Fundamentals of Computer Vision

Unit 5: Non-Linear Filtering

Jorge Bernal

Index

01

1. Median Filters

02

2. Homomorphic Filters

03

3. Mathematical Morphology

1

Median Filters

Median Filters

- **Limitation of Linear Filters**
  - Frequency shaping enhance some frequency components and suppress the others
  - For individual frequency component, we cannot differentiate its “desirable” and “undesirable” parts
- **Nonlinear Filters**
  - Cannot be expressed as convolutions
  - Cannot be expressed as frequency shaping
- **“Nonlinear” Means Everything** (other than linear)
  - Need to be more specific
  - They often use heuristics
  - We will study some “nice” ones

Median Filters

- **Order Statistics (OS)**
  - Given a set of numbers
    Denote the OS as
    such that

$$\boldsymbol{x} = \{x_1, x_2, \cdots, x_{2M+1}\}$$

max value

$$\boldsymbol{x}_{OS} = \{x_{(1)}, x_{(2)}, \cdots, x_{(2M+1)}\}$$

$$x_{(1)} \le x_{(2)} \le \cdots \le x_{(M+1)} \le \cdots \le x_{(2M+1)}$$

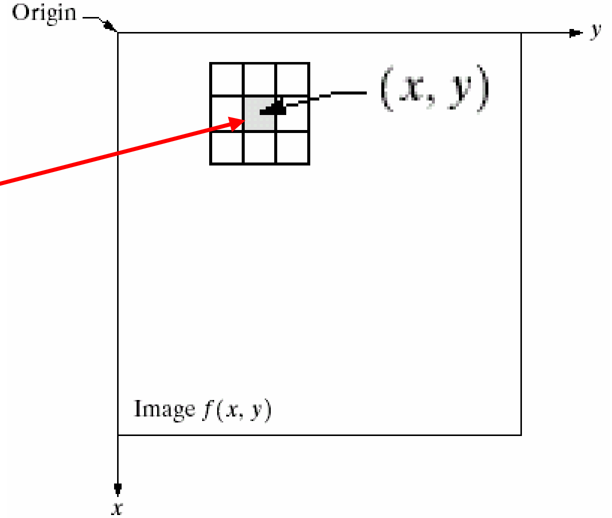
min value

middle value

- **Median**
  - Define

$$Median\{x_1, x_2, \cdots, x_{2M+1}\} = x_{(M+1)}$$

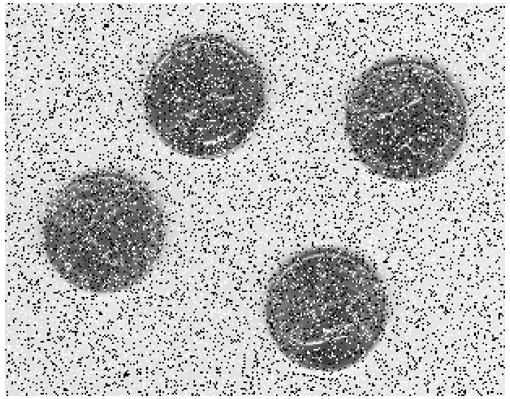
- **Applying Median Filters to Images**
  - Use sliding windows
  - Typical windows:
    3x3, 5x5, 7x7, other shapes ……



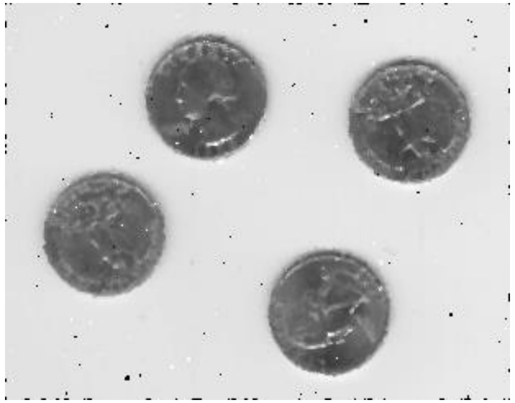
Median Filters

original

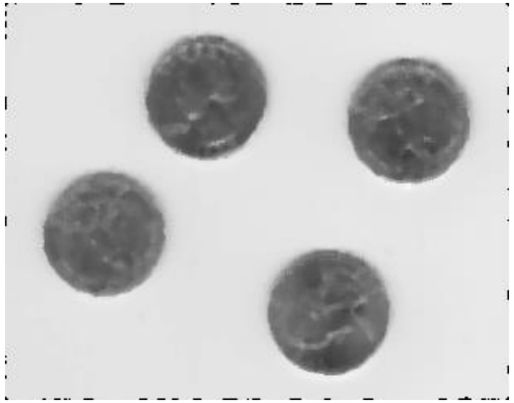
noisy ($p_a = p_b = 0.1$)



median filtered 3x3 window



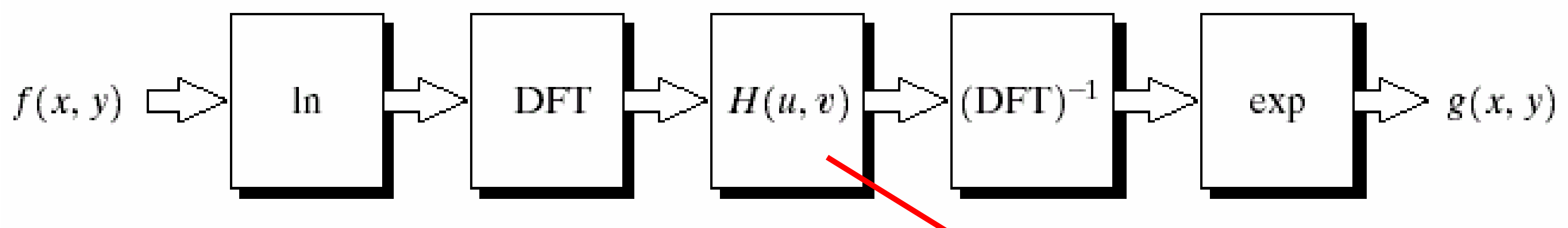
median filtered 5x5 window



2

Homomorphic Filters

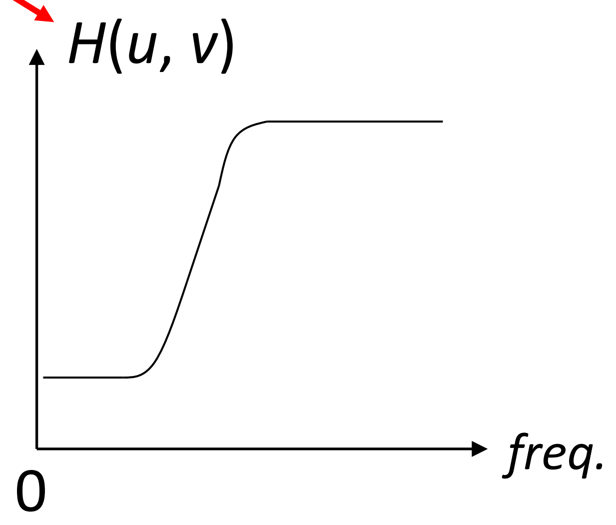
Homomorphic Filters



$f(x, y) = i(x, y)\, r(x, y)$

illumination
(slowly varying)
(low-frequency)

reflectance
(fast variation)
(high-frequency)



$\ln[f(x, y)] = \ln[i(x, y)] + \ln[r(x, y)]$

Key: linear separation

Homomorphic Filters

before

after

3

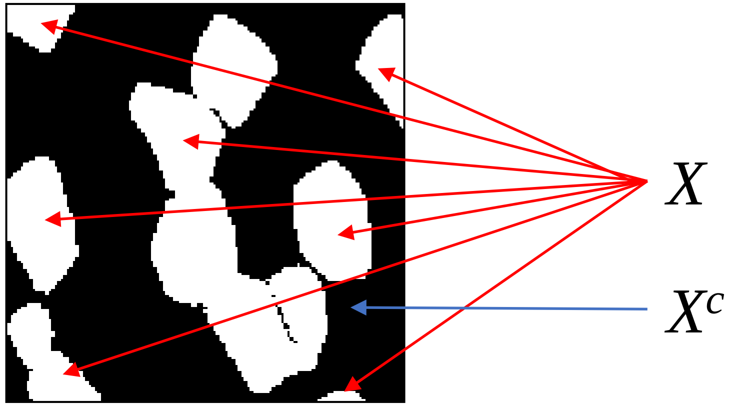
Mathematical Morphology

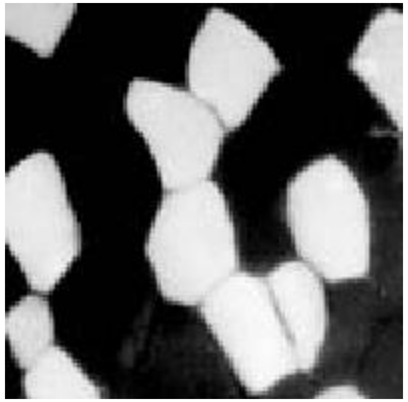
Mathematical Morphology

- Self-sufficient framework for image processing and analysis, created at the École des Mines (Fontainebleau) in 70’s by Jean Serra, Georges Mathéron, from studies in science materials
- Conceptually simple operations combined to define others more and more complex and powerful
- Operations have a clear geometrical meaning
- Powerful for image analysis

Mathematical Morphology

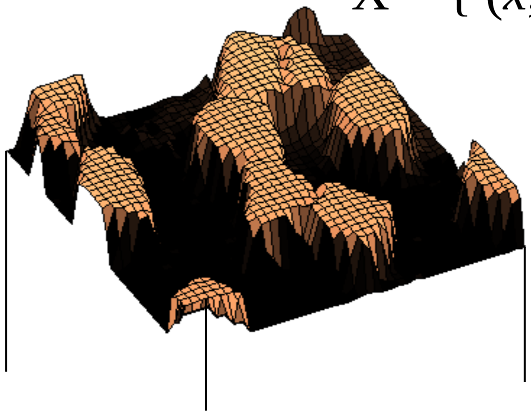
Binary and grey-level images seen as sets





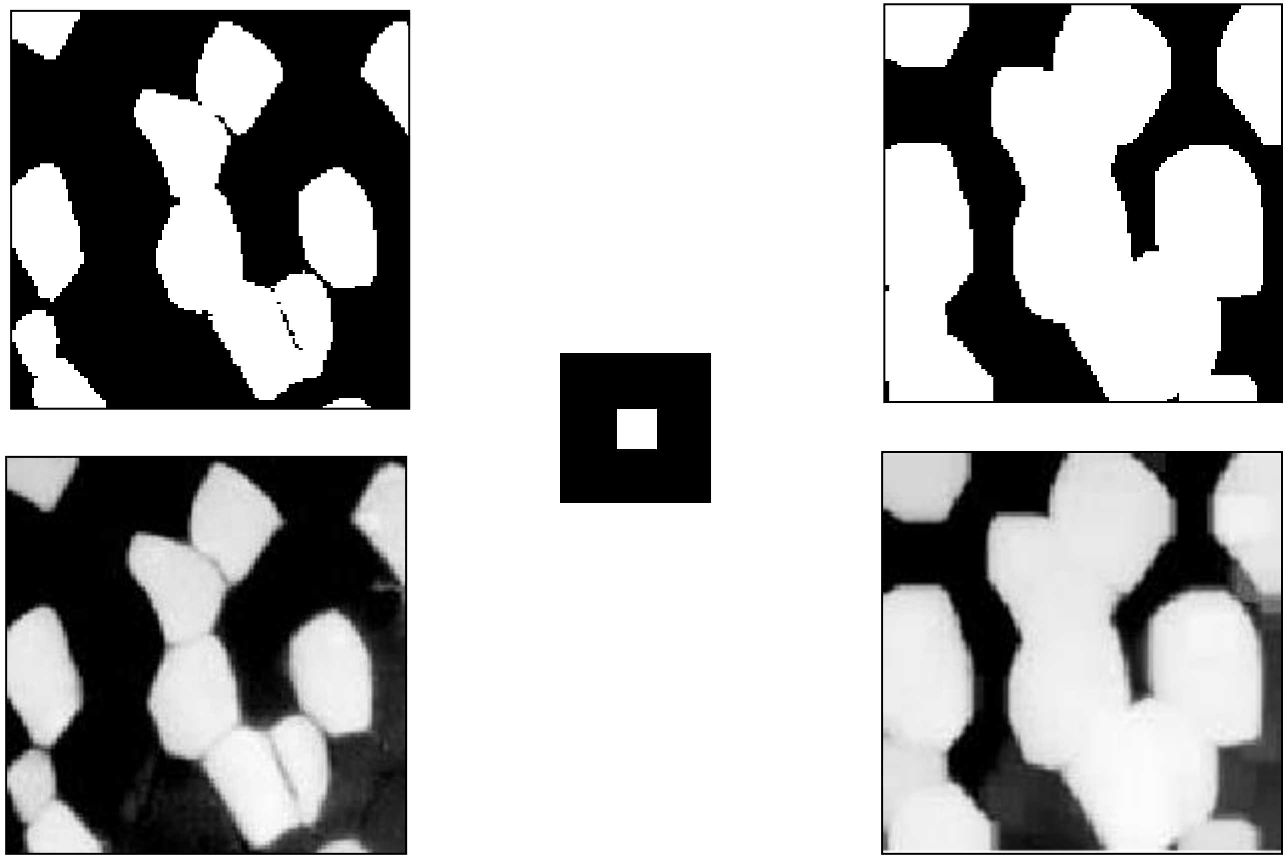
$f\,(x,y)$

$$X = \{\ (x,\, y,\, z)\ \ ,\ \ z \leq\ f\,(x,y)\ \}$$



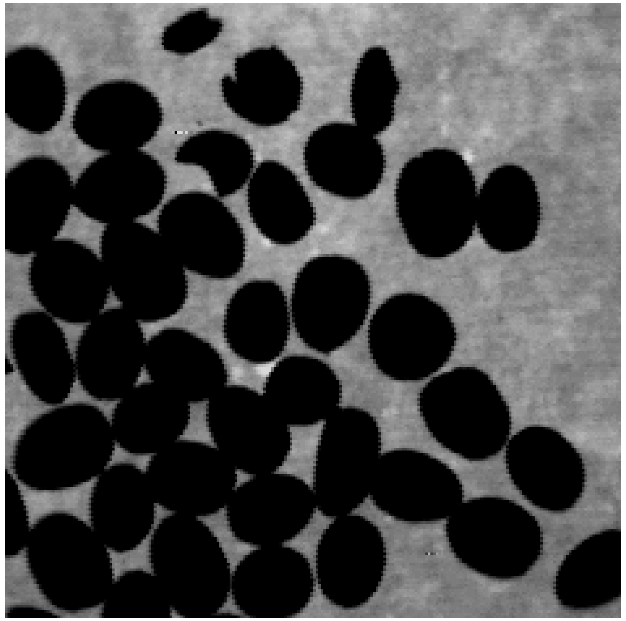
Mathematical Morphology

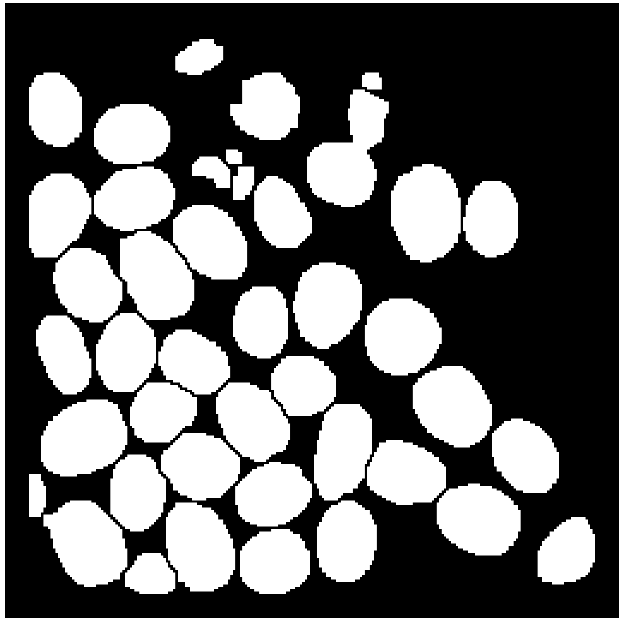
Operations defined as interaction of images with a special set, the *structuring element*



Mathematical Morphology

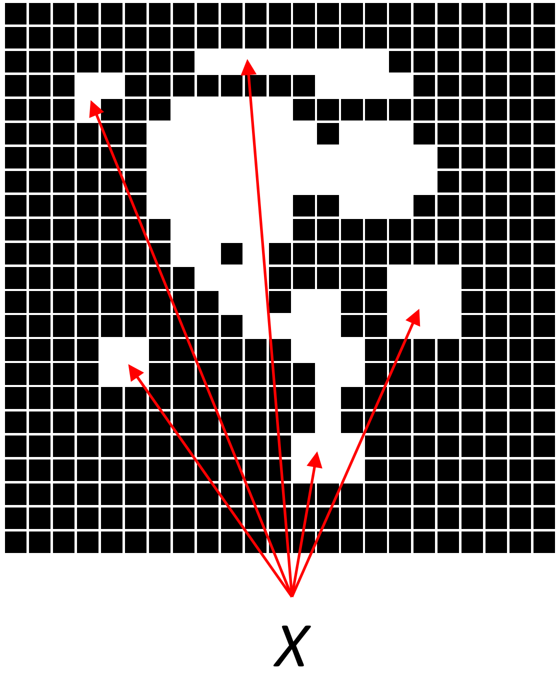
Mathematical Morphology



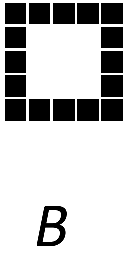


Binary Morphology

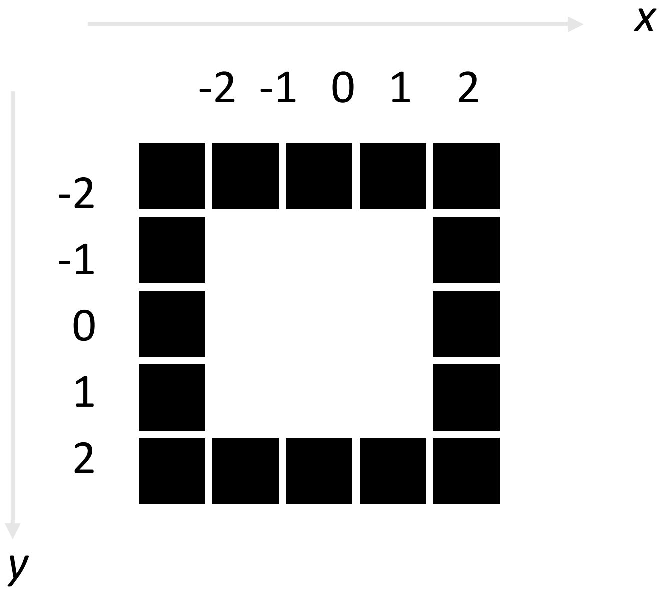
Notation



No necessarily compact nor filled



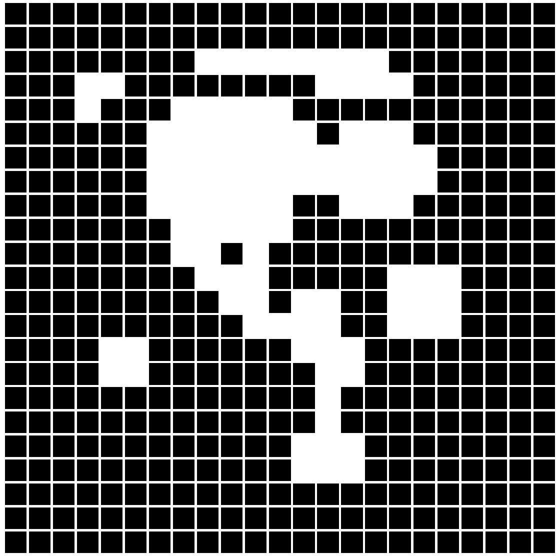
A special set : the structuring element



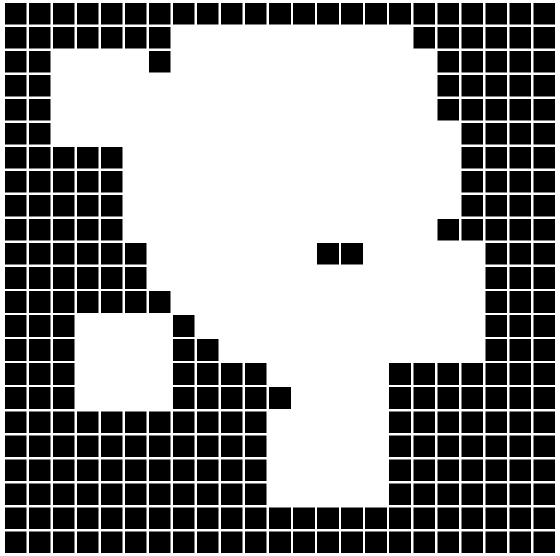
Origin at center in this case, but not necessarily centered nor symmetric

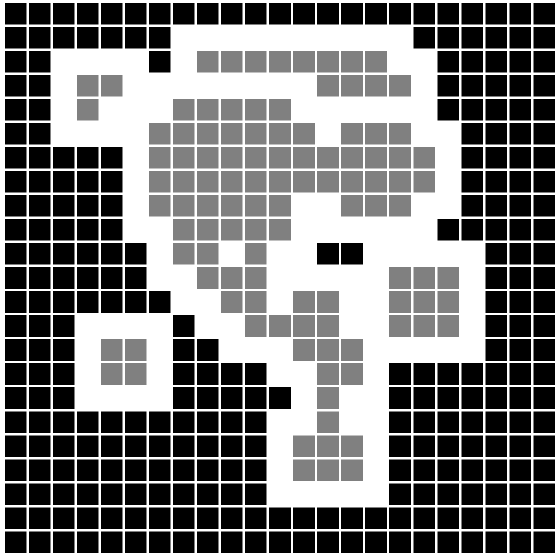
Binary Morphology: Dilate

Dilation : $x = (x_1,x_2)$ such that if we center *B* on them, then the so translated *B* intersects *X*.

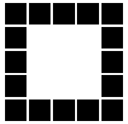


X





difference



B

Binary Morphology: Dilate

Dilation : $x = (x_1,x_2)$ such that if we center B on them,
then the so translated B intersects X.

How to formulate this definition ?

1) Literal translation

$$B_x \stackrel{def}{=} \{b + x \mid b \in B\} \qquad x, b \in \mathbf{R}^2$$

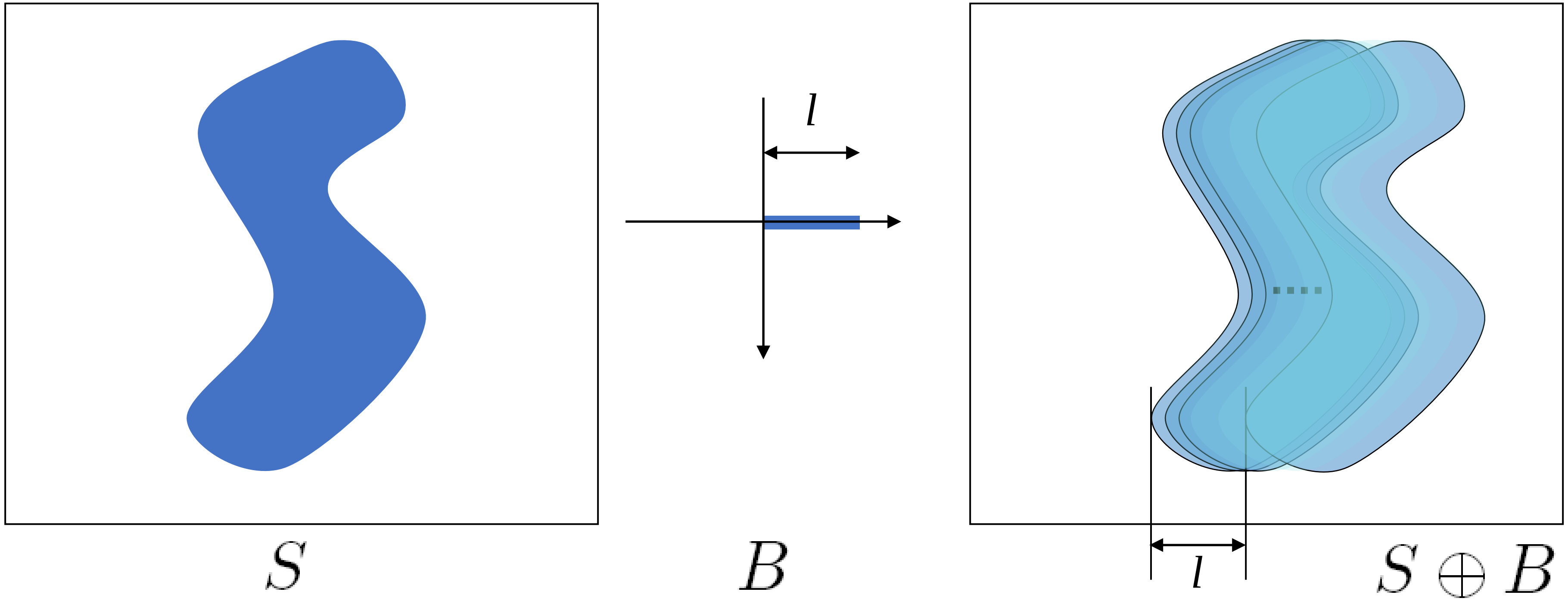
$$\{x \mid B_x \cap X \neq \emptyset\}$$

2) Better : from Minkowski's sum of sets $\quad X \oplus \check{B}$

Binary Morphology: Dilate

Minkowski's sum of sets :

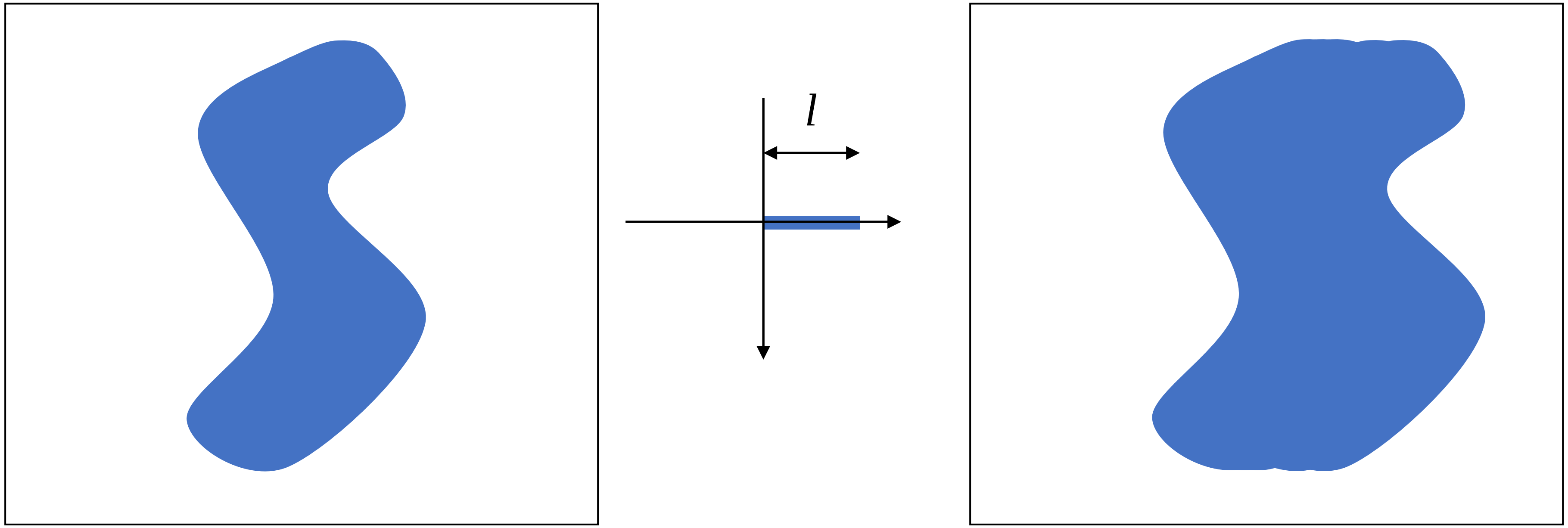
$$S \oplus B \overset{def}{=} \{s + b \mid s \in S, b \in B\} = \bigcup_{b \in B} S_b$$



Binary Morphology: Dilate

Minkowski's sum of sets :

$$S \oplus B \stackrel{def}{=} \{s + b \mid s \in S, b \in B\} = \bigcup_{b \in B} S_b$$

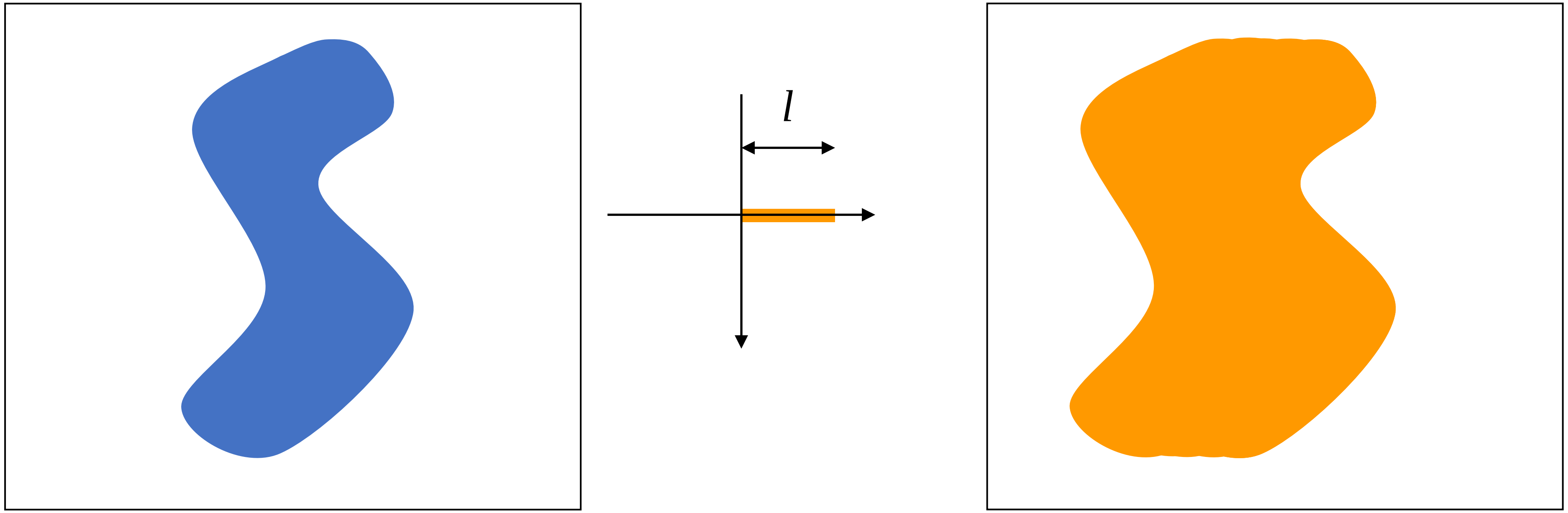


S B $S \oplus B$

Binary Morphology: Dilate

Dilation : $B_x \stackrel{def}{=} \{b + x \mid b \in B\} \quad x, b \in \mathbf{R}^2$

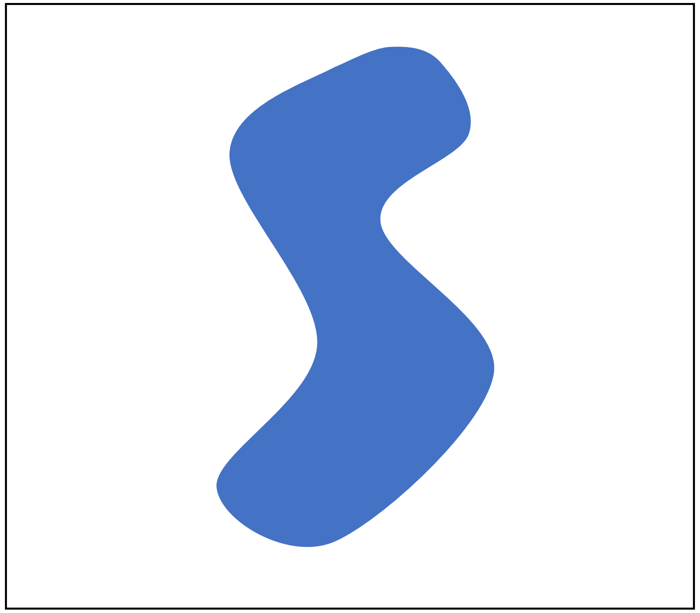
$$\{x \mid B_x \cap X \neq \emptyset\}$$



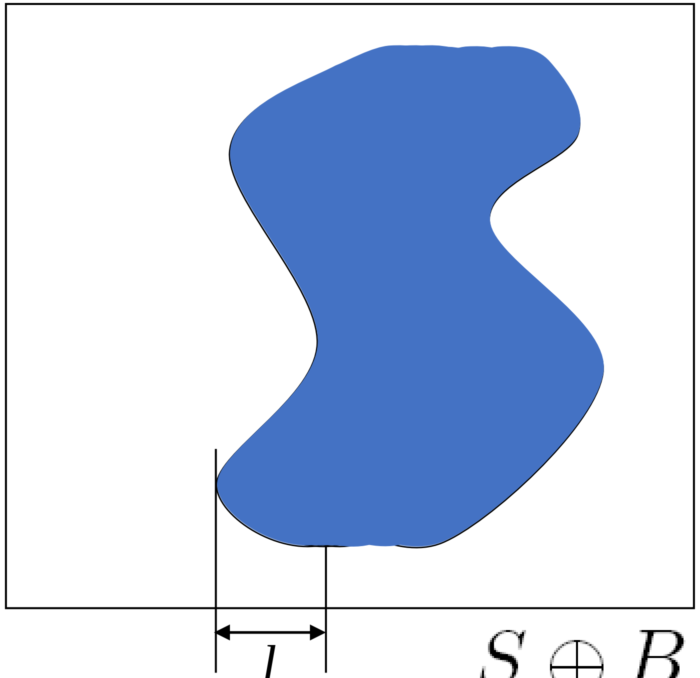
S B Dilation

Binary Morphology: Dilate

Dilation is *not* the Minkowski's sum



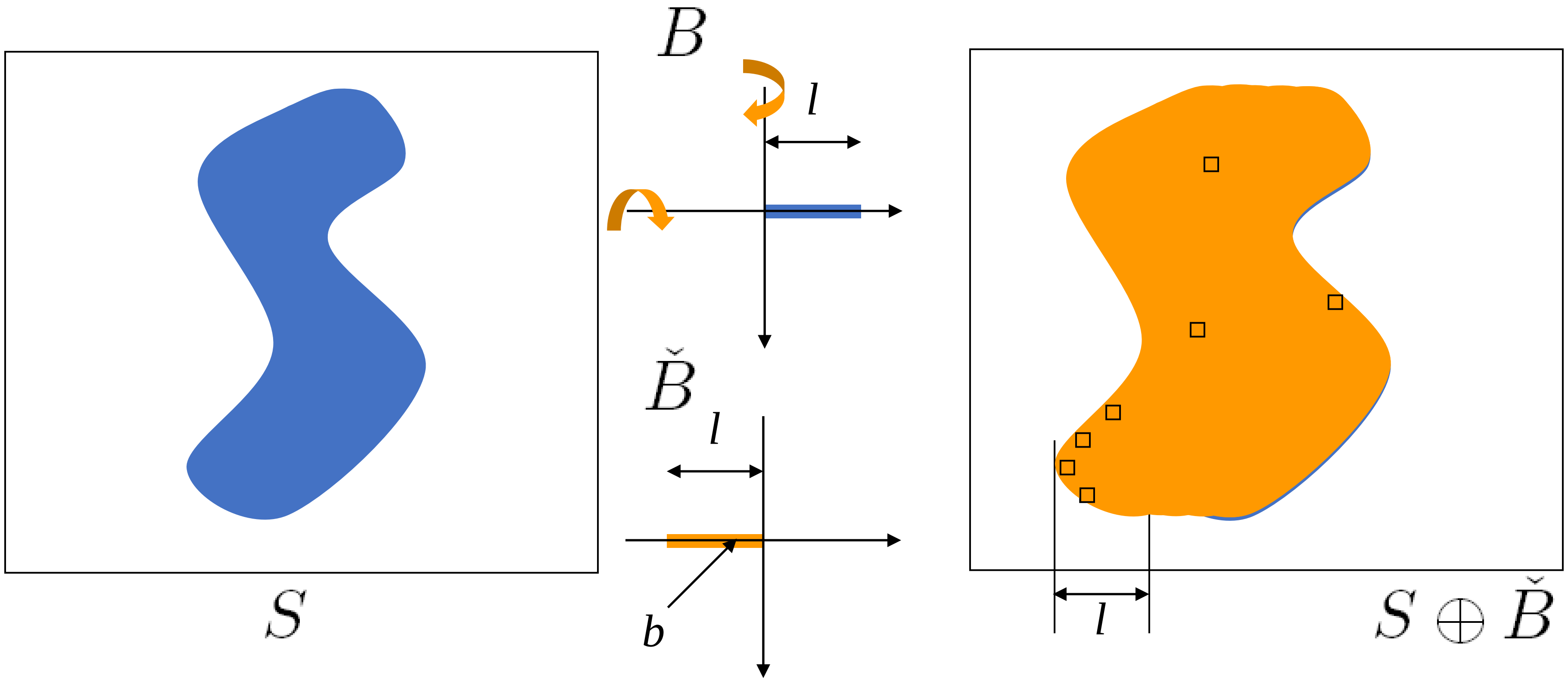
S



$S \oplus B$

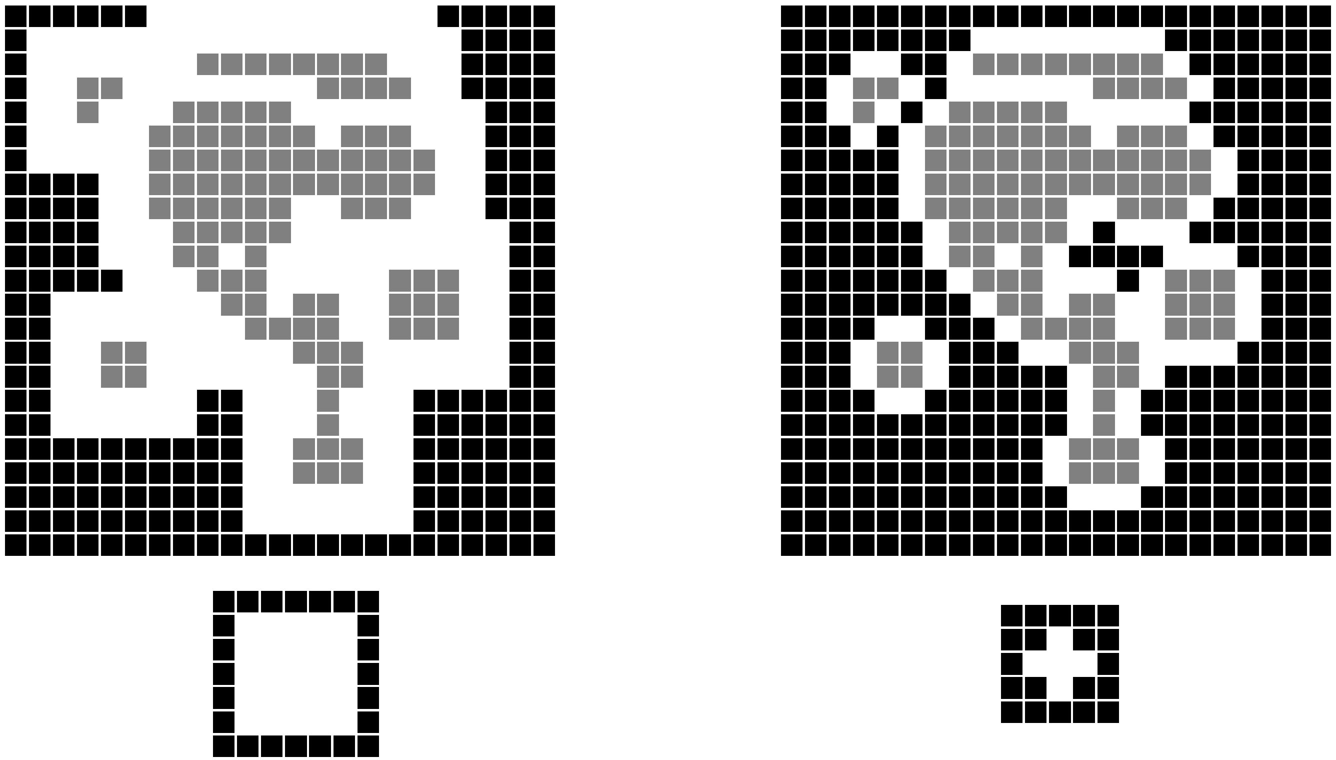
Binary Morphology: Dilate

$$\{x \mid B_x \cap X \neq \emptyset\} = \bigcup_{b \in \check{B}} X_b = X \oplus \check{B}$$



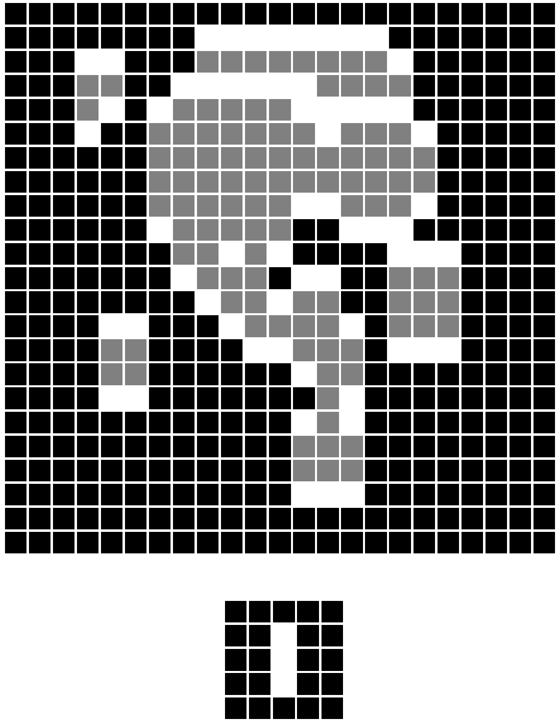
Binary Morphology: Dilate

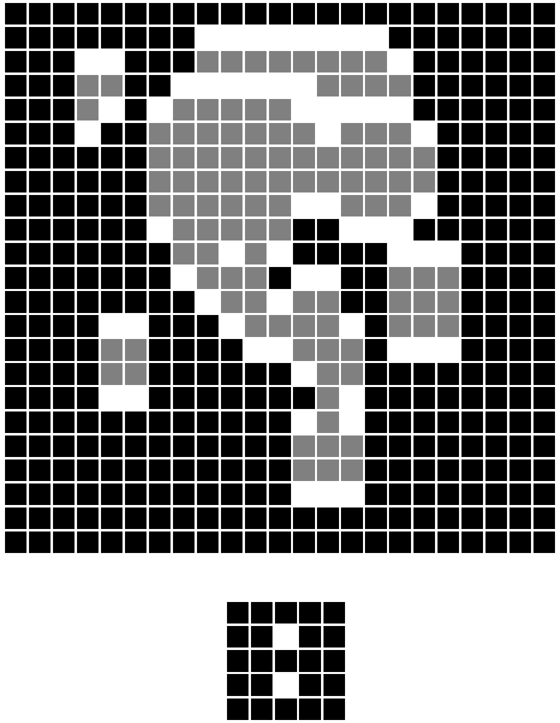
Dilation with other structuring elements



Binary Morphology: Dilate

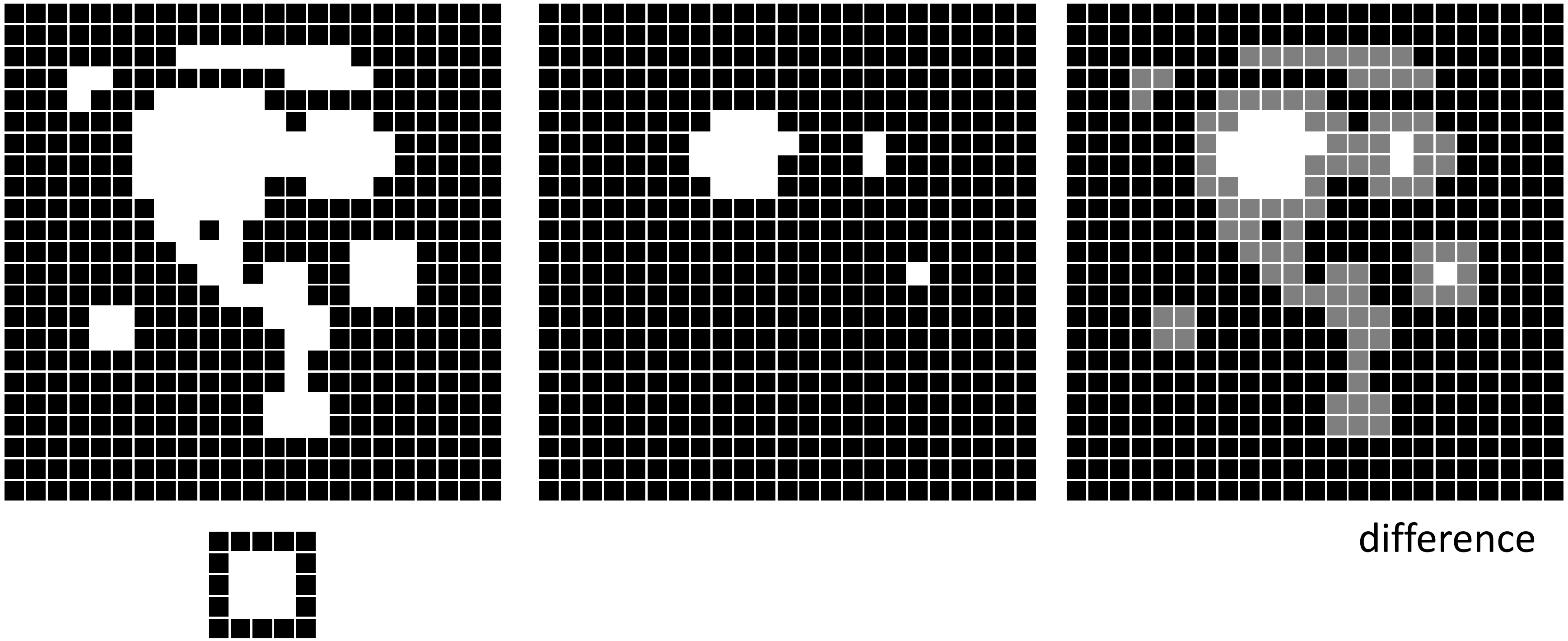
Dilation with other structuring elements





Binary Morphology: Erode

Erosion : $x = (x_1, x_2)$ such that if we center B on them,
then the so translated B is contained in X.



difference

Binary Morphology: Erode

Erosion : $x = (x_1,x_2)$ such that if we center B on them,
then the so translated B is contained in X.

How to formulate this definition ?

1) Literal translation

$$\{x \mid B_x \subseteq X\}$$

2) Better : from Minkowski's substraction of sets

$$X \ominus \check{B}$$

Binary Morphology: Erode

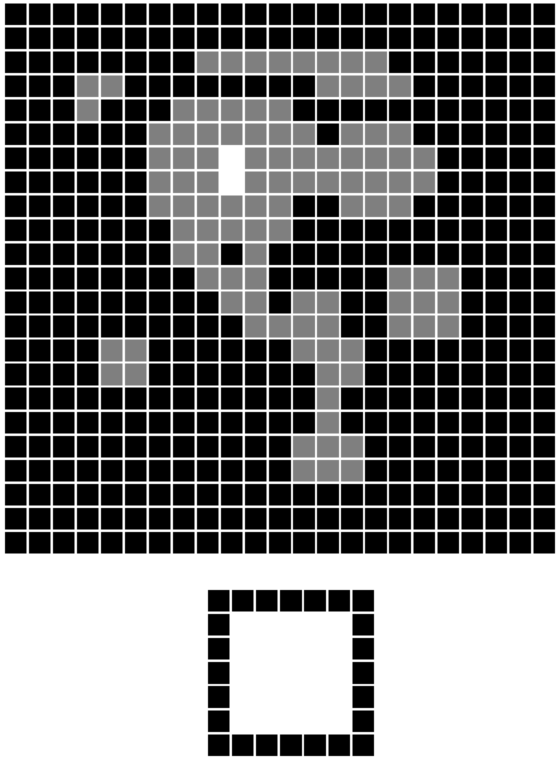
$$S \ominus B \overset{def}{=} \{y \mid y - b \in X, \;\; \forall b \in B\} = \bigcap_{b \in B} S_b$$

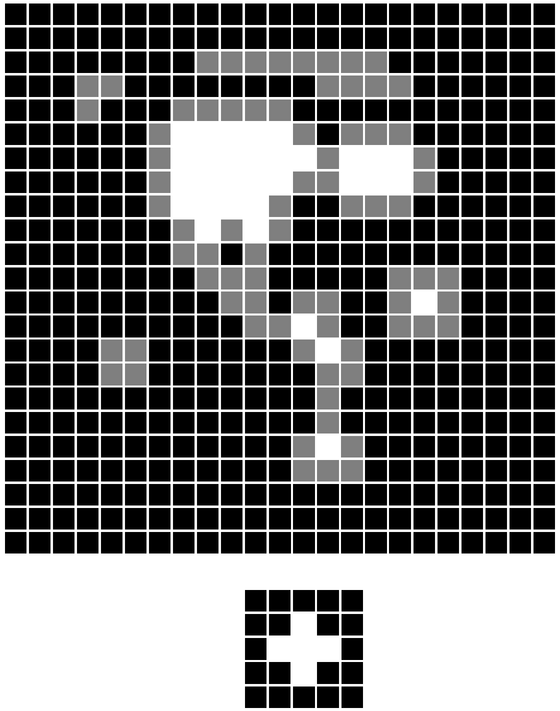
$$\{x \mid B_x \subseteq X\} = \bigcap_{b \in \check{B}} X_b$$

$$\bigcap_{b \in \check{B}} X_b = \{y \mid y - b \in X, \;\; \forall b \in \check{B}\} = X \ominus \check{B}$$

Binary Morphology: Erode

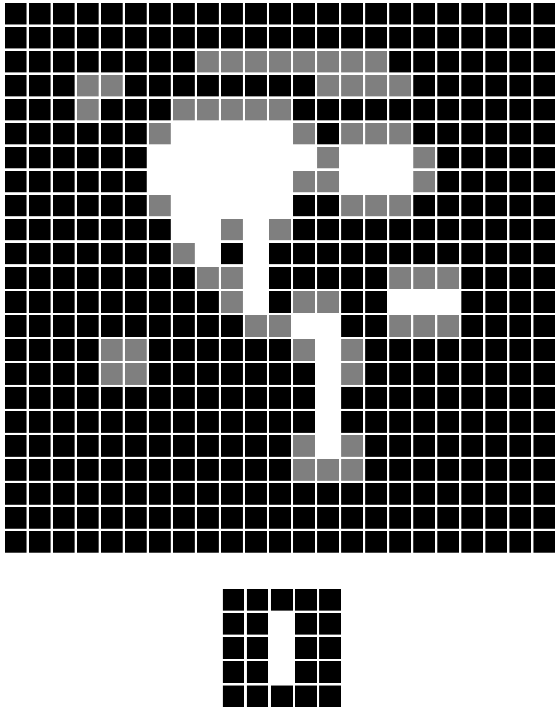
Erosion with other structuring elements

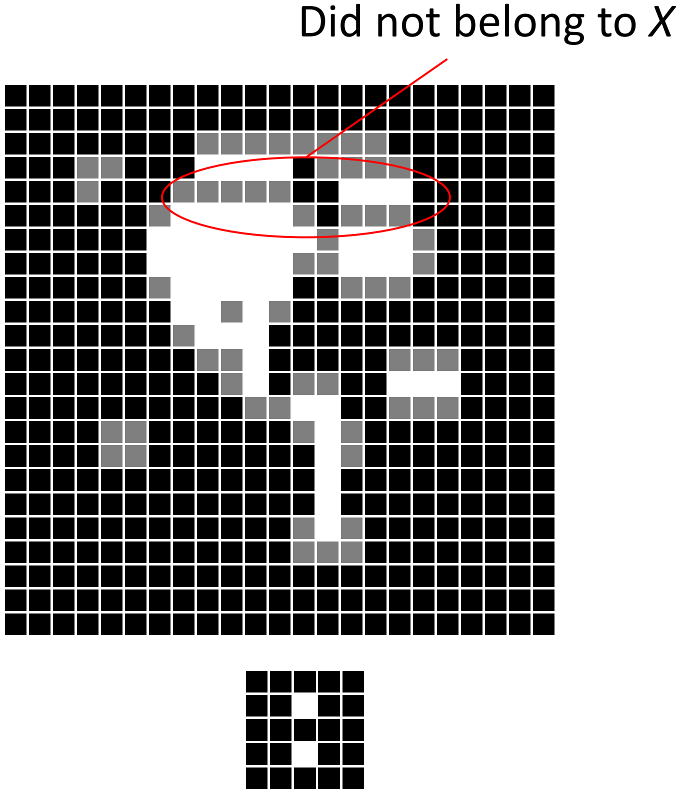




Binary Morphology: Erode

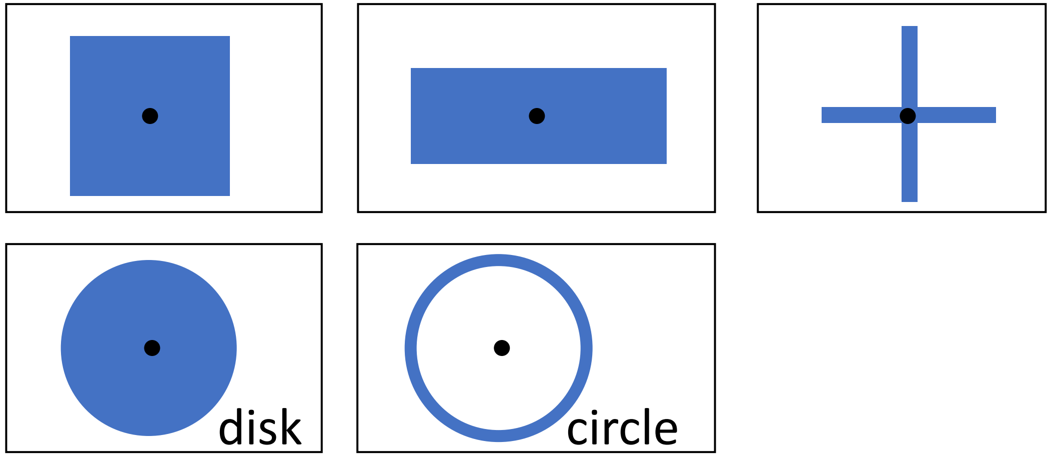
Erosion with other structuring elements

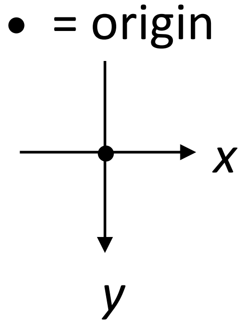


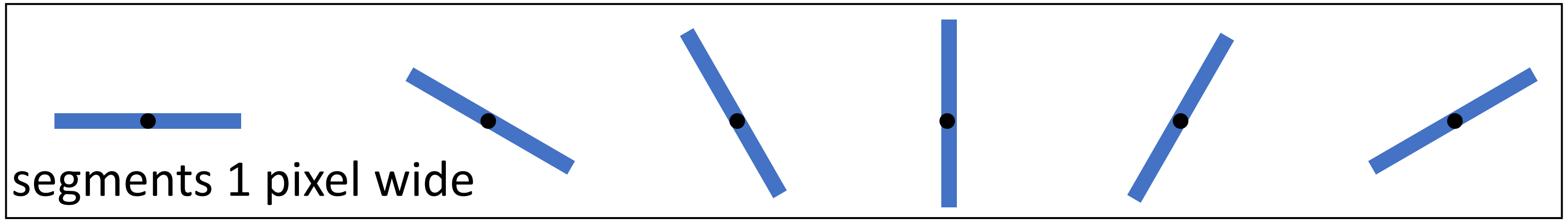


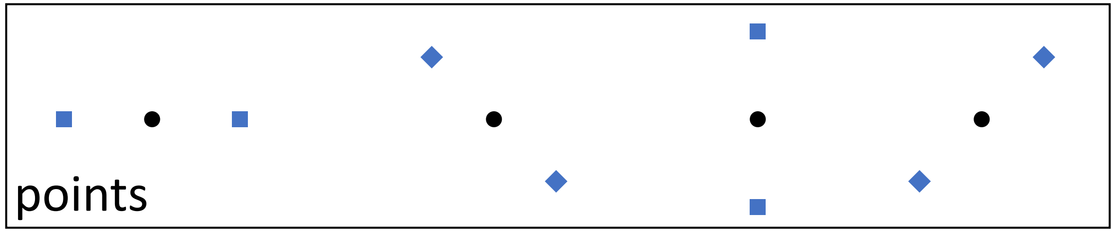
Binary Morphology: Structuring Elements

Common structuring elements shapes







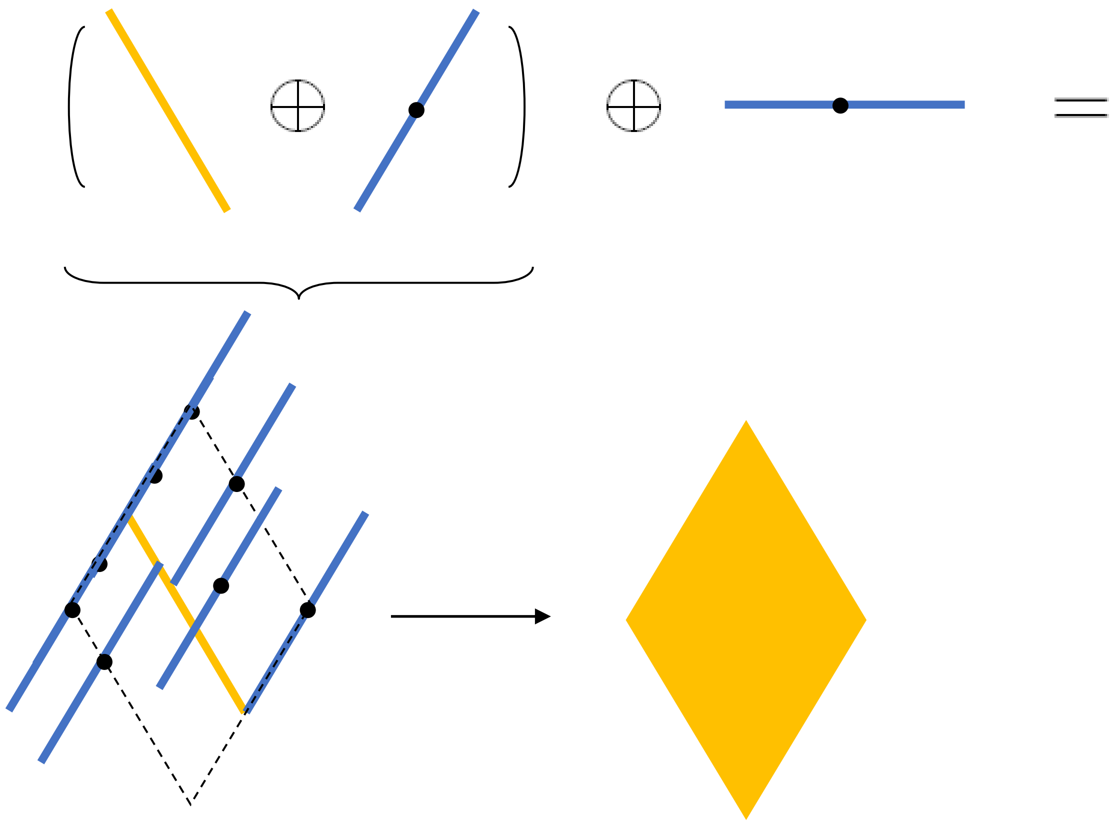


Note :

$$B = \check{B}$$

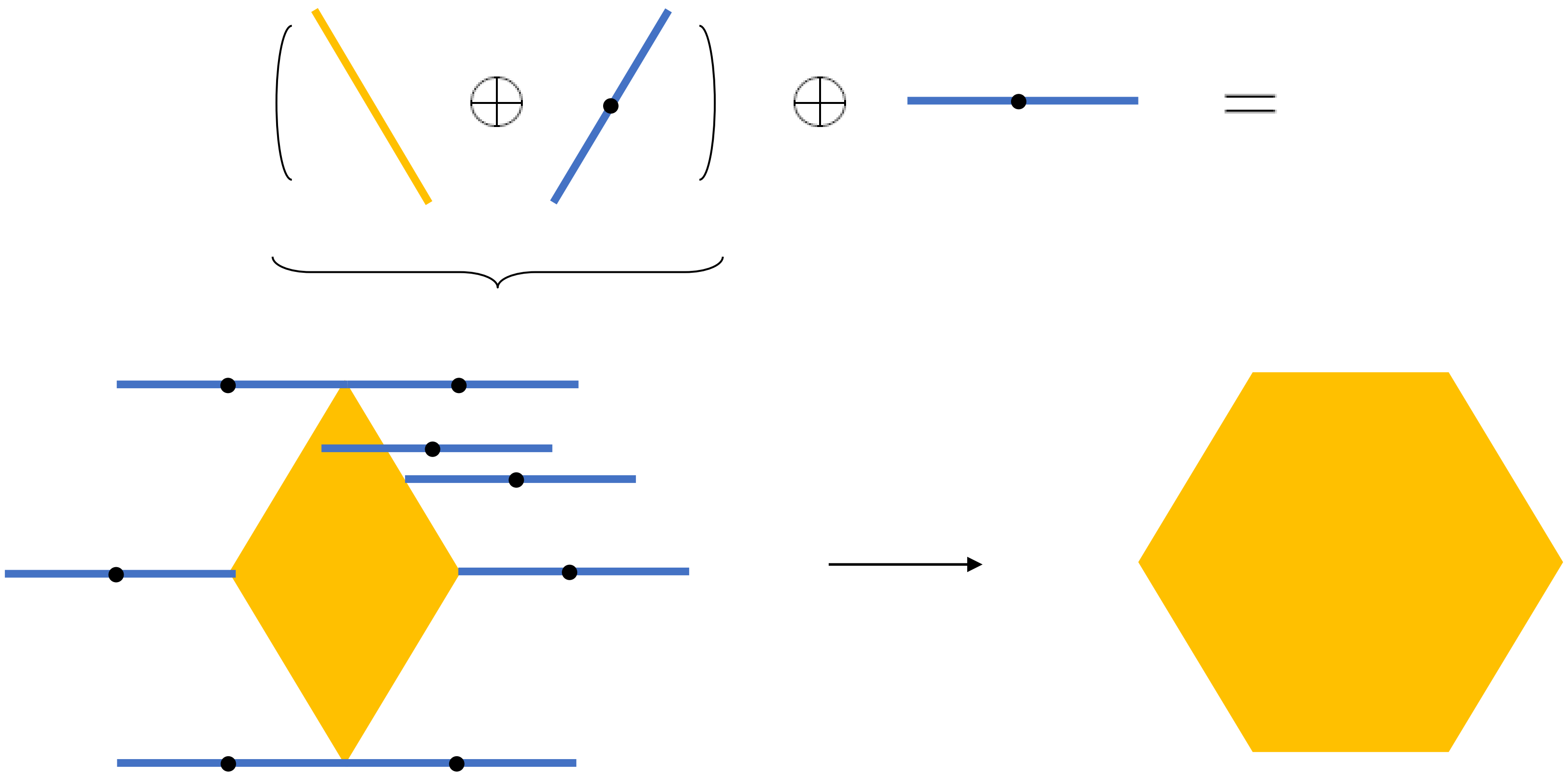
Binary Morphology

Problem :



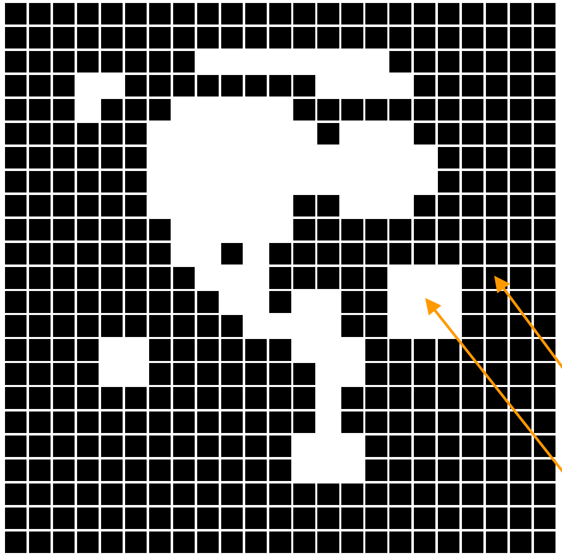
Binary Morphology

Problem :



Binary Morphology

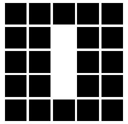
Implementation : very low computational cost



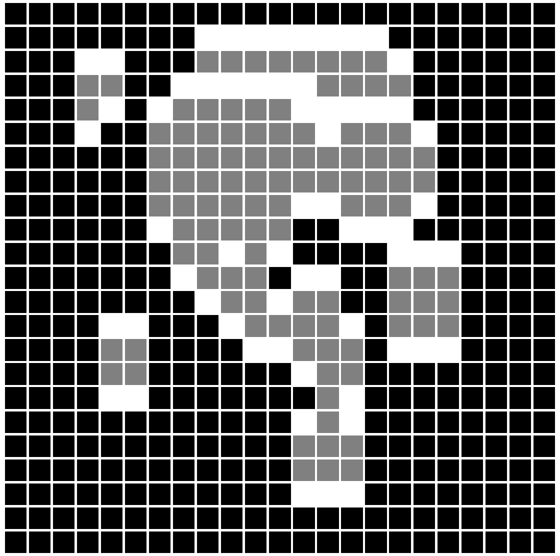
0

1 (or >0)

X



B



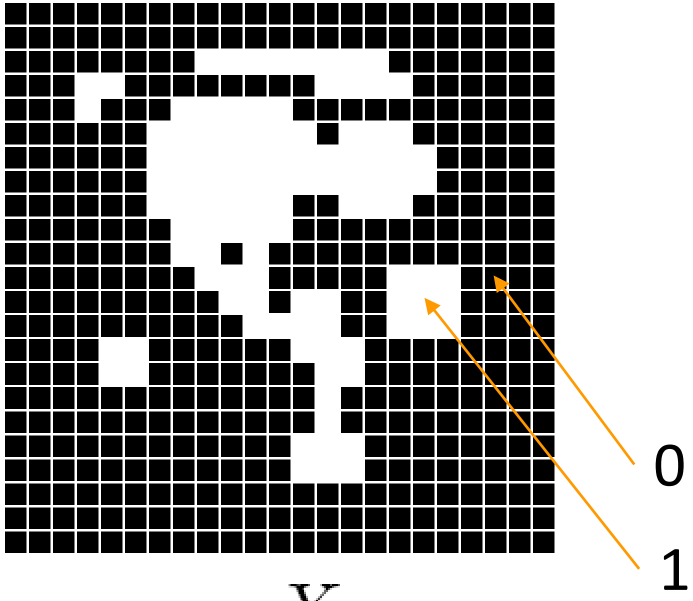
$X \oplus \check{B}$

$$
\begin{aligned}
(X \oplus \check{B})_{i,j} &= \bigvee \left(X_{i,j-1}, X_{i,j}, X_{i,j+1} \right) \\
&= \bigvee \left(X_{(i,j)+b, b \in \check{B}} \right)
\end{aligned}
$$

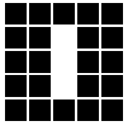
Logical *or*

Binary Morphology

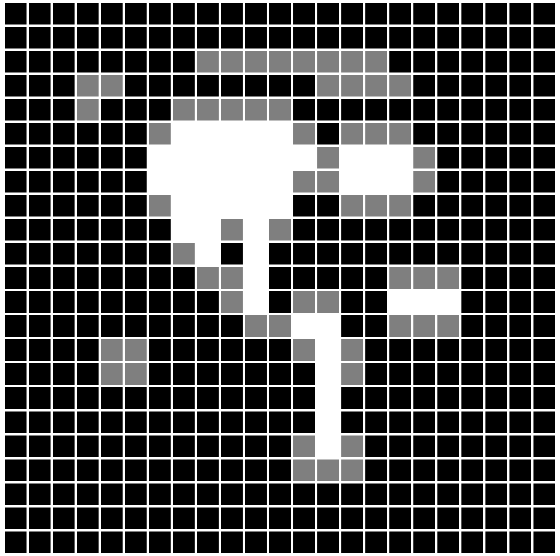
Implementation : very low computational cost



X



B



$X \ominus \check{B}$

$$(X \ominus \check{B})_{i,j} = \bigwedge (X_{i,j-1}, X_{i,j}, X_{i,j+1})$$

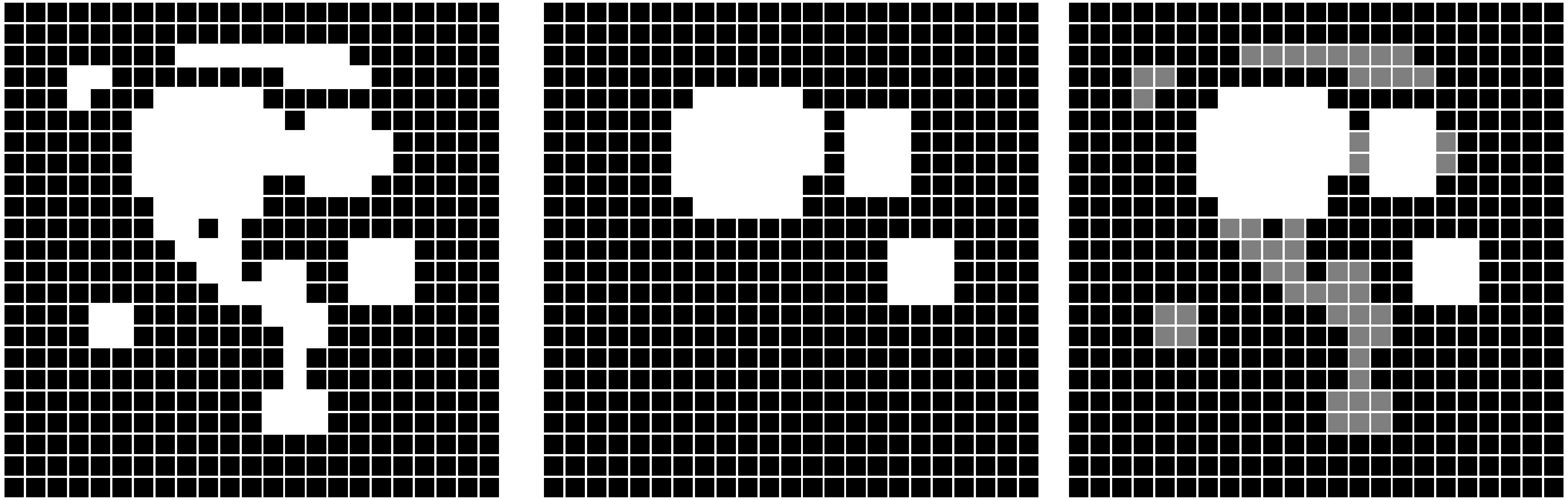
Logical *and*

$$= \bigwedge (X_{(i,j)+b,\in \check{B}})$$

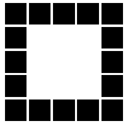
Binary Morphology: Open

Opening :

also X_B



difference

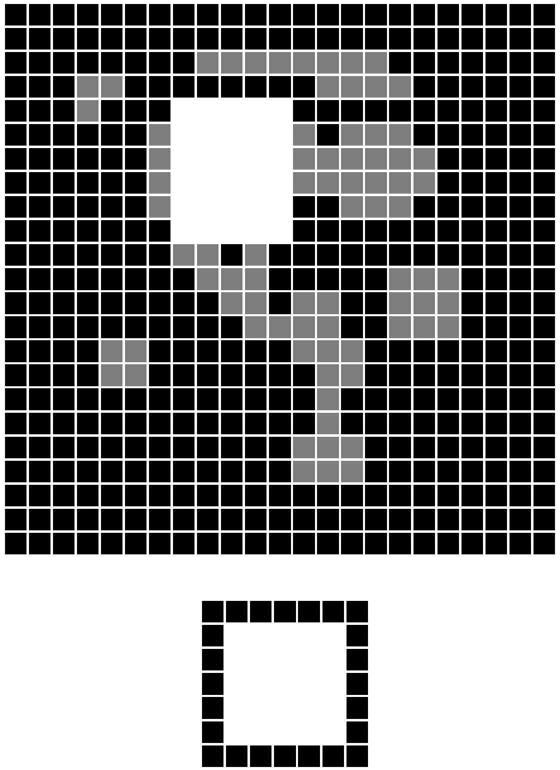


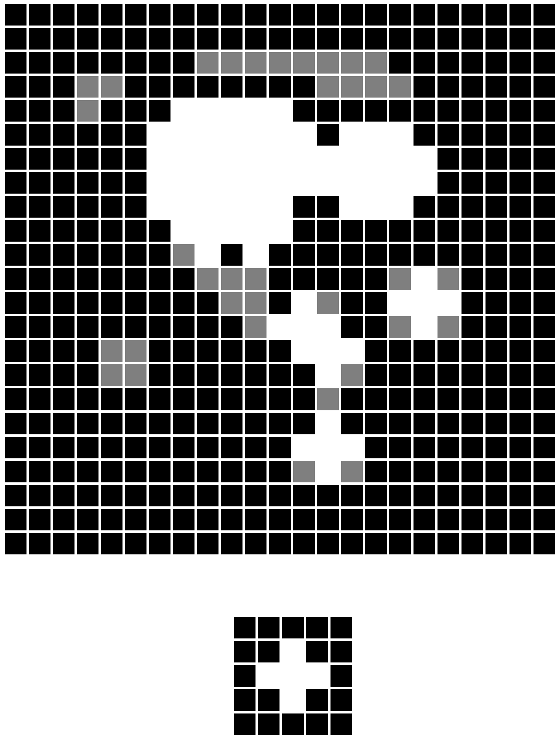
Suppresses :

- small islands
- isthmus (narrow unions)
- narrow caps

Binary Morphology: Open

Opening with other structuring elements

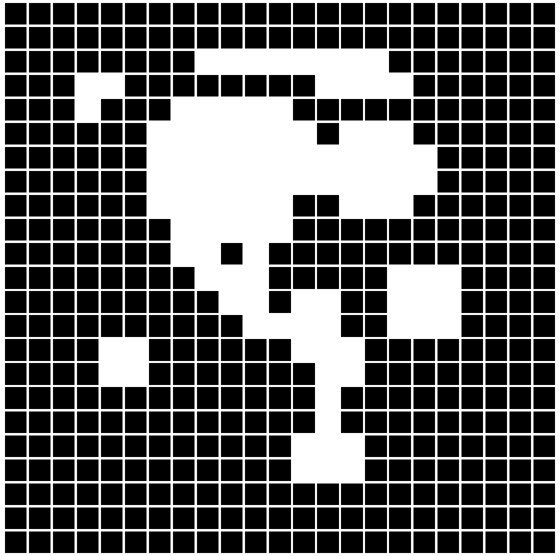


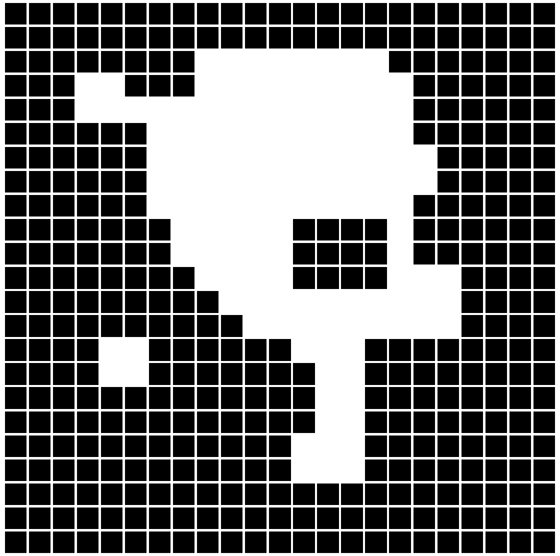


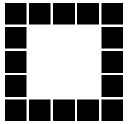
Binary Morphology: Close

Closing : $X \bullet B \stackrel{def}{=} \Big(\bigcup_{B_x \subseteq X^c} B_x \Big)^c = (X \oplus \check{B}) \ominus \check{B}$

also X^B







Suppresses :

- small lakes (holes)
- channels (narrow separations)
- narrow bays

Binary Morphology: Application

Application: shape smoothing and noise filtering

X

Binary Morphology: Application

Application: shape smoothing and noise filtering

$$X \ominus \check{B}$$

Binary Morphology: Application

Application: shape smoothing and noise filtering

X

$X \ominus \check{B}$ ✖

$X \circ B$

$(X \circ B) \oplus \check{B}$

Binary Morphology: Application

Application: shape smoothing and noise filtering

X

$X \ominus \check{B}$ ✖

$X \circ B$

$(X \circ B) \oplus \check{B}$ ✖

$(X \circ B) \bullet B$

Binary Morphology: Properties

Properties

- all of them are *increasing* :

$$X \subseteq Y \Longrightarrow \Psi(X) \subseteq \Psi(Y)$$

- opening and closing are *idempotent* :

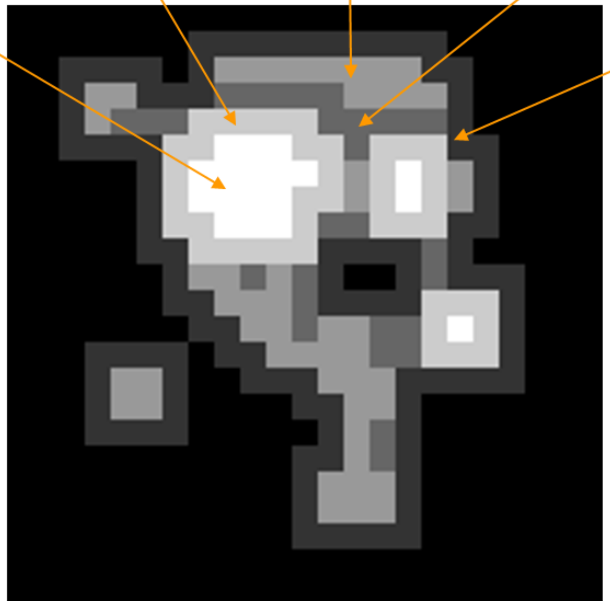
$$\Psi(X) = \Psi(\Psi(X))$$

Binary Morphology: Properties

dilation and closing are *extensive*
erosion and opening are *anti-extensive* :

$$(0,0) \in B \implies$$

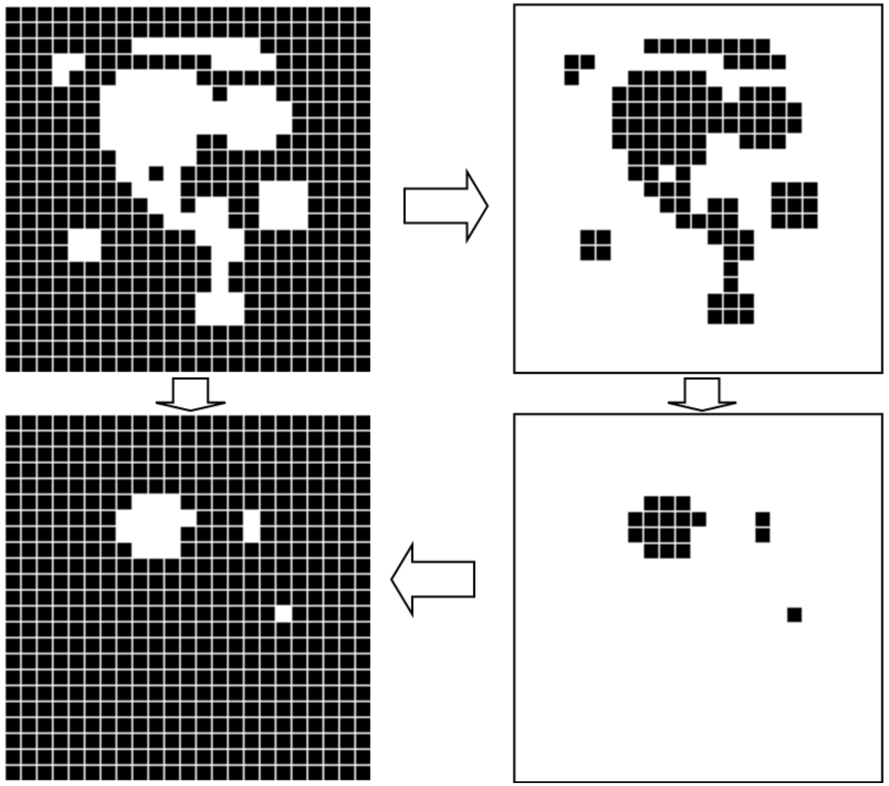
$$\boxed{X \ominus \check{B}} \subseteq \boxed{X \circ B} \subseteq \boxed{X} \subseteq \boxed{X \bullet B} \subseteq \boxed{X \oplus \check{B}}$$



Binary Morphology: Properties

- *duality* of erosion-dilation, opening-closing,...

$$\Psi, \Phi \;\; \text{duals} \quad \Psi(X) = [\Phi(X^c)]^c$$



X

X^c

$(X^c \oplus \check{B})^c$
$= X \ominus \check{B}$

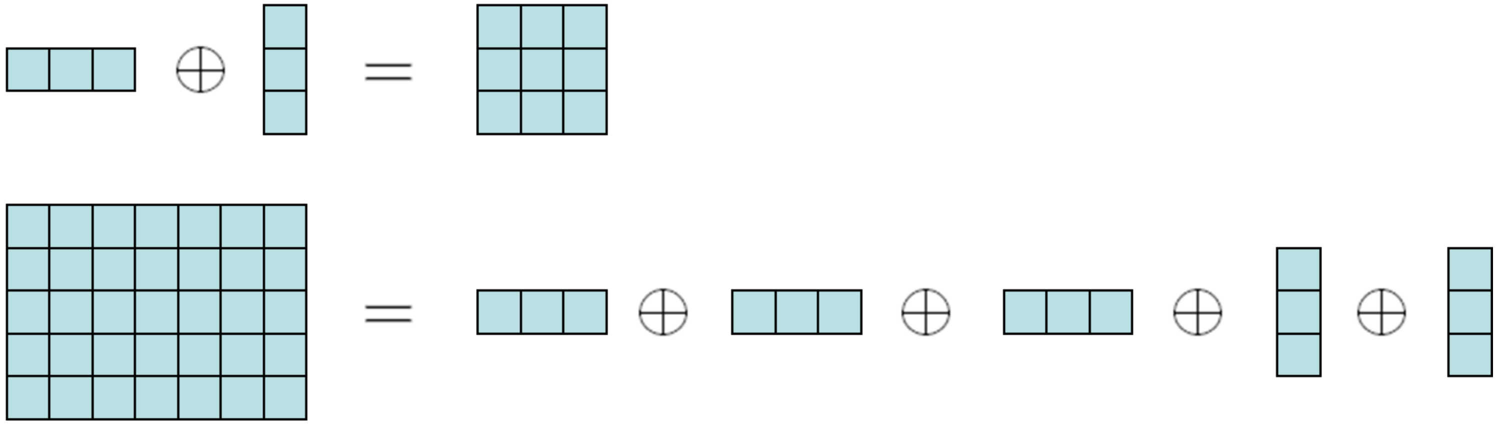
$X^c \oplus \check{B}$

Binary Morphology: Properties

- structuring elements decomposition

$$X \oplus (B_1 \oplus \check{B}_2) = (X \oplus \check{B}_1) \oplus \check{B}_2$$
$$X \ominus (B_1 \oplus \check{B}_2) = (X \ominus \check{B}_1) \ominus \check{B}_2$$

operations with big structuring elements can be done by a succession of operations with small s.e's

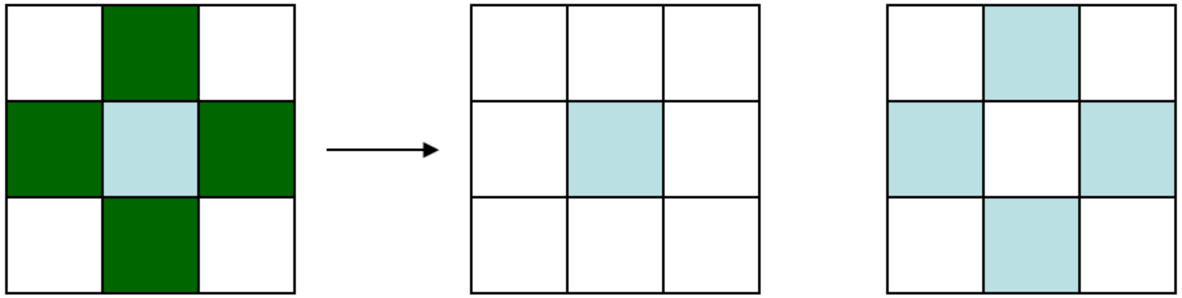


Binary Morphology: Hit-or-miss

Hit-or-miss : $$X \otimes B = (X \ominus \check{B}_1) \bigcap (X^c \ominus \check{B}_2)$$

$$B = (B_1, B_2)$$

Bi-phase structuring element



B

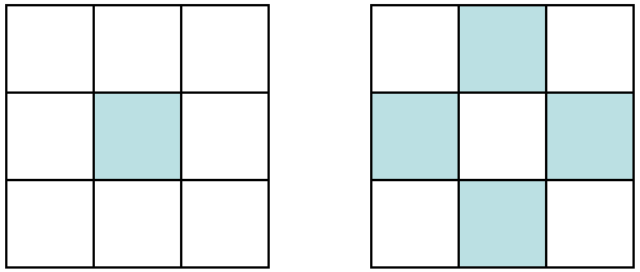
B_1
"Hit" part
(white)

B_2
"Miss" part
(black)

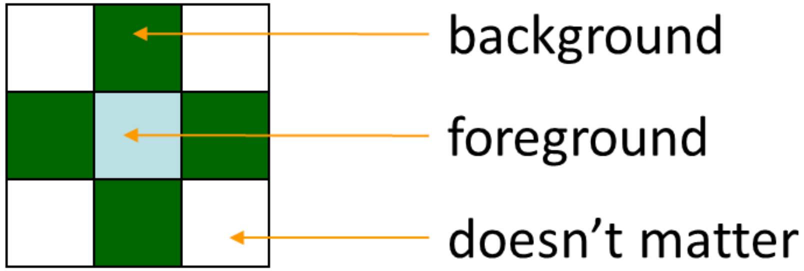
Binary Morphology: Hit-or-miss

Looks for pixel configurations :

$$\{x \mid (B_1)_x \subseteq X, (B_2)_x \subseteq X^c\}$$

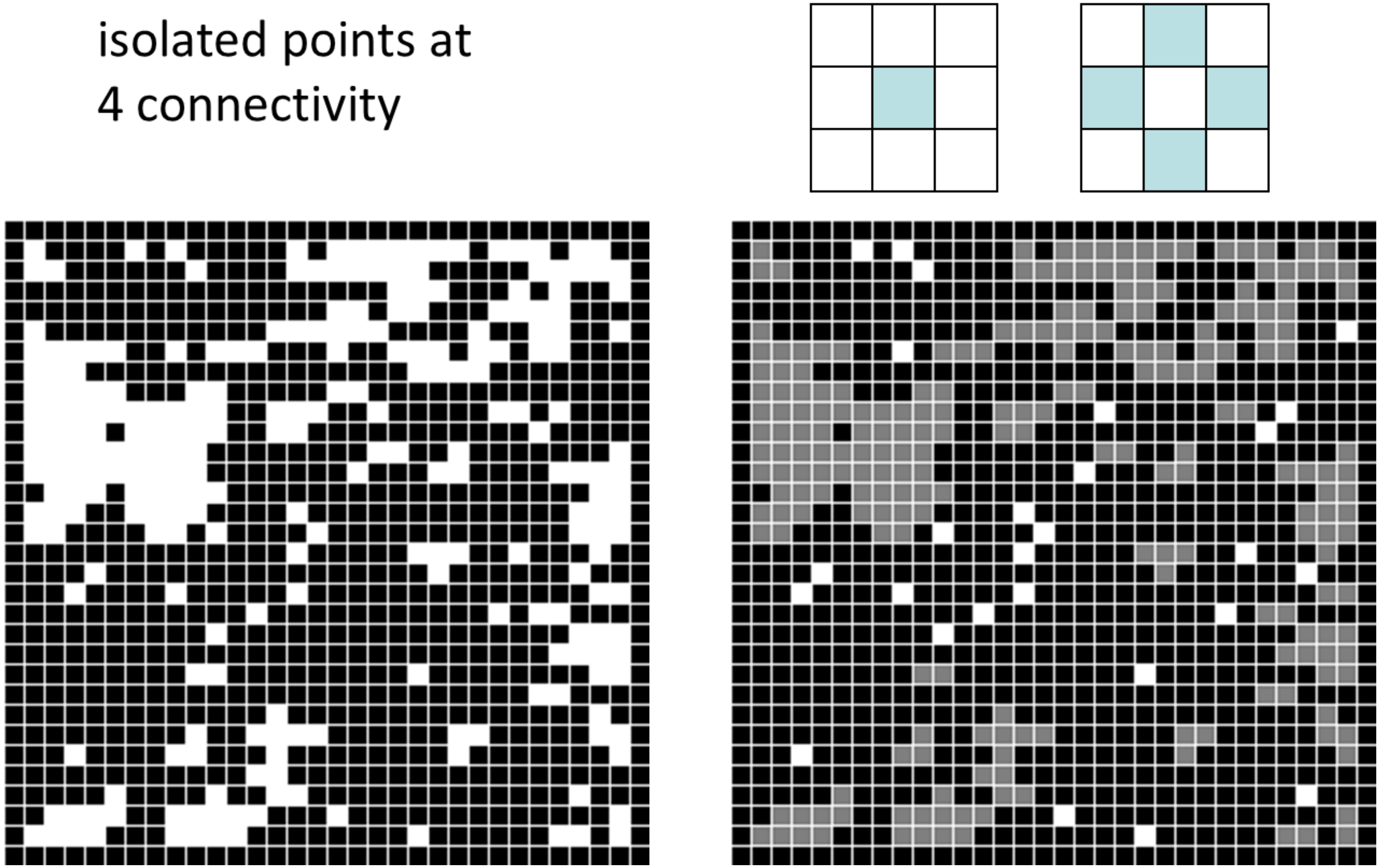


B_1 B_2



Binary Morphology: Hit-or-miss

isolated points at
4 connectivity



Binary Morphology: Hit-or-miss

Thinning : $$X \bigcirc B = X \setminus (X \otimes B)$$

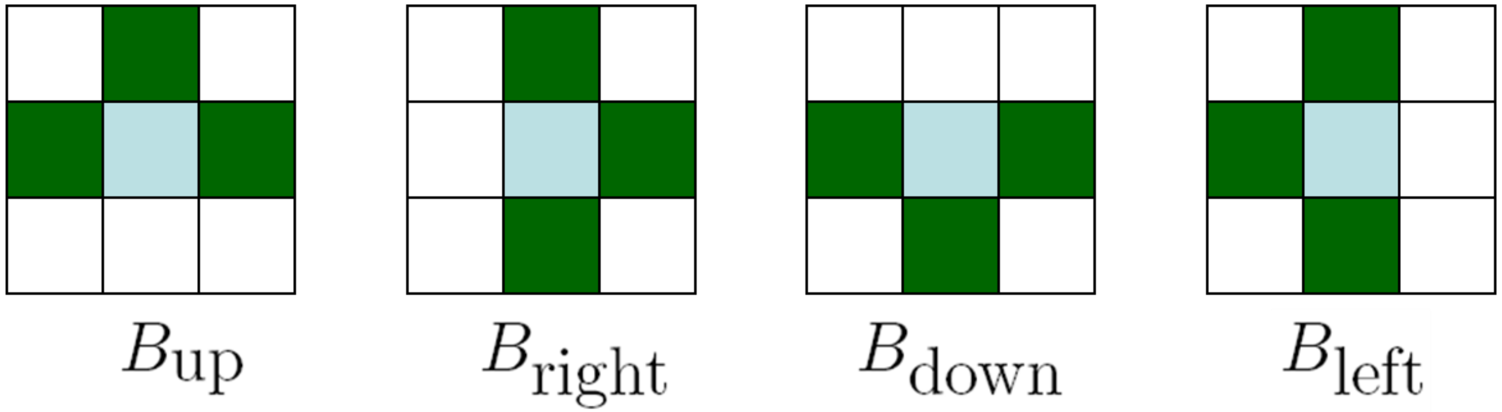
Thickenning : $$X \odot B = X \cup (X \otimes B)$$

Depending on the structuring elements (actually, series of them), very different results can be achieved :

- Prunning
- Skeletons
- Zone of influence
- Convex hull
- …

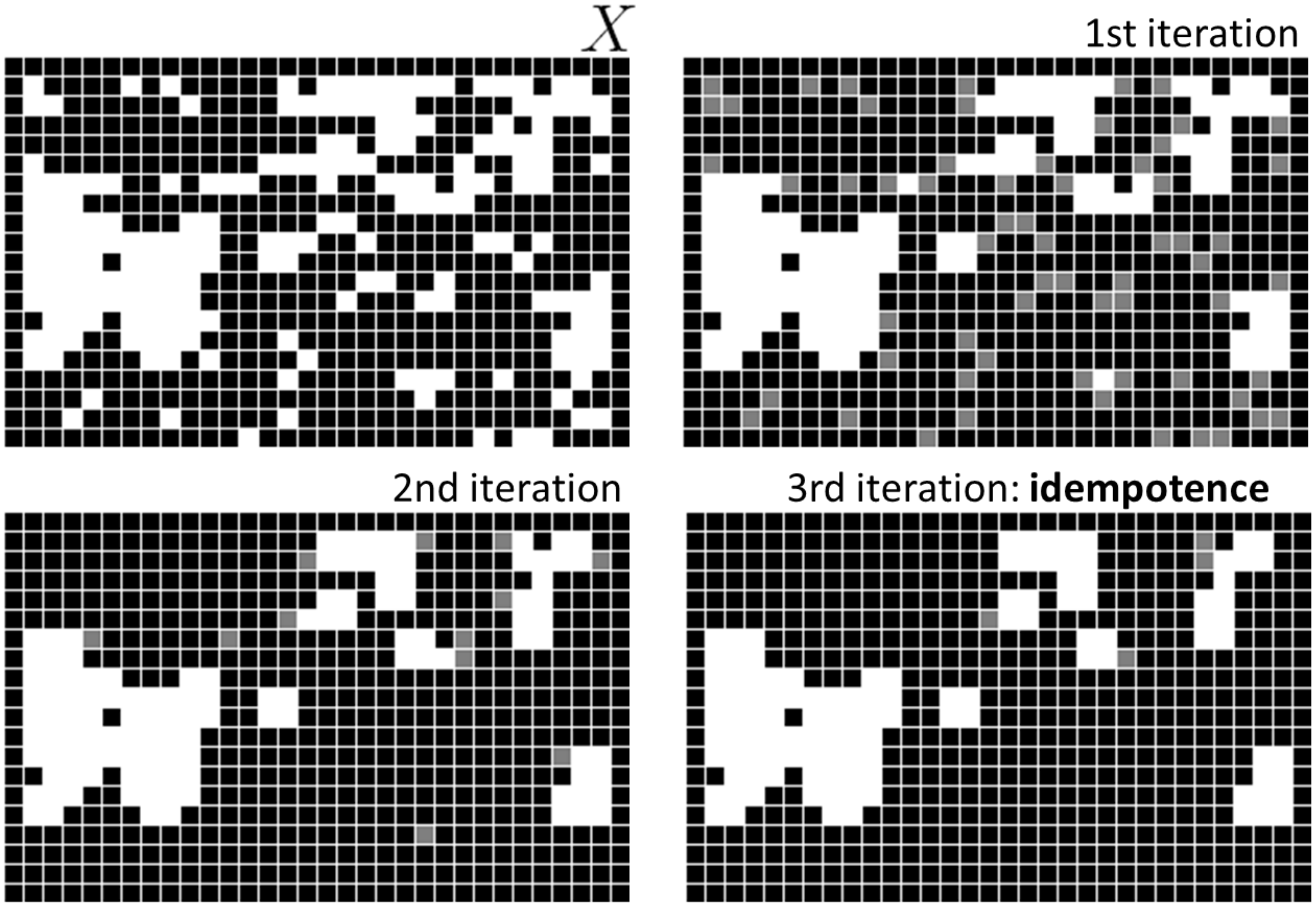
Binary Morphology: Hit-or-miss

Prunning at 4 connectivity : remove end points by a *sequence* of thinnings



1 iteration = $(((X \bigcirc B_{\text{up}}) \bigcirc B_{\text{right}}) \bigcirc B_{\text{down}}) \bigcirc B_{\text{left}}$

Binary Morphology: Hit-or-miss



Fundamentals of Computer Vision

Unit 5: Non-Linear Filtering

Jorge Bernal (Jorge.Bernal@uab.cat)