Digital Image Processing

Lecture #4

Ming-Sui (Amy) Lee

Morphological
Image Processing

Morphological Processing

- **Morphology**
  - **Morpho-: shape/form/structure**
  - **-ology: study**
- **Morphological image processing**
  - **Post-processing**
  - **Binary images → gray-level image**

F → **Morphological Processing** → G

Morphological Processing

- **For some applications**
  - ○ **Structures of objects composed by lines or arcs**
  - ○ **Care about the pattern connectivity**
  - ○ **Independent of width**
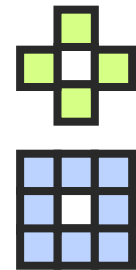

Hand-written characters

Fingerprint patterns

5

Morphological Processing

- **Binary image connectivity**
  - **Pixel bond**
    - Specify the connectivity of a pixel with its neighbors
    - **Four-connected** neighbor → bond = 2
    - **Eight-connected** neighbor → bond = 1

  - **Minimally connected**
    - Elimination of any black pixel (except boundary pixels) results in disconnection of the remaining black pixels

Morphological Processing

- **Binary hit or miss transformations**
  - Select a nxn **hit pattern** (odd-sized mask)
  - Compare with a nxn image window
    - **Match** → hit → change the central pixel value
    - Otherwise → miss → do nothing

  - **Example**
    - To clean the isolated binary noise

      0   0   0

      0   1   0   Hit or miss?

      0   0   0

Morphological Processing

- **Binary hit or miss transformations**
  - 0 → background
  - 1 → object

  $$\begin{array}{ccc} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{array}$$
  Hit or miss?

  - ○ **Logical expression**

  $$\begin{bmatrix} X_3 & X_2 & X_1 \\ X_4 & X & X_0 \\ X_5 & X_6 & X_7 \end{bmatrix} \qquad G(j,k) = X \cap (X_0 \cup X_1 \cup \cdots \cup X_7)$$

  - **Example**
    - ○ If $G(j,k) = X \cap 1$ → do nothing
    - ○ If $G(j,k) = X \cap 0$
      - If X=0 → G(j,k)=0 → do nothing
      - If X=1 → hit → G(j,k)=0

8

Morphological Processing

- **Binary hit or miss transformations**

$$G(j,k) = X \cap (X_0 \cup X_1 \cup \cdots \cup X_7)$$

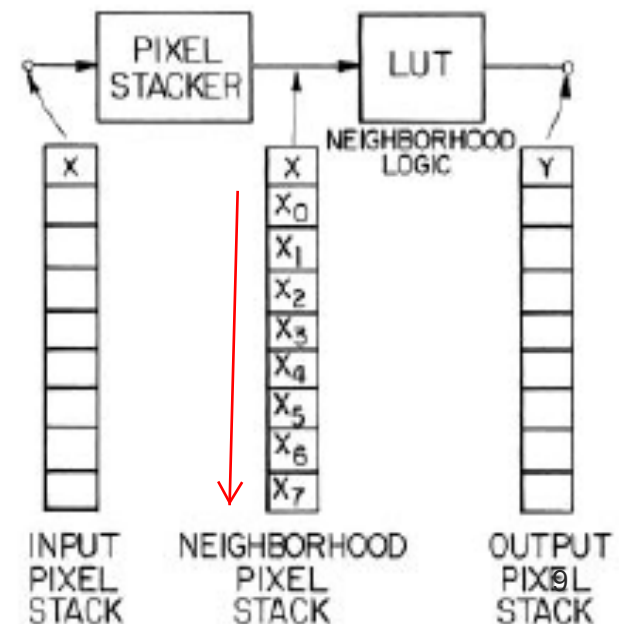$$\Longrightarrow 2^9 \text{ possible mask patterns}$$

- Implementation
  - **Pixel stack**
    - Treat the **8 neighboring** pixels as a "byte"

$$
\begin{bmatrix}
X_3 & X_2 & X_1 \\
X_4 & X & X_0 \\
X_5 & X_6 & X_7
\end{bmatrix}
\otimes
\begin{bmatrix}
2^{-4} & 2^{-3} & 2^{-2} \\
2^{-5} & 2^0 & 2^{-1} \\
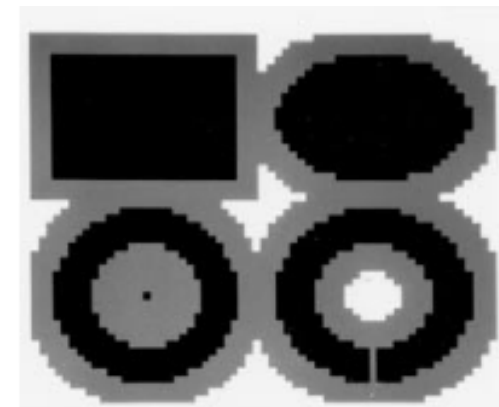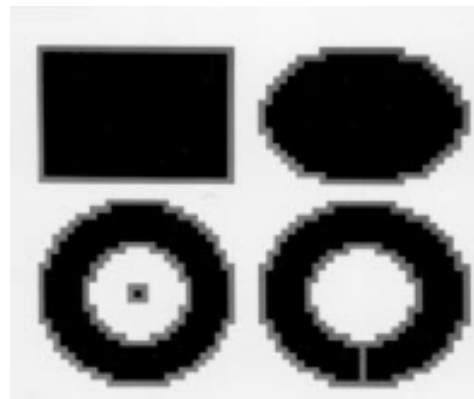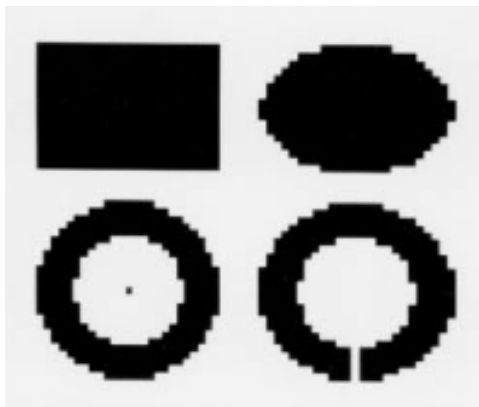2^{-6} & 2^{-7} & 2^{-8}
\end{bmatrix}
$$

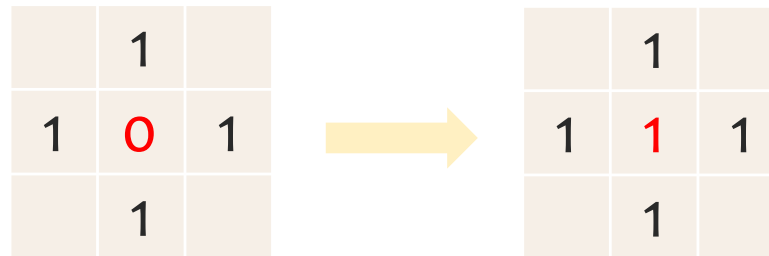 - **Look-Up-Table (LUT)**

Morphological Processing

- **Simple morphological processing based on binary hit or miss rules**
  - Additive operators (0→1)
    - Interior fill
    - Diagonal fill
    - Bridge
    - 8-neighbor dilate

Morphological Processing
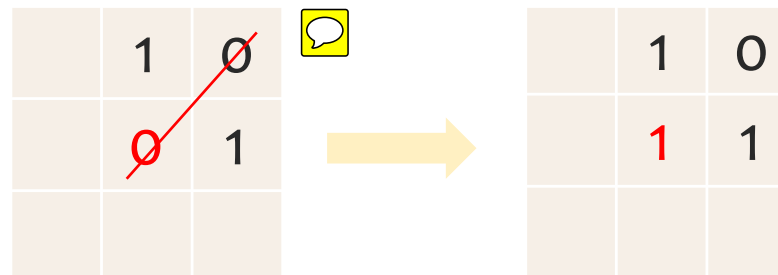
○ **Interior fill**
- Create a pixel if all four-connected neighbor pixels are white

	1	
1	0	1
	1	

➡

	1	
1	1	1
	1	

○ **Diagonal fill**
- Create a pixel if creation eliminates the eight-connectivity of the background

1	0	
0	1	

➡

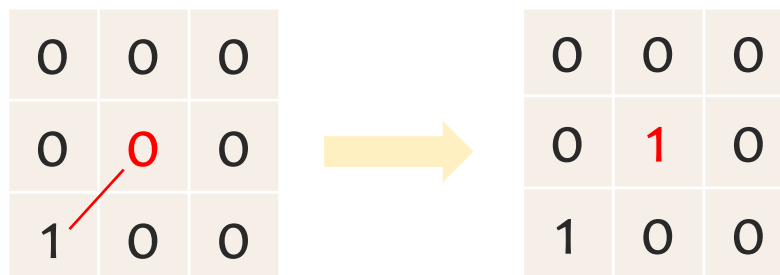| 1	0	
	1	1

Morphological Processing

○ **Bridge**
- Create a white pixel if creation results in connectivity of previously unconnected neighboring white pixels

1	0	0
0	0	1
0	0	0

→

1	0	0
0	1	1
0	0	0
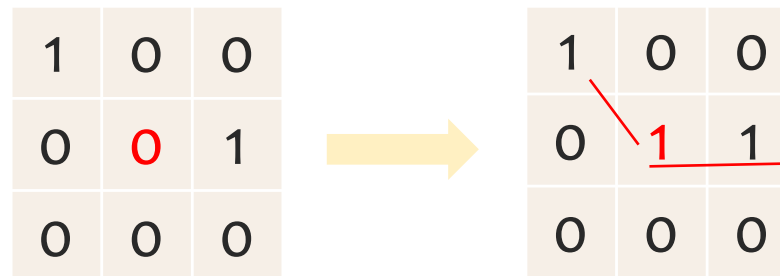
○ **8-neighbor dilate**
- Create a black pixel if at least one eight-connected neighbor pixel is white

0	0	0
0	0	0
1	0	0

→

0	0	0
0	1	0
1	0	0

12

Morphological Processing

- **Simple morphological processing based on binary hit or miss rules**
  - **Subtractive** operators (1→0)
    - **Isolated** pixel removal
    - **Spur** removal
    - **Interior** pixel removal
    - H-break / Eight-neighbor **erode**
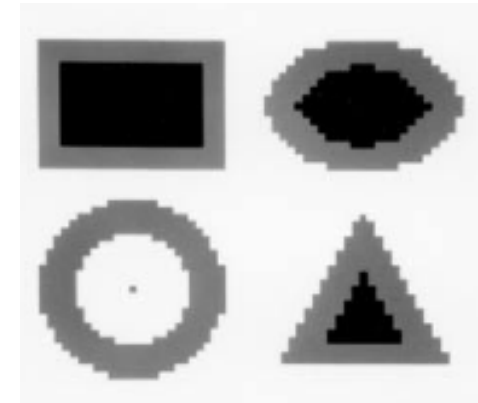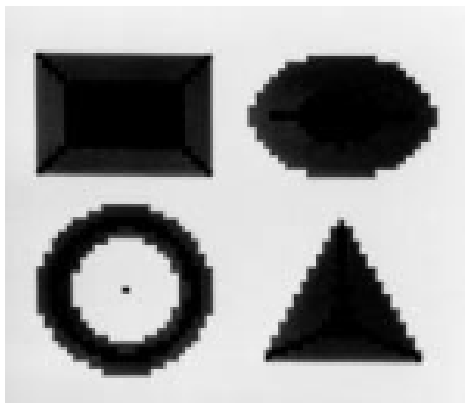
Morphological Processing

- **Isolated pixel removal**
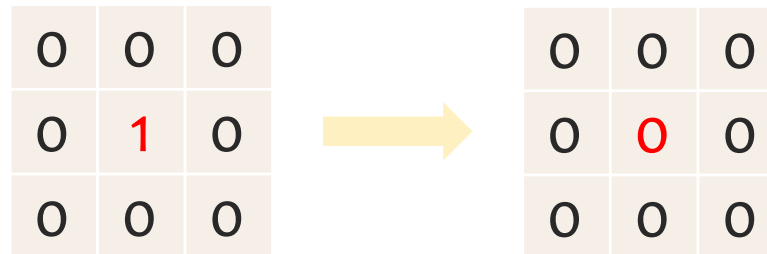 - Erase a white pixel with eight black neighbors

0	0	0
0	1	0
0	0	0

→

0	0	0
0	0	0
0	0	0

- **Spur removal**
  - Erase a white pixel with a single eight-connected neighbor

1	0	0
0	1	0
0	0	0

→

1	0	0
0	0	0
0	0	0
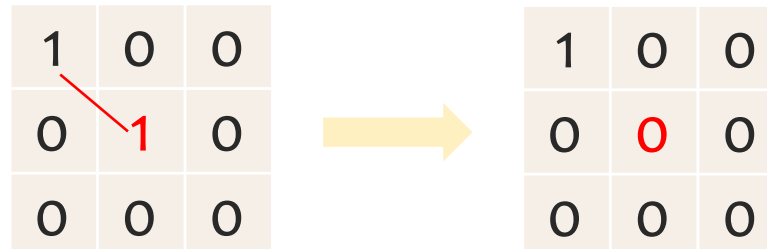
Morphological Processing

- **Interior pixel removal**
  - Erase a white pixel if all four-connected neighbors are white

	1	
1	1	1
	1	

→

0	1	0
1	0	1
0	1	0
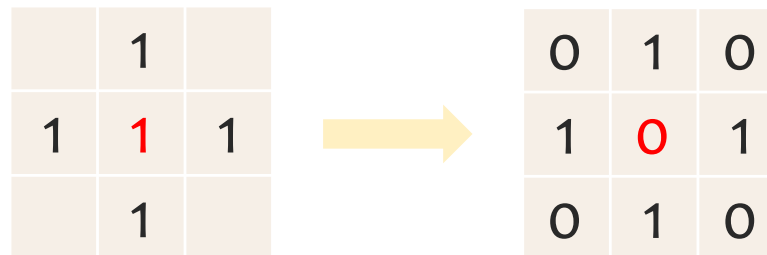
- **H-break**
  - Erase a white pixel that is H-connected

1	1	1
0	1	0
1	1	1

→

1	1	1
0	0	0
1	1	1
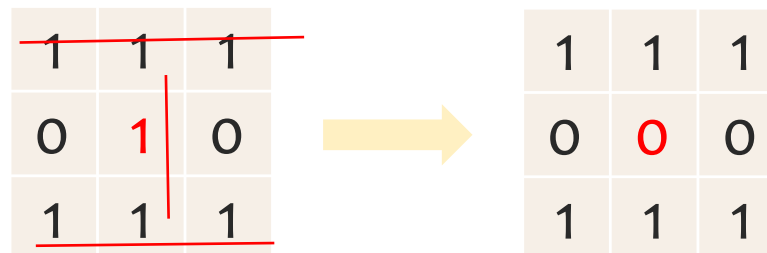
Morphological Processing

- **Eight-neighbor erode**
  - Erase a white pixel if at least one eight-connected neighbor pixel is black
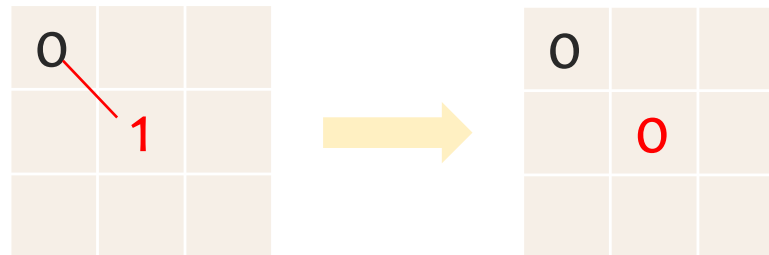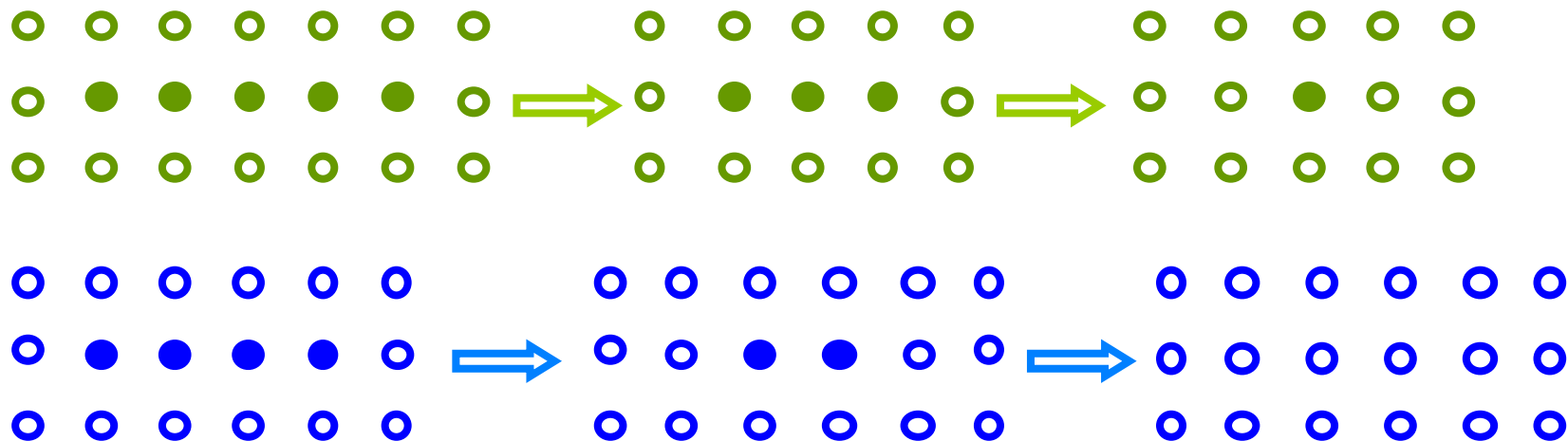
Morphological Processing

- **Example**
  - ○ **Subtractive operator**

- doesn't prevent total erasure and ensure connectivity
- In this case, only a 3x3 window does not sufficient to tell whether the final stage of iteration is reached or not

Morphological Processing

- **Solutions**
  - **Approach I**
    - **Apply a filter ==with larger size==**
      - "fairly complicated patterns", "many combinations"
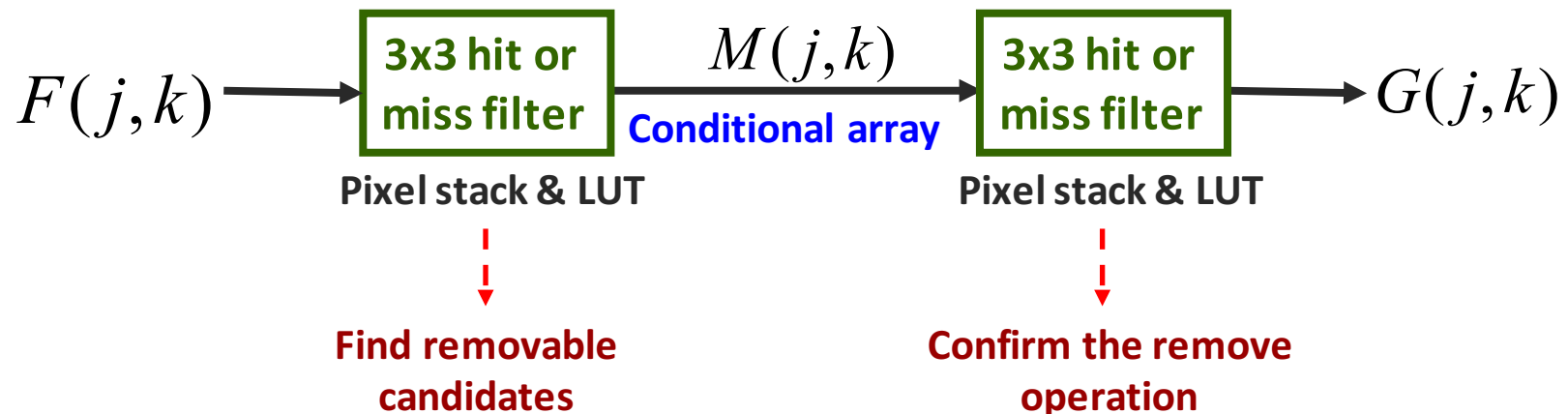
  - **Approach II**
    - **Consider a ==structural (composite) design== with 3x3 filters: ==two-stage approach==**
      - Application dependent
      - ==Thinning, shrinking, skeletonizing==
      - Share the same structure but vary in some modular details

Morphological Processing

- **Advanced morphological processing**
  - **Shrinking/Thinning/Skeletonizing**
    - **Conditional erosion**
    - **Prevent total erasure & Ensure connectivity**

$$F(j,k) \longrightarrow \boxed{\begin{array}{c}\textbf{3x3 hit or}\\ \textbf{miss filter}\end{array}} \xrightarrow{M(j,k)} \boxed{\begin{array}{c}\textbf{3x3 hit or}\\ \textbf{miss filter}\end{array}} \longrightarrow G(j,k)$$

Conditional array

Pixel stack & LUT   **Pixel stack & LUT**

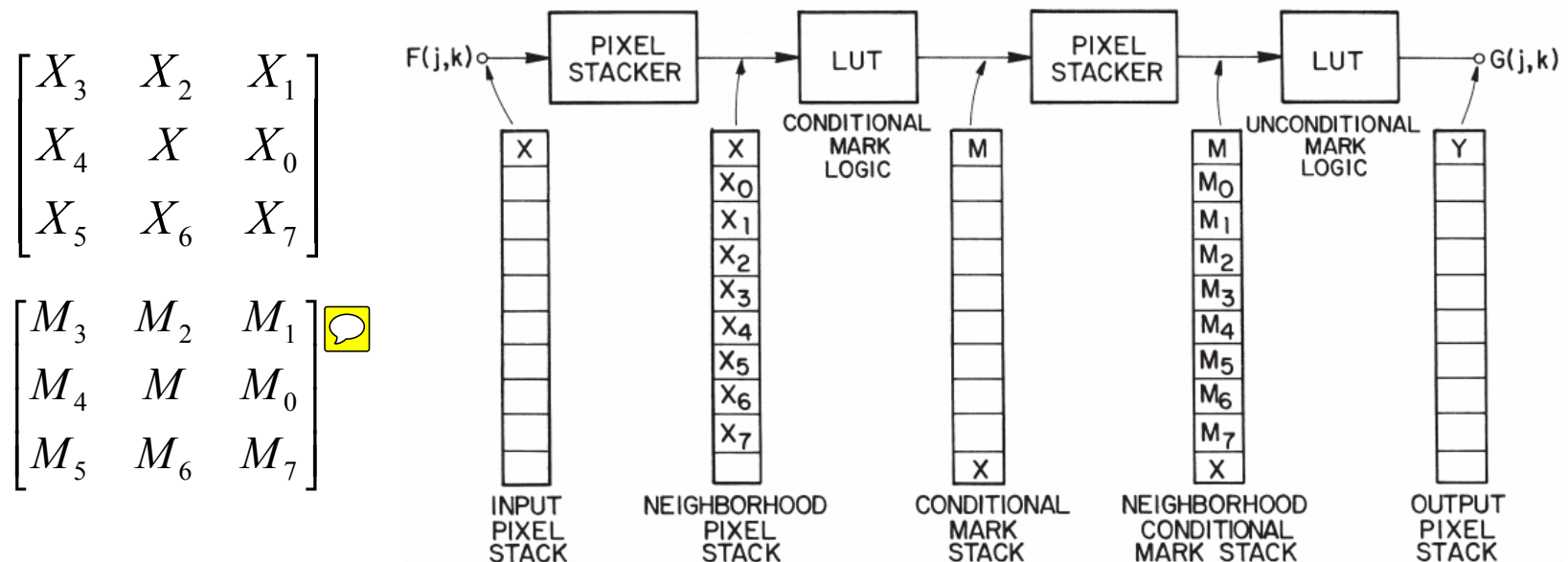**Find removable candidates**   **Confirm the remove operation**

Morphological Processing

- **Advanced morphological processing**
  - **Shrinking/Thinning/Skeletonizing**
    - **Conditional erosion**
    - **Prevent total erasure & Ensure connectivity**

$$\begin{bmatrix} X_3 & X_2 & X_1 \\ X_4 & X & X_0 \\ X_5 & X_6 & X_7 \end{bmatrix}$$

$$\begin{bmatrix} M_3 & M_2 & M_1 \\ M_4 & M & M_0 \\ M_5 & M_6 & M_7 \end{bmatrix}$$

Morphological Processing

- **Shrinking/Thinning/Skeletonizing**
  - **Stage I**
    - **Generate a binary image M(j,k) called the conditional array (or mask)**
      - If M(j,k)=1, it means (j,k) is a candidate for erasure
      - If M(j,k)=0, it means no further operation is needed on (j,k)
  - **Stage II**
    - **Based on the center pixel, X, and M(j,k) pattern, we decide whether to erase X or not in the output G(j,k)**
      - If there's a hit → do nothing
      - If there's a miss → erase the center pixel

Morphological Processing

- Stage I → Part of Table 14.3-1

TABLE 14.3-1. Shrink, Thin and Skeletonize Conditional Mark Patterns [$M = 1$ if hit]

Table	Bond	Pattern							
S	1	0 0 1 0 1 0 0 0 0	1 0 0 0 1 0 0 0 0	0 0 0 0 1 0 1 0 0	0 0 0 0 1 0 0 0 1				
S	2	0 0 0 0 1 1 0 0 0	0 1 0 0 1 0 0 0 0	0 0 0 1 1 0 0 0 0	0 0 0 0 1 0 0 1 0				
S	3	0 0 1 0 1 1 0 0 0	0 1 1 0 1 0 0 0 0	1 1 0 0 1 0 0 0 0	1 0 0 1 1 0 0 0 0	0 0 0 1 1 0 1 0 0	0 0 0 0 1 0 1 1 0	0 0 0 0 1 0 0 1 1	0 0 0 0 1 1 0 0 1
TK	4	0 1 0 0 1 1 0 0 0	0 1 0 1 1 0 0 0 0	0 0 0 1 1 0 0 1 0	0 0 0 0 1 1 0 1 0				
STK	4	0 0 1 0 1 1 0 0 1	1 1 1 0 1 0 0 0 0	1 0 0 1 1 0 1 0 0	0 0 0 0 1 0 1 1 1				

Bond: classification, narrow down the search space
Pattern: coded as an 8-bit symbol for a filter

$$\begin{bmatrix} X_3 & X_2 & X_1 \\ X_4 & X & X_0 \\ X_5 & X_6 & X_7 \end{bmatrix} \otimes \begin{bmatrix} 2^{-4} & 2^{-3} & 2^{-2} \\ 2^{-5} & 2^0 & 2^{-1} \\ 2^{-6} & 2^{-7} & 2^{-8} \end{bmatrix}$$

22

Morphological Processing

- Stage II → Part of Table 14.3-2

TABLE 14.3-2. Shrink and Thin Unconditional Mark Patterns
$[P(M, M_0, M_1, M_2, M_3, M_4, M_5, M_6, M_7) = 1$ if hit$]^a$

Pattern

Spur **Single 4-connection**

```
0 0 M   M 0 0     0 0 0   0 0 0
0 M 0   0 M 0     0 M 0   0 M M
0 0 0   0 0 0     0 M 0   0 0 0
```

$$G(j,k) = X \cap [\overline{M} \cup P(M, M_1, \ldots, M_7)]$$

where $P(M, M_1, \ldots, M_7)$ is an erasure inhibiting logical variable

L Cluster

```
0 0 M   0 M M   M M 0   M 0 0   0 0 0   0 0 0   0 0 0   0 0 0
0 M M   0 M 0   0 M 0   M M 0   M M 0   0 M 0   0 M 0   0 M M
0 0 0   0 0 0   0 0 0   0 0 0   M 0 0   M M 0   0 M M   0 0 M
```

4-Connected offset

```
0 M M   M M 0   0 M 0   0 0 M
M M 0   0 M M   0 M M   0 M M
0 0 0   0 0 0   0 0 M   0 M 0
```

$$\begin{bmatrix} M_3 & M_2 & M_1 \\ M_4 & M & M_0 \\ M_5 & M_6 & M_7 \end{bmatrix} \otimes \begin{bmatrix} 2^{-4} & 2^{-3} & 2^{-2} \\ 2^{-5} & 2^0 & 2^{-1} \\ 2^{-6} & 2^{-7} & 2^{-8} \end{bmatrix}$$

23

Morphological Processing

- **Stage II → Part of Table 14.3-2 (cont'd)**

Spur corner cluster

$$0\ A\ M \quad M\ B\ 0 \quad 0\ 0\ M \quad M\ 0\ 0$$
$$0\ M\ B \quad A\ M\ 0 \quad A\ M\ 0 \quad 0\ M\ B$$
$$M\ 0\ 0 \quad 0\ 0\ M \quad M\ B\ 0 \quad 0\ A\ M$$

Corner cluster

$$M\ M\ D$$
$$M\ M\ D$$
$$D\ D\ D$$

Tee branch

$$D\ M\ 0 \quad 0\ M\ D \quad 0\ 0\ D \quad D\ 0\ 0 \quad D\ M\ D \quad 0\ M\ 0 \quad 0\ M\ 0 \quad D\ M\ D$$
$$M\ M\ M \quad M\ M\ M \quad M\ M\ M \quad M\ M\ M \quad M\ M\ 0 \quad M\ M\ 0 \quad 0\ M\ M \quad 0\ M\ M$$
$$D\ 0\ 0 \quad 0\ 0\ D \quad 0\ M\ D \quad D\ M\ 0 \quad 0\ M\ 0 \quad D\ M\ D \quad D\ M\ D \quad 0\ M\ 0$$

$$A \cup B \cup C = 1, \quad D = 0 \cup 1, \quad A \cup B = 1$$

Morphological Processing

- Stage II → Part of Table 14.3-3

TABLE 14.3-3. Skeletonize Unconditional Mark Patterns
$[P(M, M_0, M_1, \underline{M_2, M_3, M_4}, \underline{M_5, M_6, M_7}) = 1 \text{ if hit}]^a$ $\qquad A \cup B \cup C = 1, \quad D = 0 \cup 1$

Pattern

Spur

0	0	0		0	0	0		0	0	M		M	0	0
0	M	0		0	M	0		0	M	0		0	M	0
0	0	M		M	0	0		0	0	0		0	0	0

Single 4-connection

0	0	0		0	0	0		0	0	0		0	M	0
0	M	0		0	M	M		M	M	0		0	M	0
0	M	0		0	0	0		0	0	0		0	0	0

L corner

0	M	0		0	M	0		0	0	0		0	0	0
0	M	M		M	M	0		0	M	M		M	M	0
0	0	0		0	0	0		0	M	0		0	M	0

Morphological Processing

- **Example - shrinking**

```
0   0   0   0   0   0          0   0   0   0   0   0
0   0   1   1   0   0          0   0   M   M   0   0
0   0   0   0   0   0          0   0   0   0   0   0
```
$$F(j,k)$$ $$M(j,k)$$

```
0   0   0   0   0   0          0   0   0   0   0   0
0   0   P   0   0   0          0   0   1   0   0   0
0   0   0   0   0   0          0   0   0   0   0   0
```
$$P(j,k)$$ $$G(j,k)$$

Morphological Processing

- **Example - shrinking**

```
0   0   0   0

0   1   1   0

0   1   1   0

0   0   0   0
```

$$F(j,k)$$

$$M(j,k)$$

$$P(j,k)$$

$$G(j,k)$$

Morphological Processing

- **Shrinking**
  - Erase black pixels such that an object without holes erodes to a single pixel at or near its center of mass, and an object with holes erodes to a connected ring lying midway between each hole and its nearest outer boundary
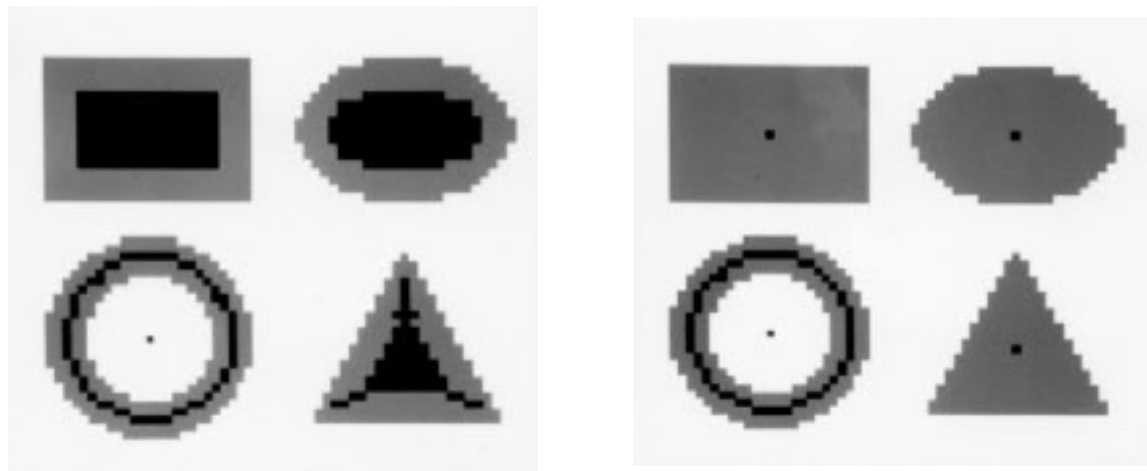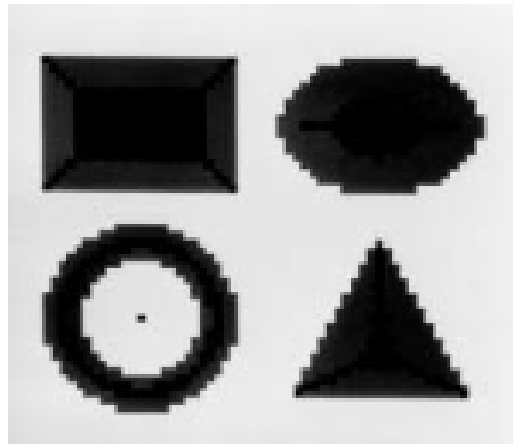
Morphological Processing

- ## Thinning

  - Erase black pixels such that an object **without holes** erodes to a **minimally connected** stroke located **equidistant from its nearest outer boundaries,** and an object with holes erodes to a minimally connected ring midway between each hole and its nearest outer boundary
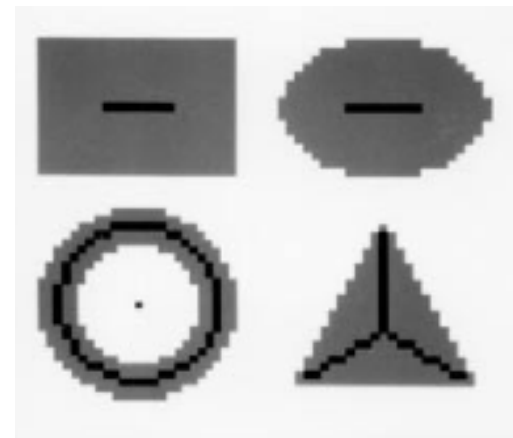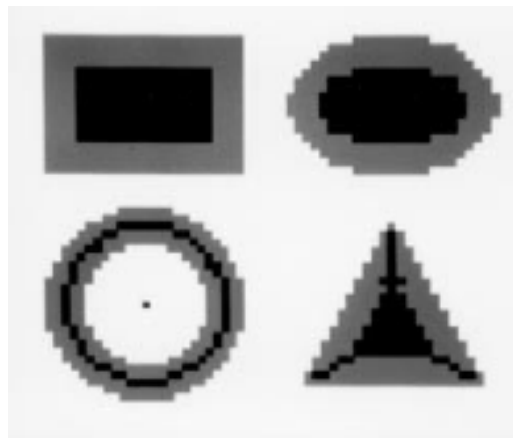
Morphological Processing

- **Skeletonizing**
  - The **medial axis skeleton** consists of the set of points that are **equally distant** from **two closest** points of an object boundary
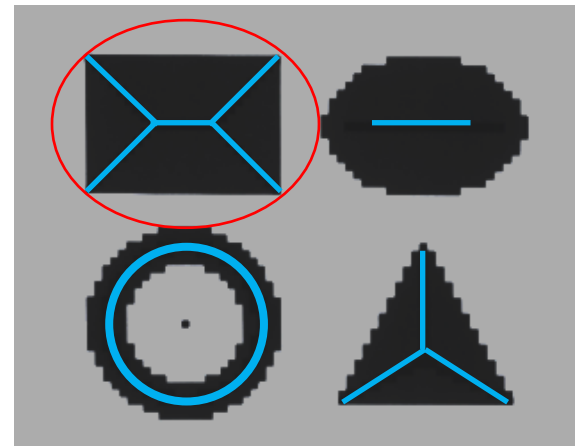
Morphological Processing

- **Algebraic operations on binary arrays**

```
0 0 0 0 0 0        0 0 0 0 0 0        1 1 1 1 1 1
0 0 1 1 0 0        0 0 0 0 0 0        1 1 0 0 1 1
0 0 1 1 0 0        0 1 1 1 1 0        1 1 0 0 1 1
0 0 1 1 0 0        0 1 1 1 1 0        1 1 0 0 1 1
0 0 1 1 0 0        0 0 0 0 0 0        1 1 0 0 1 1
0 0 0 0 0 0        0 0 0 0 0 0        1 1 1 1 1 1
      A                  B                  Ā
                                       complement
```

```
0 0 0 0 0 0        0 0 0 0 0 0        0 0 0 0 0 0
0 0 1 1 0 0        0 0 0 0 0 0        0 0 1 1 0 0
0 1 1 1 1 0        0 0 1 1 0 0        0 1 0 0 1 0
0 1 1 1 1 0        0 0 1 1 0 0        0 1 0 0 1 0
0 0 1 1 0 0        0 0 0 0 0 0        0 0 1 1 0 0
0 0 0 0 0 0        0 0 0 0 0 0        0 0 0 0 0 0
    A ∪ B              A ∩ B            A XOR B
     union           intersection     exclusive-OR
      OR                 AND               XOR
```

31

Morphological Processing

■ **Generalized dilation and erosion**

○ **Reflection and translation of a binary image**

$$F(j,k) \qquad \widetilde{F}(j,k) \qquad T_{1,2}\{F(j,k)\}$$
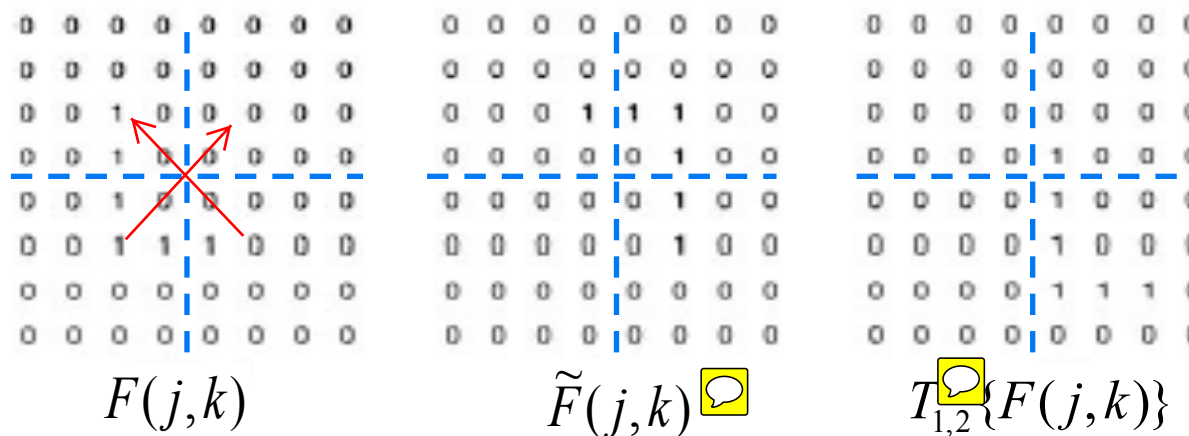
○ **Dilation**

$$G(j,k) = F(j,k) \oplus H(j,k)$$

Structuring element

○ **Erosion**

$$G(j,k) = F(j,k) \ominus H(j,k)$$

32

Morphological Processing

- **Dilation** $G(j,k) = F(j,k) \oplus H(j,k)$

  - Can be implemented in several ways
  - Minkowski addition definition

$$G(j,k) = \bigcup_{(r,c)\in H}\bigcup T_{r,c}\{F(j,k)\}$$

$$G(j,k) = T_{0,0}\{F(j,k)\}\cup T_{0,1}\{F(j,k)\}\cup T_{1,0}\{F(j,k)\}$$

$$\cup T_{1,1}\{F(j,k)\}\cup T_{2,0}\{F(j,k)\}$$

```
0 0 0 0 0        1 1 0
0 0 1 0 0        1 1 0
0 1 1 0 0        1 0 0
0 0 1 1 0
0 0 0 0 0
  F(j,k)          H(j,k)
```

```
0 0 0 0 0     · 0 0 0 0 0                                                    0 0 0 0 0 0 0
0 0 1 0 0     · 0 0 1 0 0     0 0 0 0 0     · 0 0 0 0 0                      0 0 1 1 0 0 0
0 1 1 0 0     · 0 1 1 0 0     0 0 1 0 0     · 0 0 1 0 0     0 0 0 0 0        0 1 1 1 0 0 0
0 0 1 1 0     · 0 0 1 1 0     0 1 1 0 0     · 0 1 1 0 0     0 0 1 0 0        0 1 1 1 1 0 0
0 0 0 0 0     · 0 0 0 0 0     0 0 1 1 0     · 0 0 1 1 0     0 1 1 0 0        0 1 1 1 1 0 0
                              0 0 0 0 0     · 0 0 0 0 0     0 0 1 1 0        0 0 1 1 0 0 0
                                                           0 0 0 0 0        0 0 0 0 0 0 0

$T_{0,0}\{F(j,k)\}$   $T_{0,1}\{F(j,k)\}$   $T_{1,0}\{F(j,k)\}$   $T_{1,1}\{F(j,k)\}$   $T_{2,0}\{F(j,k)\}$          $G(j,k)$
```

Morphological Processing

- **Erosion** $G(j,k) = F(j,k) \ominus H(j,k)$
  - Can be implemented in several ways
  - Dual relationship of Minkowski addition

$$G(j,k) = \bigcap_{(r,c)\in H}\bigcap T_{r,c}\{F(j,k)\}$$

$$
\begin{matrix}
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
\end{matrix}
\ominus
\begin{matrix}
1 & 1 & 1 \\
1 & 0 & 0 \\
1 & 1 & 1 \\
\end{matrix}
=
\begin{matrix}
0 & 0 & 0 \\
1 & 1 & 0 \\
0 & 0 & 0 \\
\end{matrix}
$$

$F(j,k)$ $H(j,k)$ $G(j,k)$

$$
\begin{matrix}
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 \\
\end{matrix}
\ominus
\begin{matrix}
1 & 1 & 1 \\
1 & 0 & 0 \\
1 & 1 & 1 \\
\end{matrix}
=
\begin{matrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
\end{matrix}
$$

$F(j,k)$ $H(j,k)$ $G(j,k)$

//Sternberg definition//

$G(j,k) = \bigcap\bigcap_{(r,c)\in H} T_{r,c}\{F(j,k)\}$

//Serra definition//

$G(j,k) = \bigcap\bigcap_{(r,c)\in \tilde{H}} T_{r,c}\{F(j,k)\}$
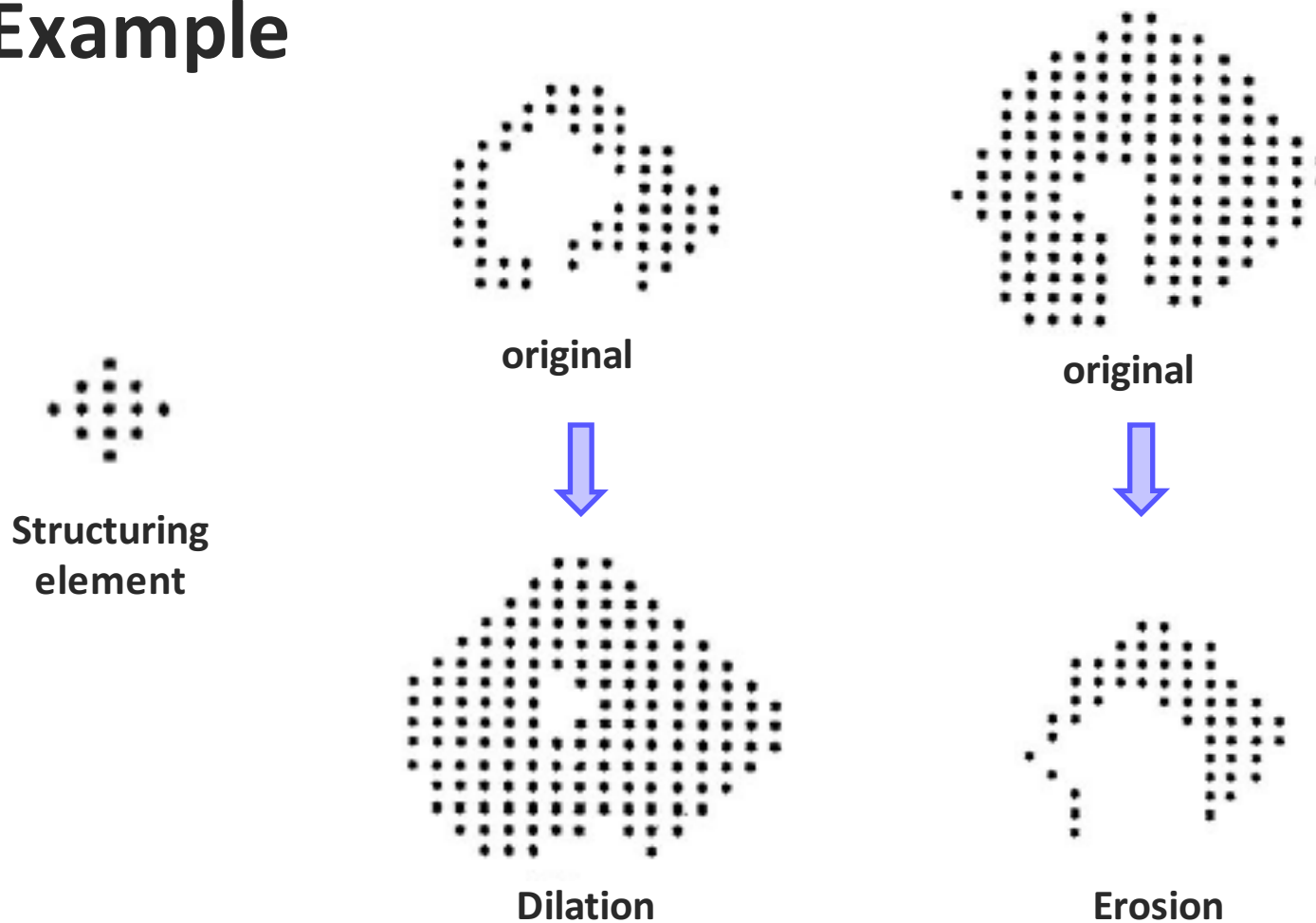
34

Morphological Processing

- **Example**

Structuring element

original

original

Dilation

Erosion

35

Morphological Processing

- **Example**

Morphological Processing

- **Example**

a. Original   b. Binary

c. Erosion   d. Dilation

e. Opening   f. Closing
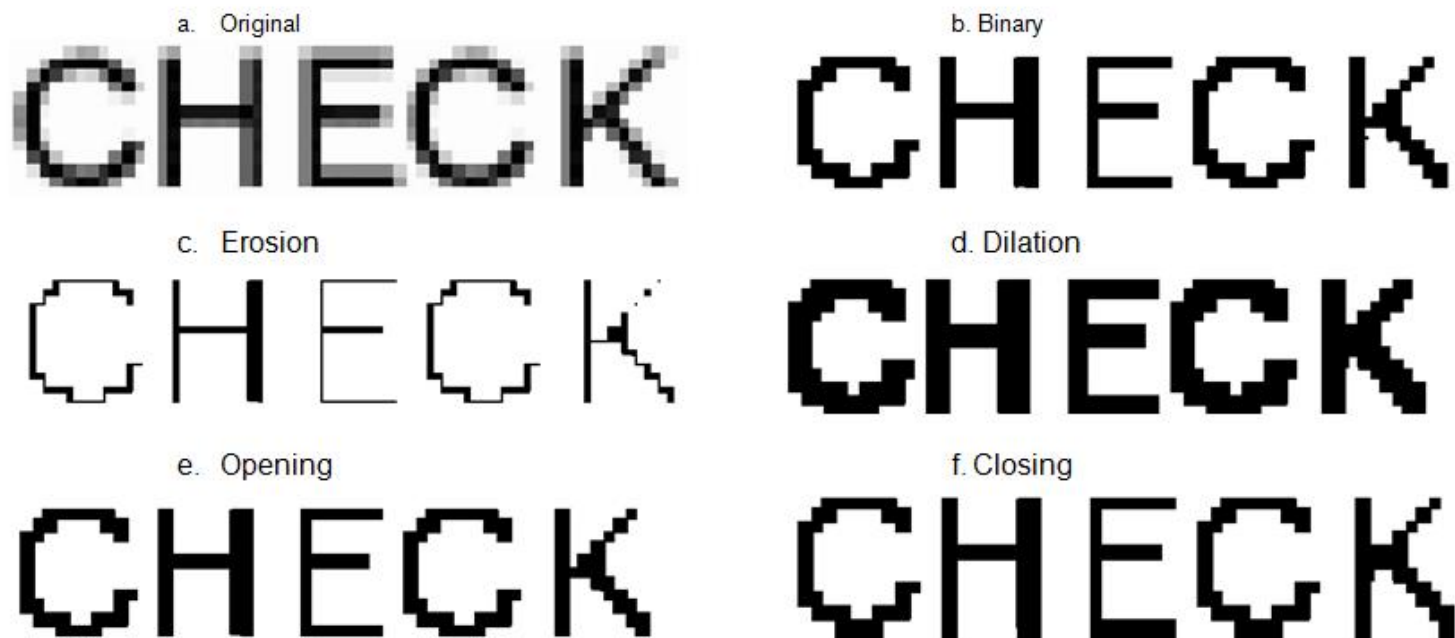
Morphological Processing

- ## **Example**

Original fingerprint	Skeletoninzed fingerprint

The original fingerprint contains ridges with width of several pixels.
The skeletonized fingerprint contains ridges only a single pixel wide.

Morphological Processing

- **Applications**
  - **Boundary Extraction**
    - Extract the boundary (or outline) of an object
  - **Hole Filling**
    - Given a pixel inside a boundary, hole filling attempts to fill that boundary with object pixels
  - **Connected Component Labeling**
    - Scan an image and groups its pixels into components based on pixel connectivity

Morphological Processing

- **Boundary Extraction**

$$\beta(F(j,k)) = F(j,k) - (F(j,k) \Theta H(j,k))$$

F

H — Origin

F Θ H

$\beta(F)$
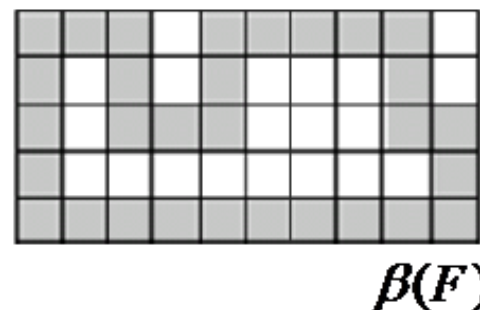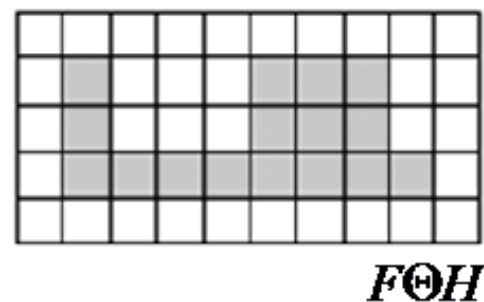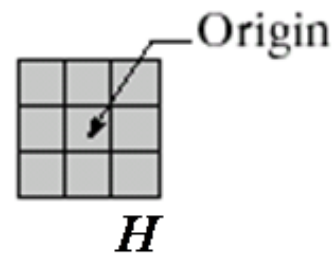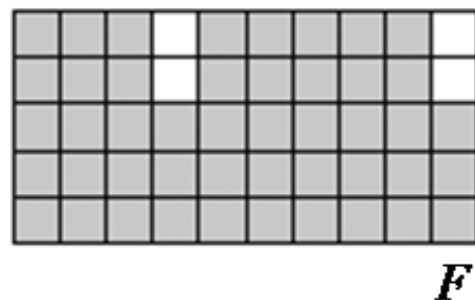
Morphological Processing

- **Example**

Original Image   Extracted Boundary

Morphological Processing

- **Hole Filling**

$$G_i(j,k) = (G_{i-1}(j,k) \oplus H(j,k)) \cap F^c(j,k) \qquad i = 1,2,3...$$

$$G(j,k) = G_i(j,k) \cup F(j,k)$$

42
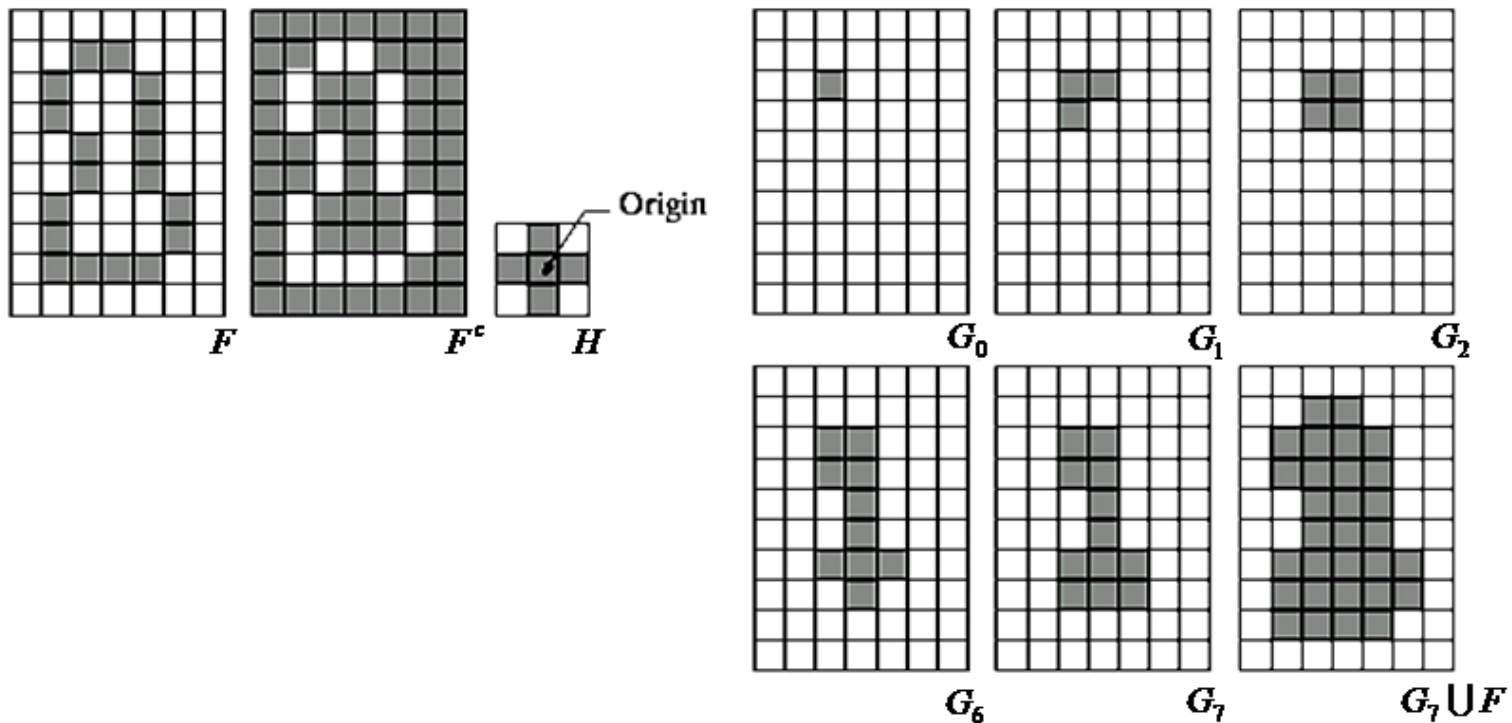
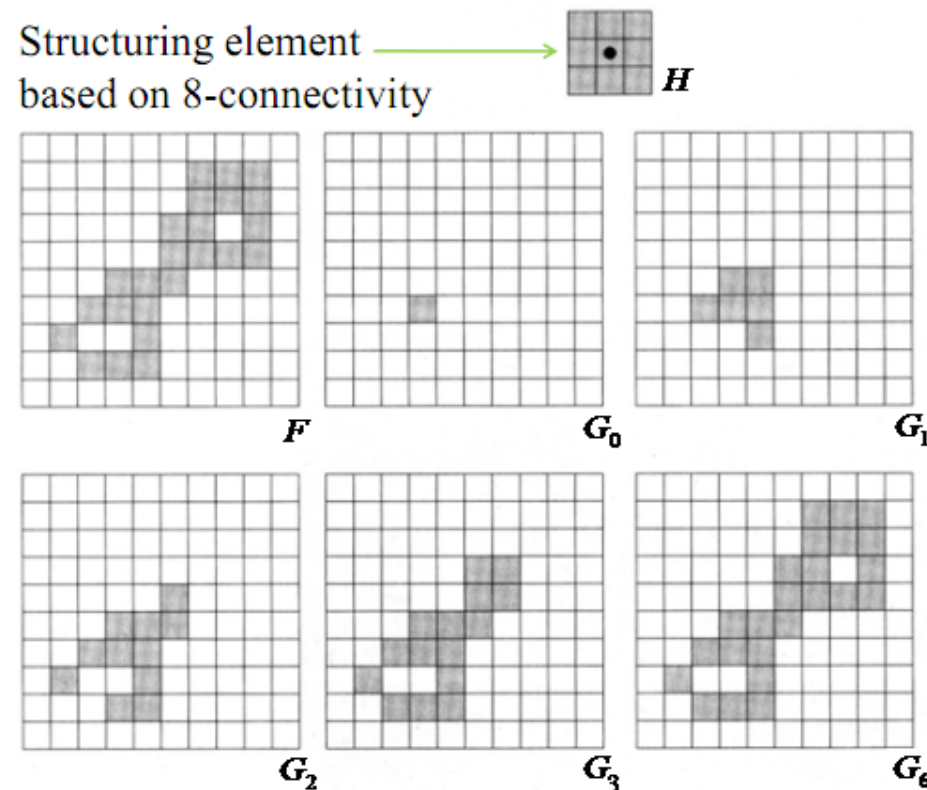# **Morphological Processing**
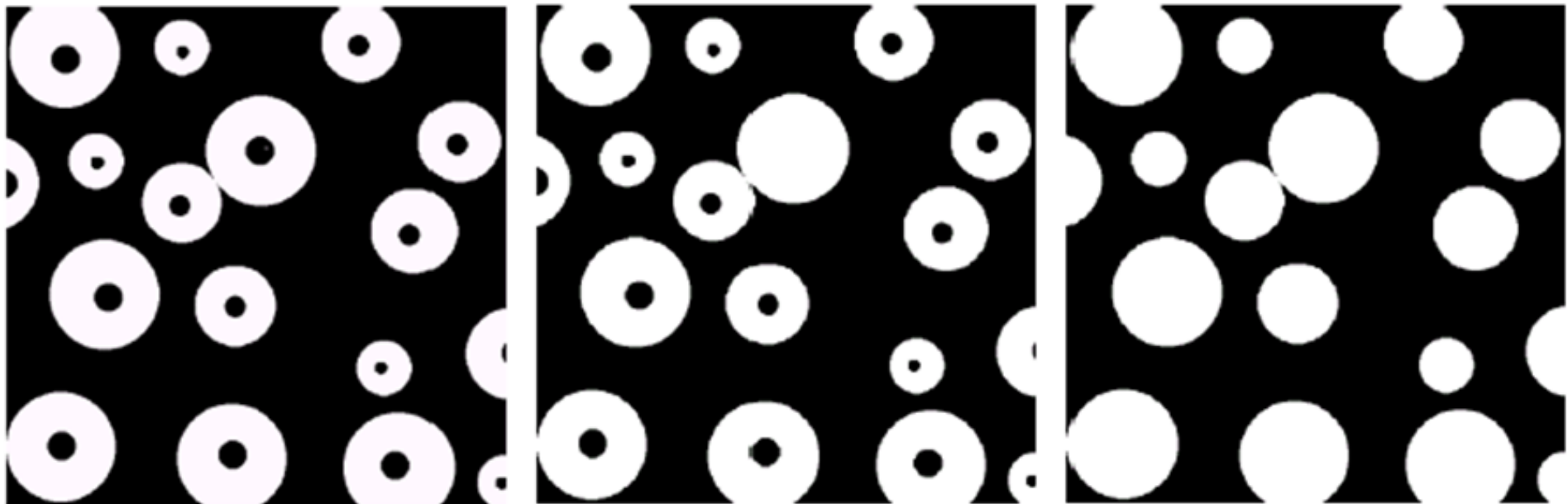
- **Example**

Original Image   One Hole Filled   All Holes Filled

Morphological Processing

- **Connected Component Labeling**

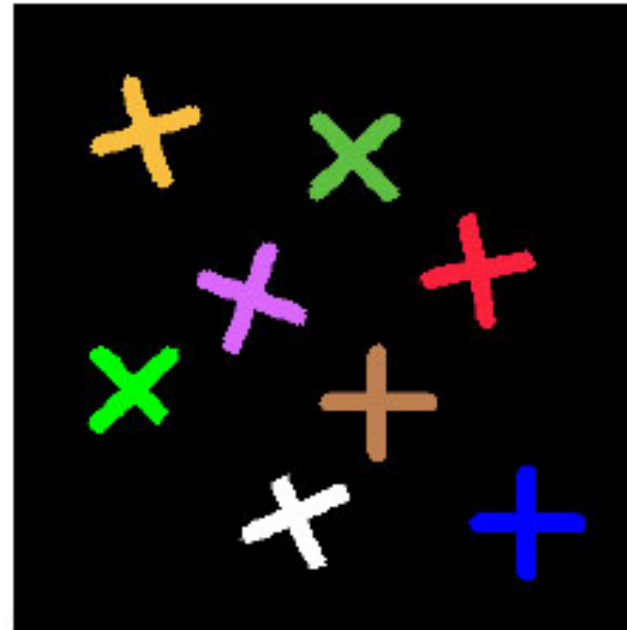$$G_i(j,k) = (G_{i-1}(j,k) \oplus H(j,k)) \cap F(j,k) \quad i = 1,2,3,...$$

Structuring element based on 8-connectivity → H

F G_0 G_1

G_2 G_3 G_6

Morphological Processing

- **Example**
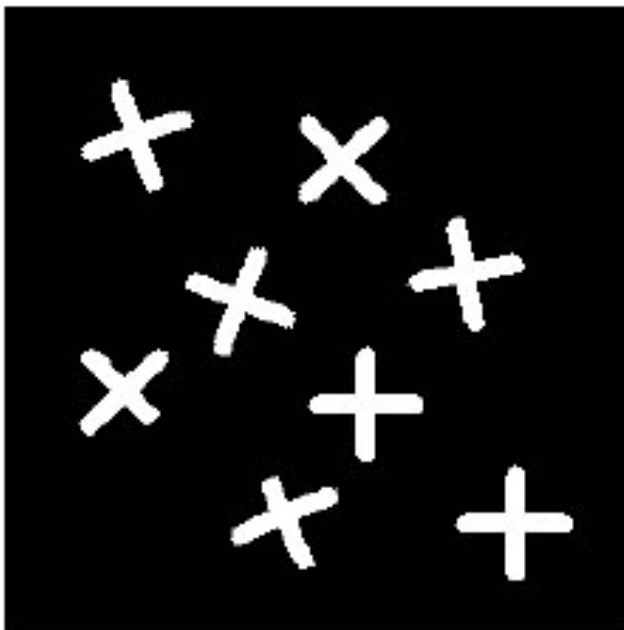
Original Image

Labelled Components

Morphological Processing

- **Applications**
  - **Open operator**

  $$G(j,k) = F(j,k) \circ H(j,k) = [F(j,k) \Theta \tilde{H}(j,k)] \oplus H(j,k)$$

    - **With a compact structuring element**
      - Smoothes contours of objects
      - Eliminates small objects
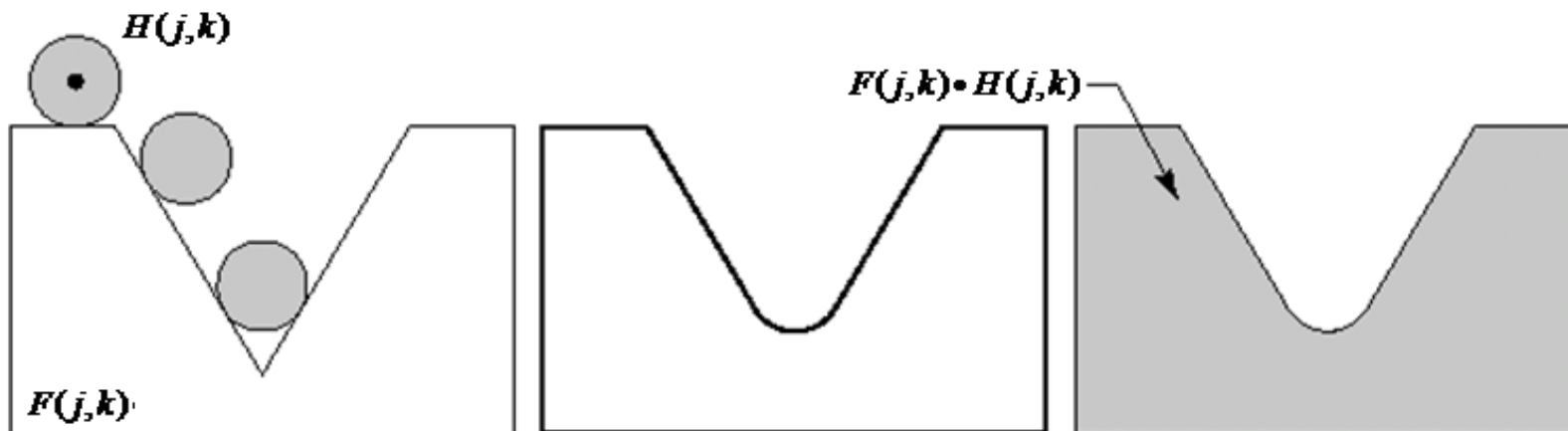   - Breaks narrow strokes
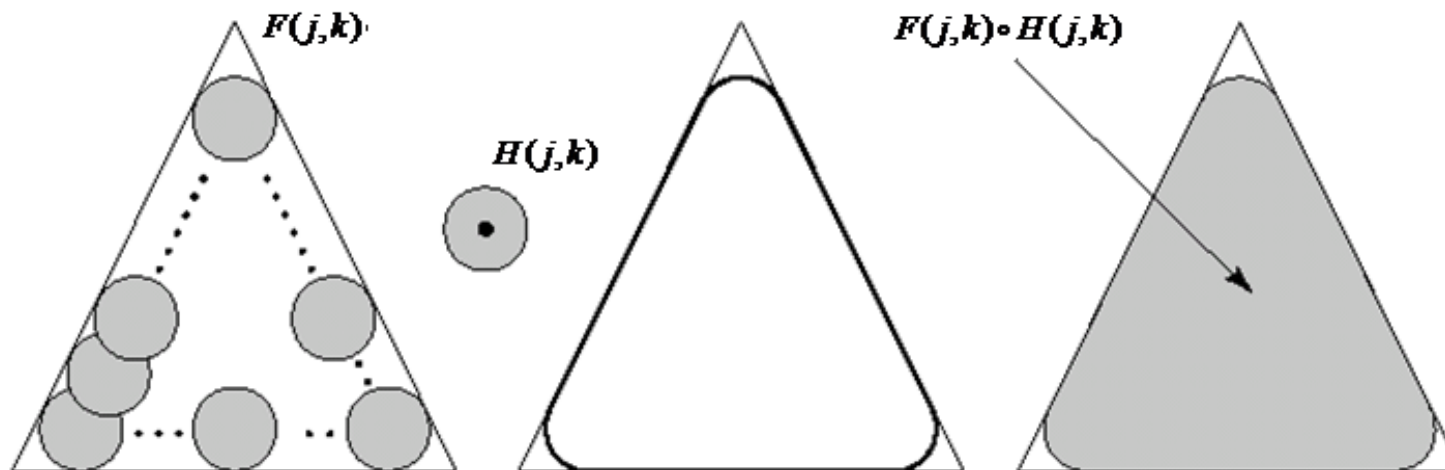
Morphological Processing

- **Applications**
  - ○ **Close operator**

  $$G(j,k) = F(j,k) \bullet H(j,k) = [F(j,k) \oplus H(j,k)] \Theta \widetilde{H}(j,k)$$

  - ■ **With a compact structuring element**
    - ○ **Smoothes contours of objects**
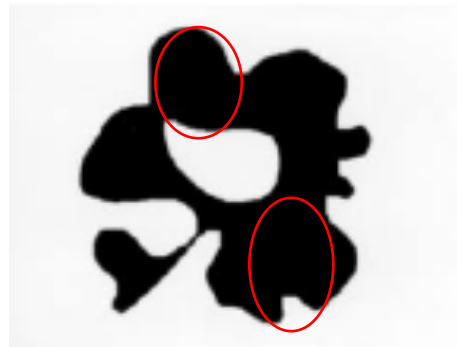   - ○ **Eliminate small holes**
    - ○ **Fuses short gaps between objects**

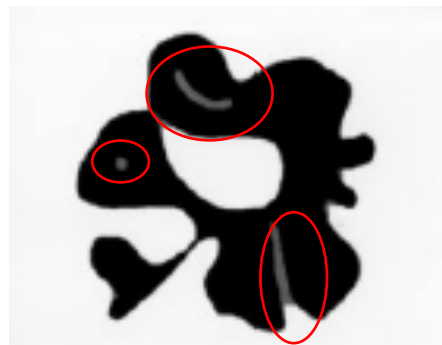# **Morphological Processing**
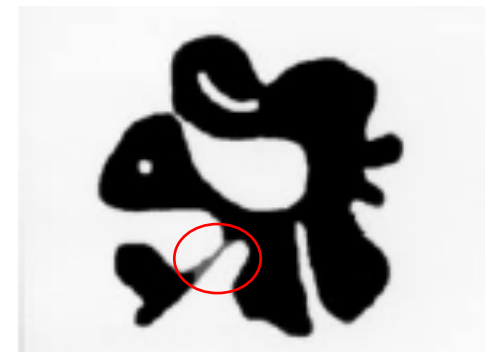
- **Example**

| original | (a) close | (b) open |

Q: repeated openings/closings?

Compare (a) with the original image

Compare (b) with the original image

MCBall

http://www.youtube.com/watch?v=5PvubZiYwOo