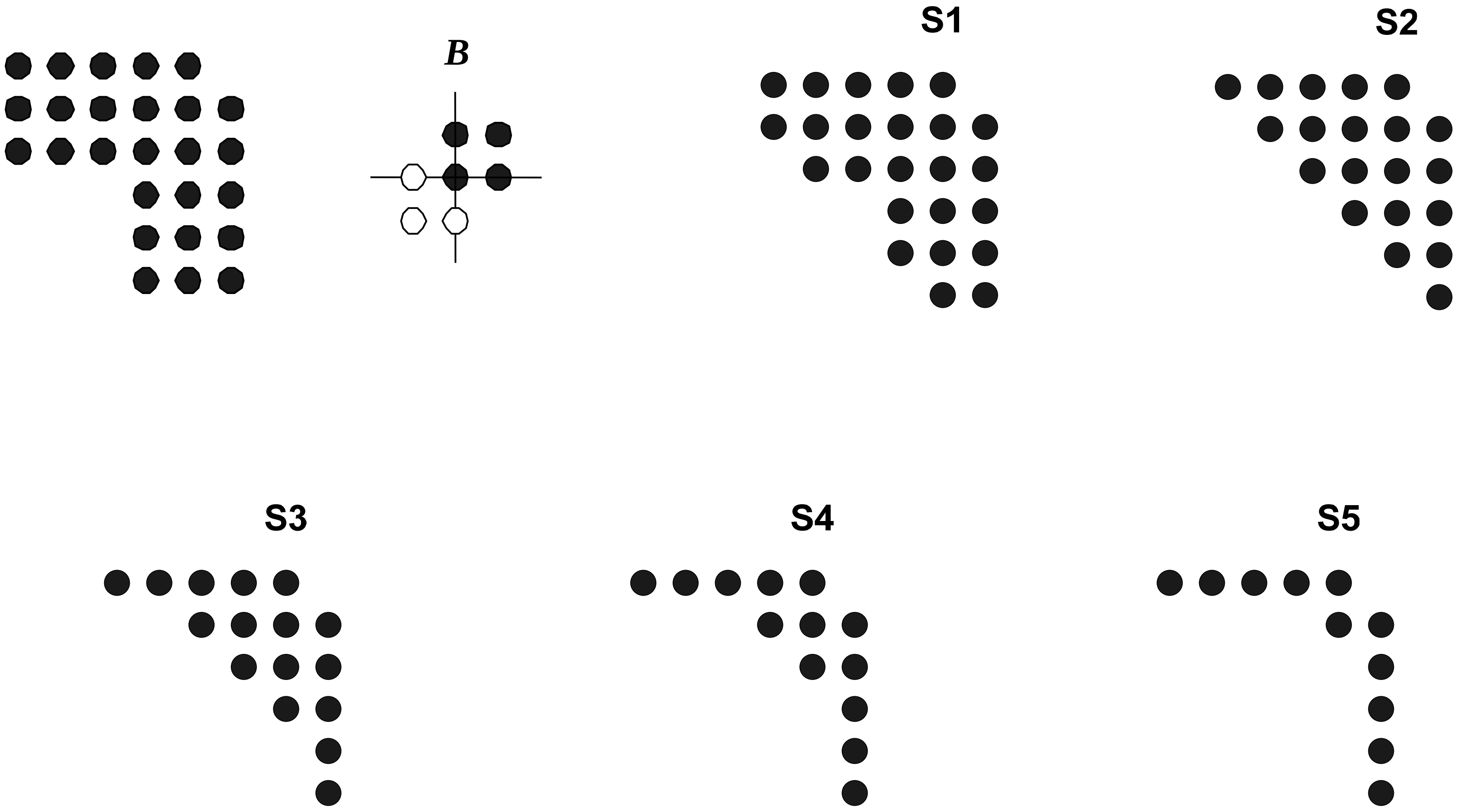
- Thinning can be defined in terms of the hit-or-miss transform, denote by:

$$A \otimes B = A - (A \circledast B) = A \cap (A \circledast B)^c$$

hit-or-miss transform: $I \circledast B_{1,2} = (A \ominus B_1) \cap (A^c \ominus B_2)$

- The process is iterative with B until no further changes occur.

Thinning



Thinning

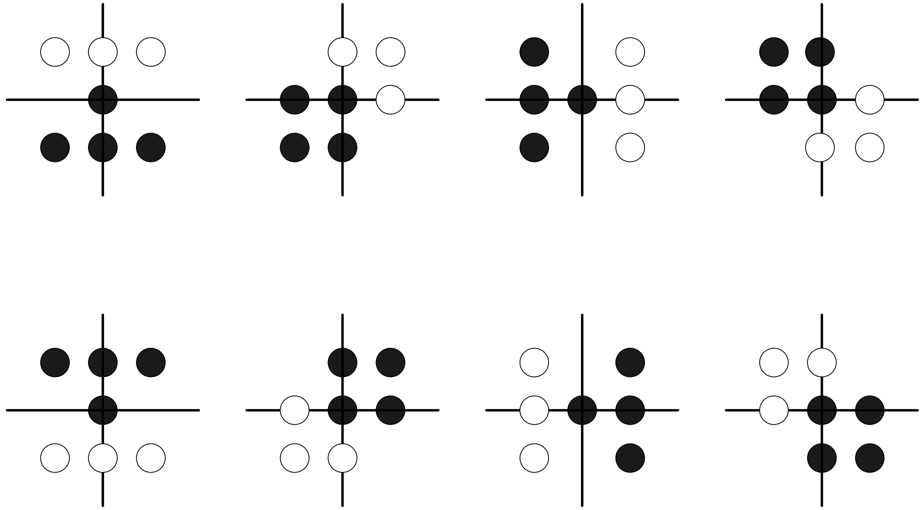
- Thinning can be defined in terms of many structuring elements, denote by

$$A \otimes \{B\} = \left(\left(\ldots\left(\left(A \otimes B^1\right) \otimes B^2\right)\ldots\right) \otimes B^n\right), \quad B = \{B^1, B^2, \ldots, B^n\}$$

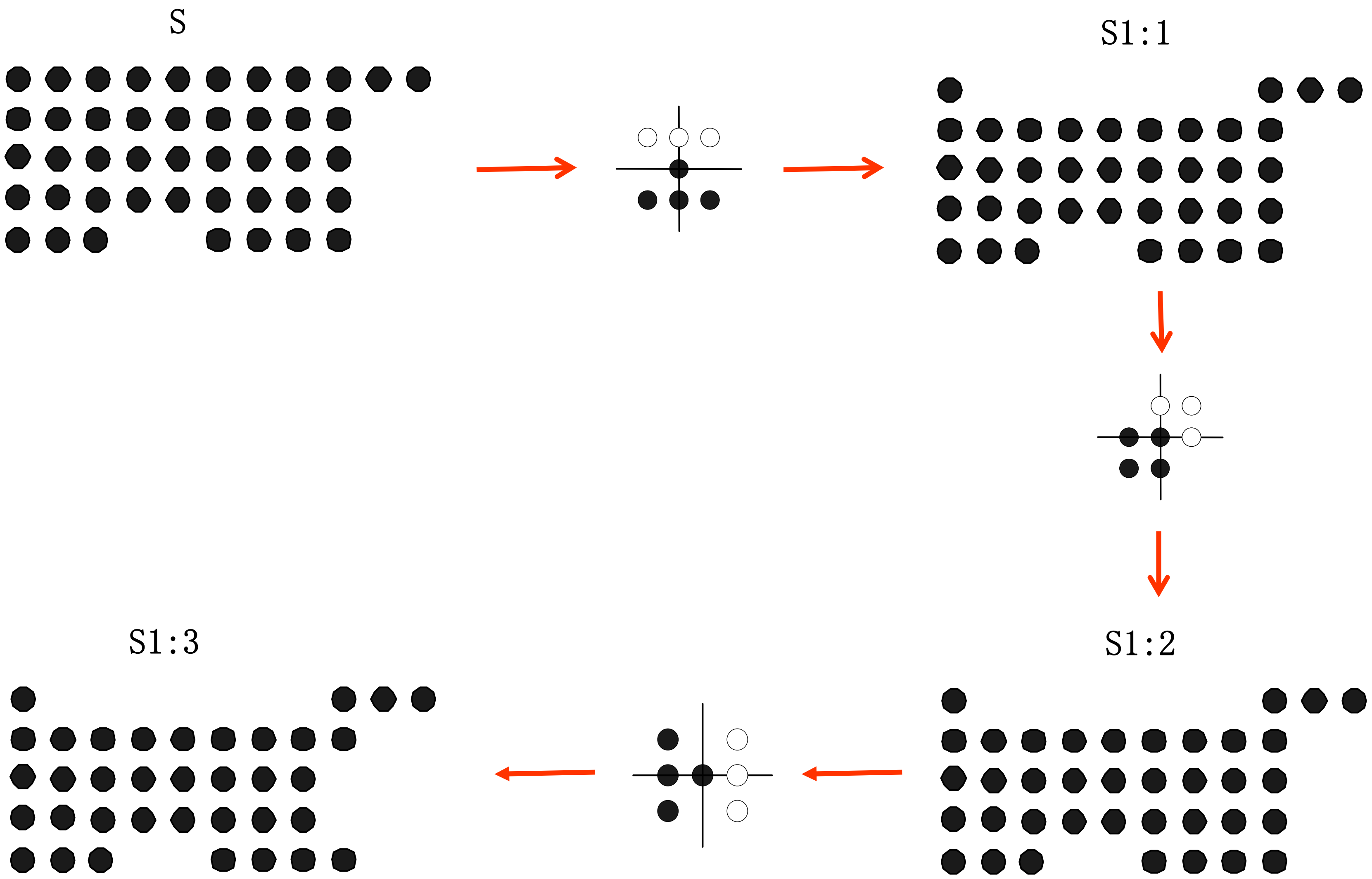
B^i can be a rotation version of B^{i-1}.

- The entire process is repeated until no further changes occur.

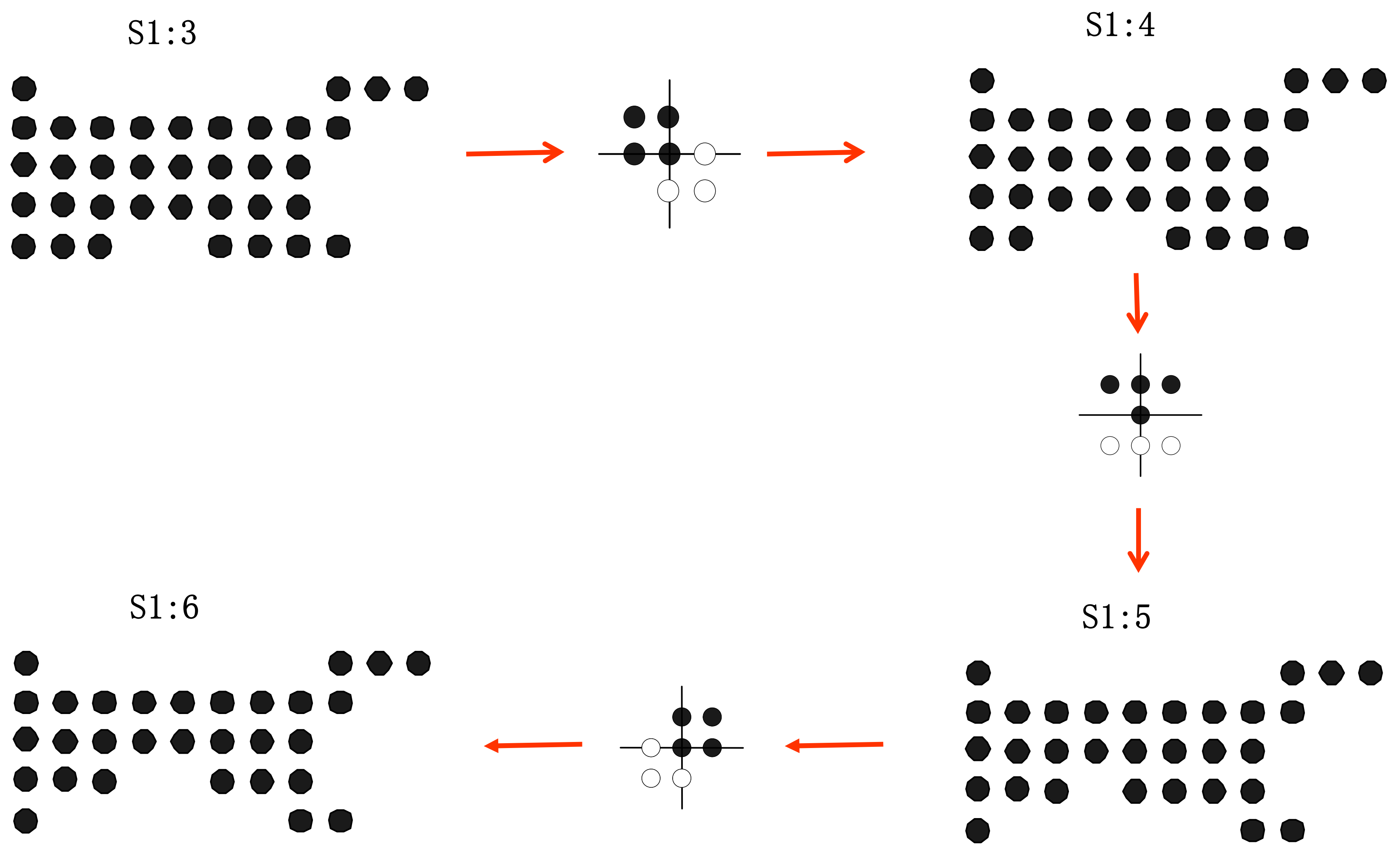
- Usually 8-direction structure elements are used for thinning.



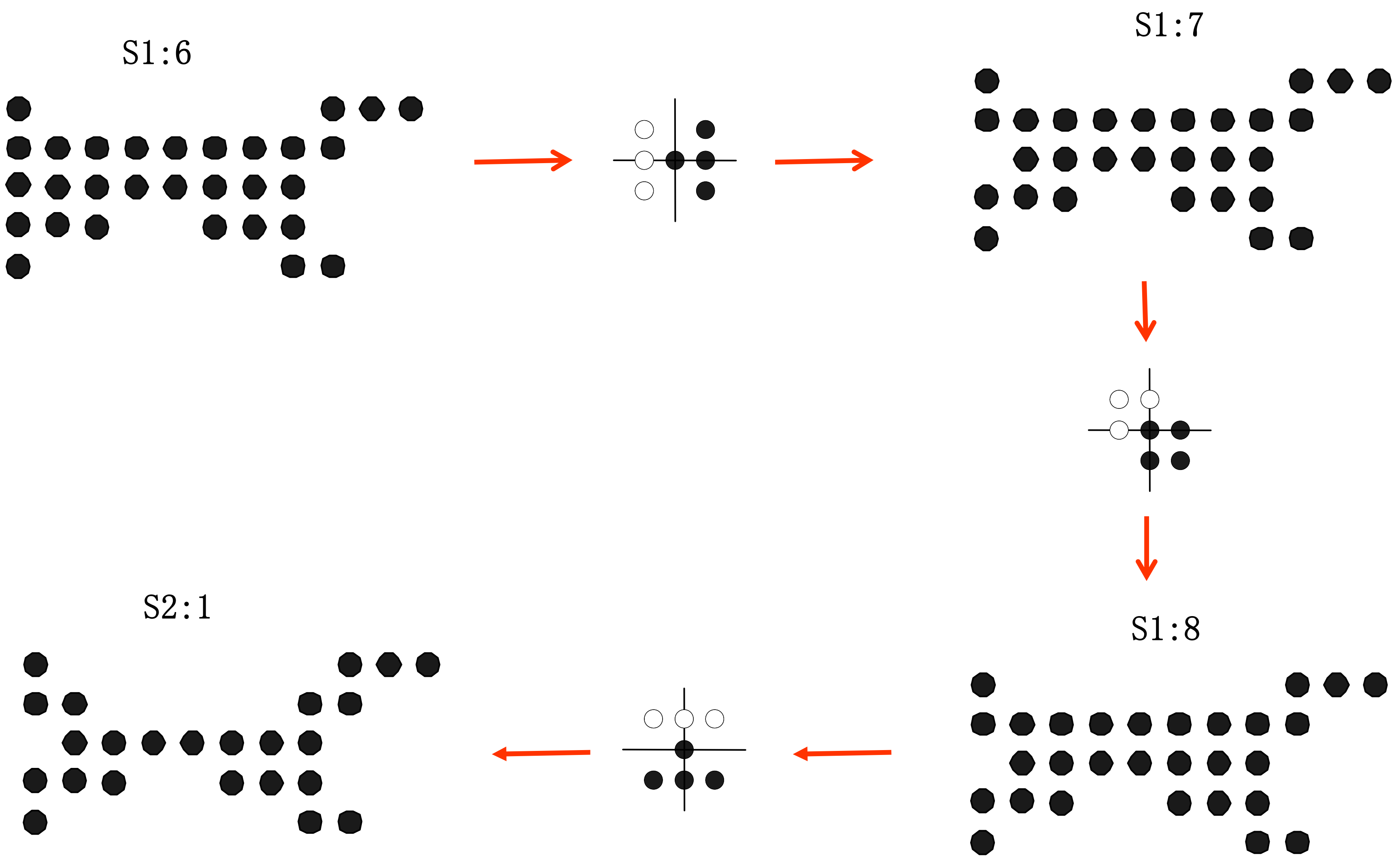
Thinning



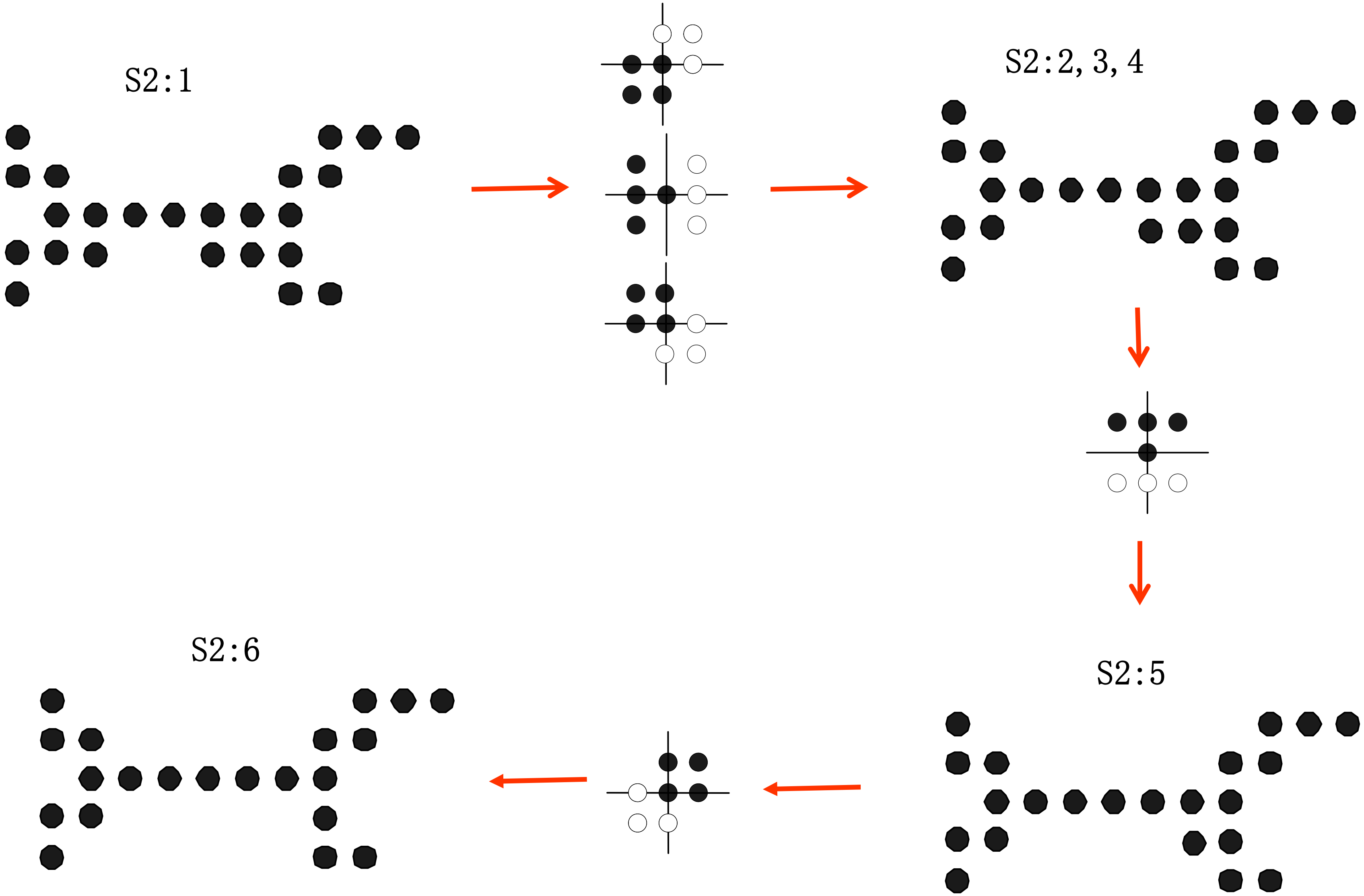
Thinning



Thinning

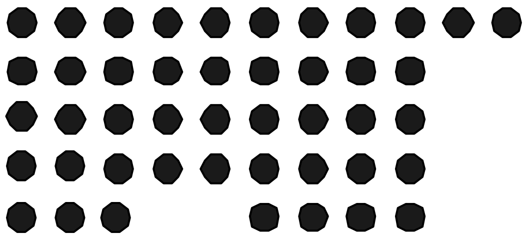


Thinning



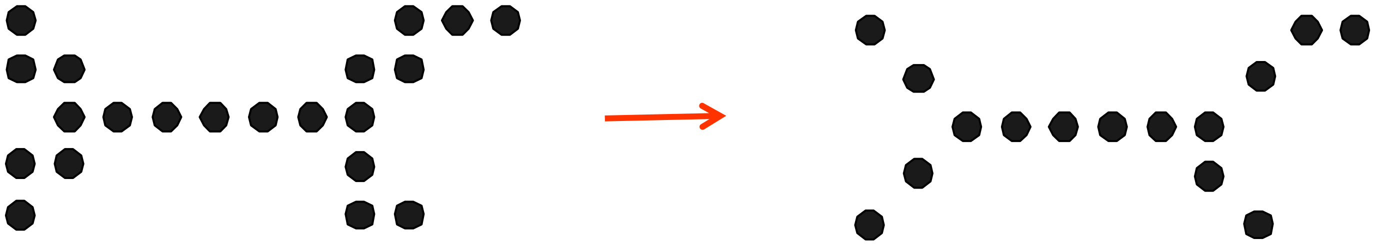
Thinning

S

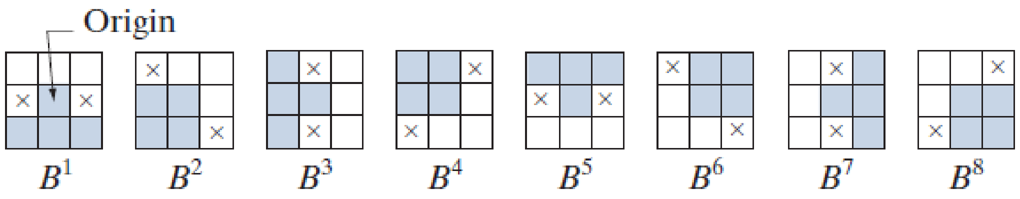


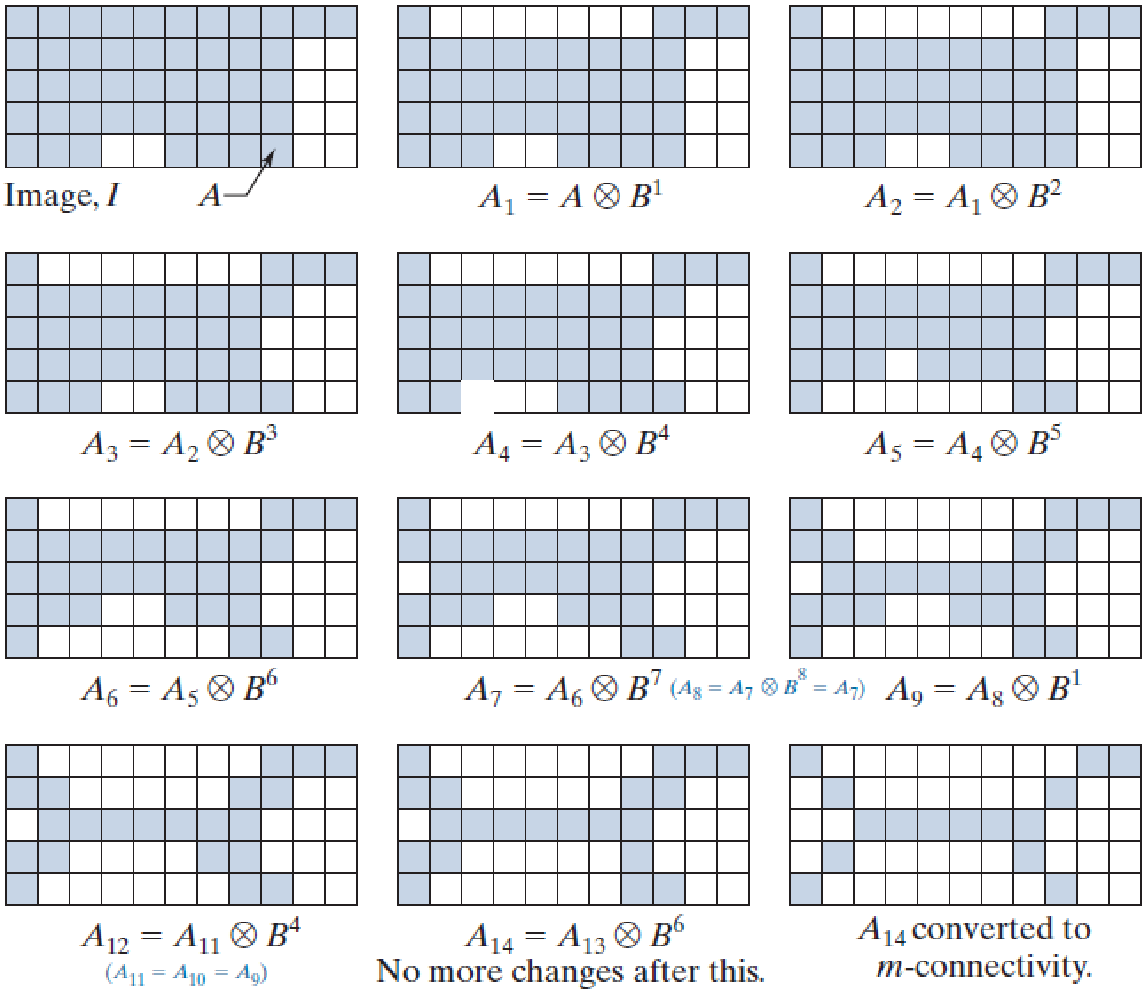
S2:6

m connection



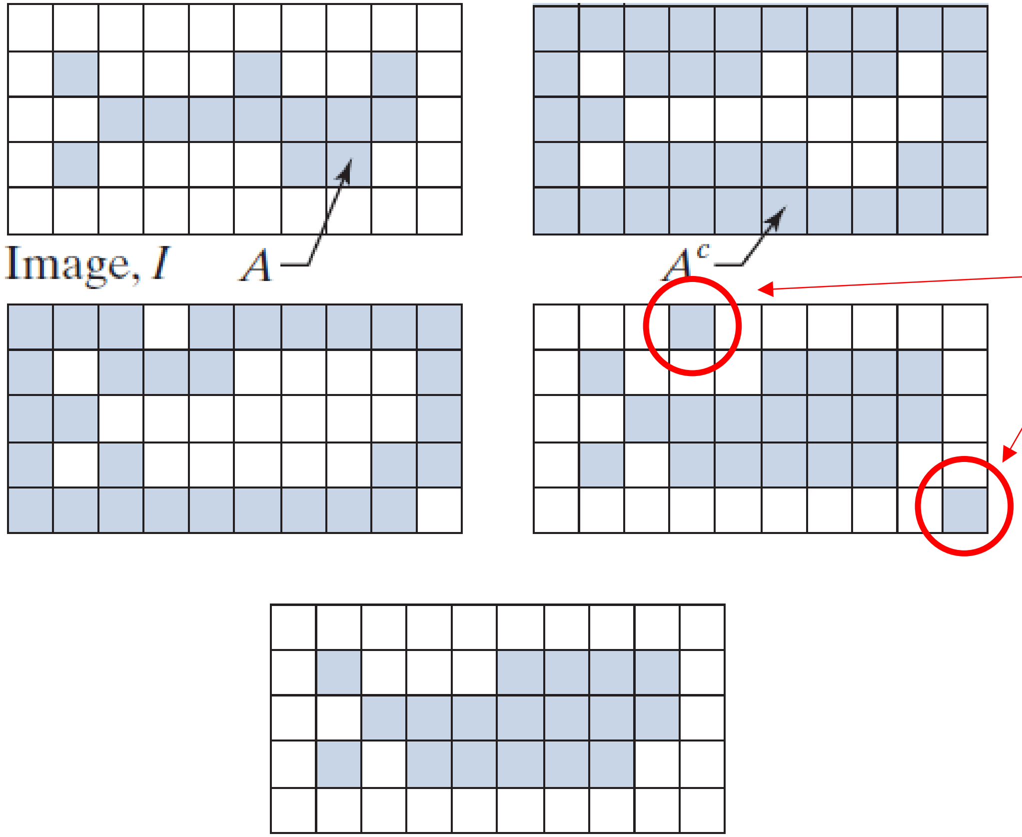
Thinning


Origin
B^1
B^2
B^3
B^4
B^5
B^6
B^7
B^8


Image, I
A
$A_1 = A \otimes B^1$
$A_2 = A_1 \otimes B^2$
$A_3 = A_2 \otimes B^3$
$A_4 = A_3 \otimes B^4$
$A_5 = A_4 \otimes B^5$
$A_6 = A_5 \otimes B^6$
$A_7 = A_6 \otimes B^7$ $(A_8 = A_7 \otimes B^8 = A_7)$
$A_9 = A_8 \otimes B^1$
$A_{12} = A_{11} \otimes B^4$
$(A_{11} = A_{10} = A_9)$
$A_{14} = A_{13} \otimes B^6$
No more changes after this.
A_{14} converted to
m-connectivity.

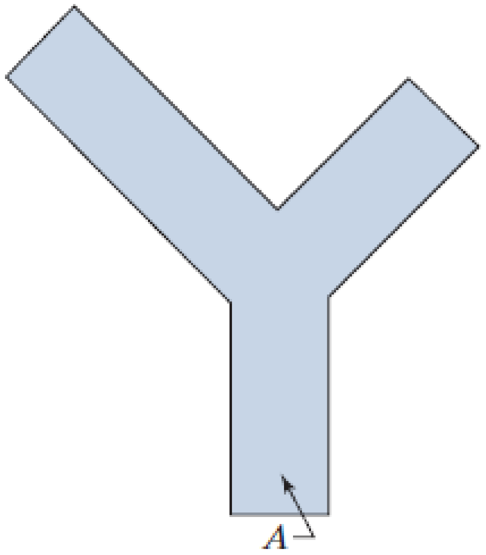
Thickening

- Thickening is seldom used in practice. The usual procedure is to thin the background, and then complement the result.
- Give A, get $C = A^c$, thin C, and then get C^c.

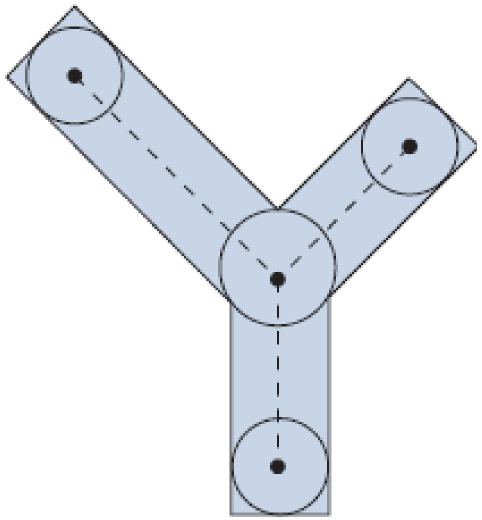


This procedure may result in some disconnected points, so it is usually followed by a simple postprocessing step.

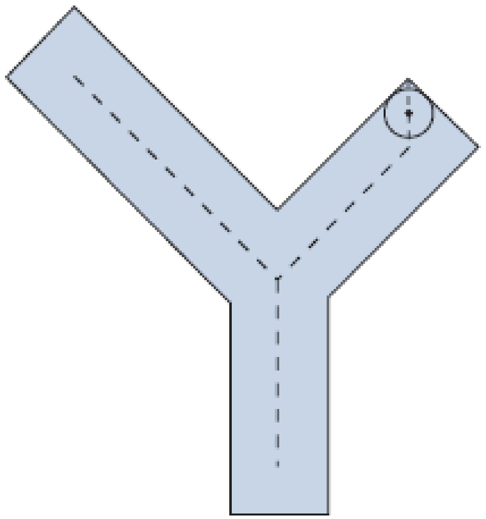
Skeletons



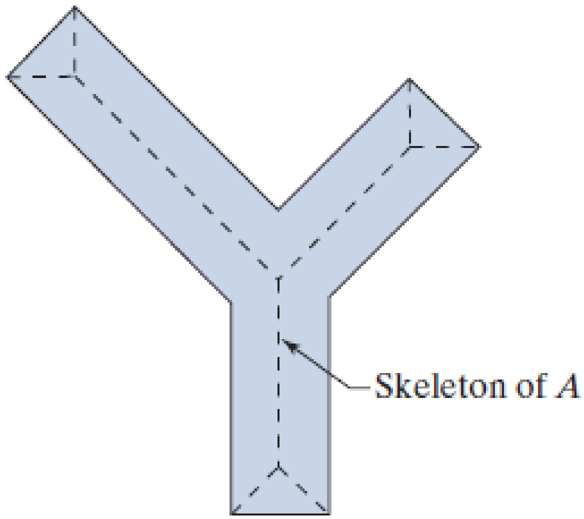
Set *A*



Various positions of maximum disks whose centers partially define the skeleton of *A*.



Another maximum disk, whose center defines a different segment of the skeleton of *A*.



Complete skeleton (dashed).

Skeletons

- Skeletons are defined in terms of erosions and openings, and denoted by:

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

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

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

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

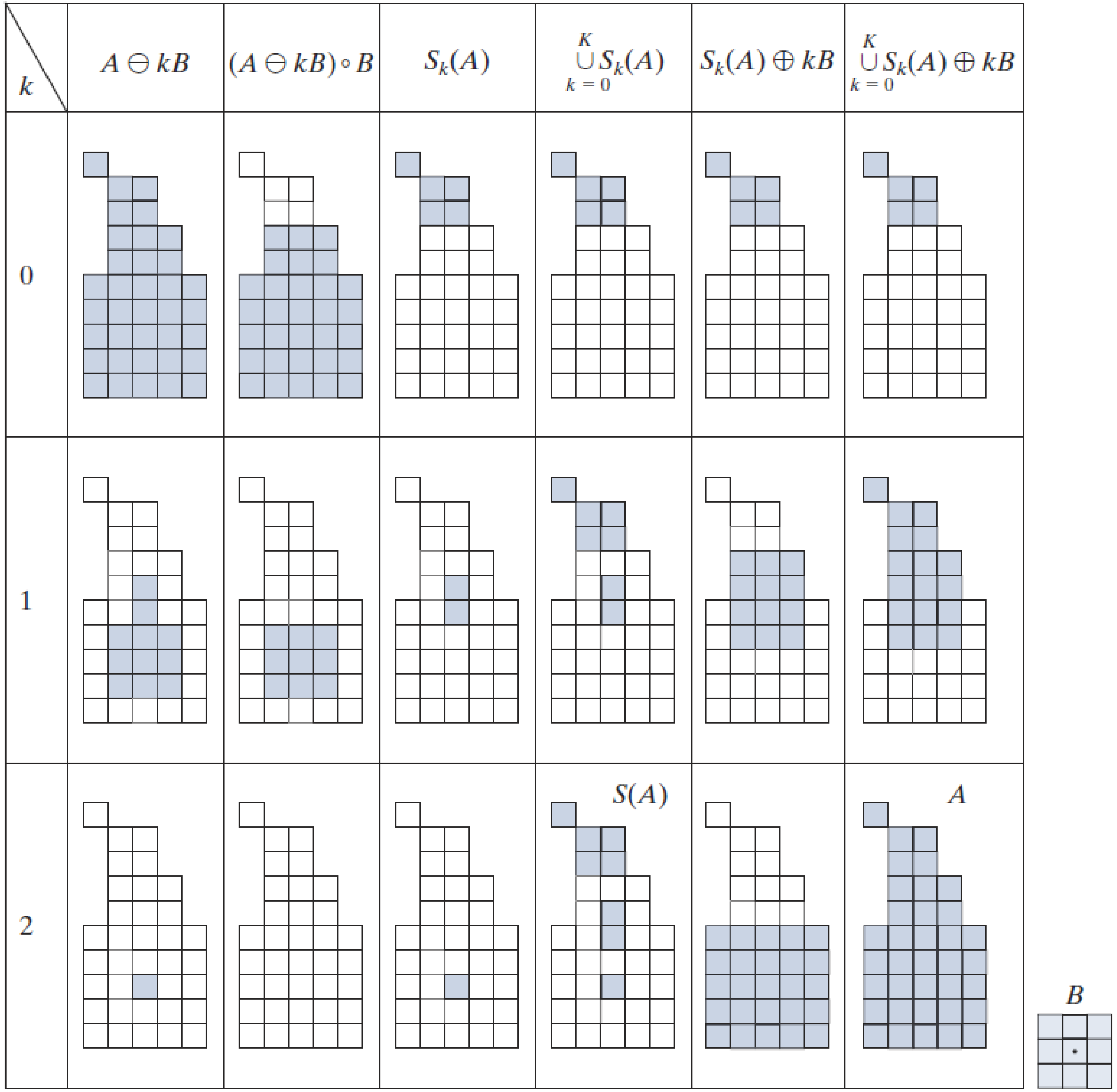
$$K = \max\{k \mid (A \ominus kB) \neq \varnothing\}$$

- Also, A can be reconstructed from these subsets:

$$A = \bigcup_{k=0}^{K} (S_k(A) \oplus kB) \qquad (S_k(A) \oplus kB) = ((\ldots(S_k(A) \oplus B) \oplus B) \oplus \ldots) \oplus B$$

Skeletons

The original set is at the top left, and its morphological skeleton is at the bottom of the fourth column. The reconstructed set is at the bottom of the sixth column.



Skeletons

Binary fingerprint image

Skeleton image

Grayscale Morphology

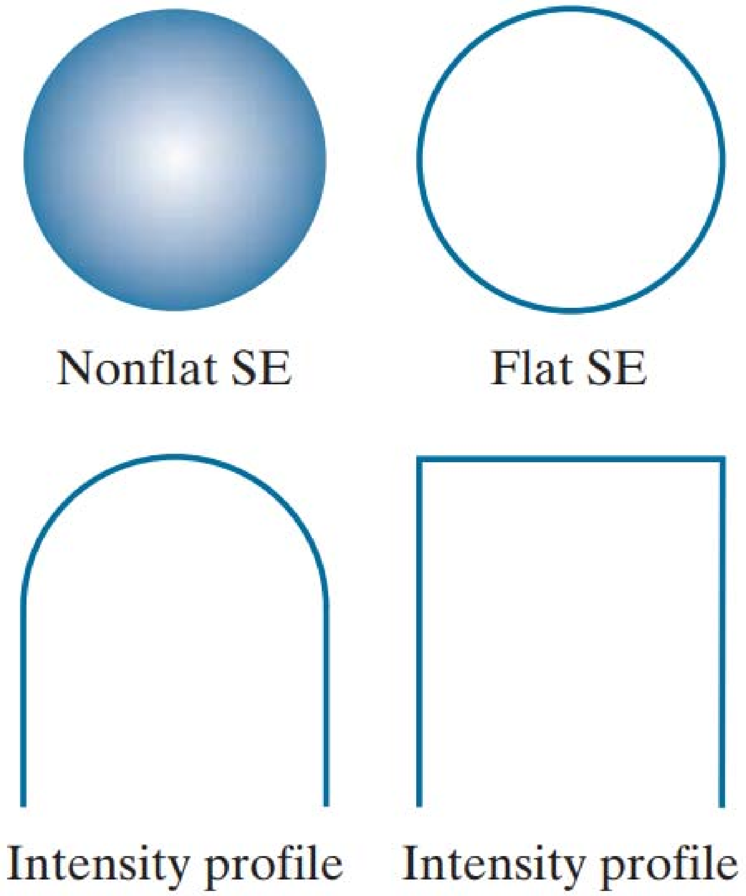
- In this lecture:
  - We firstly extend to grayscale images the basic operations of dilation, erosion, opening, and closing.
  - We then use these operations to develop several basic grayscale morphological algorithms.

Grayscale Morphology

- We deal with digital functions $f(x, y)$ and $b(x, y)$, where $f(x, y)$ is a grayscale image and $b(x, y)$ is a structuring element.
- In grayscale morphology, structuring elements are used as “probes” to examine a given image for specific properties.
- As in the binary case, the origin of grayscale structuring elements must be clearly identified.
- The reflection of an SE in grayscale morphology:

$$\hat{b}(x, y) = b(-x, -y)$$

Grayscale Morphology



Two categories of SE: nonflat and flat.

Grayscale Erosion and Dilation

- The grayscale erosion of f by a flat structuring element b at location (x, y) is defined as the minimum value of the image in the region coincident with $b(x, y)$, when the origin of b is at (x, y) .
- In equation form:

$$[f \ominus b](x, y) = \min_{(s,t)\in b}\{f(x+s, y+t)\}$$

- For example, if b is a square structuring element of size 3×3, obtaining the erosion at a point requires finding the minimum of the nine values of f contained in the 3×3 region spanned by b when its origin is at that point.

Grayscale Erosion and Dilation

25	41	37	168	194
72	249	108	10	191
141	246	235	218	101
246	125	203	240	168
248	205	246	174	44

Gray image f (5×5)

Find the minimum value of the image in the neighbor region.

1	1	1
1	1	1
1	1	1

SE b 3×3

25	25	10	10	10
25	25	10	10	10
72	72	10	10	10
125	125	125	44	44
125	125	125	44	44

Erosion

Grayscale Erosion and Dilation

25	41	37	168	194
72	249	108	10	191
141	246	235	218	101
246	125	203	240	168
248	205	246	174	44

Gray image f (5×5)

25	25	10	10	10
25	25	10	10	10
72	72	10	10	10
125	125	125	44	44
125	125	125	44	44

Erosion

Grayscale Erosion and Dilation

25	41	37	168	194
72	249	108	10	191
141	246	235	218	101
246	125	203	240	168
248	205	246	174	44

Gray image f (5×5)

25	25	10	10	10
25	25	10	10	10
72	72	10	10	10
125	125	125	44	44
125	125	125	44	44

Erosion

Grayscale Erosion and Dilation

25	41	37	168	194
72	249	108	10	191
141	246	235	218	101
246	125	203	240	168
248	205	246	174	44

Gray image f (5×5)

25	25	10	10	10
25	25	10	10	10
72	72	10	10	10
125	125	125	44	44
125	125	125	44	44

Erosion

Grayscale Erosion and Dilation

- Similarly, the grayscale dilation of f by a flat structuring element b at any location (x, y) is defined as the maximum value of the image in the window spanned by $\hat{b}$ when the origin of $\hat{b}$ is at (x, y).

- In equation form:

$$[f \oplus b](x, y) = \max_{(s,t)\in b}\{f(x-s, y-t)\}$$

Grayscale Erosion and Dilation

25	41	37	168	194
72	249	108	10	191
141	246	235	218	101
246	125	203	240	168
248	205	246	174	44

Gray image f (5×5)

Find the maximum value of the image in the neighbor region.

1	1	1
1	1	1
1	1	1

SE b 3×3

249	249	249	194	194
249	249	249	235	218
249	249	249	240	240
248	248	246	246	240
248	248	246	246	240

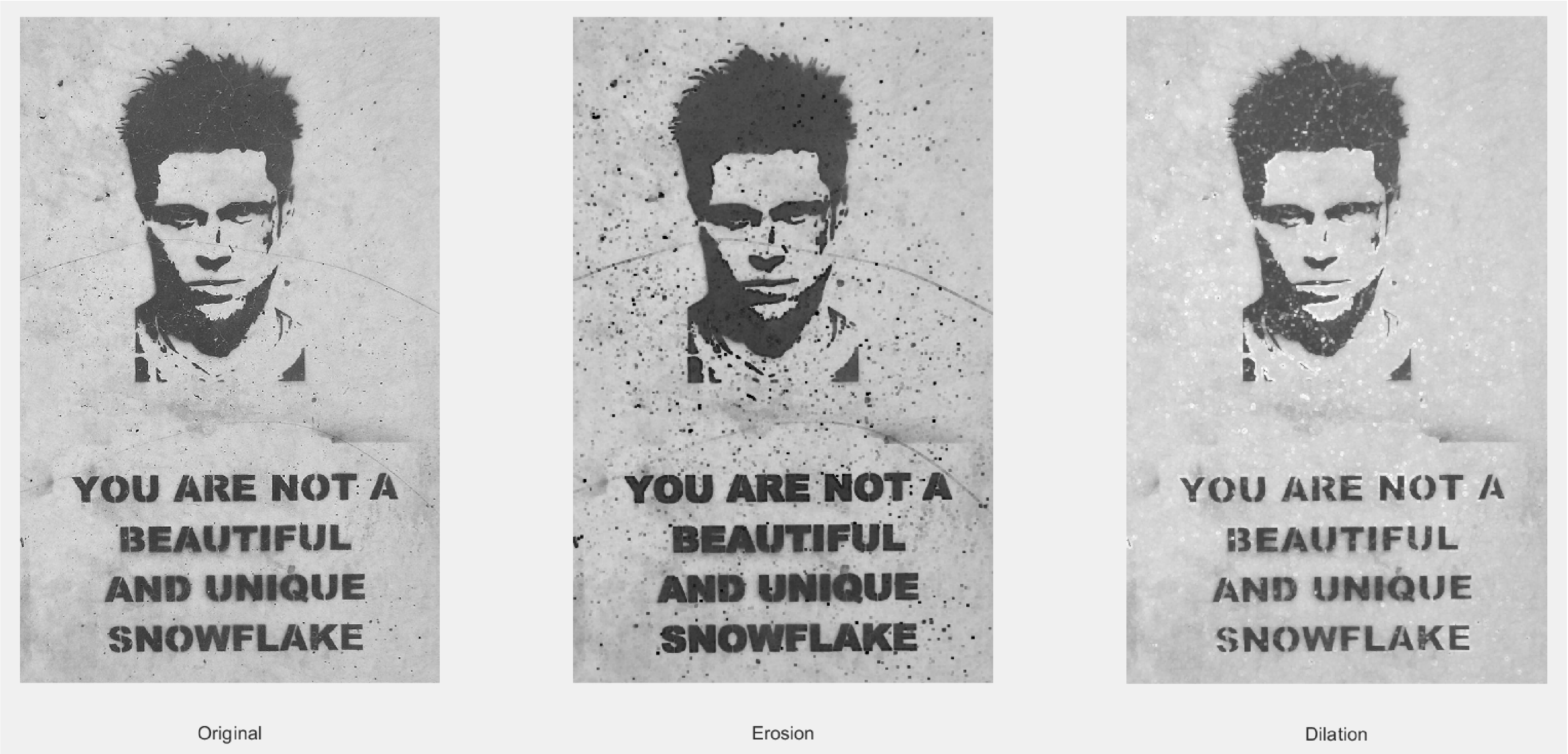
Dilation

Grayscale Erosion and Dilation

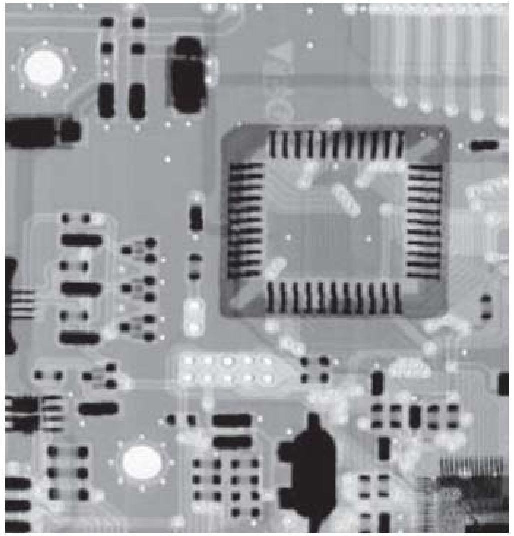
Original Erosion Dilation

Original Erosion Dilation

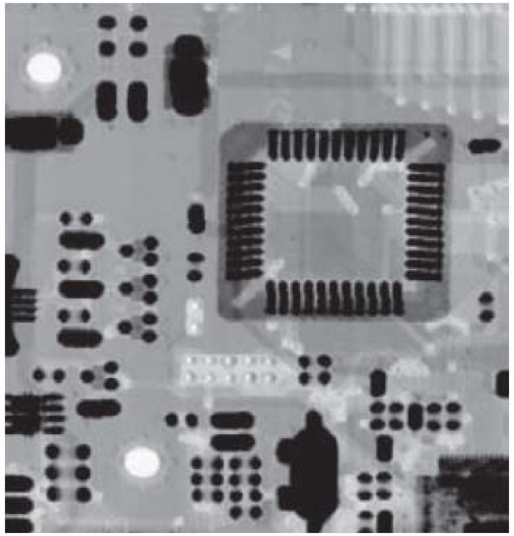
Grayscale Erosion and Dilation



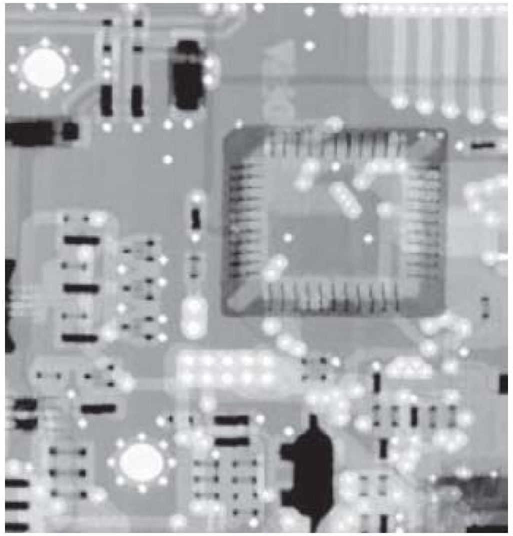
Grayscale Erosion and Dilation



Gray-scale X-ray image of size 448×425 pixels.



Erosion using a flat disk SE with a radius of 2 pixels.



Dilation using the same SE.

- The eroded grayscale image is darker than the original image, with that the sizes of bright features are reduced, and that the sizes of dark features are increased.
- The effects of dilation are the opposite of using erosion. The bright features are thickened and the intensities of the darker features are reduced. The sizes of the dark dots in dilation are reduced.

Summary

- In this lecture we have learnt:
  - Boundary extraction
  - Region filling
  - Extraction of connected components
  - Convex hull
  - Thinning/thickening
  - Skeletons
  - Grayscale morphological processing

Optional Homework

Check the Textbook!

- **Chapter 9: Problems 9.30, 9.32, 9.51(a)**

- Homework answers will be provided at the end of each week.