Xử lý ảnh - Lọc trung vị đối với ảnh màu
Hình thái toán học: ảnh nhị phân

Đỗ Thanh Hà

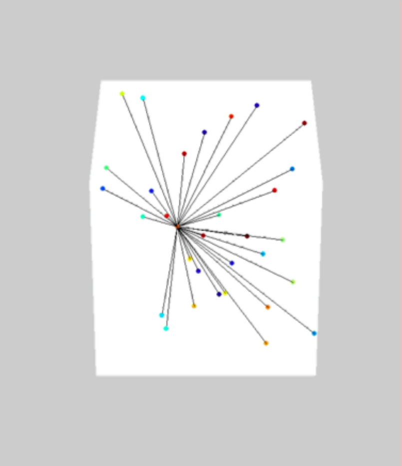
Bộ môn Tin học
Khoa Toán - Cơ - Tin học
Trường Đại học Khoa học Tự nhiên

Nội dung

- Bộ lọc trung vị cho ảnh màu
- Hình thái toán học: ảnh nhị phân

Bộ lọc trung vị cho ảnh màu

Một lọc vector trung vị được lựa chọn từ tập các vector, một vector gần nhất với tất cả các vector khác



Lọc trung vị màu

Nếu $F^n = R^3$ thì trung vị vector có thể được sử dụng như bộ lọc trung vị màu

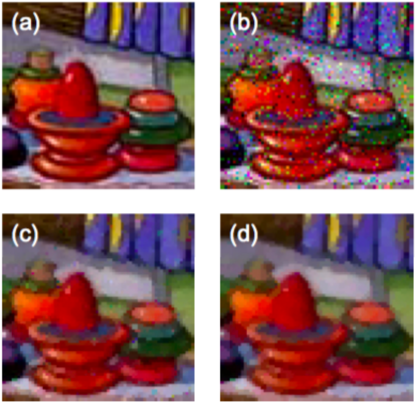
(a) Ảnh gốc

(b) Ảnh (a) với nhiễu thêm vào

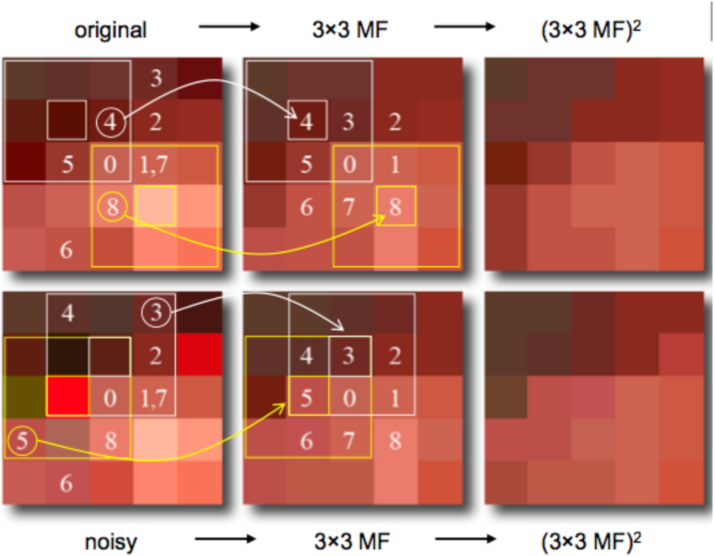
(c) Ảnh (b) được khử nhiễu bằng lọc trung vị

(d) Ảnh (c) được khử nhiễu bằng lọc trung vị

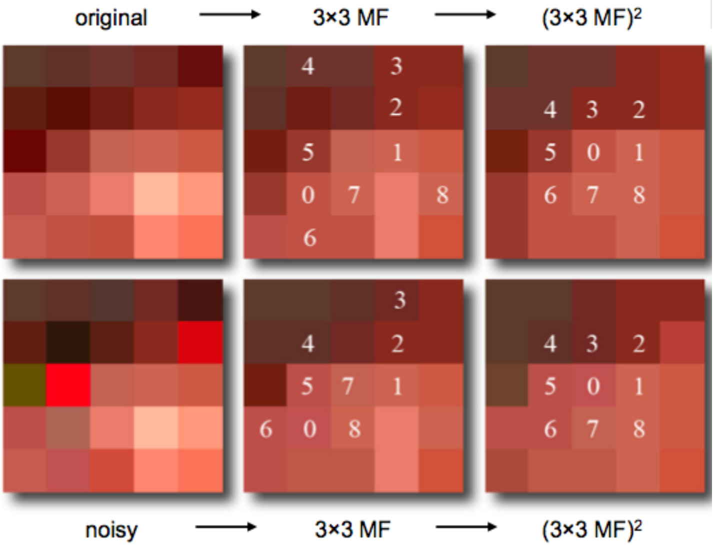
Lọc trung vị được thực hiện trên hàng xóm kích thước 3×3



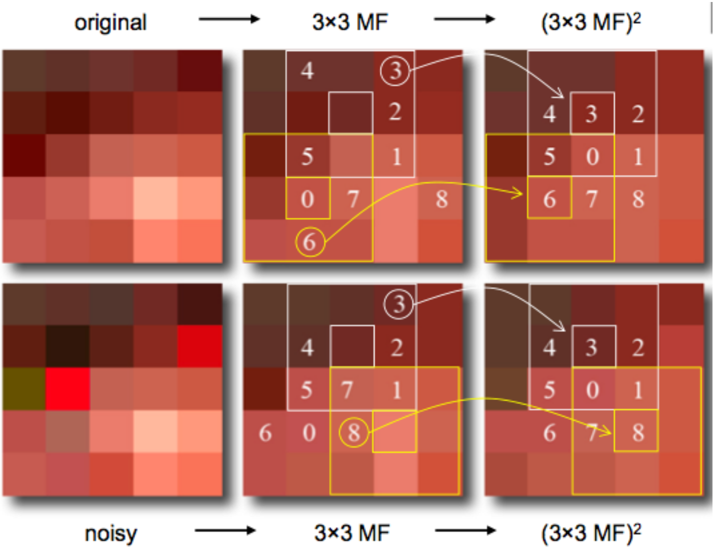
Lọc trung vị cho ảnh màu - minh hoạ



Lọc trung vị cho ảnh màu - minh hoạ



Lọc trung vị cho ảnh màu - minh hoạ



Ảnh màu **I**

ảnh nhiễu **J**

Ảnh khử nhiễu **K**

Bộ lọc trung vị màu được sử dụng có kích thước 3×3

Lọc vector trung vị (CMF) và lọc trung vị trên từng kênh màu riêng lẻ (MF)

- Tại sao ta không thực hiện lọc trung vị trên từng kênh màu của ảnh?
  - *Kết quả tại một pixel có thể là một màu mà không tồn tại trong hàng xóm của pixel trong ảnh đầu vào.*
  - *Kết quả không phải là trung vị của màu mà là trung vị của cường độ sáng của mỗi kênh màu được xử lý một cách độc lập*
- Điều này có dẫn đến vấn đề gì không?
  - *Có thể có hoặc không, và nó phụ thuộc vào ứng dụng*
  - *Ảnh kết quả của CMF và MF trên từng kênh màu nhìn sẽ khác nhau một chút*
  - *Trong trường hợp cần bảo tồn màu sắc của ảnh, thì MF trên từng kênh màu có thể không đảm bảo được điều này*

CMF và MF - Ví dụ

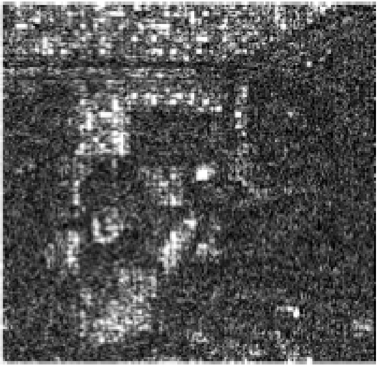
Ảnh màu **I**

ảnh nhiễu **J**

Ảnh khử nhiễu bằng CMF

Ảnh khử nhiễu bằng MF

CMF và MF - Ví dụ



Tỉ lệ pixel trong CMF^2 ảnh nhiễu giống ảnh gốc: 0.29

Tỉ lệ pixel trong MF^2 ảnh nhiễu giống ảnh gốc: 0.14

Hình thái toán học: ảnh nhị phân

Hình thái toán học là gì

- không tuyến tính
- xây dựng dựa trên lý thuyết tập hợp Minkowski
- một phần lý thuyết lưới hữu hạn
- cho phân tích ảnh dựa trên hình dáng
- thực sự hữu ích

Sử dụng của hình thái toán học

- Nâng ảnh
- Phân đoạn ảnh
- Phục hồi ảnh
- Phát hiện cạnh (edge detection)
- Phân tích texture
- Tạo đặc trưng
- Skeletonization
- Phân tích hình dáng
- Nén ảnh
- Phân tích thành phần
- Phát hiện đặc trưng
- Giảm nhiễu

Kí hiệu và định nghĩa ảnh

- Một ảnh là một ánh xạ, $\mathbf{I}$, từ tập S_p gồm các toạ độ pixel (*tập không gian ảnh*) vào tập các giá trị G thoả mãn: với mỗi toạ độ pixel $\mathbf{p} = (r, c)$ tồn tại một giá trị $\mathbf{I}(\mathbf{p}) \in G$.
- Ảnh nhị phân chỉ có 2 giá trị $G = \{v_{fg}, v_{bg}\}$ trong đó v_{fg} được gọi là giá trị *foreground* và v_{bg} được gọi là giá trị *background*.
- Thông thường $v_{fg} = 0$ và $v_{bg} = -\infty$. Ngoài ra v_{fg} và v_{bg} có thể nhận các giá trị khác, bao gồm $\{v_{fg}, v_{bg}\} = \{0, \infty\}$, $\{0, 1\}$, $\{1, 0\}$, $\{0, 255\}$, $\{255, 0\}$
- Trong bài giảng này, ta giả sử $\{v_{fg}, v_{bg}\} = \{255, 0\}$

Kí hiệu và định nghĩa ảnh (tiếp)

- Foreground của ảnh nhị phân $\mathbf{I}$ là

$$FG\{\mathbf{I}\} = \left\{\mathbf{I}(\mathbf{p}), \mathbf{p} = (r, c) \in S_p | \mathbf{I}(\mathbf{p}) = v_{fg}\right\}$$

- Background của ảnh nhị phân $\mathbf{I}$ là

$$BG\{\mathbf{I}\} = \left\{\mathbf{I}(\mathbf{p}), \mathbf{p} = (r, c) \in S_p | \mathbf{I}(\mathbf{p}) = v_{bg}\right\}$$

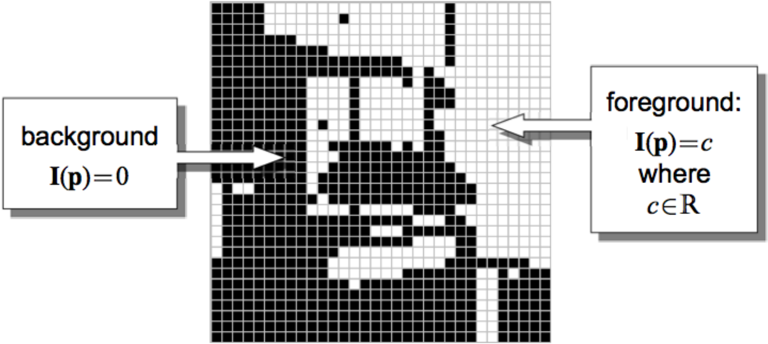
- Như vậy

$$FG\{\mathbf{I}\} \cup BG\{\mathbf{I}\} = \mathbf{I} \text{ và } FG\{\mathbf{I}\} \cap BG\{\mathbf{I}\} = \emptyset$$

- và background là phần bù của foreground và ngược lại

$$BG\{\mathbf{I}\} = \left\{FG\{\mathbf{I}\}^C\right\} \text{ và } FG\{\mathbf{I}\} = \left\{BG\{\mathbf{I}\}^C\right\}$$

Ảnh nhị phân



Ví dụ biểu diễn của một ảnh số, mỗi ô vuông là một pixel

Support của một ảnh

- Support của một ảnh **I** là tập vị trí các pixel *foreground* trong không gian ảnh

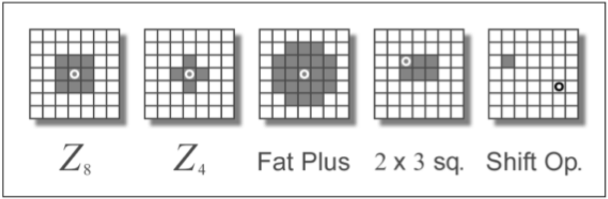
$$\text{supp}(\mathbf{I}) = \left\{\mathbf{p} = (r, c) \in S_p | \mathbf{I}(\mathbf{p}) = v_{fg}\right\}$$

- Phần bù của support là tập vị trí các pixel *background* trong không gian ảnh

$$\left\{\text{supp}(\mathbf{I})\right\}^C = \left\{\mathbf{p} = (r, c) \in S_p | \mathbf{I}(\mathbf{p}) = v_{bg}\right\}$$

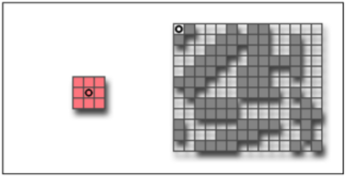
Thành phần cấu trúc là một ảnh nhỏ - được sử dụng như một cửa sổ dịch chuyển - mà nó hỗ trợ phác hoạ hàng xóm của một pixel trong không gian ảnh

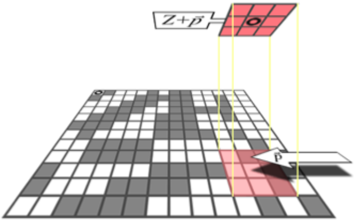
- Ví dụ về SEs, trong đó FG là màu xám và BG là màu trắng



- SE có thể có bất kì hình dáng, kích thước nào
- Trong hình trên, hình tròn là tâm của thành phần cấu trúc

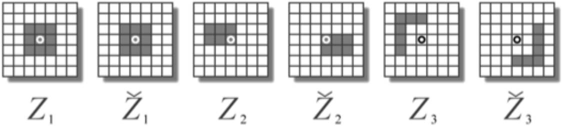
- Cho ảnh **I** và **Z** là SE
- $\mathbf{Z} + \mathbf{p}$ có nghĩa là **Z** được di chuyển sao cho tâm của nó trùng với vị trí $\mathbf{p} \in S_p$
- $\mathbf{Z} + \mathbf{p}$ là dịch chuyển của **Z** đến vị trí **p** trong S_p
- Tập các vị trí trong ảnh được phác hoạ bởi $\mathbf{Z} + \mathbf{p}$ gọi là **Z**-*hàng xóm* của **p**, kí hiệu bằng $N\{\mathbf{I}, \mathbf{Z}\}(\mathbf{p})$



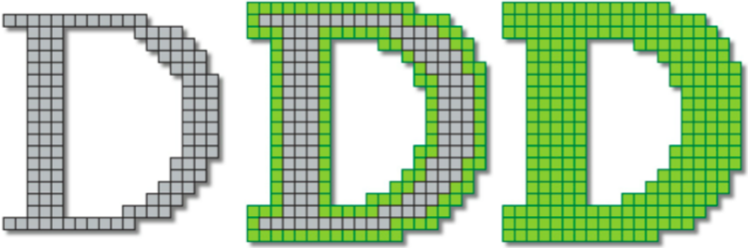


Thành phần cấu trúc phản chiếu

- Cho $\mathbf{Z}$ là SE và $\Im$ là hình vuông gồm vị trí các pixels mà chứa tập $\{(r,c),(-r,-c)|(r,c)\in \text{supp}(\mathbf{Z})\}$
- $\check{\mathbf{Z}}(\rho,\chi)=\mathbf{Z}(-\rho,-\chi)$ với mọi $(\rho,\chi)\in\Im$ là thành phần cấu trúc phản chiếu
- $\check{\mathbf{Z}}$ là $\mathbf{Z}$ quay 180 độ quanh tâm của nó



Giãn nở



Có rất nhiều định nghĩa về phép giãn nở. Ba trong số các định nghĩa đó thể áp dụng cho ảnh nhị phân là:

1. Tập tất cả các vị trí pixel **p** trong không gian ảnh mà thoả mãn giao của $\check{\mathbf{Z}} + \mathbf{p}$ với ảnh **I** là khác rỗng

$$\mathbf{I} \oplus B = \{\mathbf{p} \in S_p \big| \left[(\check{\mathbf{Z}} + \mathbf{p}) \cap \mathbf{I}\right] \neq \emptyset\}$$

2. Hợp các bản sao của SE, mỗi bản sao được dịch chuyển đến mỗi vị trí pixel trong support của ảnh

$$\mathbf{I} \oplus \mathbf{Z} = \bigcup_{\mathbf{p} \in \mathrm{supp}\{\mathbf{I}\}} (\mathbf{Z} + \mathbf{p})$$

3. Hợp các bản sao của ảnh, mỗi bản sao được dịch chuyển tới vị trí pixel trong suport của SE

$$\mathbf{I} \oplus \mathbf{Z} = \bigcup_{\mathbf{p} \in \mathrm{supp}\{\mathbf{Z}\}} (\mathbf{I} + \mathbf{p})$$

$$\{\mathbf{p} \in S_p \big| [(\check{\mathbf{Z}} + \mathbf{p}) \cap \mathbf{I}] \neq \emptyset\}$$

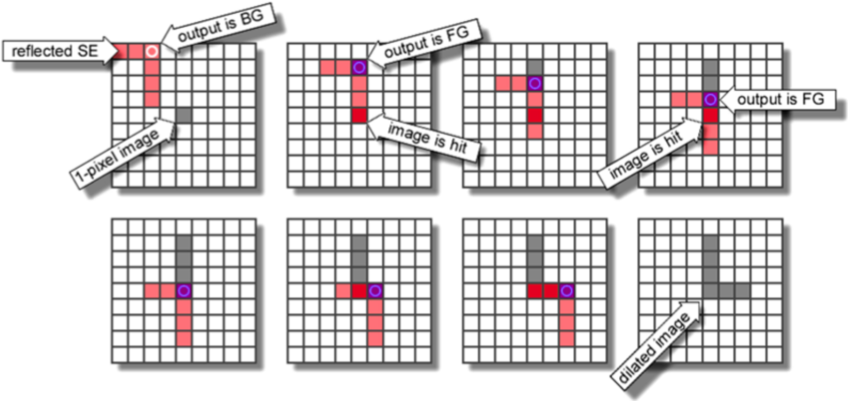
ảnh gốc

ảnh gốc /giản nở

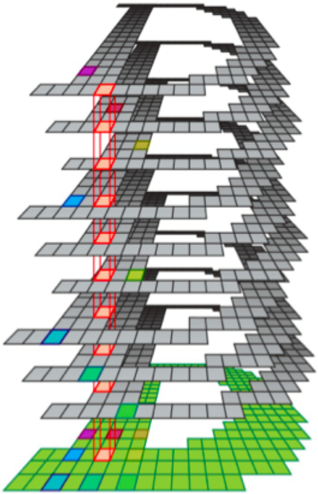
ảnh giản nở

SE $= \mathbf{Z}_8$

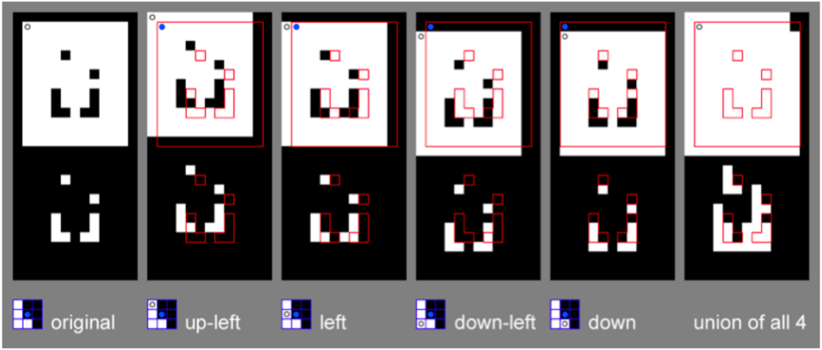
Giãn nở sử dụng SE phản chiếu



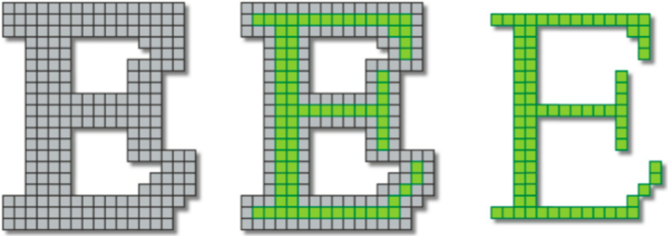
Giãn nở bằng dịch chuyển ảnh



Giãn nở bằng dịch chuyển ảnh



Co ảnh



Co ảnh nhị phân

Có rất nhiều định nghĩa về phép co. Ba trong số các định nghĩa đó thể áp dụng cho ảnh nhị phân là:

1. Tập tất cả các vị trí pixel **p** trong không gian ảnh mà thoả mãn $\mathbf{Z} + \mathbf{p}$ là tập con của **I**

$$\mathbf{I} \ominus \mathbf{Z} = \{\mathbf{p} \in S_p \big| \mathbf{Z} + \mathbf{p} \subset \mathbf{I}\}$$

2. Giao các bản sao của SE phản chiếu, mỗi bản sao được dịch chuyển đến mỗi vị trí pixel trong support của ảnh

$$\mathbf{I} \ominus \mathbf{Z} = \bigcap_{\mathbf{p} \in \mathrm{supp}\{\mathbf{I}\}} (\check{\mathbf{Z}} + \mathbf{p})$$

3. Giao các bản sao của ảnh, mỗi bản sao được dịch chuyển tới vị trí pixel trong suport của SE phản chiếu

$$\mathbf{I} \ominus \mathbf{Z} = \bigcup_{\mathbf{p} \in \mathrm{supp}\{\check{\mathbf{Z}}\}} (\mathbf{I} + \mathbf{p})$$

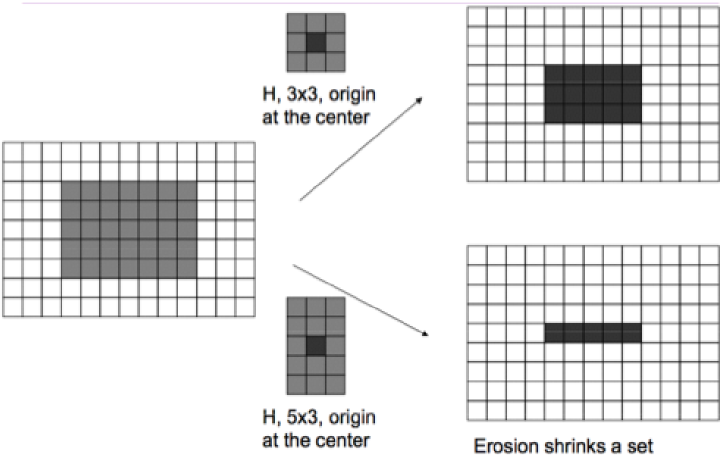
Phép co

$$\{\mathbf{p} \in S_p \big| \mathbf{Z} + \mathbf{p} \subset \mathbf{I}\}$$

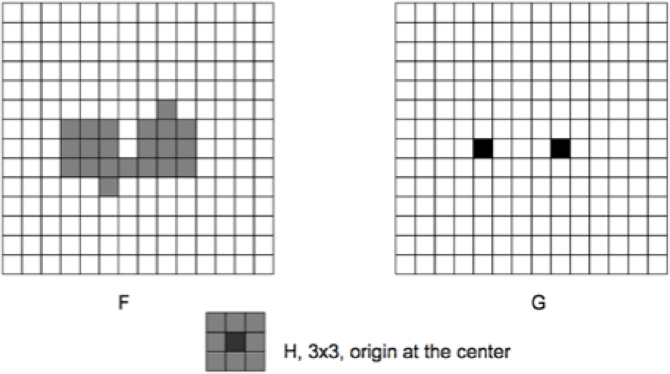
ảnh gốc co /ảnh gốc ảnh co

SE $= \mathbf{Z}_8$

Ví dụ về phép co ảnh



Ví dụ về phép co ảnh



So sánh phép co và phép co giãn nở

Ảnh chứa phép co

giãn nở chứa ảnh gốc

Co / ảnh gốc

Co / ảnh gốc /giản nở

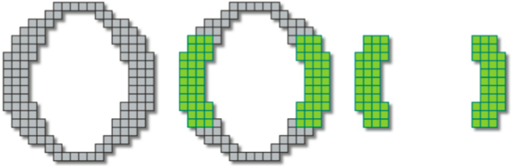
ảnh góc / giản nở

SE $= \mathbf{Z}_8$

Phép toán đóng và mở

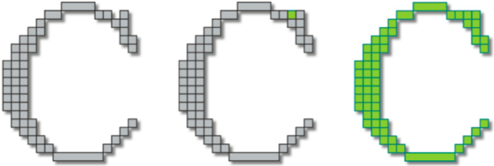
- Phép mở

$$\mathbf{I} \circ \mathbf{Z} = (\mathbf{I} \ominus \mathbf{Z}) \oplus \mathbf{Z}$$

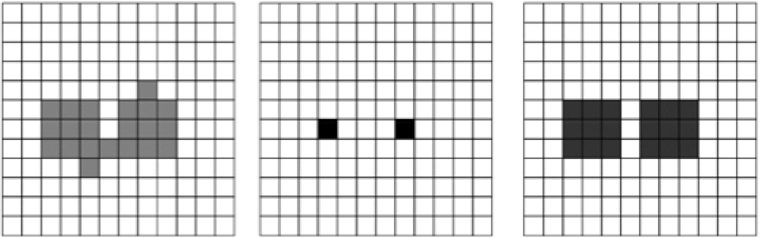


- Phép đóng

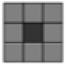
$$\mathbf{I} \bullet \mathbf{Z} = (\mathbf{I} \oplus \check{\mathbf{Z}}) \ominus \check{\mathbf{Z}}$$



Ví dụ về phép mở

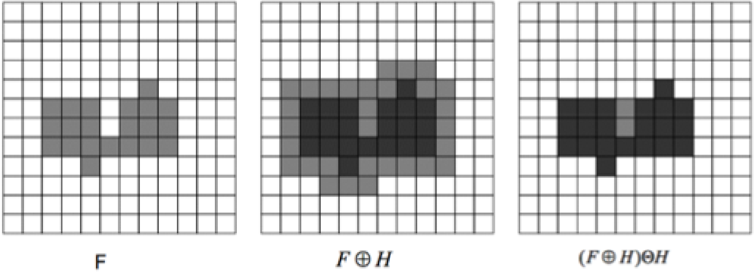


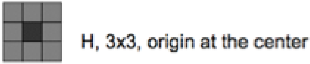
F $\qquad F \ominus H \qquad (F \ominus H) \oplus H$



H, 3x3, origin at the center

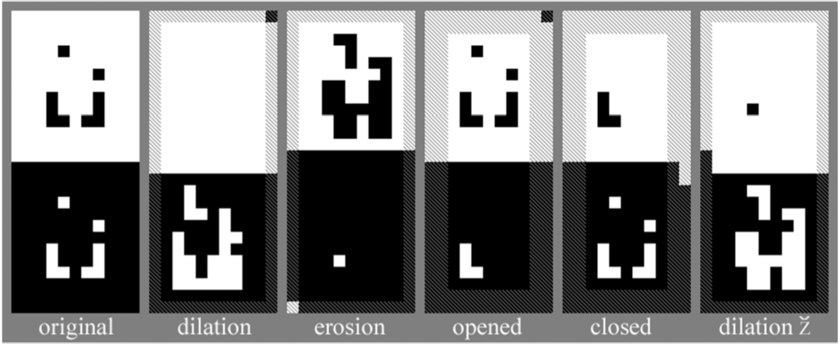
Ví dụ về phép đóng





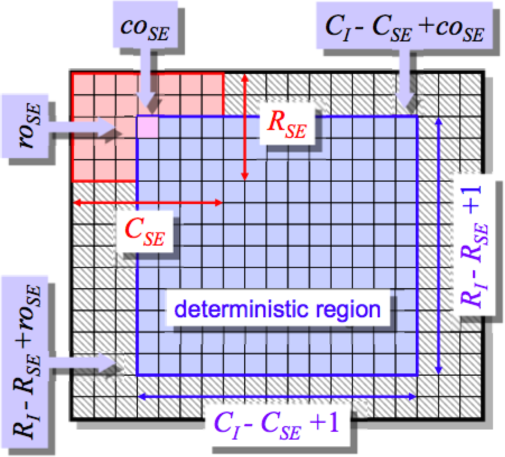
Phép toán nhị phân với SEs không đối xứng

- SE có hình dạng chữ L, kích thước 3×3, tâm ở chính giữa
- FG = các pixel màu trắng, BG = các pixel màu đen
- Các pixel gạch chéo là không rõ



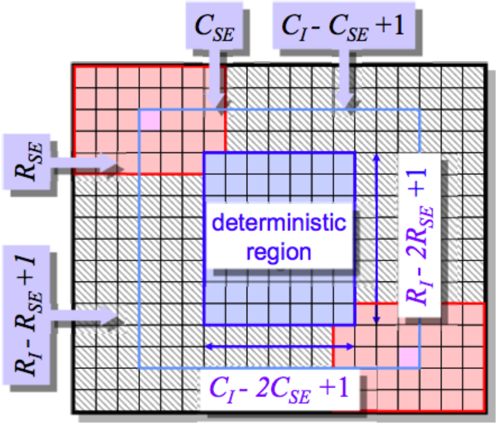
Hiệu ứng trên đường biên: Co và giãn nở

Do phép toán hình thái là phép toán trên hàng xóm, nên tồn tại các pixel xung quanh đường biên trên ảnh kết quả mà giá trị của nó chưa xác định (*nó phụ thuộc vào từng thuật toán được sử dụng*)



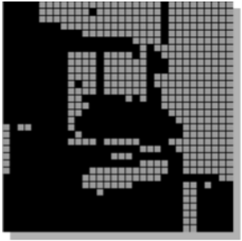
Hiệu ứng trên đường biên: Đóng và mở

Do phép toán đóng/mở là kết quả từ phép toán giãn nở và co nên đường biên của vùng xác định xa gấp 2× đường biên của ảnh giãn nở hoặc co



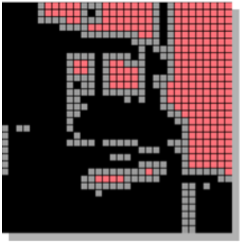
Trích chọn đường biên

ảnh nhị phân



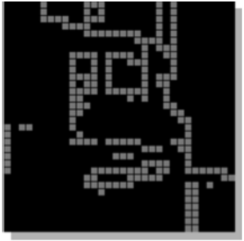
ảnh gốc

8-connected SE



co bởi hình vuông

4-conn bên trong biên



khác nhau

Đây là một phần của một ảnh lớn, nên hiệu ứng biên là không có

Khôi phục ảnh nhị phân

Loại bỏ các vùng nhỏ mà không liên kết với các đối tượng lớn hơn nhưng không làm biến dạng tính năng nhỏ của các đối tượng lớn

original

opened

reconstructed

Thuật toán cho khôi phục ảnh nhị phân

1. $\mathbf{J} = \mathbf{I} \circ \mathbf{Z}$ với $\mathbf{Z}$ là SE bất kỳ
2. $\mathbf{T} = \mathbf{J}$
3. $\mathbf{J} = \mathbf{J} \oplus \mathbf{Z}_k$ với $k = 4$ hoặc $k = 8$
4. $\mathbf{J} = \mathbf{I}$ AND $\mathbf{J}$ [chỉ lấy những pixels từ $\mathbf{J}$ mà cũng thuộc $\mathbf{I}$]
5. Nếu $\mathbf{J} \neq \mathbf{T}$ thì chuyển đến bước 2.
6. Nếu không thì dừng [$\mathbf{J}$ là ảnh khôi phục]

Bài tập

Cho ảnh $\mathbf{I} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$ và SE $= \begin{bmatrix} 1 \\ \color{red}{1} \\ 1 \end{bmatrix}$ Tính

- phản chiếu của SE
- $\mathbf{I} \oplus SE$
- $\mathbf{I}^C \ominus SE$

Tìm thành phần cấu trúc SE và phép toán tương ứng để từ ảnh ban đầu (a) ta thu được ảnh (b)

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

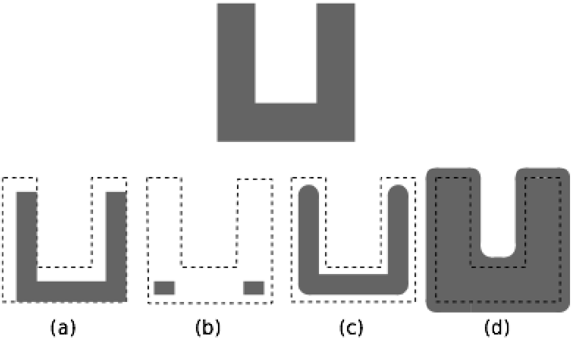
(a)

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

(b)

Bài tập

1. Tìm thành phần cấu trúc SE và phép toán hình thái tương ứng mà tạo ra các ảnh từ (a) đến (d).
2. Hãy cho biết tâm của các thành phần cấu trúc này



Chú ý: Đường đứt nét chỉ ra biên của tập ban đầu