

XỬ LÝ ẢNH SỐ

Chương 7. Xử lý ảnh hình thái

❑ Thuật toán xử lý ảnh hình thái được sử dụng để loại bỏ những khiếm khuyết trong bức ảnh sau khi được phân vùng và bổ sung thêm các thông tin bị mất cho bức ảnh.

❑ Các nội dung chính

- Khái niệm xử lý ảnh hình thái (morphology)
- Các phép toán xử lý hình thái đơn giản
- Các phép toán xử lý hình thái ghép
- Các thuật toán xử lý hình thái

Khái niệm xử lý ảnh hình thái

- ❑ Xử lý ảnh hình thái (*morphology*) mô tả một loạt các kỹ thuật xử lý trên hình dạng của các đối tượng trên bức ảnh.
- ❑ Các phép toán hình thái thường được sử dụng để loại bỏ những khiếm khuyết của đối tượng trong quá trình xử lý phân vùng ảnh.



Image after segmentation



Image after segmentation and
morphological processing

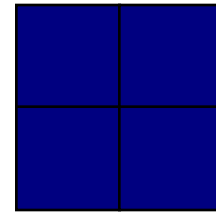
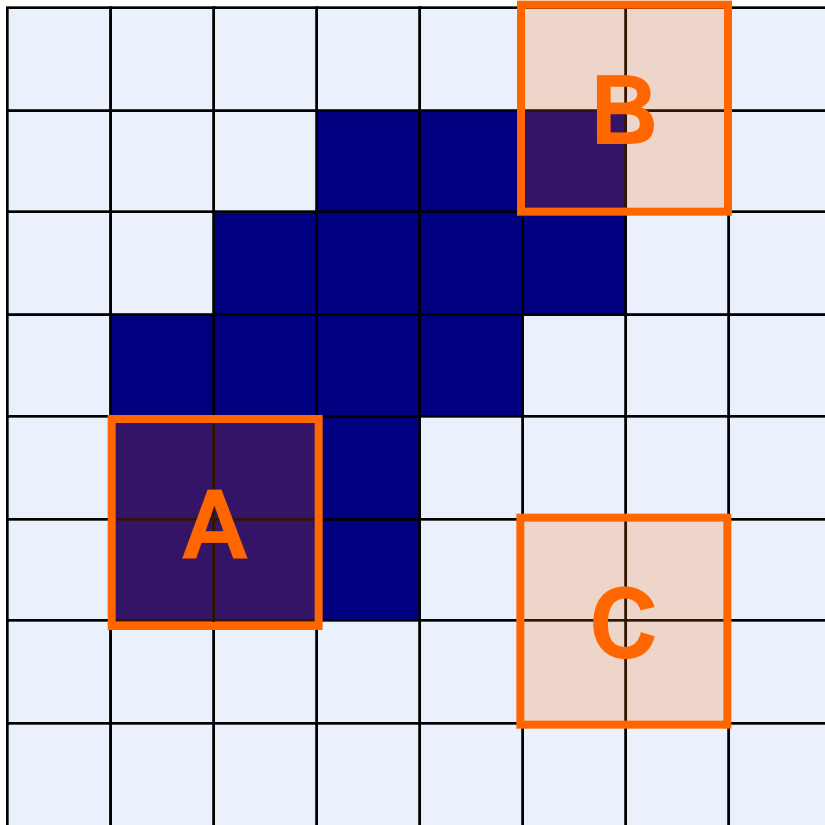
Các phần tử có cấu trúc (*Structuring Elements*)

- ❑ Các SE - phần tử cấu trúc- là các mảng (ma trận) có cấu trúc được tạo bởi 2 mức xám 0 và 1
- ❑ Thông thường SEs có dạng đối xứng với trọng tâm ở giữa.

1	1	1
1	1	1
1	1	1

0	1	0
1	1	1
0	1	0

0	0	1	0	0
0	1	1	1	0
1	1	1	1	1
0	1	1	1	0
0	0	1	0	0



Structuring Element

Fit: Fit xảy ra khi tất cả pixel của SE trùng với các pixel của bức ảnh

Hit: Hit xảy ra khi bất kỳ pixel nào của SE trùng với pixel của ảnh

Tất cả các phép toán xử lý hình thái đều được dựa trên 2 hiện tượng trên!

Fitting & Hitting

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	1	B	1	1	1	0	C	0	0	0
0	1	1	1	1	1	1	1	0	0	0	0
0	1	1	1	1	1	1	1	0	0	0	0
0	0	1	1	1	1	1	1	0	0	0	0
0	0	1	1	1	1	1	1	1	1	0	0
0	0	1	1	1	1	1	A	1	1	1	0
0	0	0	0	0	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0

1	1	1
1	1	1
1	1	1

Structuring
Element 1

0	1	0
1	1	1
0	1	0

Structuring
Element 2

Các phép xử lý hình thái cơ bản

- ❖ Phép xử lý ảnh hình thái về cơ bản tương tự như phép lọc không gian.
- ❖ SE được dịch chuyển qua tất cả các pixel của ảnh gốc để tạo ra bức ảnh mới.
- ❖ Giá trị của các pixel mới phụ thuộc vào phép toán xử lý hình thái
- ❖ Hai phép toán xử lý hình thái cơ bản là: **co** (**erosion**, shrink, reduce) và **giãn** (**dilation**, grow, expand)

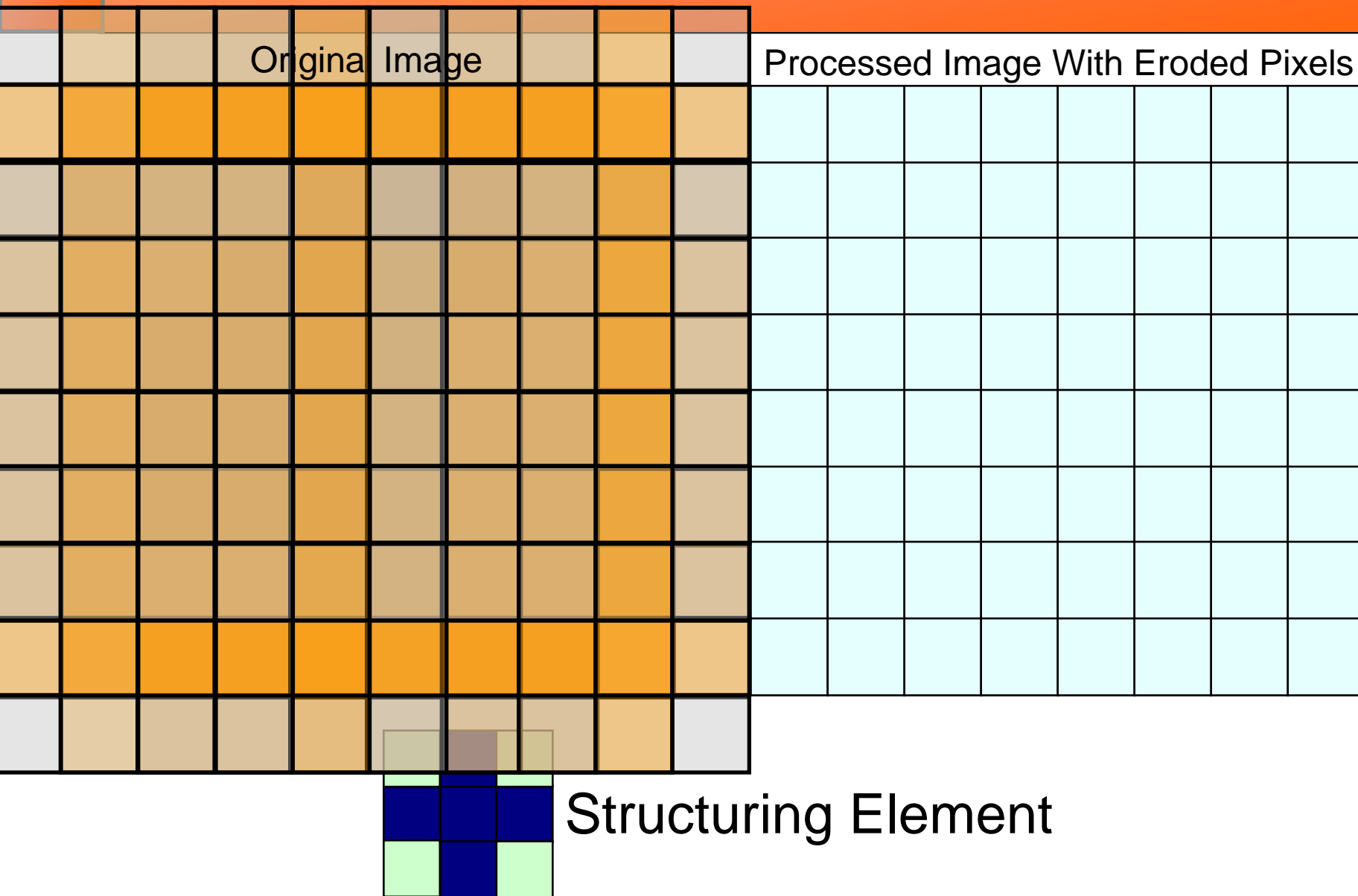
❑ Phép co bức ảnh f bởi SE s được biểu diễn bởi phép toán: $f \ominus s$

❑ Mô tả phép toán:

Giả sử SE s đang ở vị trí (x, y) . Pixel mới sau khi thực hiện phép toán có giá trị như sau:

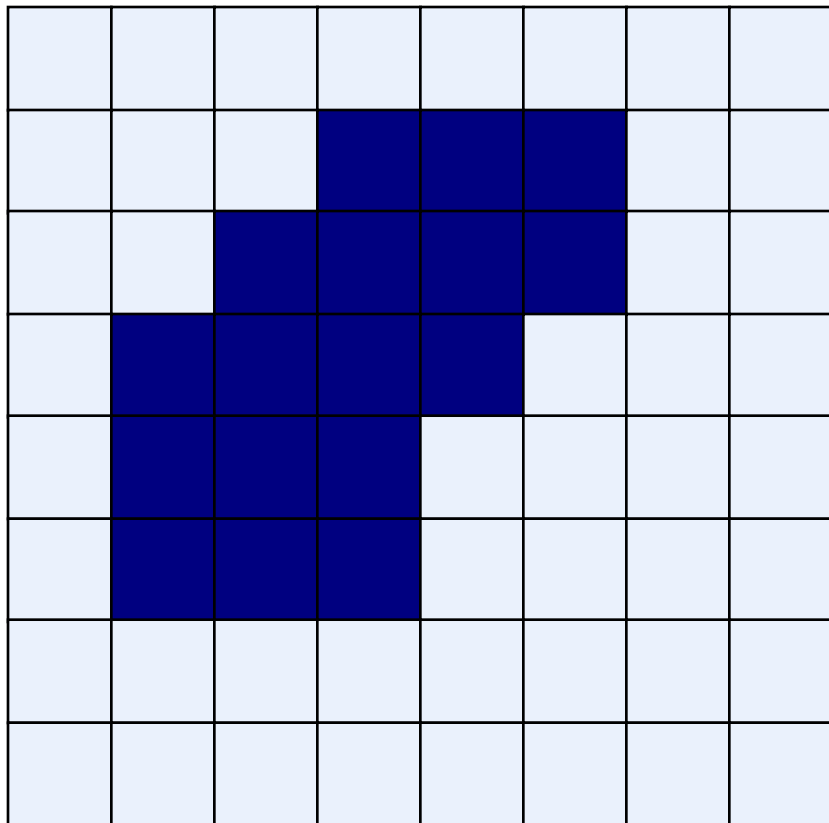
$$g(x, y) = \begin{cases} 1 & \text{if } s \text{ fits } f \\ 0 & \text{otherwise} \end{cases}$$

Phép co: Ví dụ 1

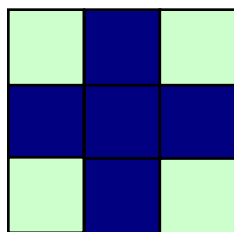
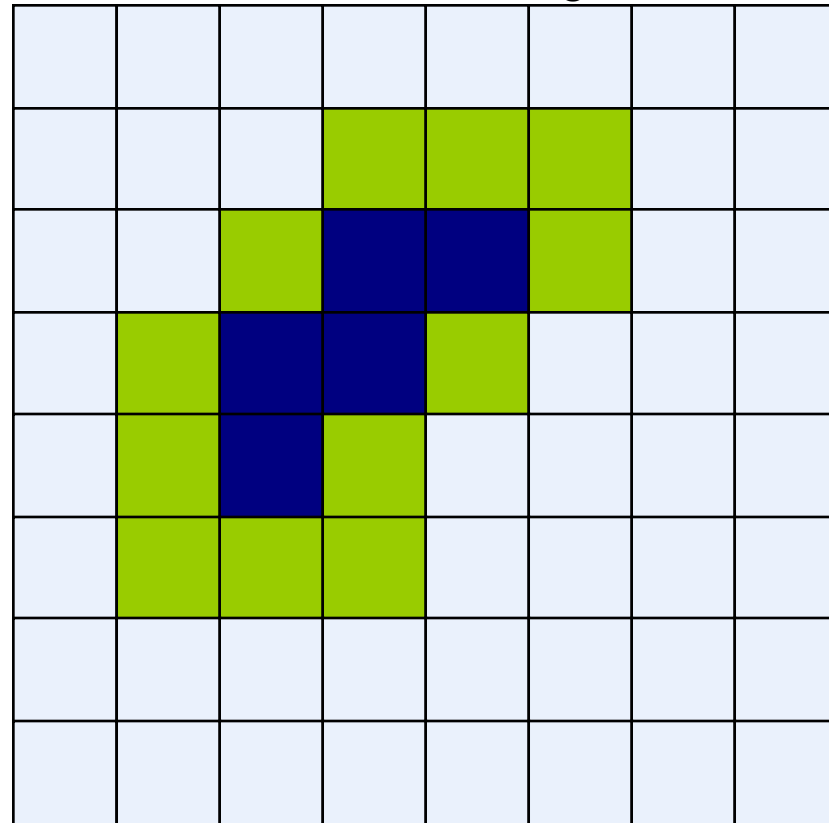


Phép co: Ví dụ 1

Original Image



Processed Image

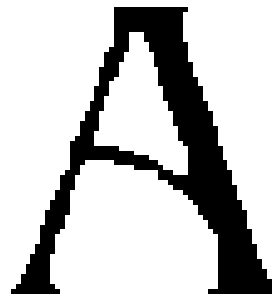


Structuring Element

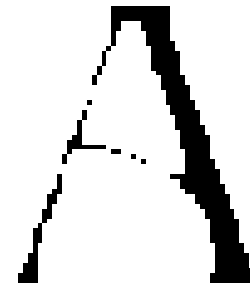
Phép co: Ví dụ 2



Original image

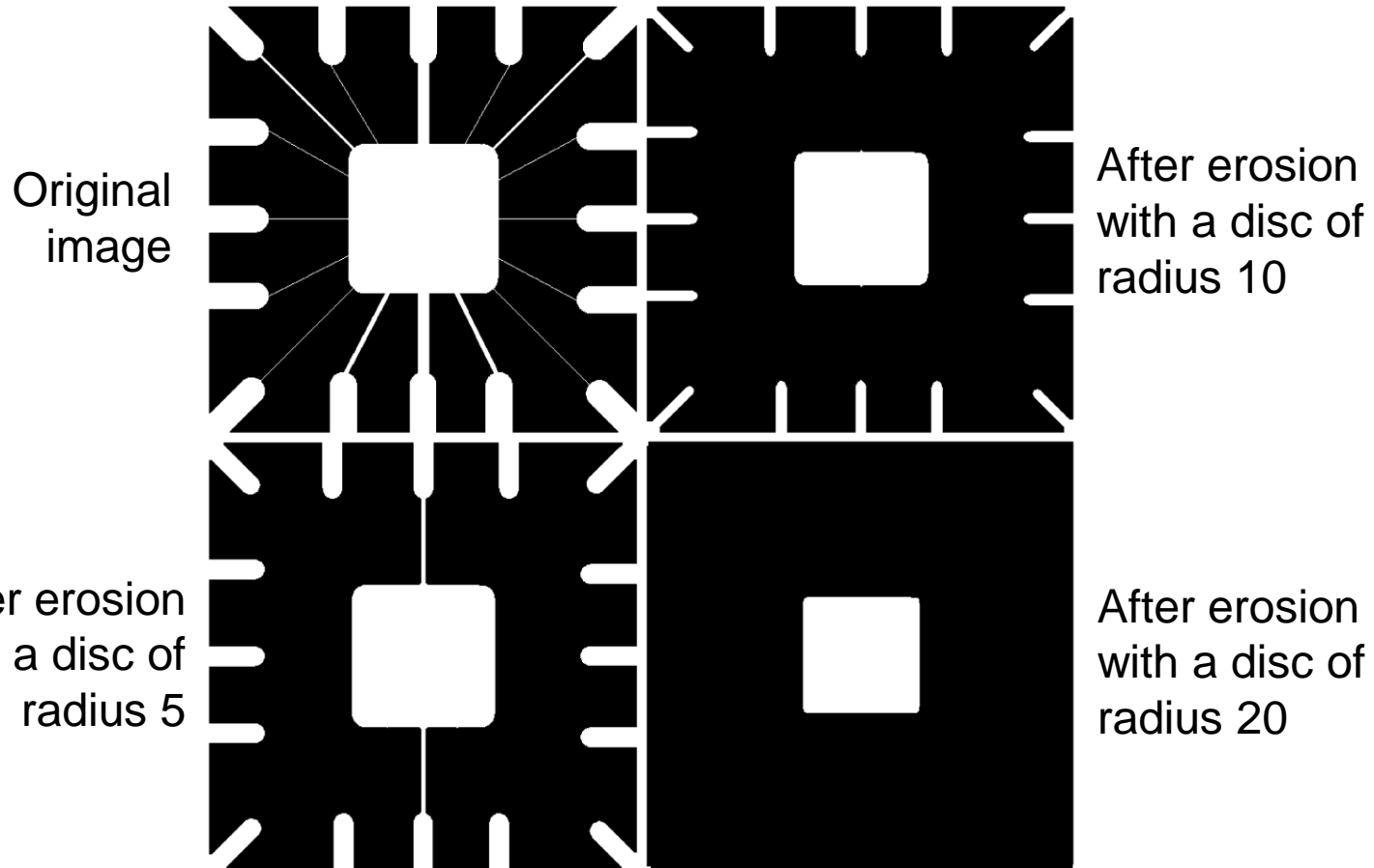


Erosion by 3*3
square structuring
element



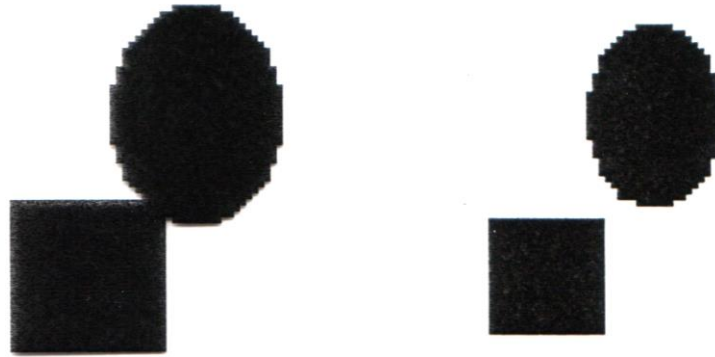
Erosion by 5*5
square structuring
element

Phép co: Ví dụ 3

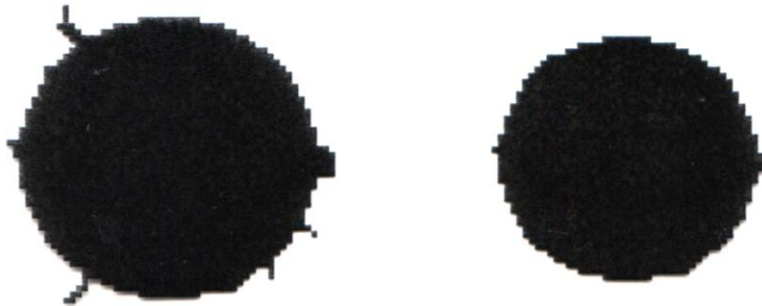


Ứng dụng của phép co hình

- ❑ Phép co hình làm tách rời các đối tượng gần nhau



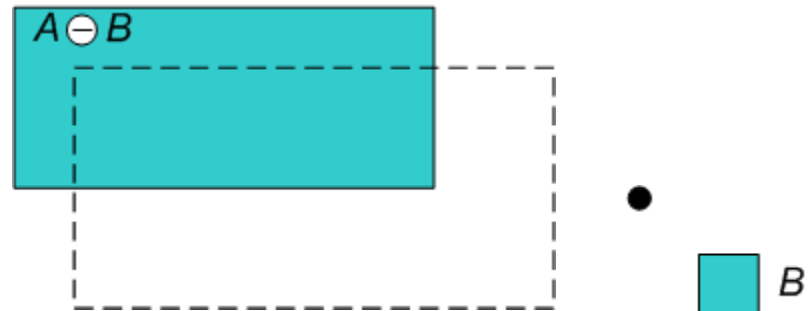
- ❑ Phép co hình cắt bớt các phần dư thừa của đối tượng



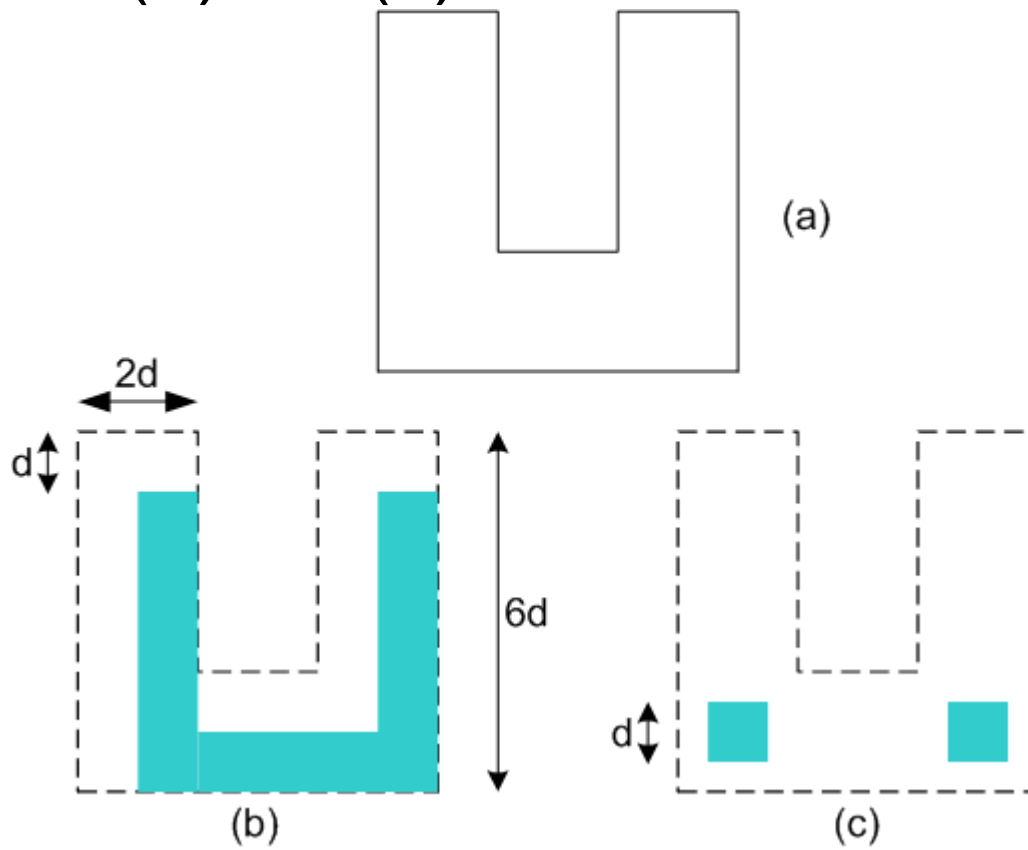
Chú ý: Phép co hình có tác dụng thu nhỏ đối tượng

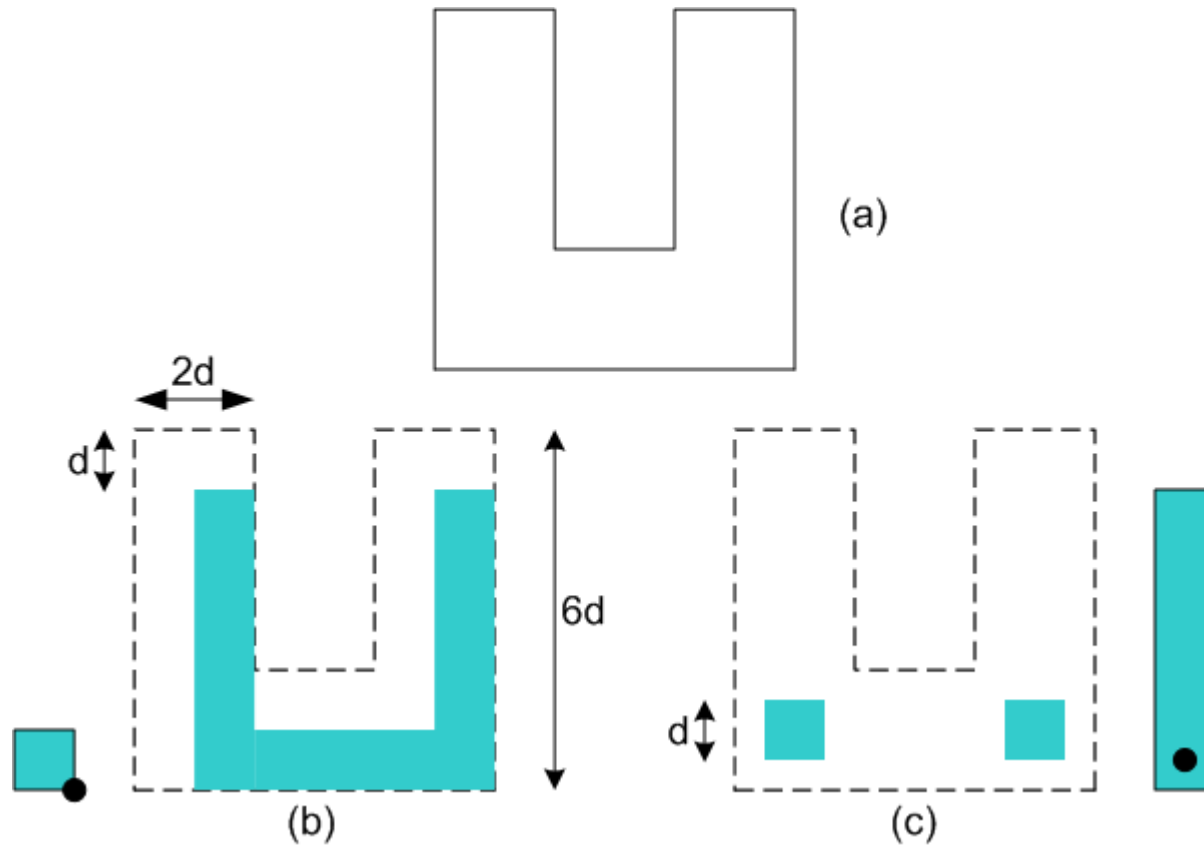
Cho tập A và phần tử cấu trúc B như hình vẽ. Tìm kết quả của phép co A với cấu trúc B .





Cho tập A như hình vẽ (a). Tìm các phần tử cấu trúc để sau phép co, tập A có dạng như hình (b) và (c).





Phép giãn hình- Dilation

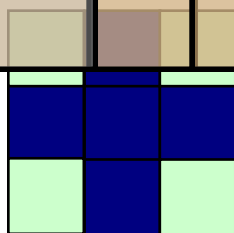
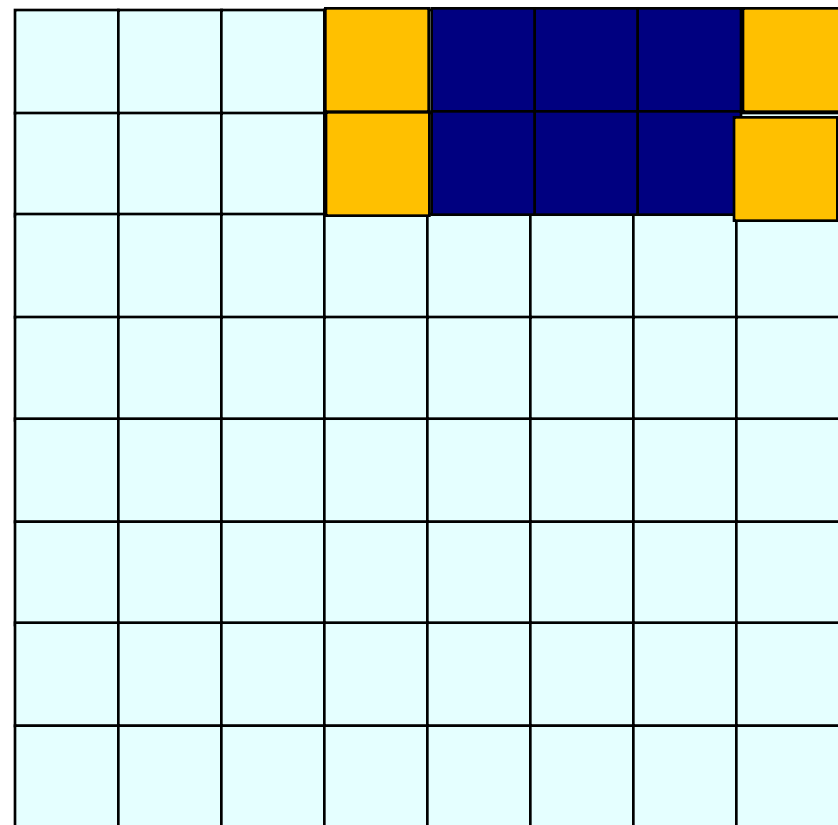
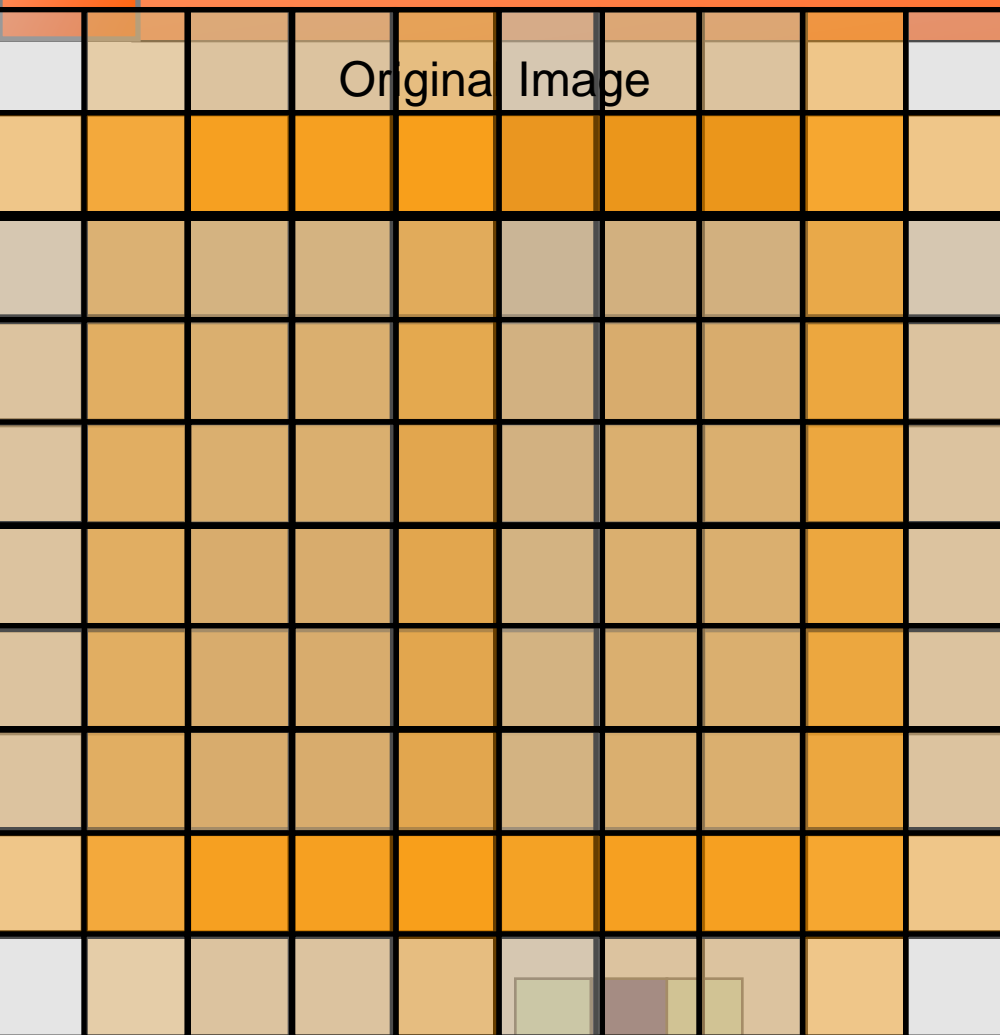
□ Phép giãn hình f bởi phần tử cấu trúc s được cho bởi công thức $f \oplus s$

□ Mô tả phép toán:

Nếu phần tử cấu trúc s tại pixel có tọa độ (x, y) của ảnh gốc thì sau phép giãn hình, pixel đó có giá trị:

$$g(x, y) = \begin{cases} 1 & \text{if } s \text{ hits } f \\ 0 & \text{otherwise} \end{cases}$$

Phép giãn hình: Ví dụ 1



Structuring Element

Phép giãn hình: Ví dụ 2



Original image



Dilation by 3*3
square structuring
element



Dilation by 5*5
square structuring
element

Phép giãn hình: Ví dụ 3

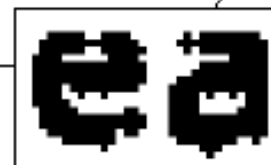
Original image

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.



After dilation

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.



0	1	0
1	1	1
0	1	0

Structuring element

Ứng dụng của phép giãn hình

- ❑ Phép giãn hình có thể sửa những đối tượng bị đứt gãy

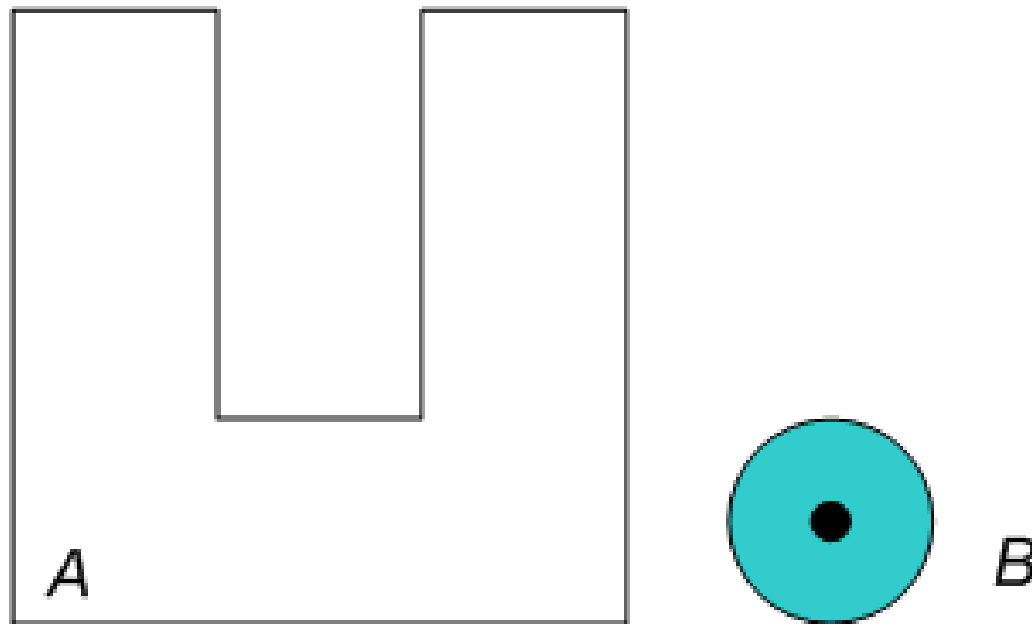


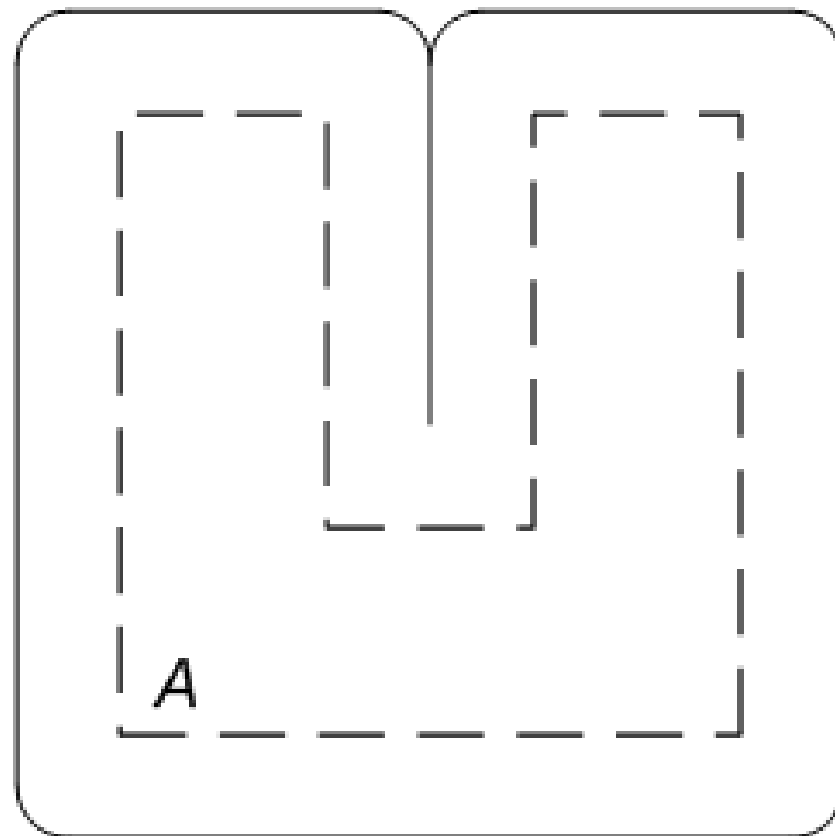
- ❑ Phép giãn hình có thể bổ sung những phần thiếu của đối tượng



Lưu ý: Phép giãn hình có tác dụng phóng to các đối tượng

Cho tập A và phần tử cấu trúc B như hình vẽ. Tìm kết quả của phép co A với cấu trúc B .





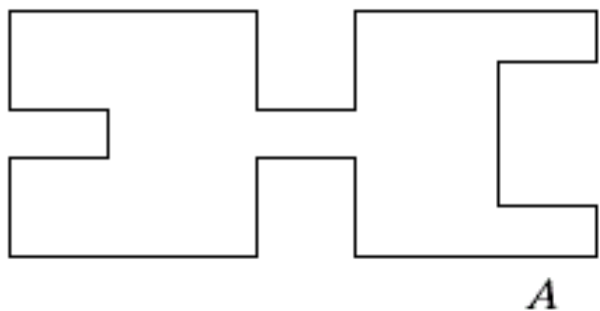
Các phép xử lý hình thái kết hợp

- Hai phép toán xử lý hình thái co và giãn có thể kết hợp với nhau để tạo ra phép xử lý hình thái mới:
 - Phép mở (Opening)
 - Phép đóng (Closing)

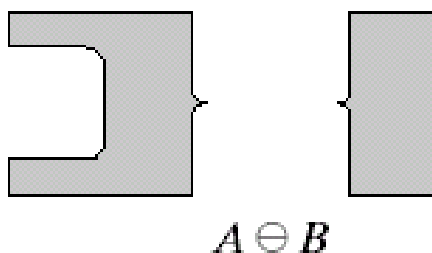
Phép mở - Opening

□ Phép toán mở ảnh f bởi phần tử cấu trúc s , được biểu diễn bởi công thức $f \circ s$, bao gồm phép co hình và theo sau bởi phép giãn:

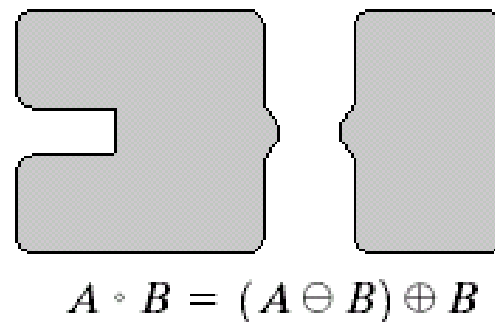
$$f \circ s = (f \ominus s) \oplus s$$



Original shape



After erosion



After dilation
(opening)

Note a disc shaped structuring element is used

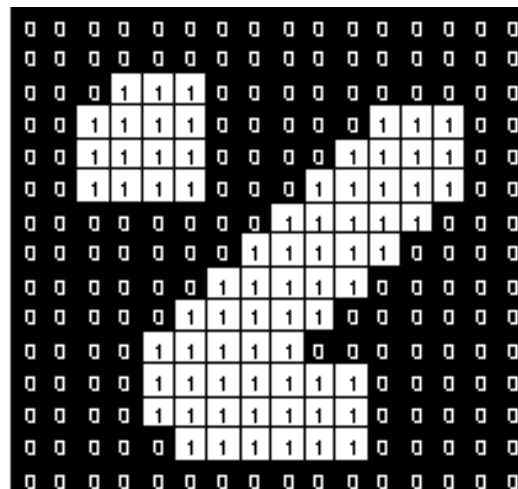
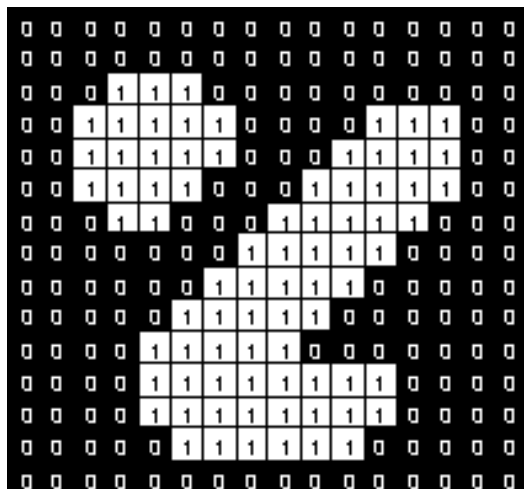
Phép mở ảnh: Ví dụ



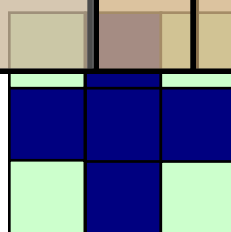
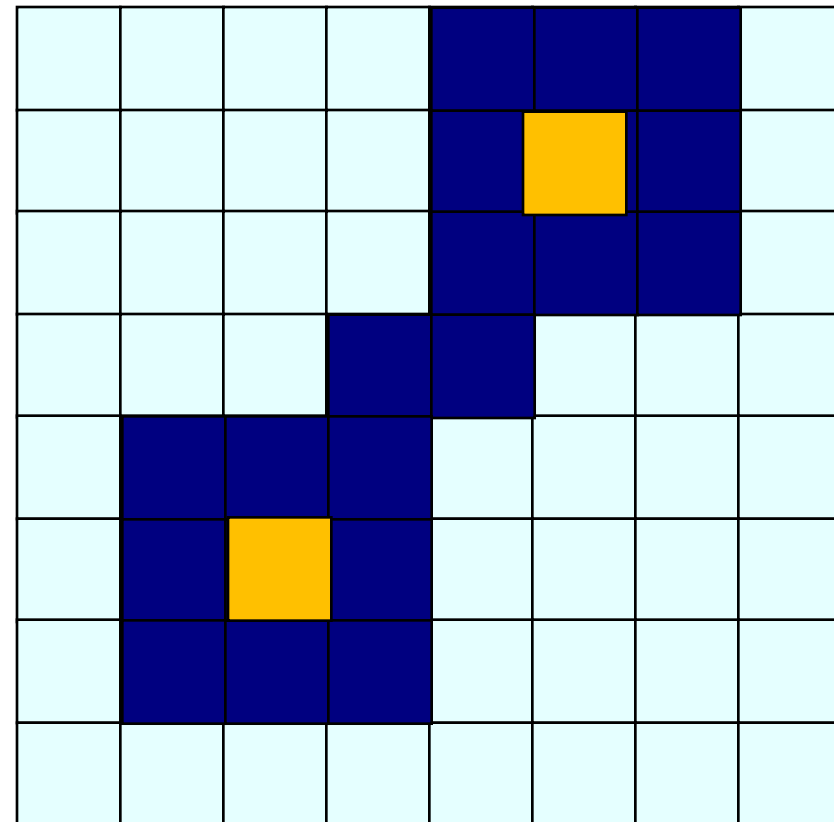
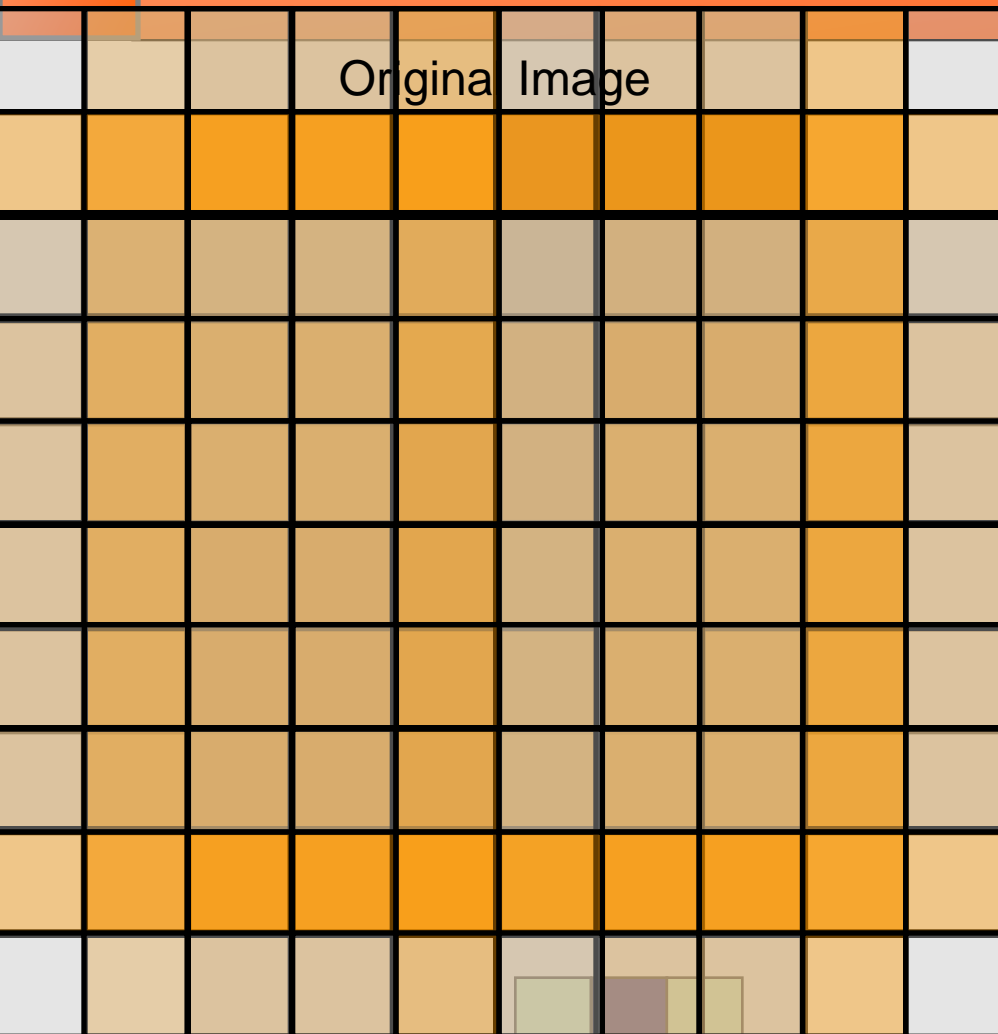
Original Image



Image After Opening

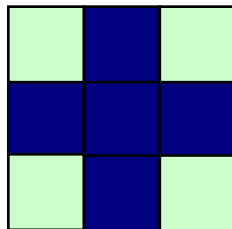
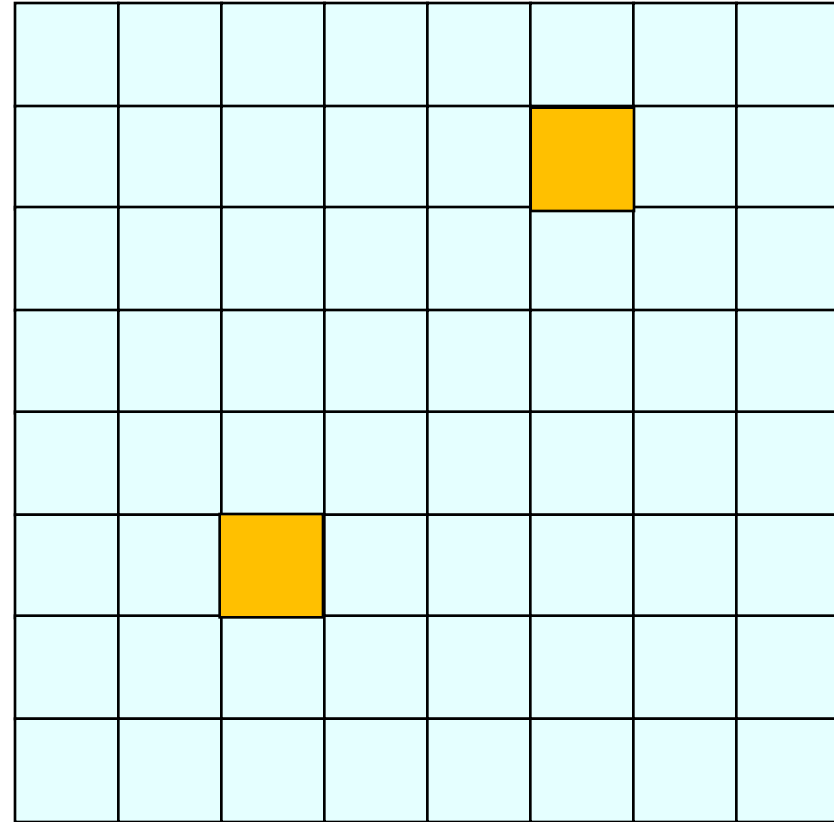
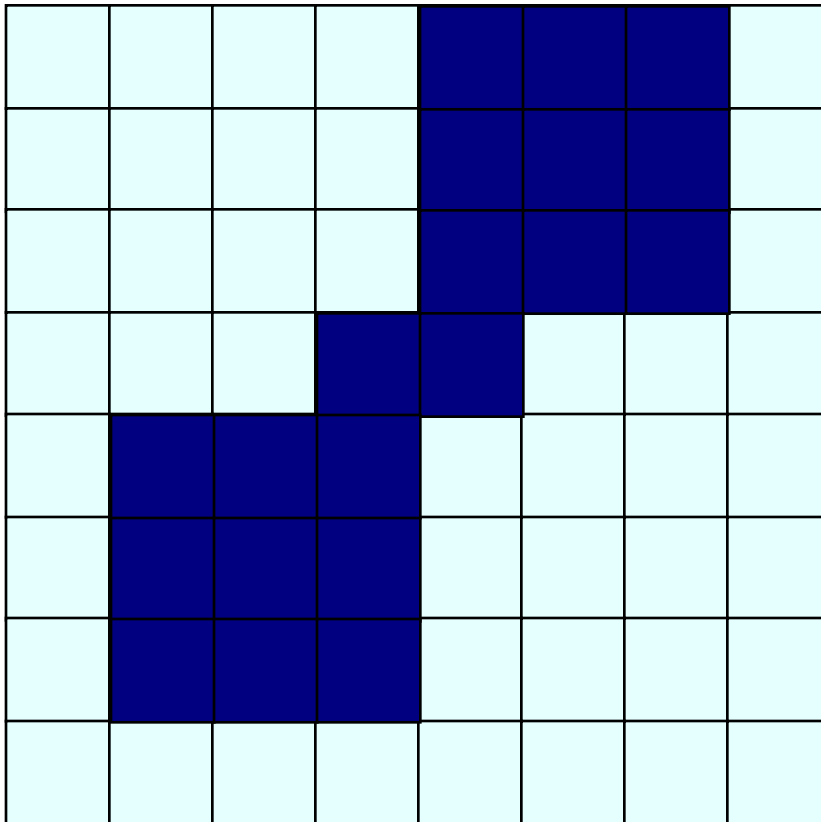


Phép mở ảnh: ví dụ



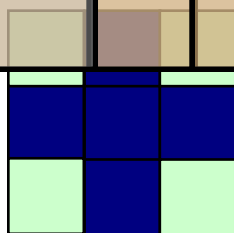
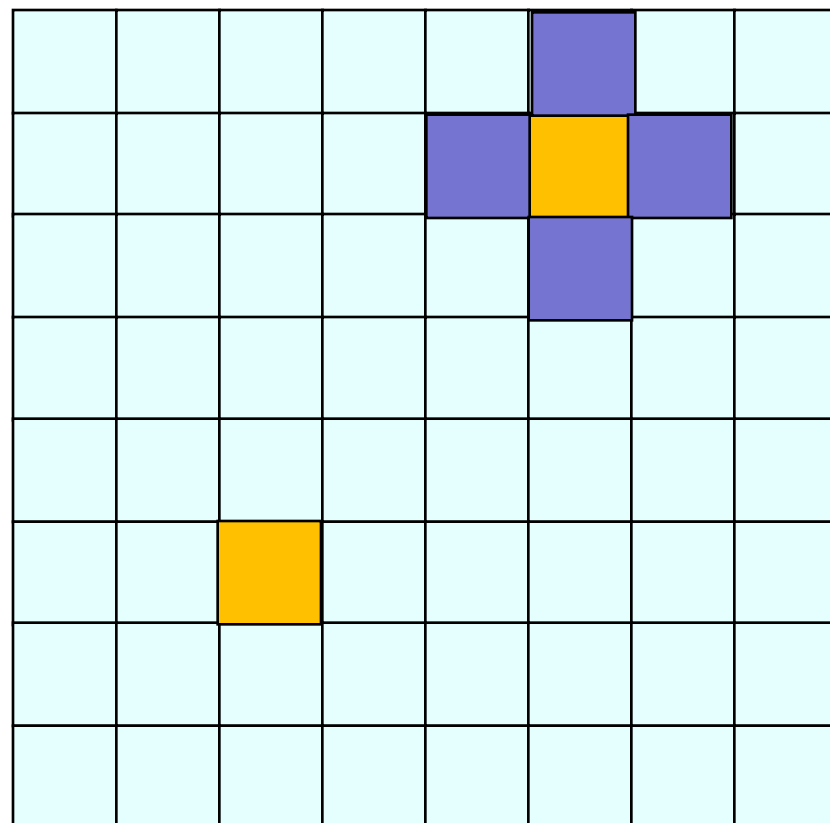
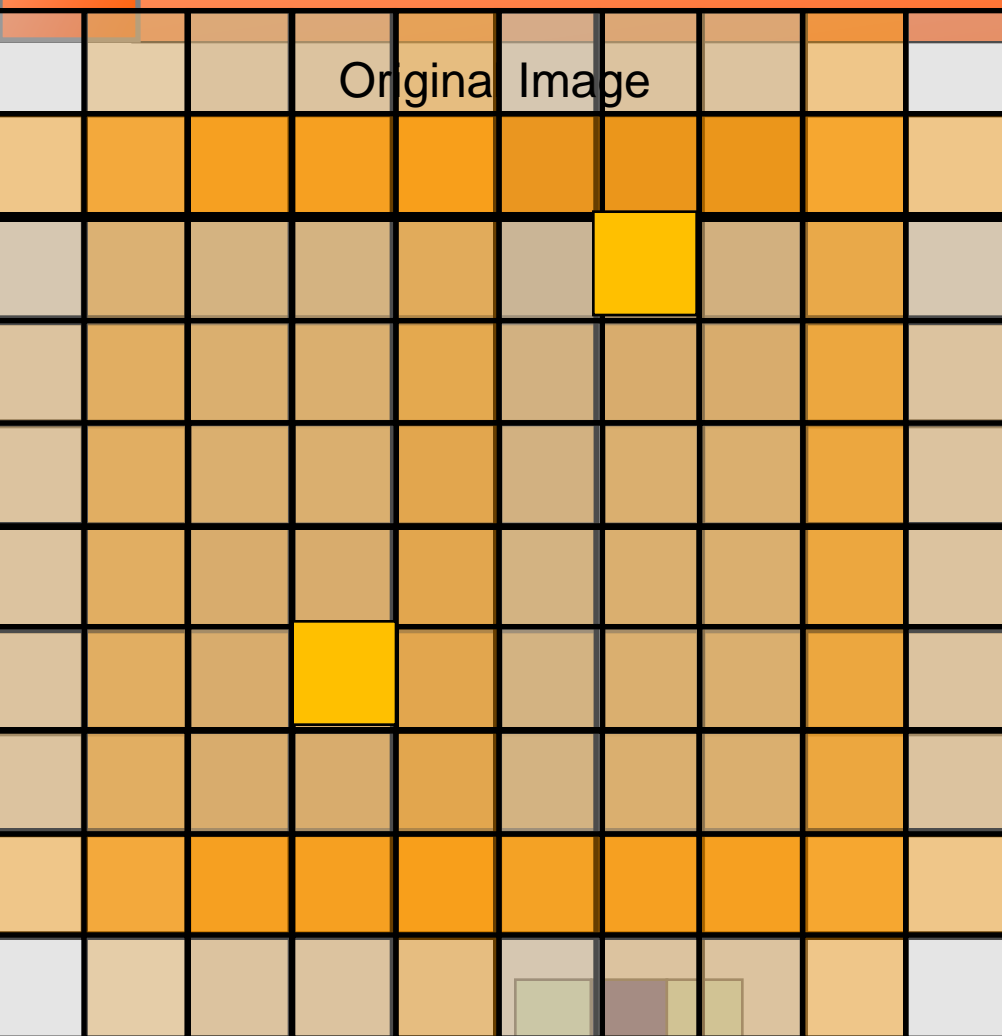
Structuring Element

Phép mở ảnh: Ví dụ



Structuring Element

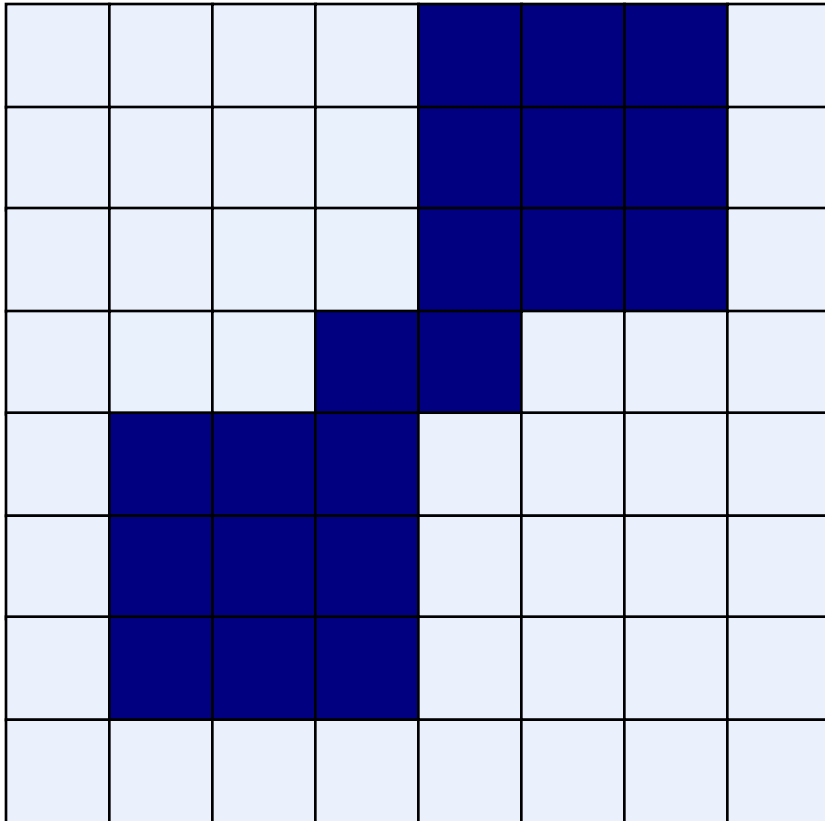
Phép mở ảnh: Ví dụ



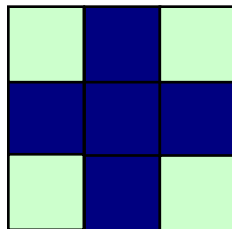
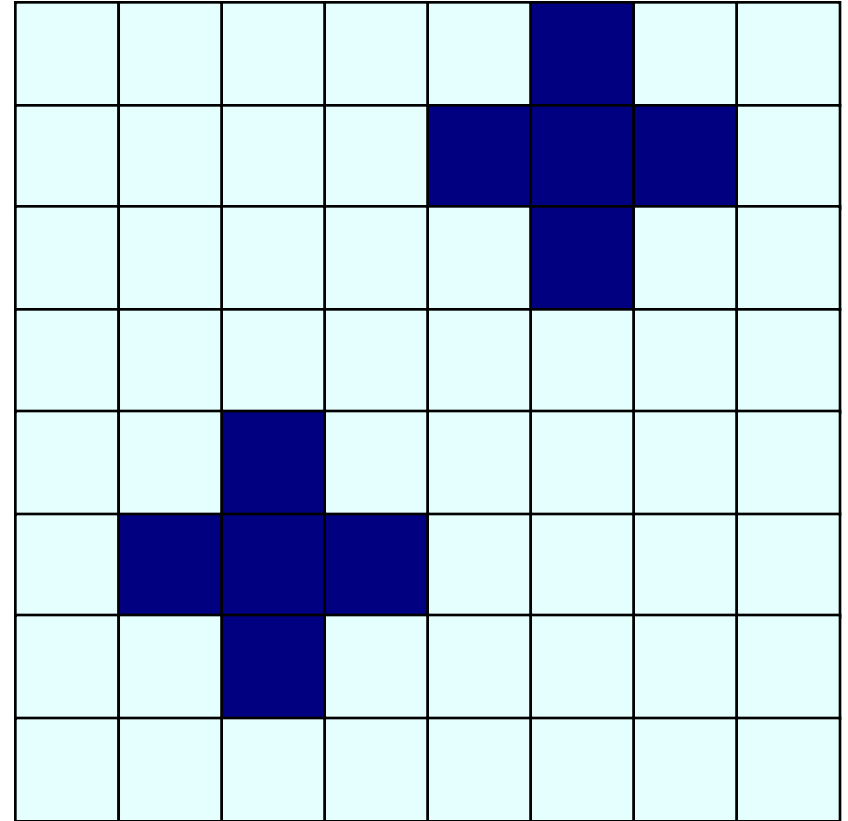
Structuring Element

Phép mở ảnh: Ví dụ

Original Image



Processed Image



Structuring Element

Ứng dụng của phép mở ảnh

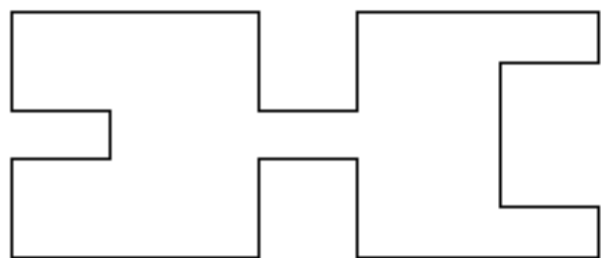
Phép mở ảnh có tác dụng:

- Làm trơn các đường gãy khúc.
- Cắt đứt các dải eo hẹp nối giữa các đối tượng
- Loại bỏ các mảnh nhỏ và các góc nhọn trên ảnh

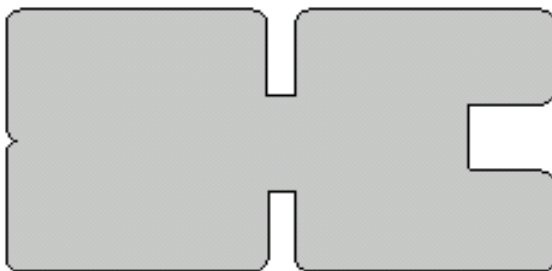
Phép đóng - Closing

□ Phép đóng hình f bởi phần tử cấu trúc s , được biểu diễn bởi công thức $f \cdot s$, bao gồm phép giãn hình và theo sau bởi phép co hình

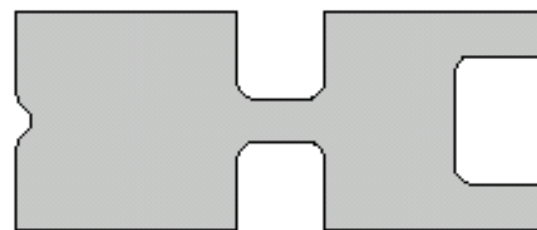
$$f \cdot s = (f \oplus s) \ominus s$$

 A

Original shape

 $A \oplus B$

After dilation

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

After erosion
(closing)

Note a disc shaped structuring element is used

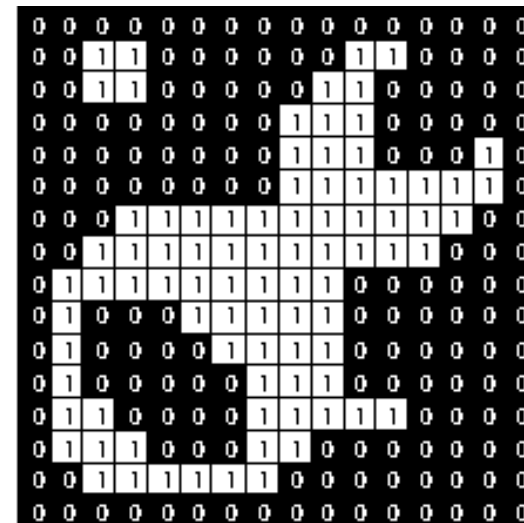
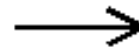
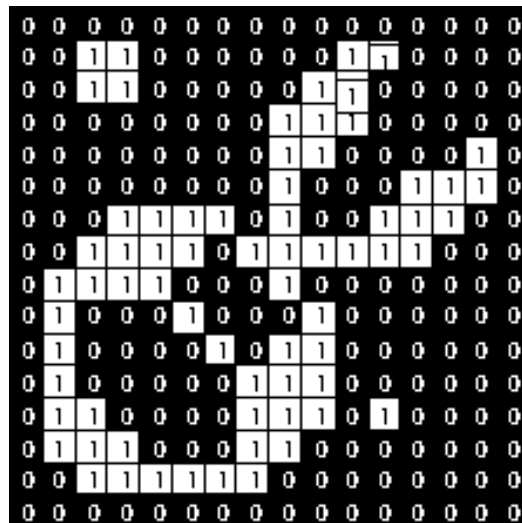
Phép đóng ảnh: Ví dụ



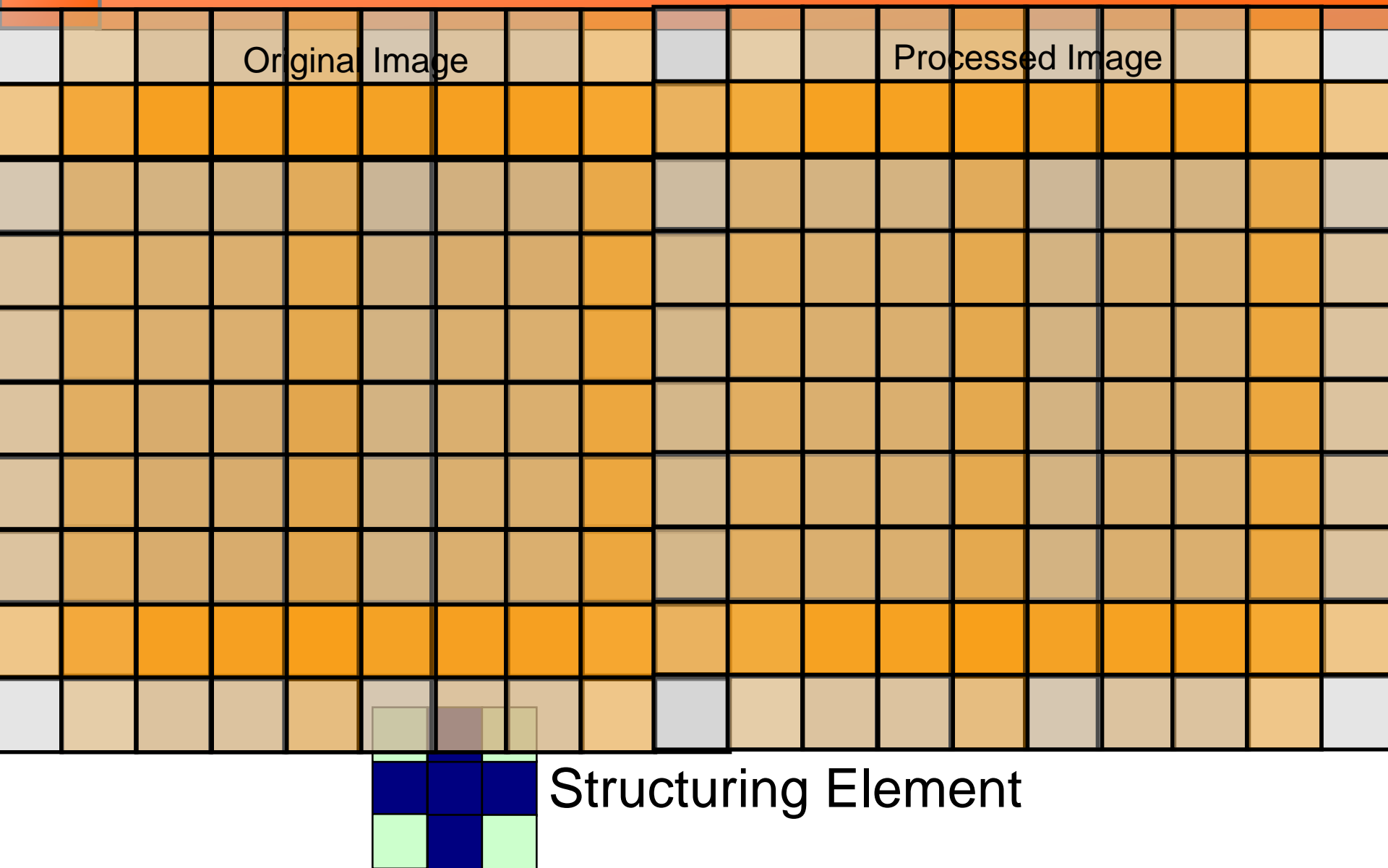
Original Image



Image After Closing

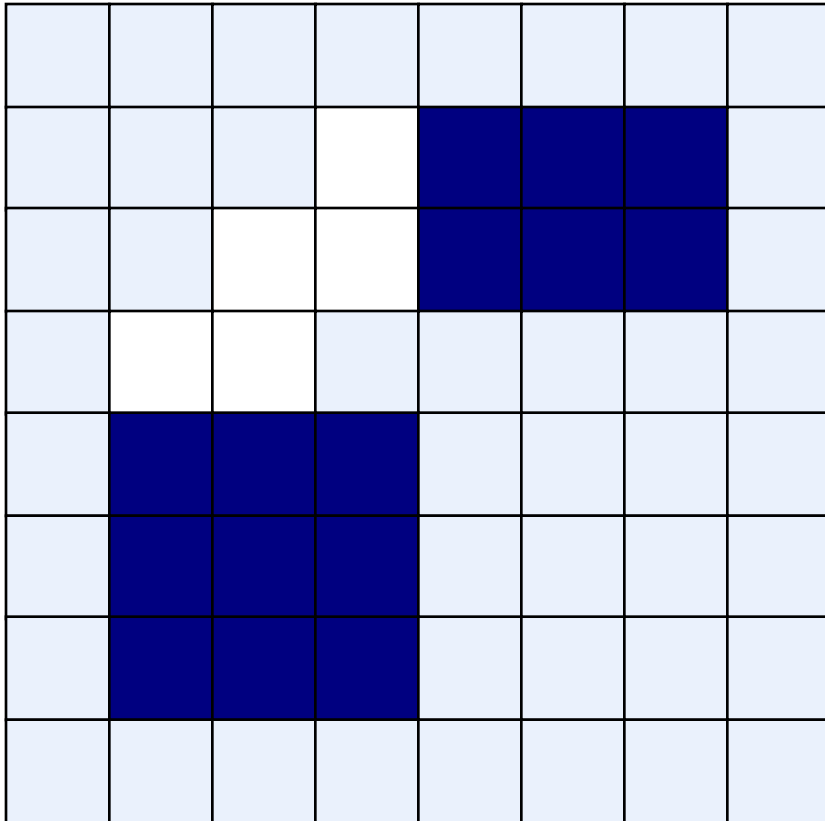


Phép đóng ảnh: Ví dụ

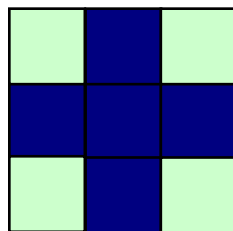
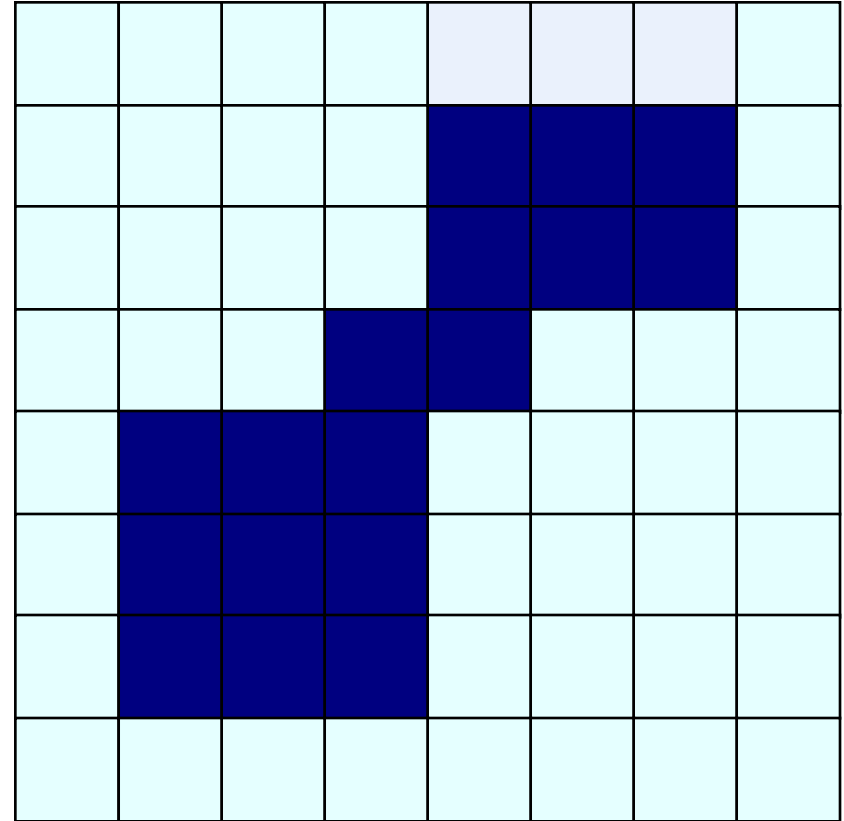


Phép đóng ảnh: Ví dụ

Original Image



Processed Image



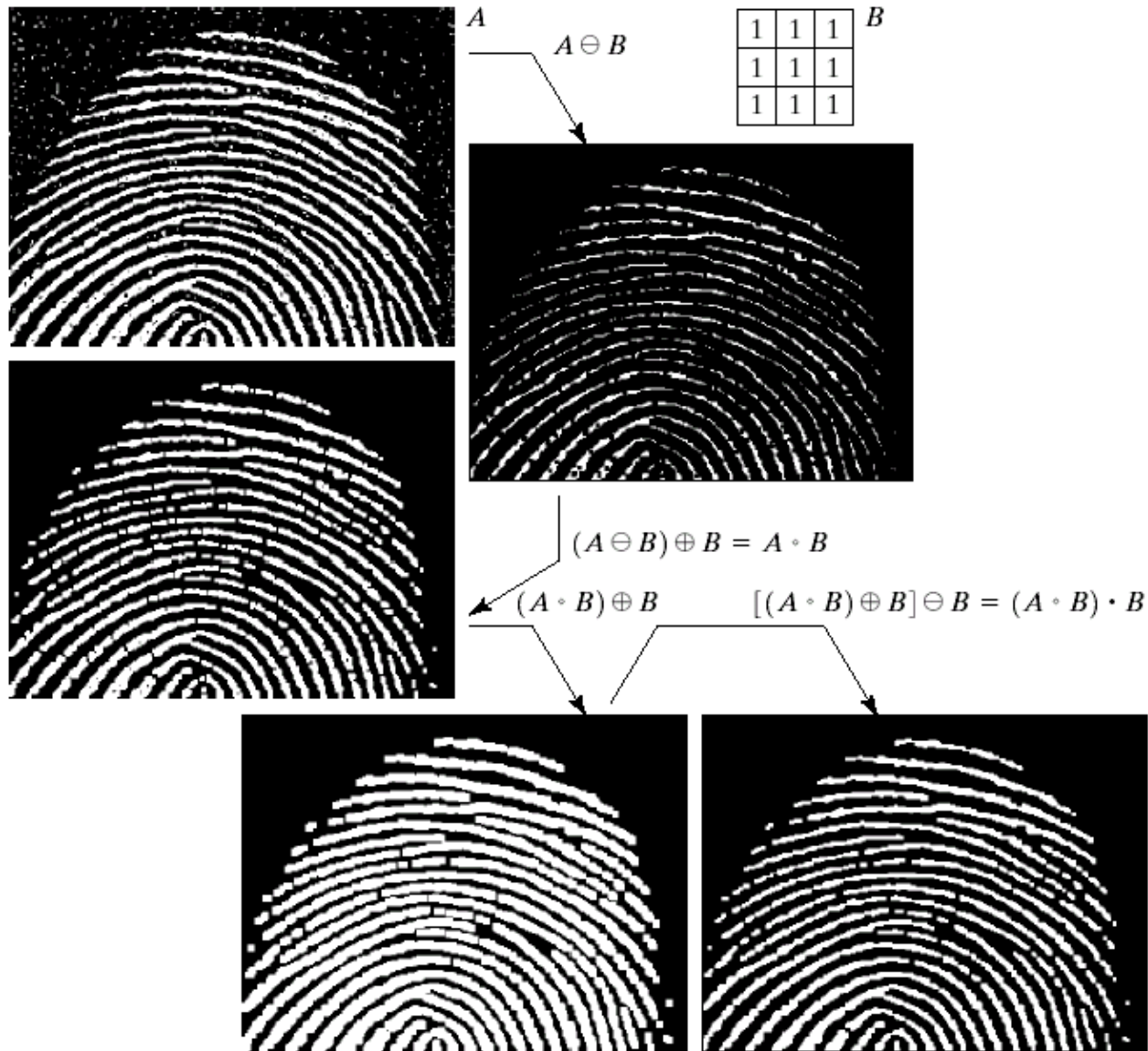
Structuring Element

Ứng dụng của phép đóng ảnh

Phép đóng ảnh có tác dụng:

- ❖ Làm trơn các đường gãy khúc.
- ❖ Làm nổi bật các dải eo hẹp nối giữa các đối tượng
- ❖ Loại bỏ các khoảng trống nhỏ trên ảnh

Ví dụ về xử lý ảnh hình thái

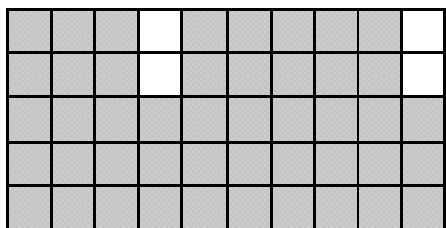
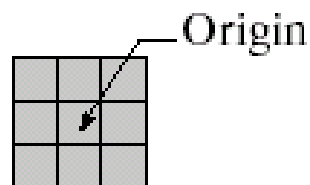
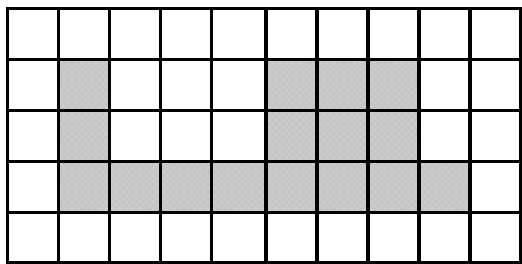
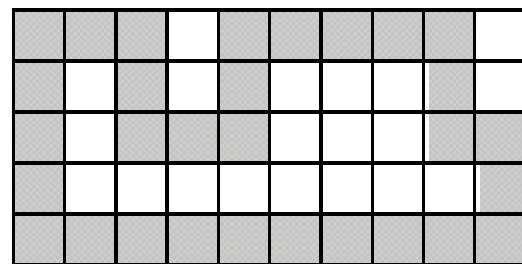


THUẬT TOÁN XỬ LÝ HÌNH THÁI

Tách biên

Thuật toán tách biên được cho bởi công thức:

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

*A**B**A ⊖ B**β(A)*

Cho bức ảnh sau A. Thực hiện phép co ảnh A bởi phần tử cấu trúc có dạng 3x3. Tách biên của đối tượng trong ảnh A.

$$A = \begin{bmatrix} 1110111110 \\ 1110111110 \\ 1111111111 \\ 1111111111 \\ 1111111111 \end{bmatrix}$$

```
>>se=strel('square',3)
```

```
>>E=imerode(A,se)
```

```
>>Ec=not(E)
```

```
>>beta=Ec&A
```

$$E^c = \begin{bmatrix} 111111111 \\ 1011100011 \\ 1011100011 \\ 1000000001 \\ 111111111 \end{bmatrix} \quad A = \begin{bmatrix} 1110111110 \\ 1110111110 \\ 111111111 \\ 111111111 \\ 111111111 \end{bmatrix}$$

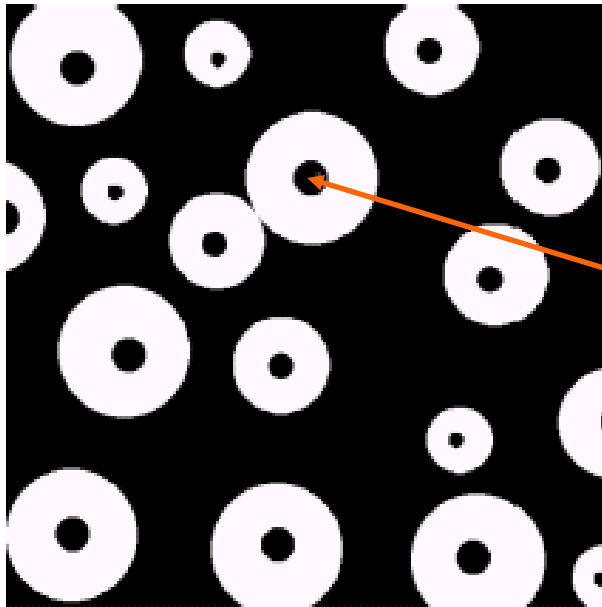
$$E = A \ominus B = \begin{bmatrix} 000000000 \\ 0100011100 \\ 0100011100 \\ 0111111110 \\ 000000000 \end{bmatrix}$$

$$\beta(A) = A \cap E^c = \begin{bmatrix} 1110111110 \\ 1010100010 \\ 1011100011 \\ 1000000001 \\ 111111111 \end{bmatrix}$$

THUẬT TOÁN XỬ LÝ HÌNH THÁI

Làm đầy vùng

❖ Thuật toán làm đầy vùng là quá trình tìm tất cả các pixel bên trong của một vùng và gán các pixel tìm được giá trị bằng 1



Cho 1 điểm bên trong hình tròn, liệu có thể tô toàn bộ hình tròn này không?

- ❖ Thuật toán : Xác định 1 điểm bên trong vùng, sau đó thực hiện liên tục các phép giãn hình



$$X_0 = p$$

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

Until $X_k = X_{k-1}$

Bound the growth

THUẬT TOÁN XỬ LÝ HÌNH THÁI

Làm đầy vùng (cont...)

- Công thức của thuật toán:

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

Trong đó X_0 là điểm bắt đầu trong vùng, B là phần tử cấu trúc và A^c là phần bù của A

- Phương trình được lặp cho đến khi X_k bằng X_{k-1}
- Kết quả cuối cùng là hợp của X_k và A

THUẬT TOÁN XỬ LÝ HÌNH THÁI

Làm đầy vùng (cont...)

a	b	c
d	e	f
g	h	i

FIGURE 9.15

Region filling.

(a) Set A .

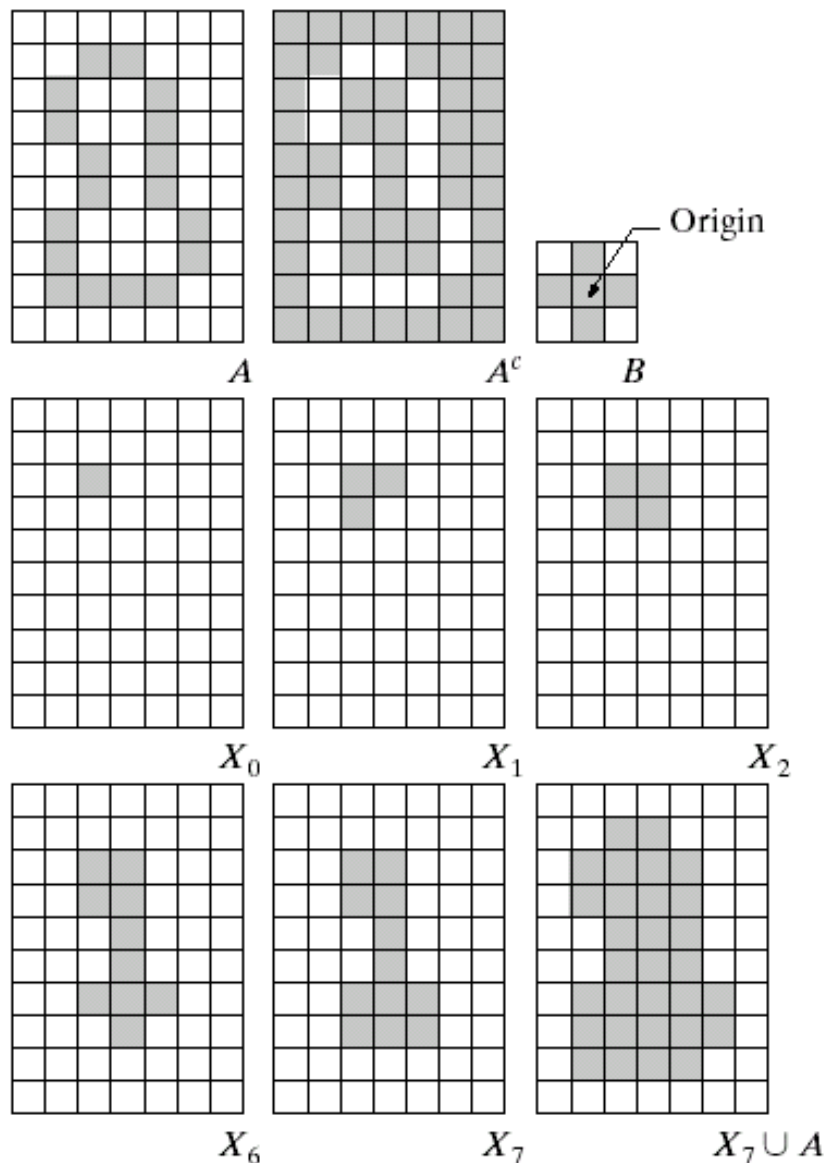
(b) Complement of A .

(c) Structuring element B .

(d) Initial point inside the boundary.

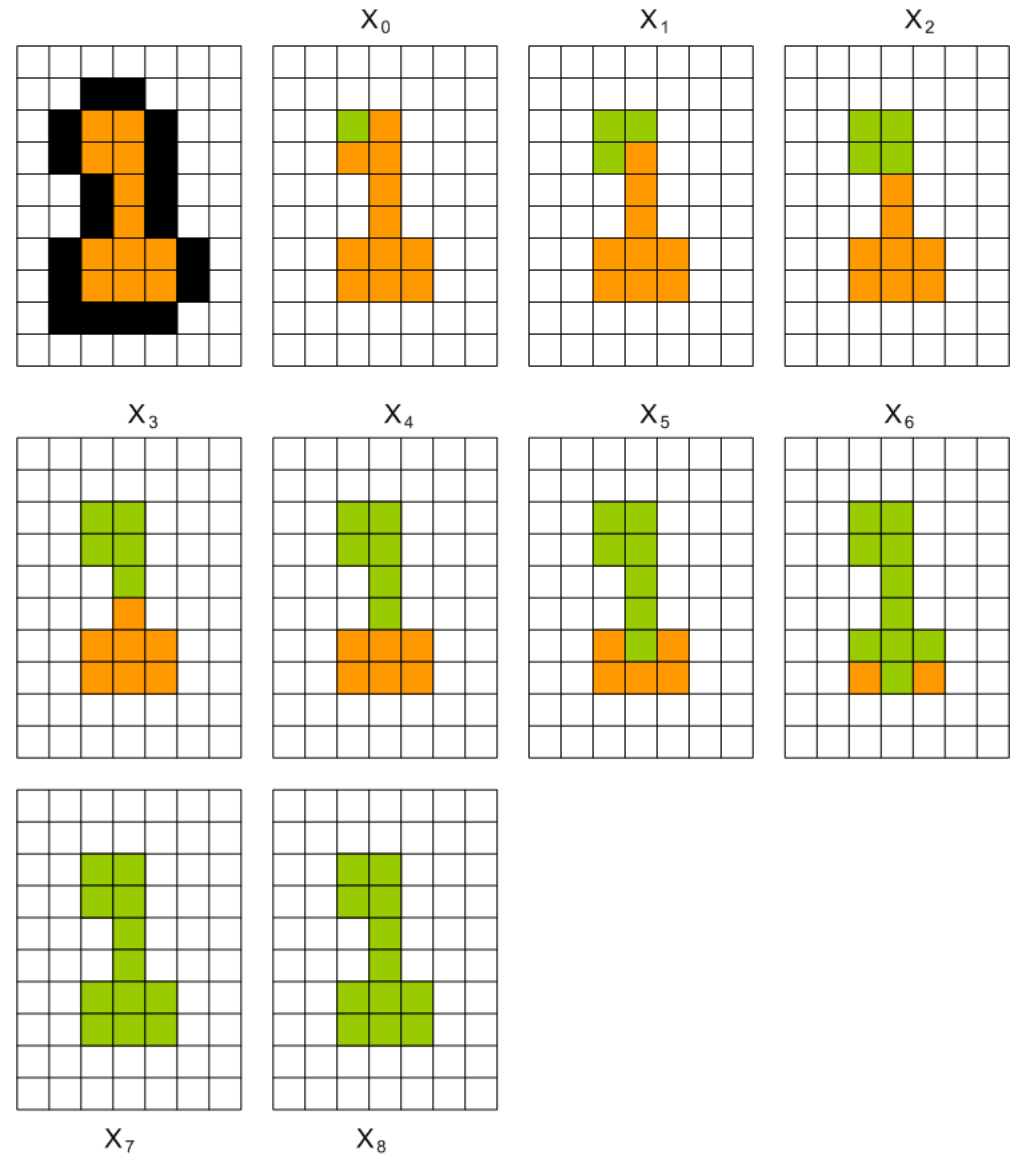
(e)–(h) Various steps of Eq. (9.5-2).

(i) Final result [union of (a) and (h)].



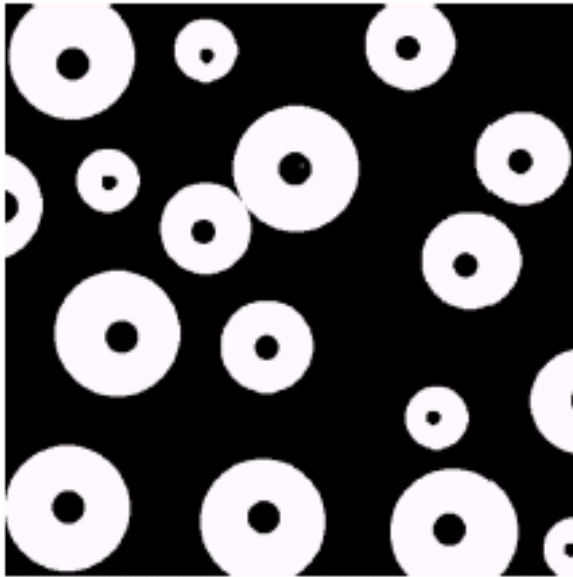
THUẬT TOÁN XỬ LÝ HÌNH THÁI

Làm đầy vùng (cont...)

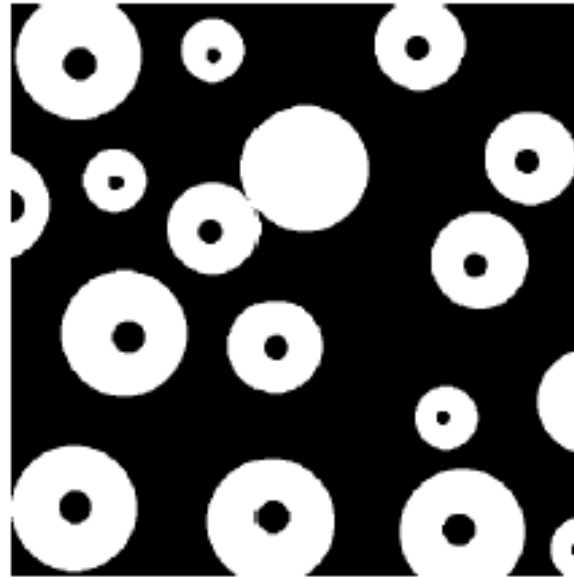


THUẬT TOÁN XỬ LÝ HÌNH THÁI

Làm đầy vùng (cont...)



Original Image



One Region
Filled



All Regions
Filled

Cho bức ảnh sau A. Thực hiện các bước làm đầy vùng cho đối tượng trong ảnh A bởi cấu trúc B

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

- Phép co và phép giãn
- Phép đóng và phép mở ảnh
- Các thuật toán xử lý ảnh hình thái