

XỬ LÝ ẢNH

IMAGE PROCESSING

NỘI DUNG

- Tổng quan về xử lý ảnh
- Xử lý nâng cao chất lượng ảnh
- Các phương pháp phát hiện biên
- Phân vùng ảnh
- Nén ảnh

TÀI LIỆU THAM KHẢO

- *Digital Image Processing*, Rafael C. Gonzalez and Richard E. Woods, 3rd edition, Pearson Education.
- *Digital Image Processing Using MATLAB*, Rafael C. Gonzalez, Richard E. Woods and Steven L. Eddins, 2nd edition, Pearson Prentice Hall.
- *Fundamentals of Digital Image Processing*, A. K. Jain, Prentice Hall.

ĐÁNH GIÁ MÔN HỌC

- Đánh giá quá trình: 50%
 - Tham gia, trong lớp tích cực, làm bài tập về nhà
 - Kiểm tra quá trình
- Đánh giá kết thúc môn học: 50%

Chương 1

TỔNG QUAN VỀ XỬ LÝ ẢNH

INTRODUCTION TO IMAGE PROCESSING

TỔNG QUAN VỀ XỬ LÝ ẢNH

Mục đích:

- Trình bày các khái niệm, các vấn đề cơ bản của hệ thống xử lý ảnh số
- Tổng quan của một số ý tưởng, phương pháp và một số kỹ thuật sử dụng trong xử lý ảnh.

NỘI DUNG BÀI GIẢNG

- Tổng quan về xử lý ảnh số
 - Ảnh số, xử lý ảnh số là gì?
 - Lịch sử của xử lý ảnh số
 - Các giai đoạn chính trong xử lý ảnh số
 - Các ứng dụng tiêu biểu của xử lý ảnh số
- Các khái niệm cơ bản trong xử lý ảnh
- Các mối quan hệ cơ bản giữa các điểm ảnh
- Các nội dung cơ bản của xử lý ảnh

GIỚI THIỆU

“One picture is worth more than ten thousand words”

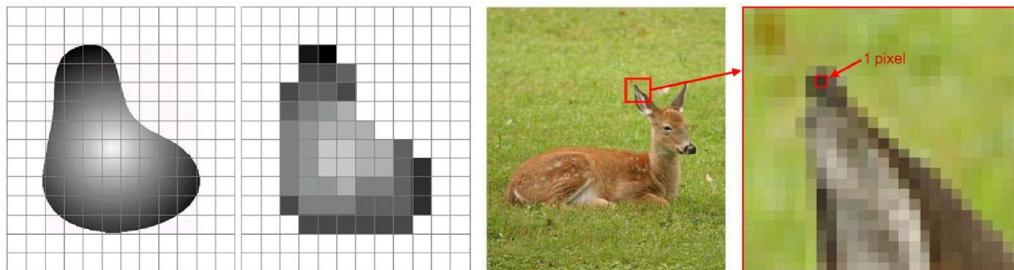
Anonymous

Ảnh số là gì?

- **Ảnh (image):** một đại diện của một đối tượng quan sát hoặc cảnh trong không gian 3 chiều kết hợp với tầm nhìn của một con người.
- **Ảnh số (digital image):** được biểu diễn bởi một hình ảnh hai chiều như là một tập hữu hạn các giá trị số, được gọi là các phần tử/yếu tố hình ảnh (picture elements) hoặc là các điểm ảnh (pixels).

Ảnh số là gì? (cont.)

- Các giá trị điểm ảnh (pixel) thường đại diện cho mức xám (gray level), màu sắc (colour)



Ảnh số là gì? (cont.)

- **Ảnh nói chung là một ma trận dữ liệu, có thể được định nghĩa:**

$$f(x, y) = \begin{bmatrix} f(0,0) & \dots & f(0, n-1) \\ \dots & \dots & \dots \\ f(m, 0) & \dots & f(m, n) \end{bmatrix}$$

- Trong đó:

- (x, y) tọa độ trong miền không gian,
- f tại bất kỳ tọa độ (x, y) gọi là cường độ.

Ảnh số là gì? (cont.)

- Các định dạng chung hình ảnh bao gồm:

- Một mẫu cho một điểm ảnh (B&W hay Grayscale)
- Ba mẫu cho mỗi điểm ảnh (Red, Green, Blue)
- Bốn mẫu cho mỗi điểm ảnh (Red, Green, Blue và "alpha" hay còn gọi là độ mờ đục - Opacity)



Xử lý ảnh số là gì?

- Xử lý ảnh là một ngành khoa học nhằm trang bị ý tưởng, phương pháp luận và các kỹ thuật để trang bị cho máy tính có khả năng xử lý ảnh, và hiểu được ảnh như con người.
- Xử lý là một quá trình phức tạp và đa dạng liên quan đến nhiều lĩnh vực khác nhau.

Xử lý ảnh số là gì? (Cont.)

- Thu nhận ảnh – biểu diễn ảnh mức thấp



123	33	234	45
67	90	12	134
56	89	54	67
111	56	67	90
34		



Những con số đại diện cho độ sáng, màu sắc của những đối tượng, nhưng chúng ta không hề có bất kỳ thông tin nào về đối tượng này, chẳng hạn như: những quyển sách, màn hình mà những con số này đại diện – vì thế được gọi là biểu diễn mức thấp.

Xử lý ảnh số là gì? (Cont.)

- Các chủ đề chính:

- Thu nhận ảnh – biểu diễn ảnh mức thấp (low-level) trong thế giới thực
- Xử lý ảnh – lọc nhiễu, làm mịn, sắc nét, độ tương phản ảnh, chuyển đổi biểu diễn
- Nén ảnh – tăng cường hiệu quả lưu trữ ảnh

Xử lý ảnh số là gì? (Cont.)

- Thu nhận ảnh – biểu diễn ảnh mức thấp



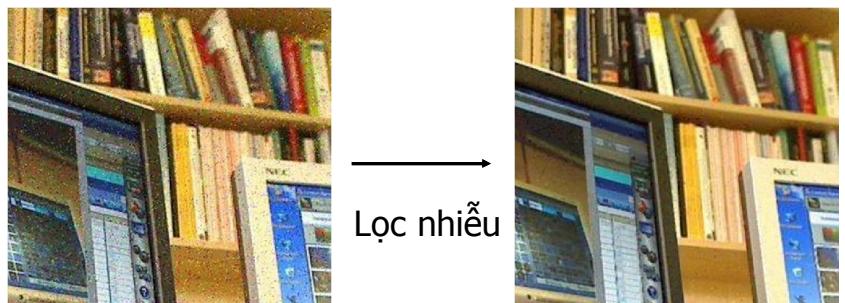
123	33	234	45
67	90	12	134
56	89	54	67
111	56	67	90
34		



- Những con số này là gì?
- Bao nhiêu số?
- Giá trị lớn/nhỏ của mỗi số là như thế nào?

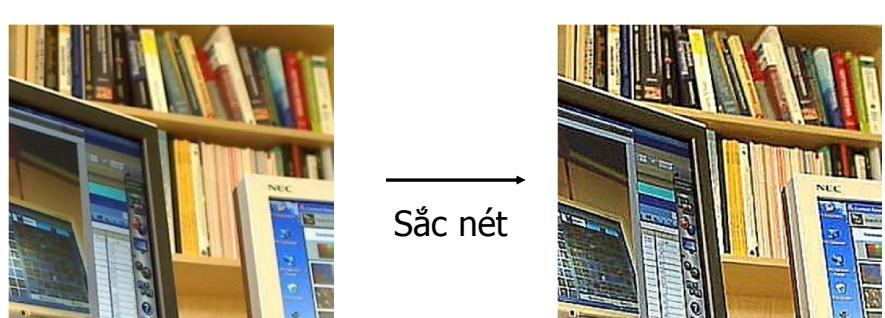
Xử lý ảnh số là gì? (Cont.)

- Xử lý ảnh – lọc nhiễu, làm mịn, sắc nét, độ tương phản ảnh, chuyển đổi biểu diễn



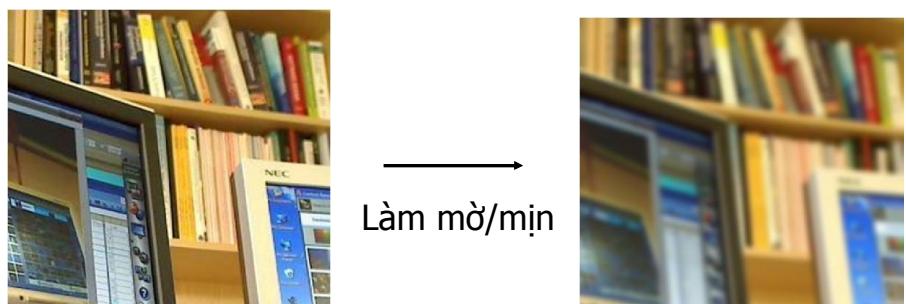
Xử lý ảnh số là gì? (Cont.)

- Xử lý ảnh – lọc nhiễu, làm mịn, sắc nét, độ tương phản ảnh, chuyển đổi biểu diễn



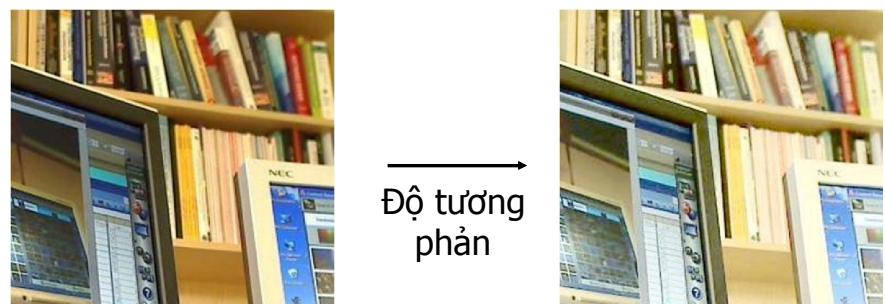
Xử lý ảnh số là gì? (Cont.)

- Xử lý ảnh – lọc nhiễu, làm mịn, sắc nét, độ tương phản ảnh, chuyển đổi biểu diễn



Xử lý ảnh số là gì? (Cont.)

- Xử lý ảnh – lọc nhiễu, làm mịn, sắc nét, độ tương phản ảnh, chuyển đổi biểu diễn



Xử lý ảnh số là gì? (Cont.)

- Xử lý ảnh – lọc nhiễu, làm mịn, sắc nét, độ tương phản ảnh, chuyển đổi biểu diễn



Trường Đại học Trà Vinh (TVU)

21

Xử lý ảnh số là gì? (Cont.)

- Nén ảnh – tăng hiệu quả lưu trữ và truyền thông ảnh



245,760 bytes

69,632 bytes

5,951 bytes

Trường Đại học Trà Vinh (TVU)

22

Lịch sử xử lý ảnh số

- Một trong những ứng dụng đầu tiên của ảnh số là trong ngành công nghệ báo chí, lần đầu tiên ảnh được gửi từ London về Mỹ thông qua hệ thống cáp ngầm vào những năm 1920, rút ngắn thời gian truyền tải ảnh từ vài tuần xuống còn vài giờ.

Chất lượng ảnh thấp

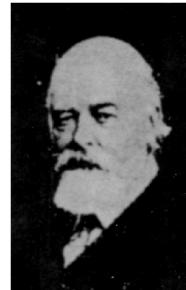


Trường Đại học Trà Vinh (TVU)

23

Lịch sử xử lý ảnh số (Cont.)

- Giữa đến cuối những năm 1920, cải tiến của hệ thống "Bartlane" cho ra những hình ảnh với chất lượng cao hơn



Chất lượng ảnh cao hơn,
Ảnh được tạo ra năm 1922



Ảnh số với 15 tông màu
Ảnh được tạo ra năm 1929

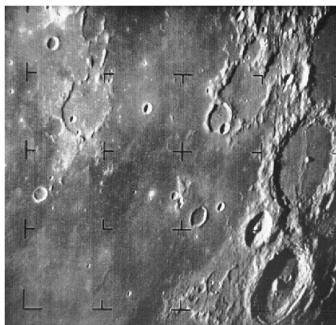
Trường Đại học Trà Vinh (TVU)

24

Lịch sử xử lý ảnh số (Cont.)

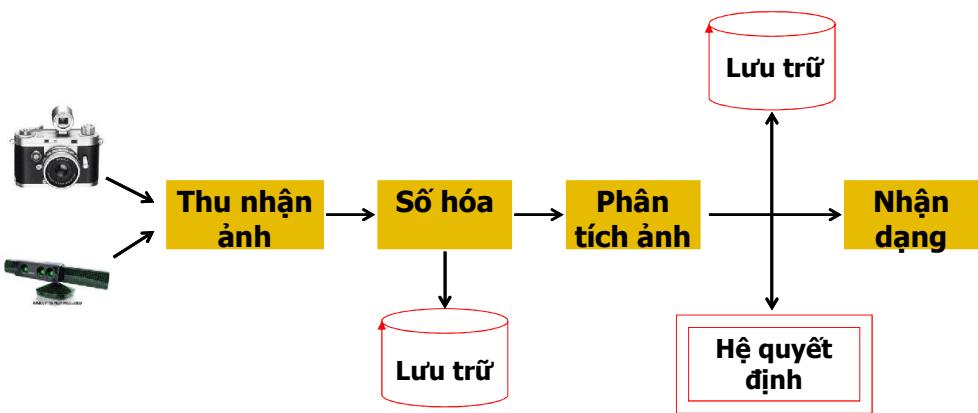
- Năm 1960, cải tiến của công nghệ điện toán và bắt đầu cuộc chạy đua không gian đã dẫn đến sự đột biến của công việc xử lý hình ảnh kỹ thuật số

- 1964: Máy tính được sử dụng để cải thiện chất lượng hình ảnh của mặt trăng thực hiện thăm dò bởi Ranger 7
- Các kỹ thuật như vậy đã được sử dụng trong các sứ mệnh không gian bao gồm cuộc đổ bộ của phi thuyền Apollo



Ảnh từ mặt trăng –Ranger 7

Các giai đoạn chính của xử lý ảnh



Lịch sử xử lý ảnh số (Cont.)

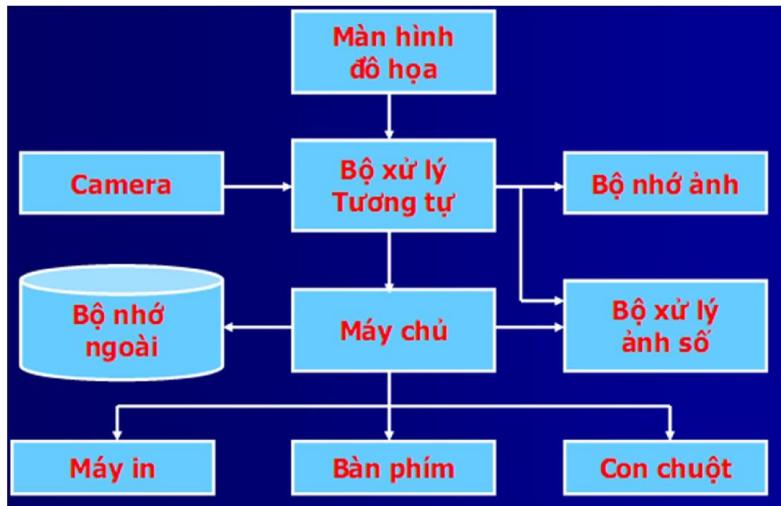
- Song song với các ứng dụng trong khám phá không gian, các kỹ thuật xử lý ảnh cũng đã bắt đầu vào cuối những năm 1960 và đầu những năm 1970 trong y học, theo dõi tài nguyên trái đất và thiên văn học.
- Đến nay xử lý ảnh đã có một bước tiến dài trong nhiều ngành khoa học, từ những ứng dụng đơn giản đến phức tạp.

Hệ thống xử lý ảnh số

- Hệ thống xử lý ảnh cho phép tiếp cận khung cảnh ảnh hoặc ảnh số ở đầu vào, thực hiện một dãy các phép xử lý đa dạng để tạo ra một ảnh ở đầu ra mang thông tin cần thiết, phân tích rút ra được các đặc trưng để cho phép hiểu được khung cảnh và ảnh.
- Các phương pháp và kỹ thuật xử lý ảnh số nhằm tạo ra các máy tính có khả năng nhìn và hiểu được ảnh như con người có thể “nhìn” ở những nơi mà con người không thể đến được

Hệ thống xử lý ảnh số (Cont.)

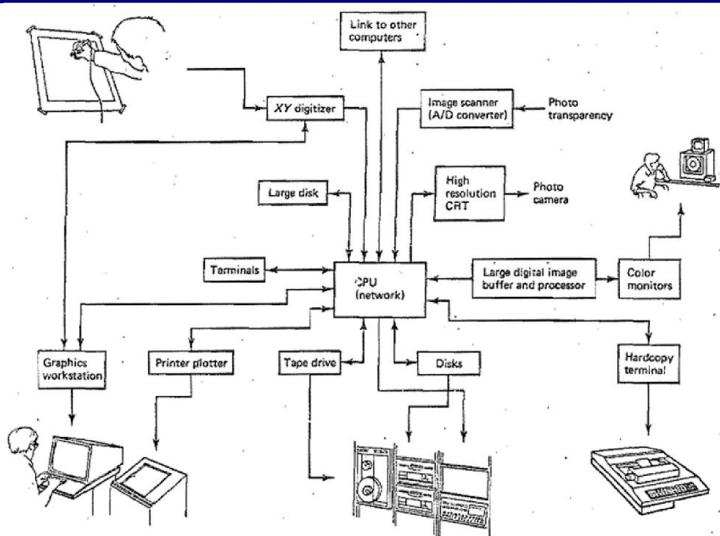
Các thành phần chính của hệ thống xử lý ảnh số



Trường Đại học Trà Vinh (TVU)

29

Hệ thống xử lý ảnh số (Cont.)



A digital image processing system (Signal and Image Processing Laboratory, University of California, Davis).

30

Các khái niệm cơ bản

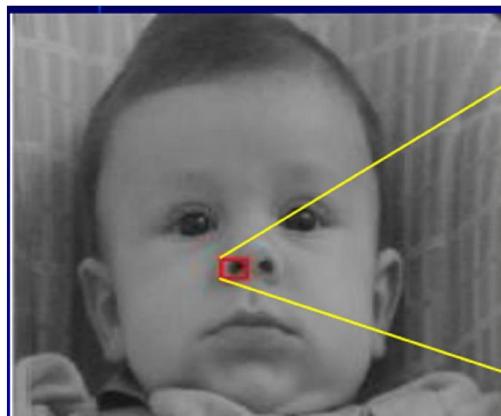
- **Ảnh (Image):** tập hợp các điểm ảnh thông qua mảng 2 chiều $I(m \times n)$, m là dòng, n là cột, lúc đó số lượng điểm ảnh là $m \times n$
 - Một hàm hai biến $f(x,y)$, trong đó (x,y) là tọa độ trong không gian hai chiều, f là hàm cường độ.

Trường Đại học Trà Vinh (TVU)

31

Các khái niệm cơ bản (Cont.)

- ## ■ Ví dụ:



9	71	61	51	49	40	35	53	86	99
8	74	53	56	48	46	48	72	85	102
1	69	57	53	54	52	64	82	88	101
7	82	64	63	59	60	81	90	93	100
4	93	76	69	72	85	94	99	95	99
7	108	94	92	97	101	100	108	105	99
6	114	109	106	105	108	108	102	107	110
5	113	109	114	111	111	113	108	111	115
0	113	111	108	106	108	110	115	120	122
3	107	106	108	108	114	120	124	124	132

Các giá trị điểm ảnh trong vùng màu đỏ trong ảnh bên trái

Trường Đại học Trà Vinh (TVU)

32

Các khái niệm cơ bản (Cont.)

- **Pixel (picture element):** điểm ảnh, mang một giá trị số $f(x,y)$ với x,y là những số nguyên chỉ vị trí pixel, f là hàm cường độ sáng hay độ xám



Mối quan hệ giữa các điểm ảnh

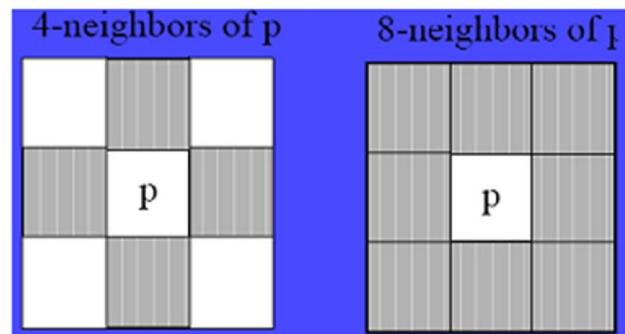
- Tính lân cận (Neighborhood)
- Tính liền kề (Adjacency)
- Tính liên thông (Connectivity)
- Đường đi (Path)
- Vùng và đường biên (Region and boundaries)

Các khái niệm cơ bản (Cont.)

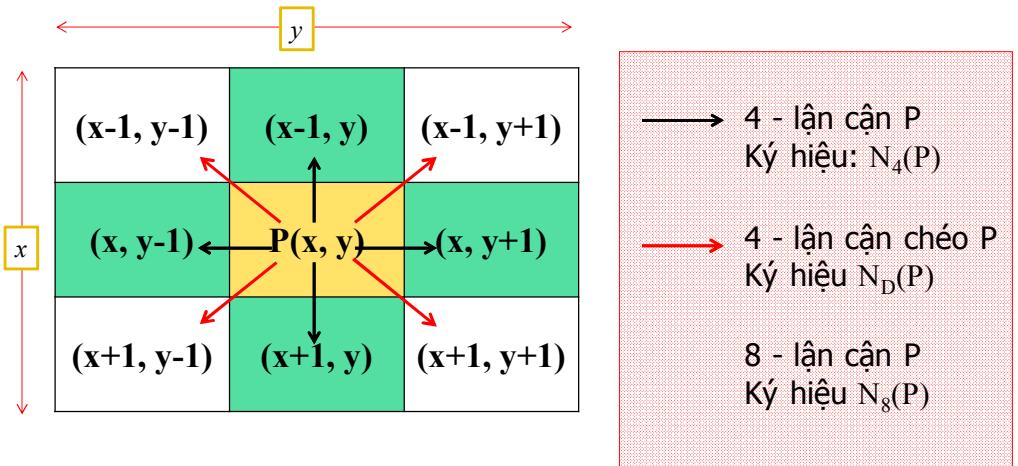
- **Bit và hệ nhị phân:** Độ sáng của mỗi pixel được biểu diễn bởi một số bit. Nếu dùng k bit cho mỗi pixel, ta có 2^k cấp độ sáng từ 0 đến 2^k-1 .
- **Mức xám (Gray level):** Mức xám là kết quả của việc mã hóa ứng với một cường độ sáng của mỗi điểm ảnh với một giá trị số. Thông thường ảnh được mã hóa dưới dạng 16, 32, 64 hay 256 mức.

Tính lân cận của điểm ảnh

- Quan hệ lân cận của điểm ảnh

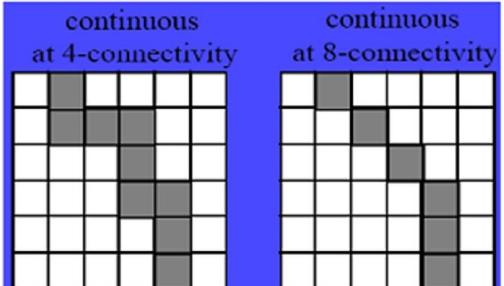


Tính lân cận của điểm ảnh



Tính liền kề

- Cho V là tập các giá trị cấp xám, ta có:
 - 4 liền kề: Hai điểm ảnh p và q với các giá trị từ V là 4 liền kề nếu q thuộc tập $N_4(p)$.
 - 8 liền kề: Hai điểm ảnh p và q với các giá trị từ V là 8 liền kề nếu q thuộc tập $N_8(p)$.



Tính liền kề (Cont.)

- Ví dụ: Cho tập dữ liệu sau đây và $V = \{1\}$, tìm 4 liền kề của điểm $p = (2,2)$

- 4 lân cận của điểm $(2,2)$ - $N_4(p)$ là: $(1,2), (2,1), (2,3)$ và $(3,2)$.
- 4 liền kề của điểm $(2,2)$ là: $(2,1)$ và $(2,3)$

	0	1	2	3	4
0	1	0	1	1	0
1	0	1	0	1	0
2	0	1	1	1	0
3	1	1	0	0	1
4	1	1	0	0	1

Tính liền kề (Cont.)

- Ex: Cho tập dữ liệu sau đây và $V = \{1, 2, 5\}$, tìm 4 liền kề $p=(4,2)$

- 4 lân cận của điểm $(4, 2)$ - $N_4(p)$ là: $(3,2), (4,1), (4,3), (5,2)$.
- 4 liền kề của điểm $p=(4,2)$ là: $(4,3)$ và $(5,2)$.

	0	1	2	3	4	5	6	7
0	1	0	2	2	4	5	6	6
1	1	1	3	2	7	4	5	6
2	6	2	4	4	6	7	6	4
3	5	7	3	3	7	3	0	6
4	1	4	2	2	2	5	0	1
5	3	5	5	3	1	0	2	3
6	3	3	5	7	1	0	2	3

Tính liên kết (Cont.)

- Ví dụ: Cho tập dữ liệu sau đây và $V = \{1\}$, tìm 8 liên kết của điểm $p=(2,2)$

- 8 lân cận của điểm $(2, 2)$
- $N_8(p)$ là: $(1,1), (1,2), (1,3), (2,1), (2,3), (3,1), (3,2), (3,3)$.
- 8 liên kè của điểm $(2,2)$
là: $(1,1), (1,3), (2,1), (2,3), (3,1)$

	0	1	2	3	4
0	1	0	1	1	0
1	0	1	0	1	0
2	0	1	1	1	0
3	1	1	0	0	1
4	1	1	0	0	1

Tính liên kết (Cont.)

- Ex: Cho tập dữ liệu sau đây và $V = \{1, 2, 5\}$, tìm 8 liên kè của điểm $p = (4,2)$

- 8 lân cận của điểm $(4, 2)$
- $N_8(p)$ là: $(3,1), (3,2), (3,3), (4,1), (4,3), (5,1), (5,2), (5,3)$
- 8 liên kè của điểm $(4,2)$
là: $(4,3), (5,1), (5,2)$

	0	1	2	3	4	5	6	7
0	1	0	2	2	4	5	6	6
1	1	1	3	2	7	4	5	6
2	6	2	4	4	6	7	6	4
3	5	7	3	3	7	3	0	6
4	1	4	2	2	5	0	1	
5	3	5	5	3	1	0	2	3
6	3	3	5	7	1	0	2	3

Tính liên kết (Cont.)

- Ex: Cho tập dữ liệu sau đây và $V = \{1, 2, 5\}$, tìm 8 liên kè của điểm $p = (4,2)$

	0	1	2	3	4	5	6	7
0	1	0	2	2	4	5	6	6
1	1	1	3	2	7	4	5	6
2	6	2	4	4	6	7	6	4
3	5	7	3	3	7	3	0	6
4	1	4	2	2	5	0	1	
5	3	5	5	3	1	0	2	3
6	3	3	5	7	1	0	2	3

Liên kết hỗn hợp

- Cho V là tập các giá trị cấp xám, ta có:
 - m liên kè (liên kè hỗn hợp): Hai điểm ảnh p và q với các giá trị từ V là m liên kè nếu:
 - q thuộc $N_4(p)$ hoặc
 - q thuộc $N_D(p)$ và tập $(N_4(p) \cap N_4(q))$ rỗng - không có điểm ảnh nào có giá trị từ V . (Tập các điểm ảnh là 4 lân cận của p và q giao nhau không có giá trị nào từ V)

Liên kết hỗn hợp (Cont.)

- Ví dụ: Cho tập dữ liệu sau đây và $V = \{1\}$, tìm m liên kết của điểm $p = (2,2)$

- 4 lân cận của điểm $(2, 2)$ – $N_4(p)$ là: $(1,2), (2,1), (2,3), (3,1)$.
- 4 lân cận chéo của $(2,2)$ – $N_D(p)$ là: $(1,1), (1,3), (3,1), (3,3)$.
- m liên kè của $(2,2)$ là: $(1,1), (2,3), (3,1)$.

0	1	2	3	4	
0	1	0	1	1	0
1	0	1	0	1	0
2	0	0	1	1	0
3	1	1	0	0	1
4	1	1	0	0	1

Trường Đại học Trà Vinh (TVU)

45

Liên kết hỗn hợp (Cont.)

- Cho tập dữ liệu sau đây và $V = \{1, 2, 5\}$, tìm m liên kè của điểm $p = (4,2)$

- 4 lân cận của điểm $(4, 2)$ – $N_4(p)$ là: $(3,2), (4,1), (4,3), (5,2)$.
- 4 lân cận chéo của $(2,4)$ là: $(3,1), (3,3), (5,1), (5,3)$.
- m liên kè của $(4,2)$ là: $(3,1), (4,3), (5,2)$.

0	1	2	3	4	5	6	7	
0	1	0	2	2	4	5	6	6
1	1	1	3	2	7	4	5	6
2	6	2	4	4	6	7	6	4
3	5	2	3	1	7	3	0	6
4	1	4	2	2	2	5	0	1
5	3	5	5	3	1	0	2	3
6	3	3	5	7	1	0	2	3

Trường Đại học Trà Vinh (TVU)

47

Liên kết hỗn hợp (Cont.)

- Cho tập dữ liệu sau đây và $V = \{1, 2, 5\}$, tìm m liên kè của điểm $p = (4,2)$

0	1	2	3	4	5	6	7	
0	1	0	2	2	4	5	6	6
1	1	1	3	2	7	4	5	6
2	6	2	4	4	6	7	6	4
3	5	2	3	1	7	3	0	6
4	1	4	2	2	2	5	0	1
5	3	5	5	3	1	0	2	3
6	3	3	5	7	1	0	2	3

Trường Đại học Trà Vinh (TVU)

46

Liên kết hỗn hợp (Cont.)

- Ex: Cho tập dữ liệu sau đây và $V = \{1, 2, 5\}$, tìm 8 liên kè của điểm $(4,2)$

0	1	2	3	4	5	6	7	
0	1	0	2	2	4	5	6	6
1	1	1	3	2	7	4	5	6
2	6	2	4	4	6	7	6	4
3	5	7	3	3	7	3	0	6
4	1	4	2	2	2	5	0	1
5	3	5	5	3	1	0	2	3
6	3	3	5	7	1	0	2	3

Trường Đại học Trà Vinh (TVU)

48

Liên kết hỗn hợp (Cont.)

- Ex: Cho tập dữ liệu sau đây và $V = \{1, 2, 5\}$, tìm 8 liên kết của điểm (4,2)

- 8 lân cận của điểm (4,2)
 - $N_8(p)$ là: (3,1), (3,2), (3,3), (4,1), (4,2), (5,1), (5,2), (5,3).
- 8 liền kề của điểm (4,2) là: (4,3), (5,1), (5,2).

	0	1	2	3	4	5	6	7
0	1	0	2	2	4	5	6	6
1	1	1	3	2	7	4	5	6
2	6	2	4	4	6	7	6	4
3	5	7	3	3	7	3	0	6
4	1	4	2	2	2	5	0	1
5	3	5	5	3	1	0	2	3
6	3	3	5	7	1	0	2	3

Ví dụ về tính liên kết của điểm ảnh

Ví dụ: Cho $V = \{1\}$, p là điểm ảnh trung tâm.

Các điểm ảnh	0	1	1
	0	1	0
	0	0	1

8 liền kề của p	0	1	-1
	0	1	0
	0	0	-1

4 liền kề của p	0	1	1
	0	1	0
	0	0	1

m liền kề của p	0	1	-1
	0	1	0
	0	0	-1

Đường đi

Đường đi giữa p và q :

- Một đường đi từ điểm p có tọa độ (x, y) đến điểm q có tọa độ (s, t) là một chuỗi tuần tự các điểm ảnh phân biệt nhau có tọa độ: $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$. Trong đó $(x_0, y_0) = (x, y)$ và $(x_n, y_n) = (s, t)$ và (x_i, y_i) liền kề với (x_{i-1}, y_{i-1}) với $1 \leq i \leq n$.
- n là độ dài của đường đi
- Nếu $(x_0, y_0) = (x_n, y_n)$ thì đường đi được gọi là một đường đi đóng hay chu trình.
- Có thể sử dụng các định nghĩa 4, 8, m liền kề tùy thuộc vào từng mục đích cụ thể.

Đường đi (Cont.)

Ví dụ:

- Tìm đường đi từ $p = (1,1)$ đến $q = (4,5)$ theo quan hệ 4 liền kề với $V = \{1\}$ trong tập S .
- Tìm đường đi từ $p = (0,6)$ đến $q = (5,1)$ theo quan hệ 8 liền kề với $V = \{1\}$ trong tập S .

	0	1	2	3	4	5	6	7
0	1	0	0	0	1	0	1	0
1	1	1	1	1	1	1	0	1
2	1	1	0	0	0	1	1	0
3	0	1	1	1	0	1	0	1
4	1	1	0	1	1	1	0	0
5	1	1	0	0	1	0	1	0

Đường đi (Cont.)

- Đường đi từ $p = (1,1)$ đến $q = (4,5)$ theo quan hệ 4 liền kề với $V = \{1\}$ trong tập S

	0	1	2	3	4	5	6	7
0	1	0	0	0	1	0	1	0
1	1	1	1	1	1	1	0	1
2	1	1	0	0	0	1	1	0
3	0	1	1	1	0	1	0	1
4	1	1	0	1	1	1	0	0
5	1	1	0	0	1	0	1	0

Trường Đại học Trà Vinh (TVU)

53

Đường đi (Cont.)

- Đường đi từ $p = (0,6)$ đến $q = (5,1)$ theo quan hệ 8 liền kề với $V = \{1\}$ trong tập S

	0	1	2	3	4	5	6	7
0	1	0	0	0	1	0	1	0
1	1	1	1	1	1	1	0	1
2	1	1	0	0	0	1	1	0
3	0	1	1	1	0	1	0	1
4	1	1	0	1	1	1	0	0
5	1	1	0	0	1	0	1	0

Trường Đại học Trà Vinh (TVU)

54

Tính liên thông của các điểm ảnh

- Cho tập con S của ảnh, hai điểm p và q trong S được gọi là liên thông nếu tồn tại một đường đi từ p đến q với tất cả các điểm trung gian nằm trong S .

	0	1	2	3	4	5	6
0	0	0	0	1	0	0	0
1	1	1	0	1	0	0	1
2	1	0	0	1	0	1	1
3	1	0	1	1	1	0	0
4	1	1	1	1	1	0	0

$p=(0,3), q=(3,2) \rightarrow$ liên thông
theo 4 liền kề với $V = \{1\}$

	0	1	2	3	4	5	6
0	0	0	0	1	0	0	0
1	1	1	0	1	0	0	1
2	1	0	0	1	0	1	1
3	1	0	1	1	1	0	0
4	1	1	1	1	1	0	0

$p=(1,1), q=(3,2) \rightarrow$ Không liên
Thông theo 4 liền kề với $V = \{1\}$

Trường Đại học Trà Vinh (TVU)

55

Khoảng cách giữa các điểm ảnh

- Định nghĩa: khoảng cách $D(p,q)$ giữa 2 điểm ảnh p tọa độ (x,y) , q tọa độ (s,t) là hàm khoảng cách:

■ $D(p,q) \geq 0$ (với $D(p,q)=0$ nếu và chỉ nếu $p=q$)

■ $D(p,q) = D(q,p)$

■ $D(p,z) \leq D(p,q) + D(q,z); z$ là một điểm ảnh khác

- Khoảng cách Euclidean:

$$De = \sqrt{(x-s)^2 + (y-t)^2}$$

Trường Đại học Trà Vinh (TVU)

56

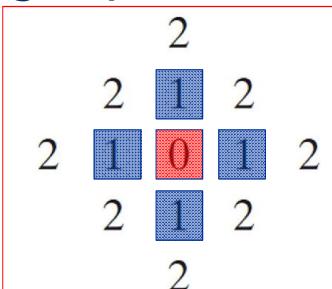
Khoảng cách khối D_4

- Khoảng cách D_4 giữa hai điểm p và q .

$$D_4(p, q) = |x - s| + |y - t|$$

- Các điểm ảnh có khoảng cách D_4 đối với điểm (x, y) nhỏ hơn hoặc bằng r tạo thành một hình thoi tâm (x, y) .

- $D_4 = 1$ là 4 lân cận của điểm (x, y)

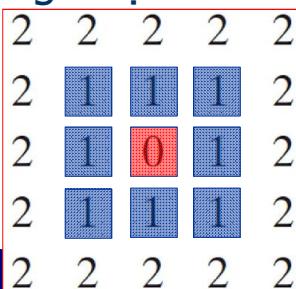


Khoảng cách khối D_8

- Khoảng cách D_8 (hay khoảng cách bàn cờ) giữa hai điểm p và q

$$D_8(p, q) = \max(|x - s|, |y - t|)$$

- Các điểm ảnh có khoảng cách D_8 đối với điểm (x, y) nhỏ hơn hoặc bằng r tạo thành một hình vuông tâm (x, y)



Câu hỏi ôn tập

1. Tại sao phải số hóa ảnh? Trình bày cách biểu diễn ảnh số trên máy tính?
2. Nêu khái niệm và định nghĩa điểm ảnh?
3. Trình bày định nghĩa mức xám
4. Trình bày mối quan hệ giữa các điểm ảnh
5. Trình bày về khoảng cách và đo khoảng cách giữa 2 điểm ảnh.

Bài tập

1. Cho 2 tập con S_1 và S_2 ,

Với quan hệ nào dưới đây thì S_1 và S_2 liền kề với nhau?

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

- a) Quan hệ 4 liền kề
- b) Quan hệ 8 liền kề
- c) Quan hệ m liền kề

Chương 2

NÂNG CAO CHẤT LƯỢNG ẢNH IMAGE ENHANCEMENT

Trường Đại học Trà Vinh (TVU)

1

NỘI DUNG

- Kiến thức cơ sở
- Một số biến đổi cấp xám cơ bản
- Xử lý lược đồ xám và cân bằng lược đồ xám
- Tăng cường ảnh dựa trên toán tử số học và logic
- Các bộ lọc ảnh làm trơn ảnh trong miền không gian
- Các bộ lọc ảnh làm nét ảnh trong miền không gian

Reading: Textbook, Chapter 1

Trường Đại học Trà Vinh (TVU)

2

Kiến thức cơ sở

- Nâng cao chất lượng là bước cần thiết trong xử lý ảnh nhằm hoàn thiện một số đặc tính của ảnh.
- Nâng cao chất lượng ảnh gồm hai công đoạn khác nhau: **tăng cường ảnh** và **khôi phục ảnh**. Tăng cường ảnh nhằm hoàn thiện các đặc tính của ảnh như:
 - Lọc nhiễu, hay làm trơn ảnh,
 - Tăng độ tương phản, điều chỉnh mức xám của ảnh,
 - Làm nổi biên ảnh.

Trường Đại học Trà Vinh (TVU)

3

Kiến thức cơ sở (Cont.)

- Miền không gian:
 - là tập hợp các điểm ảnh trong ảnh
 - phương pháp miền không gian là một thủ tục tác động lên các điểm ảnh trong miền không gian đó
- Các phép xử lý trong miền không gian tác động trực tiếp lên điểm ảnh, được ký hiệu:

$$g(x,y) = T[f(x,y)]$$

Trường Đại học Trà Vinh (TVU)

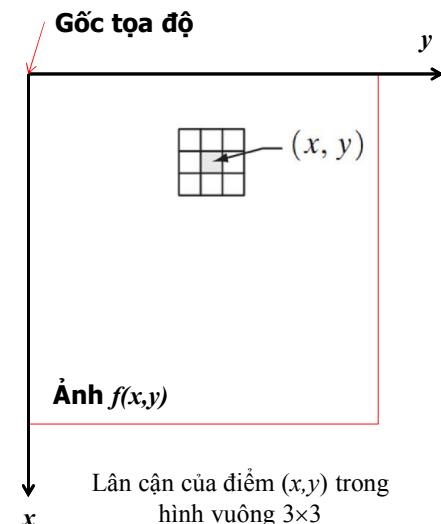
4

Kiến thức cơ sở (Cont.)

■ Trong đó:

- $f(x,y)$ là ảnh đầu vào
- $g(x,y)$ là ảnh đầu ra
- T là toán tử tác động lên f trong lân cận của điểm (x,y) .

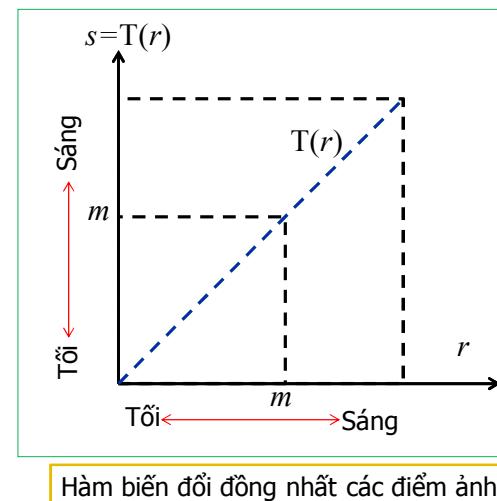
■ Tâm của lân cận (x,y) di chuyển theo từng điểm ảnh từ gốc tọa độ



Biến đổi đồng nhất

■ Hàm biến đổi đồng nhất các điểm ảnh

- Hàm biến đổi đồng nhất $T(r)$. Ảnh kết quả có độ tương phản giống ảnh gốc



Kỹ thuật xử lý điểm

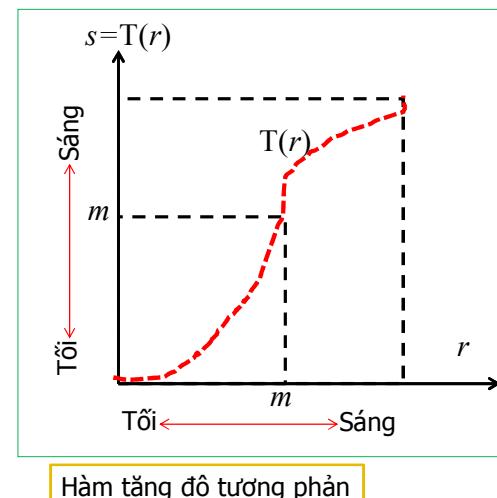
■ Toán tử **T** hoạt động tại mỗi vùng lân cận của vị trí điểm ảnh (x,y) trong ảnh f để cho ảnh đầu ra g tương ứng.

■ **T** tác động lên vùng lân cận có kích thước 1×1 (tác động lên điểm đơn) $\rightarrow g$ chỉ phụ thuộc vào giá trị của f tại điểm (x,y) , và **T** trở thành hàm biến đổi cấp xám có dạng: $s = T(r)$

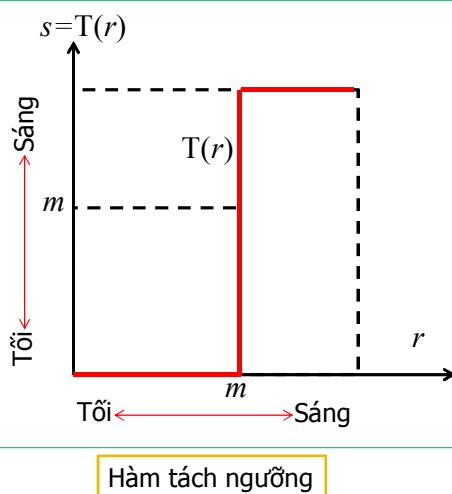
- $r = f(x,y)$
- $s = g(x,y)$

Tăng độ tương phản

- Hàm tăng độ tương phản $T(r)$. Ảnh kết quả có độ tương phản cao hơn ảnh gốc nhờ làm tối những mức xám nhỏ hơn m và tăng độ sáng những cấp xám lớn hơn m



Tách ngưỡng



Hàm tách ngưỡng

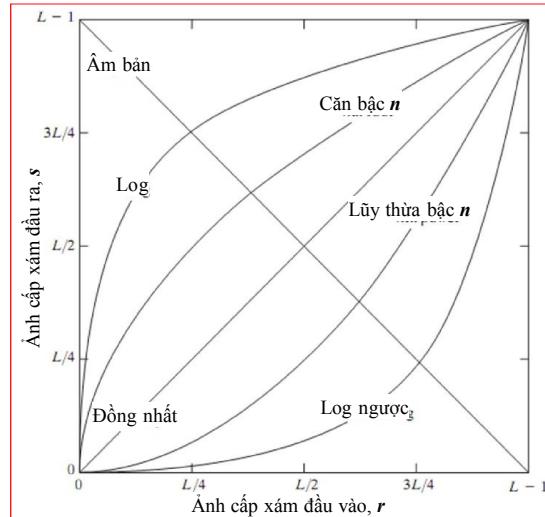
■ Hàm tách ngưỡng $T(r)$ cho kết quả là ảnh có hai mức xám (ảnh nhị phân). Những điểm ảnh có cấp xám nhỏ hơn m sẽ được quy về màu đen, những điểm ảnh có giá trị lớn hơn hoặc bằng m được quy về màu trắng.

Một số phép biến cơ bản trên ảnh xám

NHẮC LẠI MỘT SỐ KÝ HIỆU:

- Các giá trị điểm trước khi xử lý là r .
- Các giá trị điểm sau khi xử lý là s
- r và s quan hệ với nhau qua biểu thức $s=T(r)$

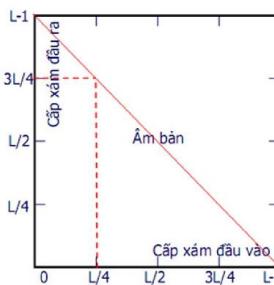
Một số phép biến cơ bản trên ảnh xám (Cont.)



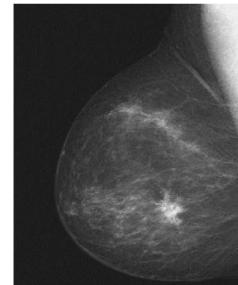
Một vài phép biến đổi mức xám để tăng cường ảnh

Phép biến đổi âm bản – phủ định

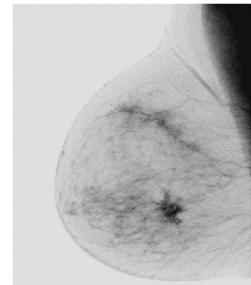
■ Phủ định của một ảnh với các cấp xám nằm trong phạm vi $[0, L-1]$ có được bằng cách sử dụng phép biến đổi âm bản: $s = L - 1 - r$



Phép biến đổi âm bản



Ảnh gốc, r



Ảnh âm bản, s

Phép biến đổi âm bản – phủ định (Cont.)

Mã lệnh giả

```
{
    for i=1:dòng
        for j=1:cột
            s(i,j)=L-1-r(i,j)
}
```

7	5	4	3	7	4	2	1	0
2	5	2	1	0	0	7	4	3
5	4	3	6	1	5	6	7	3
0	3	1	5	4	0	6	4	4
5	4	6	7	3	2	1	5	2
0	7	6	5	0	7	7	7	4
3	2	5	3	2	1	0	7	4
5	6	1	4	3	2	1	5	0
4	1	5	2	4	0	7	4	6

Ví dụ: Cho ảnh xám đa cấp r , với mức xám 3 bits, tìm ảnh âm bản s của r

0	2	3	4	0	3	5	6	7
5	2	5	6	7	7	0	3	4
2	3	4	1	6	2	1	0	4
7	4	6	2	3	7	1	3	3
2	3	1	0	4	5	6	2	5
7	0	1	2	7	0	0	0	3
4	5	2	4	5	6	7	0	3
2	1	6	3	4	5	6	2	7
3	6	2	5	3	7	0	3	1

Trường Đại học Trà Vinh (TVU)

13

Phép biến đổi logarit

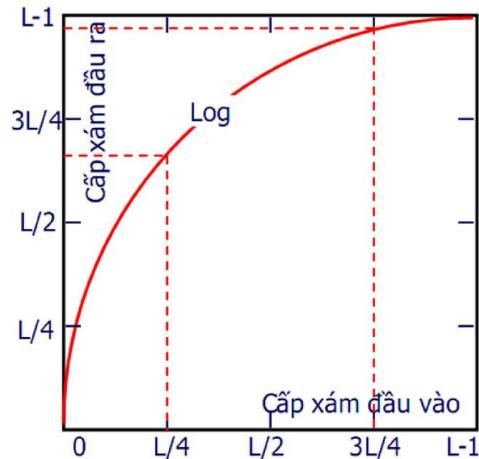
■ Dạng tổng quát của phép biến đổi logarit là:

$$s = c \times \log(1+r)$$

■ Trong đó:

- c là hằng số,
- $r \geq 0$

Phép biến đổi logarit và logarit ngược



Trường Đại học Trà Vinh (TVU)

14

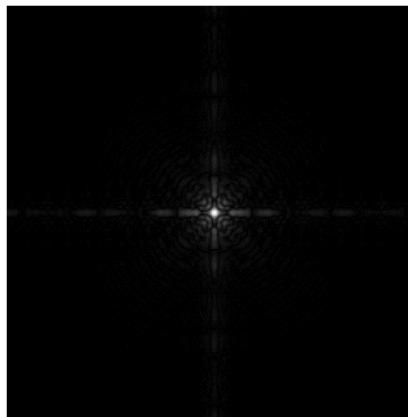
Phép biến đổi logarit (Cont.)

- Phép biến đổi Logarit ánh xạ một khoảng hẹp các giá trị cấp xám thấp trong ảnh đầu vào thành một khoảng rộng hơn các giá trị cấp xám của ảnh đầu ra.
- Ngược lại nó ánh xạ một khoảng rộng các giá trị cấp xám cao trong ảnh đầu vào thành một khoảng hẹp hơn các giá trị cấp xám của ảnh đầu ra.

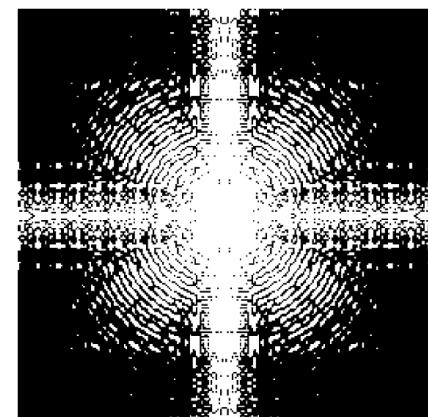
Trường Đại học Trà Vinh (TVU)

15

Phép biến đổi logarit (Cont.)



Phô Fourier



Phép biến đổi log với $c=1$

Trường Đại học Trà Vinh (TVU)

16

Phép biến đổi logarit (Cont.)

Ví dụ: Cho ảnh đa cấp xám I, với các cấp xám nằm trong đoạn $[0, 255]$. Dùng biến đổi $s = c \times \text{Log}(1+r)$ để tìm ảnh đầu ra, với hằng số $c=1$.

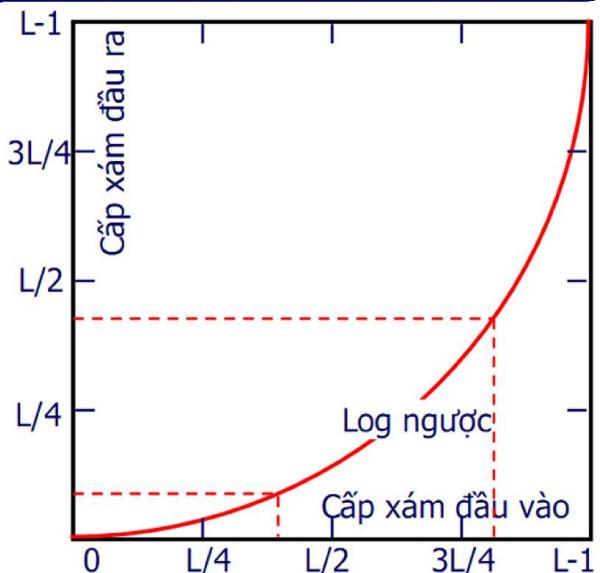
10	10	10	10	10	10	10	10	10
10	20	20	20	20	20	20	20	10
10	20	130	130	130	130	20	10	10
10	20	130	250	250	130	20	10	10
10	20	130	250	250	130	20	10	10
10	20	130	130	130	130	20	10	10
10	20	20	20	20	20	20	10	10
10	10	10	10	10	10	10	10	10



1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1
1	1	2	2	2	2	1	1	1
1	1	2	2	2	2	1	1	1
1	1	2	2	2	2	1	1	1
1	1	2	2	2	2	1	1	1
1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1

Phép biến đổi logarit ngược

- Ngược với phép biến đổi logarit

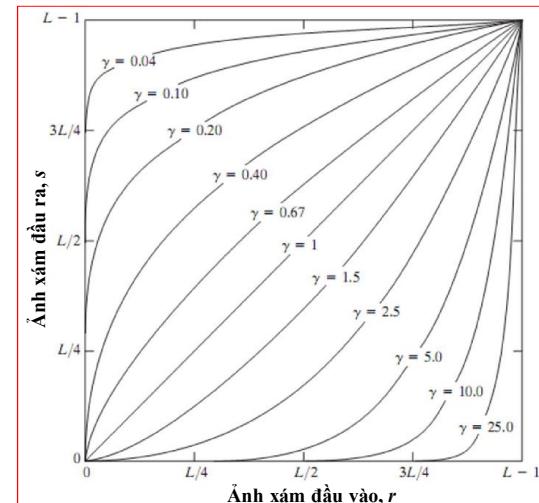


Phép biến đổi logarit ngược (Cont.)

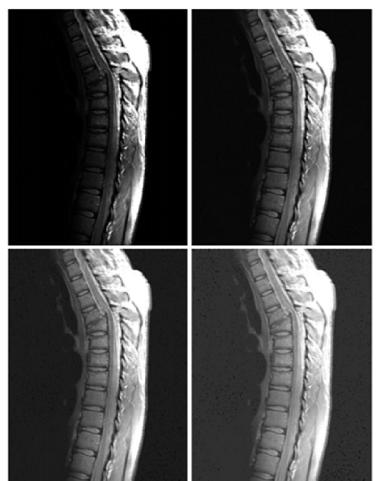
- Phép biến đổi Log ngược ánh xạ một khoảng rộng các giá trị cấp xám thấp trong ảnh đầu vào thành một khoảng rộng hơn các giá trị cấp xám của ảnh đầu ra.
- Ngược lại nó ánh xạ một khoảng hẹp các giá trị cấp xám cao trong ảnh đầu vào thành một khoảng hẹp hơn các giá trị cấp xám của ảnh đầu ra.

Phép biến đổi lũy thừa

- Dạng chung của phép biến đổi lũy thừa là: $s = c \times r^\gamma$
- Với c, γ : số dương
 - Hình bên chỉ ra các cung tương ứng của phép biến đổi lũy thừa với γ từ nhỏ đến lớn và $c = 1$.
 - Khi $\gamma=1$, phép biến đổi đồng nhất



Phép biến đổi lũy thừa (Cont.)



a
b
c
d

- (a) Ảnh cộng hưởng từ chụp xương sống người.
- (b-d) Kết quả sau khi áp dụng phép biến đổi theo phương trình $s=c \times r^\gamma$ với:
 - $c=1$ và $\gamma=\{0.6, 0.4, 0.3\}$ tương ứng hình (b), (c) và (d)

Phép biến đổi lũy thừa (Cont.)

■ $\gamma > 1$

- nén những giá trị tối
- mở rộng những giá trị sáng

■ $\gamma < 1$

- mở rộng những giá trị tối
- nén những giá trị sáng

Phép biến đổi lũy thừa (Cont.)



a
b
c
d

- (a) Ảnh chụp từ trên cao một vùng đất.
- (b-d) Kết quả sau khi áp dụng phép biến đổi theo phương trình $s=c \times r^\gamma$ với
 - $c = 1$ và $\gamma=\{3.0, 4.0, 5.0\}$ tương ứng hình (b), (c) và (d)

Phép biến đổi lũy thừa (Cont.)

- **Ví dụ:** Cho ảnh đa cấp xám I, với các cấp xám nằm trong đoạn [0, 255]. Dùng biến đổi $s = r^{0.3}$ để tìm ảnh đầu ra.

25	26	45	18	90	45	54
15	2	25	214	97	54	54
18	154	14	201	98	65	54
19	254	13	201	48	32	24
200	210	254	231	47	201	8
21	218	217	120	102	156	58
0	236	208	10	12	95	4



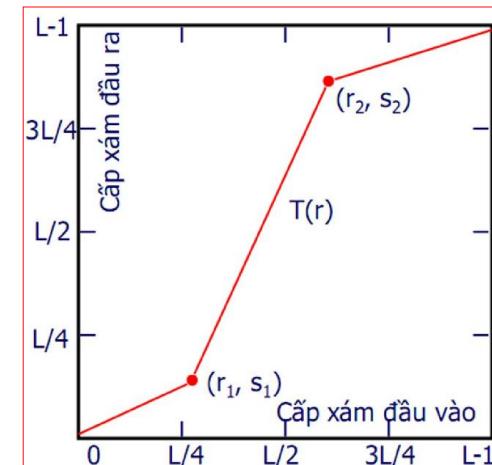
3	3	3	2	4	3	3
2	1	3	5	4	3	3
2	5	2	5	4	3	3
2	5	2	5	3	3	3
5	5	5	5	3	5	2
2	5	5	4	4	5	3
0	5	5	2	2	4	2

Các phép biến đổi tuyến tính

- Giãn độ tương phản
- Làm mỏng cấp xám
- Làm mỏng mặt phẳng bit

Giãn độ tương phản (Cont.)

- Vị trí của điểm $(r_1, s_1), (r_2, s_2)$ sẽ quy định hình dạng của phép biến đổi.
- Kết quả ảnh đầu ra của phép biến đổi bên:
 - Độ xám trong khoảng $(0, 0)$ đến (r_1, s_1) giảm.
 - Độ xám trong khoảng từ (r_1, s_1) đến (r_2, s_2) tăng.
 - Độ xám trong khoảng (r_2, s_2) đến $(L-1, L-1)$ được điều hòa.



Giãn độ tương phản

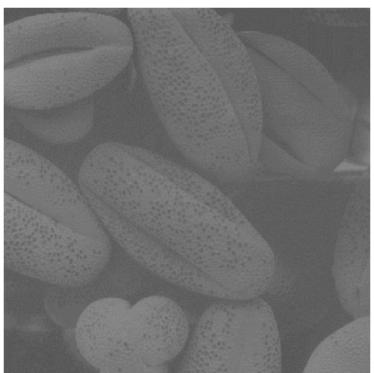
- Các ảnh có độ tương phản thấp có thể do cường độ ánh sáng kém, hoặc do bộ cảm ứng không tốt.
- Giãn độ tương phản là gia tăng các khoảng cấp xám trong ảnh.

Giãn độ tương phản (Cont.)

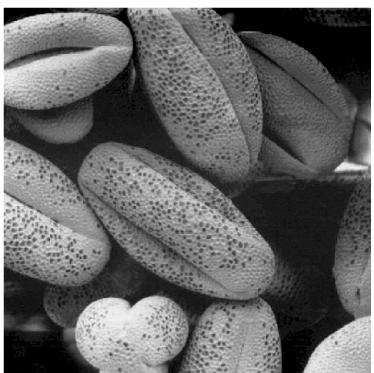
- Một thuật toán đơn giản cho việc tăng độ tương phản cho ảnh xám
 - gán a, b là giá trị nhỏ và lớn nhất trong cấp độ xám của ảnh $a/b=\min/\max(r)$
 - quét ảnh tìm giá trị nhỏ và lớn nhất gán tương ứng cho c và d
 - $s = (r - c) \times \frac{(b-a)}{(d-c)} + a$

Giãn độ tương phản (Cont.)

■ Ví dụ:



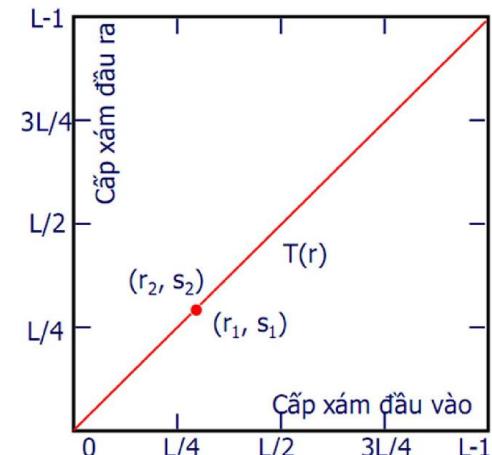
Ảnh gốc có độ tương phản thấp



Ảnh sau khi giãn độ tương phản

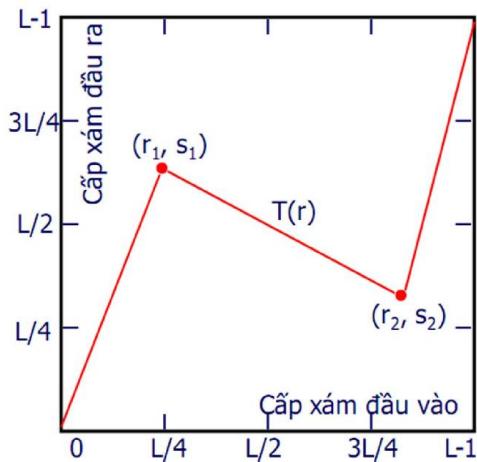
Giãn độ tương phản (Cont.)

- Nếu $r_1 = s_1$, $r_2 = s_2$, lúc đó phép biến đổi trở thành phép đồng nhất.
- Kết quả của ảnh đầu ra không thay đổi so với ảnh đầu vào.



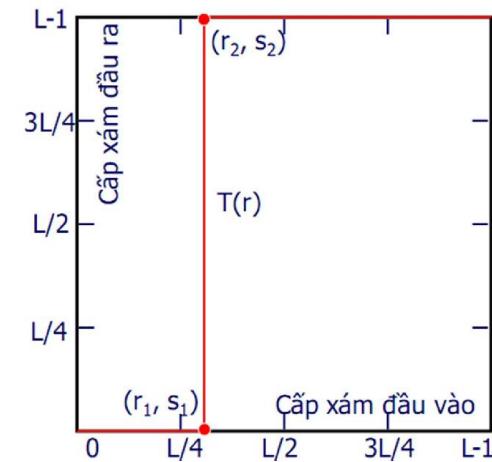
Giãn độ tương phản (Cont.)

- Vị trí của điểm (r_1, s_1) , (r_2, s_2) sẽ quy định hình dạng của phép biến đổi.
- Kết quả ảnh đầu ra của phép biến đổi bên:
 - Độ xám trong khoảng $(0, 0)$ đến (r_1, s_1) tăng.
 - Độ xám trong khoảng từ (r_1, s_1) đến (r_2, s_2) giảm.
 - Độ xám trong khoảng (r_2, s_2) đến $(L-1, L-1)$ được điều hòa.



Giãn độ tương phản (Cont.)

- Nếu $s_1 = 0$, $s_2 = L-1$, lúc đó phép biến đổi trở thành hàm phân ngưỡng.
- Kết quả ảnh đầu ra sẽ là ảnh nhị phân.



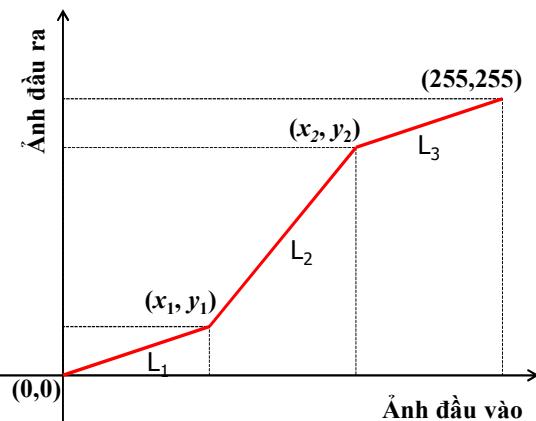
Giãn độ tương phản (Cont.)

Tóm lại

$$L_1: y = \frac{y_1}{x_1}x$$

$$L_2: y = \frac{y_2 - y_1}{x_2 - x_1} \cdot x + y_1$$

$$L_3: y = \frac{255 - y_2}{255 - x_2} \cdot x + y_2$$



$$y = \begin{cases} \frac{y_1}{x_1}x, & 0 \leq x \leq x_1 \\ \frac{y_2 - y_1}{x_2 - x_1}x + y_1, & x_1 < x < x_2 \\ \frac{255 - y_2}{255 - x_2}x + y_2, & x_2 < x < 255 \end{cases}$$

Làm mỏng mức xám

Chiếu sáng cao trong một dãy đặc trưng các cấp xám đối với một ảnh là việc thường xảy ra.

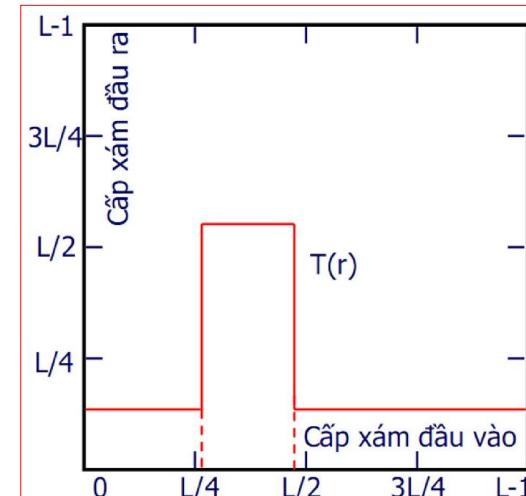
Có hai cách tiếp cận để làm mỏng mức xám:

- Hiển thị một giá trị cao cho tất cả các mức xám trong dãy cần quan tâm và hiển thị một giá trị thấp cho những mức xám còn lại.

- Hiển thị một giá trị cao cho tất cả các mức xám trong dãy cần quan tâm và giữ nguyên các mức xám còn lại.

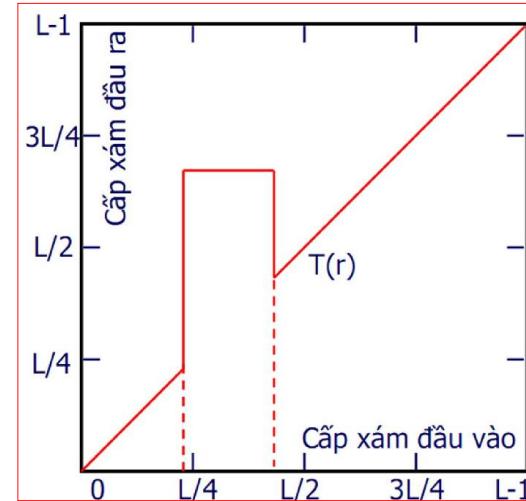
Làm mỏng mức xám (Cont.)

Hiển thị một giá trị cao cho tất cả các mức xám trong dãy cần quan tâm và hiển thị một giá trị thấp cho những mức xám còn lại.



Làm mỏng mức xám (Cont.)

Hiển thị một giá trị cao cho tất cả các mức xám trong dãy cần quan tâm và giữ nguyên các mức xám còn lại.

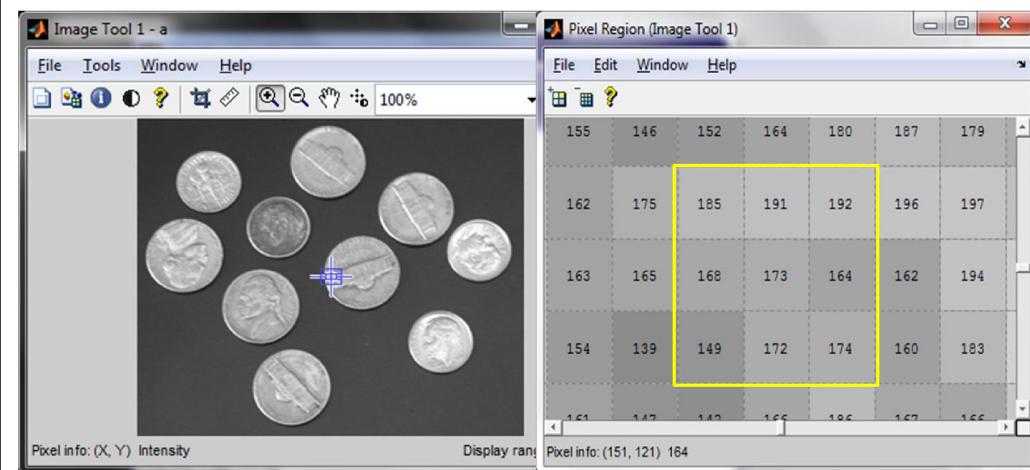


Làm mờng mặt phẳng BIT

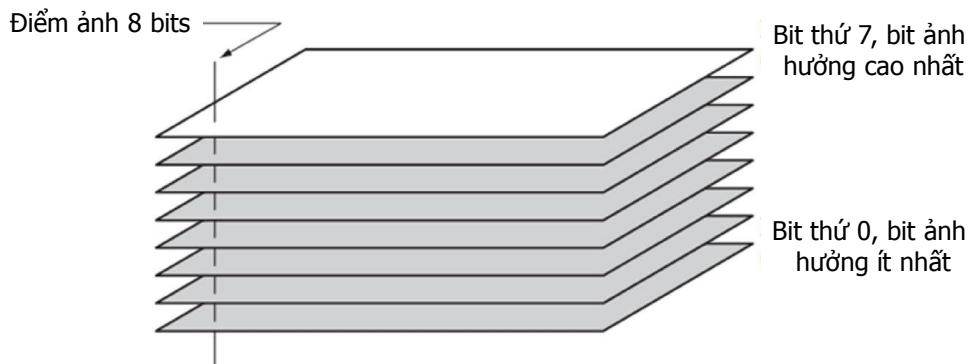
■ Như ta biết, mỗi điểm ảnh thường được mã hóa trên B bits. Nếu $B = 8$, ta có ảnh $2^8 = 256$ mức xám (ảnh nhị phân $B = 1$). Được phân định như sau:

- Bit bậc thấp có ý nghĩa ít nhất
- Bit bậc cao mang nhiều ý nghĩa thông tin ảnh

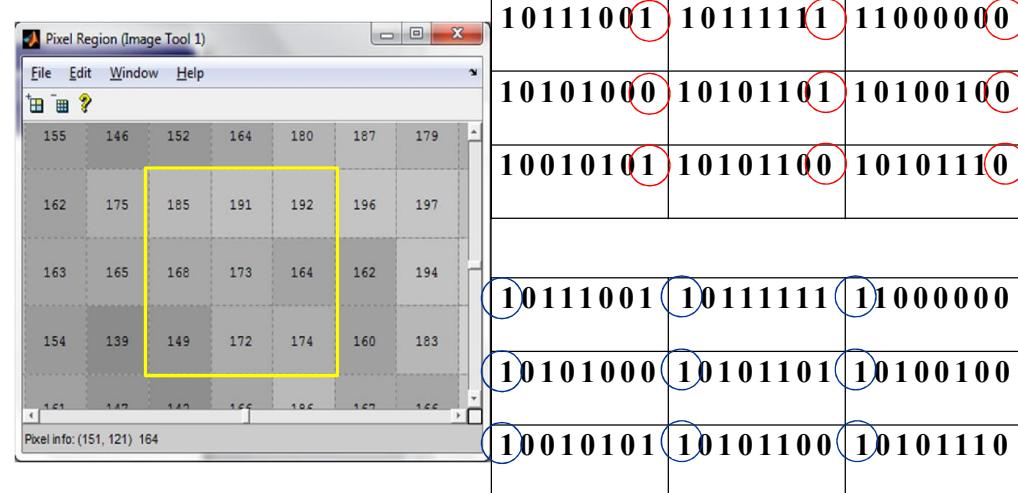
Làm mờng mặt phẳng BIT (Cont.)



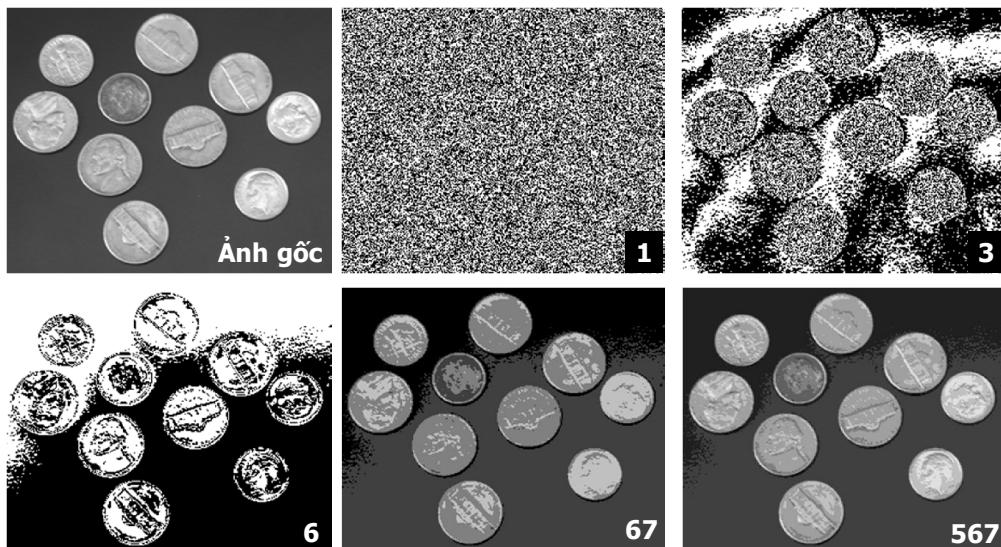
Làm mờng mặt phẳng BIT (Cont.)



Làm mờng mặt phẳng BIT (Cont.)



Làm mờ mặt phẳng BIT (Cont.)



Trường Đại học Trà Vinh (TVU)

41

Lược đồ xám của ảnh

- Lược đồ xám là một trong những đặc trưng cơ bản được sử dụng trong xử lý ảnh số.
- Lược đồ xám dùng để phục vụ cho việc nén ảnh và phân đoạn ảnh.
- Việc tính toán lược đồ xám rất đơn giản, cho nên nó được sử dụng trong các công cụ xử lý ảnh thời gian thực.

Trường Đại học Trà Vinh (TVU)

42

Lược đồ xám của ảnh (Cont.)

■ Lược đồ (histogram) xám của một ảnh với các mức xám nằm trong khoảng $[0, L-1]$ là một hàm rời rạc: $h(k)=n_k$.

■ Trong đó:

- k là mức độ xám, $k=0, 1, 2, \dots, L-1$
- n_k số lượng điểm ảnh trong ảnh có mức xám là k
- $h(k)$ là lược đồ cường độ sáng của ảnh với cấp xám bằng k

Trường Đại học Trà Vinh (TVU)

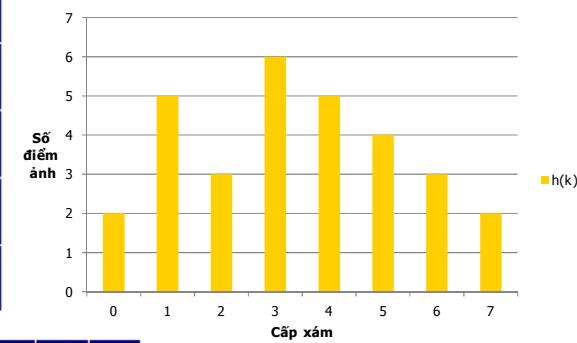
43

Lược đồ xám của ảnh (Cont.)

■ Ví dụ: Cho ảnh sau I, hãy tính lược đồ xám của ảnh I.

0	1	1	3	5	4
5	1	4	7	4	6
3	2	1	3	2	5
1	3	4	5	6	7
0	2	3	4	3	6

Lược đồ xám của ảnh I



Trường Đại học Trà Vinh (TVU)

44

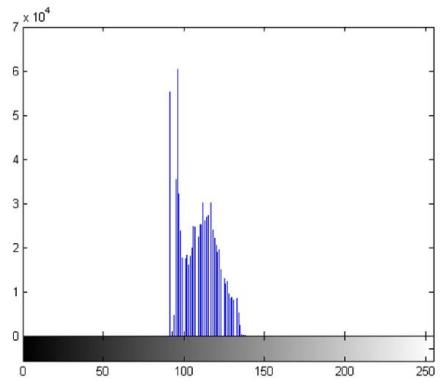
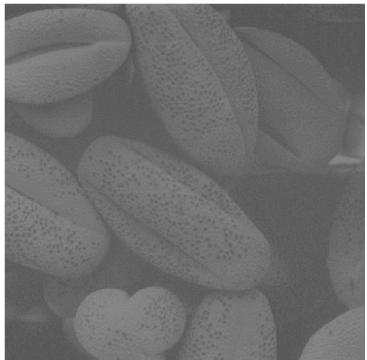
Chuẩn hóa lược đồ xám

- Để chuẩn hóa lược đồ xám ta chia giá trị n_k cho tổng số điểm ảnh trong ảnh (ký hiệu là n). Vì vậy, lược đồ xám chuẩn hóa được cho bởi công thức: $p(k)=n_k/n$.
- $p(k)$ là ước lượng xác suất xảy ra cấp xám thứ k
- Tổng các thành phần của lược đồ xám chuẩn hóa bằng 1

$$\sum_{k=0}^{L-1} p(k) = 1$$

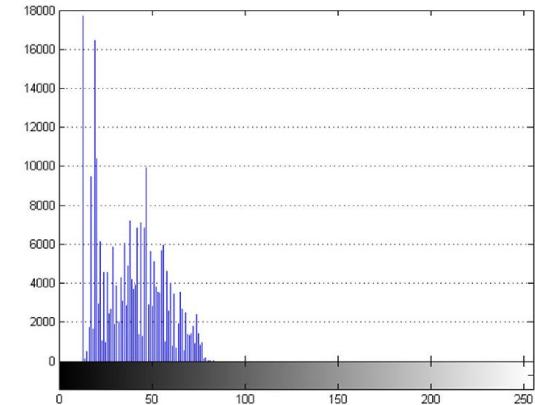
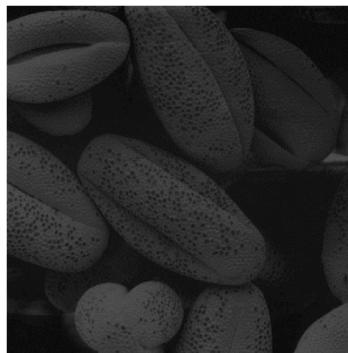
Xử lý lược đồ xám

- Trục hoành tương ứng với các giá trị cấp xám k .
- Trục tung tương ứng với các giá trị của $h(k)$ hoặc $p(k)$.



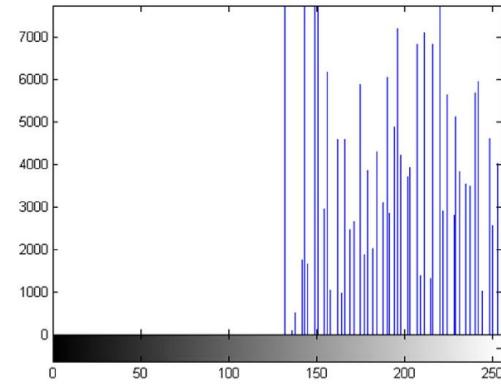
Xử lý lược đồ xám (Cont.)

- Đối với ảnh tối, lược đồ xám của ảnh sẽ phân bố chủ yếu về phía giá trị cấp xám thấp, tức là phía bên trái của đồ thị.



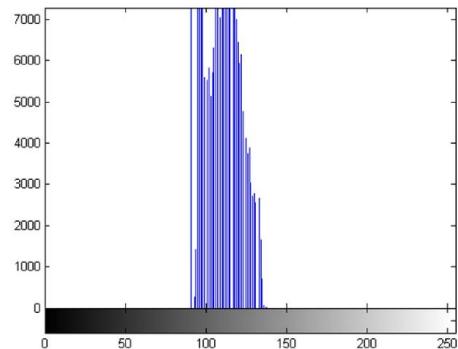
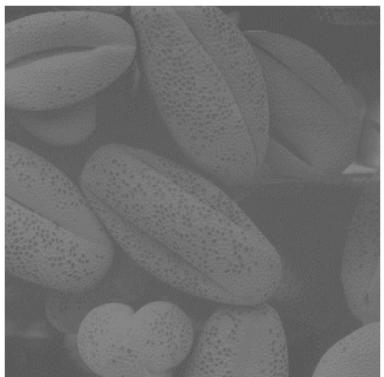
Xử lý lược đồ xám (Cont.)

- Đối với ảnh sáng, lược đồ xám của ảnh sẽ phân bố chủ yếu về phía giá trị cấp xám cao, tức là phía bên phải của đồ thị.



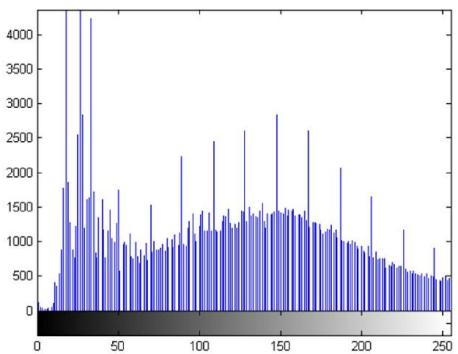
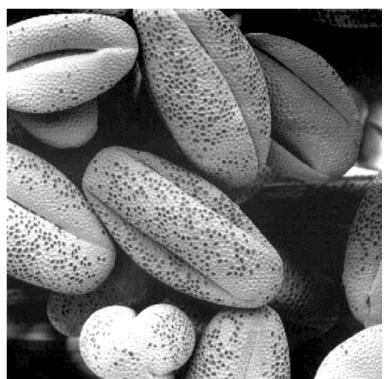
Xử lý lược đồ xám (Cont.)

- Đối với ảnh có độ tương phản thấp, lược đồ xám của ảnh sẽ phân bố chủ yếu ở các giá trị cấp xám trung bình, tức là phía giữa của đồ thị.



Xử lý lược đồ xám (Cont.)

- Đối với ảnh có độ tương phản cao, lược đồ xám của ảnh sẽ phân bố đều trên tất cả các giá trị cấp xám, tức là đều trên toàn đồ thị.



Cân bằng lược đồ xám

- Ảnh đầu vào có thể:
 - Tối → không nhìn rõ nét,
 - Sáng → mờ,
 - Độ tương phản thấp → khó nhìn thấy các đối tượng.
- Xử lý để ảnh đầu ra rõ hơn, có nhiều thông tin hơn.
- Quá trình xử lý là ánh xạ mỗi điểm ảnh với cấp xám k trong ảnh đầu vào thành điểm ảnh tương ứng với cấp xám s_k trong ảnh đầu ra.

Phương pháp cân bằng lược đồ

- Vào:** Ảnh I có L cấp xám, $k=0, 1, \dots, L-1$. Số điểm ảnh trong ảnh I là n .
- Ra:** Ảnh J với các cấp xám $s_k \in [0, L-1]$, $k=0, 1, \dots, L-1$.
- Giải thuật:**
 - Tính số điểm ảnh có cấp xám k , ký hiệu n_k ($k=0, 1, \dots, L-1$) trong ảnh I .
 - Tính xác suất $p(k)$ xảy ra đối với cấp xám k trong ảnh I :

$$p(k) = \frac{n_k}{n}$$

Phương pháp cân bằng lược đồ (Cont.)

3. Tính $T(k)$ là xác suất xảy ra cấp xám nhỏ hơn hoặc bằng k :

$$T(k) = \sum_{j=0}^k p(j) = \frac{1}{n} \sum_{j=0}^k n_j$$

4. Cấp xám s_k ảnh đầu ra J tương ứng với cấp xám k của ảnh đầu vào I được tính theo công thức:

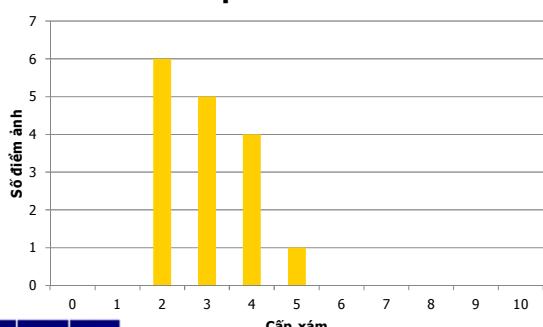
$$s_k = \text{round} [(L-1)T(k)]$$

5. Xây dựng ảnh đầu ra J bằng cách ánh xạ tương ứng cấp xám k của ảnh đầu vào thành cấp xám s_k của ảnh đầu ra.

Phương pháp cân bằng lược đồ (Cont.)

- Ví dụ: Cho ảnh đầu vào I có 10 cấp xám. Cân bằng lược đồ xám ảnh I

2	3	3	2
4	2	4	3
3	2	3	5
2	4	2	4



k	0	1	2	3	4	5	6	7	8	9
n_k	0	0	6	5	4	1	0	0	0	0

Phương pháp cân bằng lược đồ (Cont.)

- Ví dụ (tiếp theo):

k	0	1	2	3	4	5	6	7	8	9
n_k	0	0	6	5	4	1	0	0	0	0
$\sum_{j=0}^k n_j$	0	0	6	11	15	16	16	16	16	16
T(k)	0	0	6/16	11/16	15/16	16/16	16/16	16/16	16/16	16/16
$s_k = 9 \times T(k)$	0	0	3.3 ~3	6.1 ~6	8.4 ~8	9	9	9	9	9

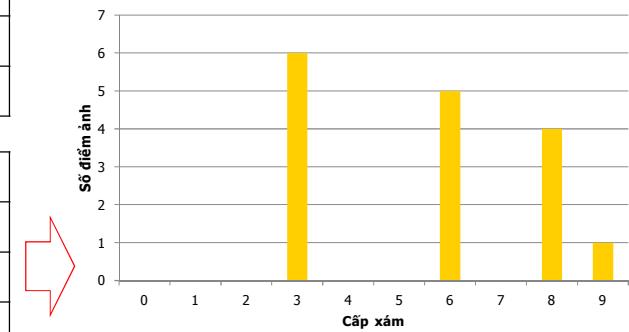
Phương pháp cân bằng lược đồ (Cont.)

k	0	1	2	3	4	5	6	7	8	9
s_k	0	0	3	6	8	9	9	9	9	9

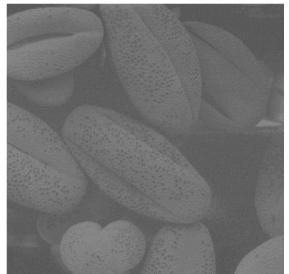
2	3	3	2
4	2	4	3
3	2	3	5
2	4	2	4

3	6	6	3
8	3	8	6
6	3	6	9
3	8	3	8

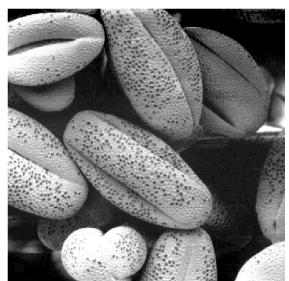
Lược đồ ngưỡng ảnh I sau khi cân bằng



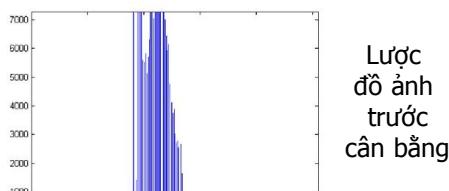
Phương pháp cân bằng lược đồ (Cont.)



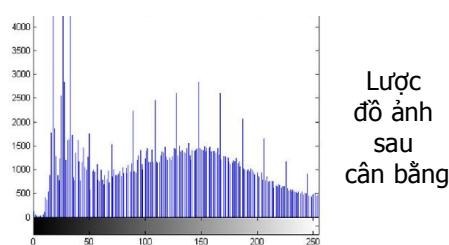
Ảnh
trước
cân
bằng



Ảnh
sau
cân
bằng



Lược
đồ ảnh
trước
cân
bằng



Lược
đồ ảnh
sau
cân
bằng

Phương pháp cân bằng lược đồ (Cont.)

- Ex: Cho ảnh I sau với 8 cấp xám, hãy cân bằng lược đồ ảnh I.

0	1	2	3	2
2	2	2	3	4
3	0	2	3	3
0	2	0	2	0
1	3	2	2	2

Toán tử số học và logic

■ Toán tử số học:

- Có 4 phép toán: **+**, **-**, ***** và **/**

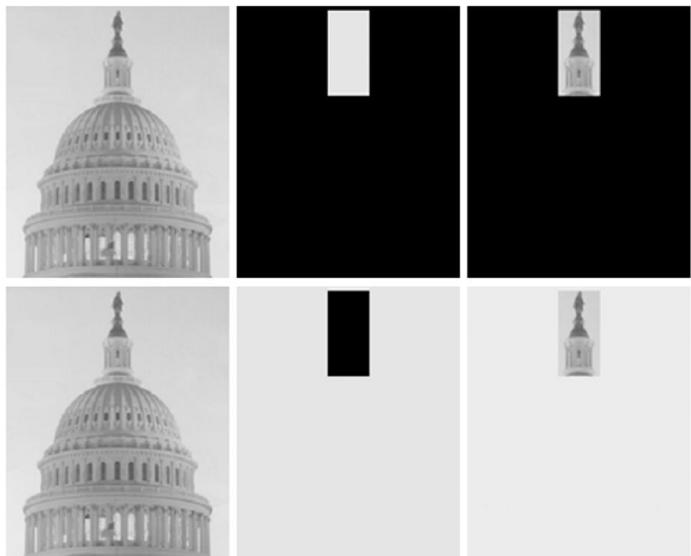
■ Toán tử logic

- Có 4 phép toán: **NOT**, **AND**, **OR**, **XOR**

Nâng cao ảnh dựa vào các phép toán số học và logic

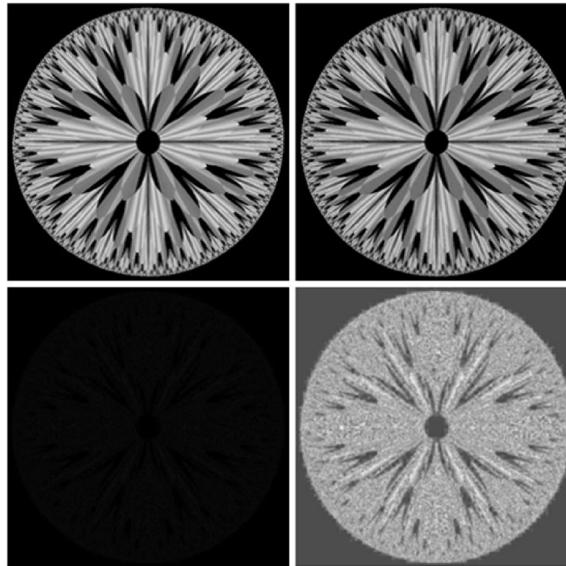
- Toán tử số học và logic đòi hỏi phải thực hiện trên từng điểm giữa hai hay nhiều ảnh.
- Ngoại trừ phép toán **NOT** thực hiện trên một ảnh.
- Toán tử logic thực hiện trên ảnh cấp xám, các điểm ảnh được xử lý như là các số nhị phân.
- Màu trắng được biểu diễn bởi số 1 và màu đen được biểu diễn bởi số 0.
- Toán tử NOT đồng nghĩa với phép biến đổi âm bản.

Nâng cao ảnh dựa vào các phép toán logic



- | | | |
|---|---|---|
| a | b | c |
| d | e | f |
- (a) **Ảnh gốc**
(b) ảnh mặt nạ
AND
(c) kết quả khi áp
dụng phép toán
AND trên ảnh (a)
và (b).
- (d) **Ảnh gốc**
(e) mặt nạ ảnh OR,
(f) kết quả ảnh sau
khi dùng phép toán
OR trên ảnh (d) và
(e).

Ví dụ về phép trừ ảnh



- | | |
|---|---|
| a | b |
| c | d |
- (a) **Ảnh gốc**
(b) Kết quả khi đặt
4 bit thấp của ảnh
gốc về 0
(c) Ảnh kết quả khu
dùng ảnh (a) trừ
ảnh (b)
(d) Ảnh kết quả sau
khi cân lược đồ
xám ảnh (c).

Phép trừ ảnh

- Sự khác biệt giữa hai ảnh $f(x, y)$ và $h(x, y)$ được tính bởi công thức:

$$g(x, y) = f(x, y) - h(x, y)$$

- Như vậy sự khác biệt của hai ảnh chính là sự khác biệt tương ứng của từng cặp điểm ảnh.
- Ứng dụng của phép trừ ảnh được sử dụng rộng rãi trong lĩnh vực y học để tìm ra sự sai khác của các ảnh X-quang.

Trung bình ảnh

- Xét ảnh nhiễu $g(x, y)$ với các thành phần nhiễu $\eta(x, y)$ nằm lẫn trong ảnh gốc $f(x, y)$ như sau:

$$g(x, y) = f(x, y) + \eta(x, y)$$

- Để giảm nhiễu trong ảnh, ta thêm vào các ảnh nhiễu để có tập các ảnh nhiễu $\{g_i(x, y)\}$, sau đó tính giá trị trung bình của tập ảnh nhiễu này.

Trung bình ảnh (Cont.)

- Với K ảnh nhiễu khác nhau $g_i(x, y)$, $i = 1, 2, \dots, K$, ta có:

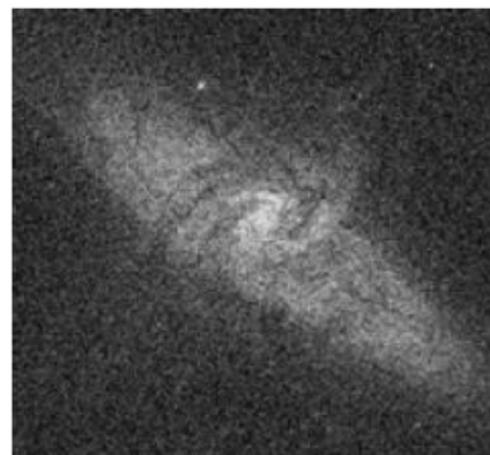
$$\bar{g}(x, y) = \frac{1}{K} \sum_{i=1}^K g_i(x, y)$$

Lúc đó: $E\{\bar{g}(x, y)\} = f(x, y)$

Khi K lớn thì $\bar{g}(x, y)$ tiến đến $f(x, y)$

- Trung bình ảnh được ứng dụng nhiều lĩnh vực thiên văn học.

Ví dụ về trung bình ảnh (2/3)



Hình ảnh của thiên hà NGC 3314 đã được sửa đổi thông qua phép làm nhiễu Gaussian với trung bình các giá trị nhiễu bằng 0.

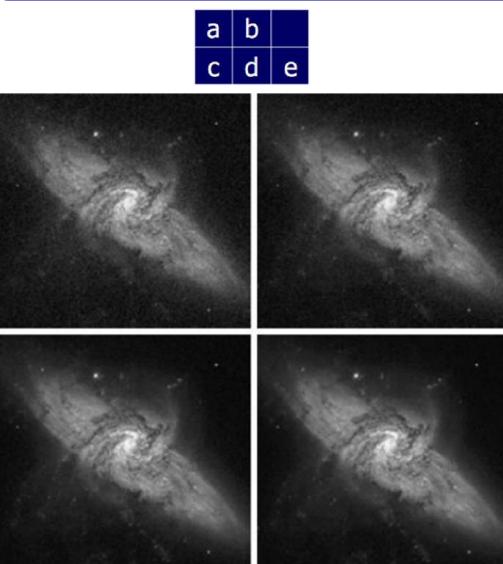
Ví dụ về trung bình ảnh (1/3)



Hình ảnh của thiên hà NGC 3314, do kính viễn vọng không gian Hubble của NASA chụp.

Thiên hà này nằm cách trái đất khoảng 140 triệu năm ánh sáng.

Ví dụ về trung bình ảnh (3/3)



- Ảnh (a) \rightarrow (d) kết quả của việc lấy trung bình 8, 16, 64, 128 ảnh nhiễu.
- Kết quả ảnh trung bình đạt được với $K = 128$, ảnh (d) gần giống với ảnh gốc (e).

Lọc không gian

- Bộ lọc là một hình vuông hoặc hình chữ nhật chứa các giá trị vô hướng. Bộ lọc còn được gọi bằng các thuật ngữ khác như: mặt nạ, nhân, mẫu, cửa sổ.
- Các giá trị trong bộ lọc được gọi là các hệ số của bộ lọc.
- Thông thường người ta sử dụng các bộ lọc có kích thước là số lẻ như: 3×3 , 5×5 , ...

Lọc không gian

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

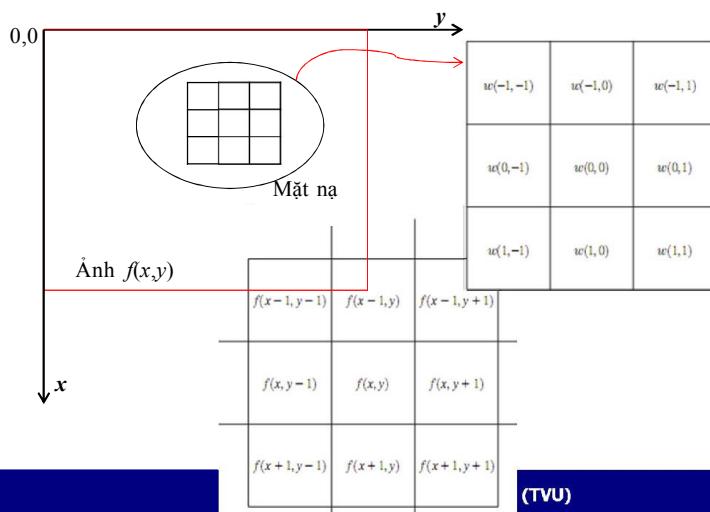
Ví dụ minh họa bộ lọc có kích thước 3×3 , w_5 là tâm của bộ lọc

Cơ chế của lọc không gian

- Quá trình xử lý là quá trình di chuyển mặt nạ lọc theo từng điểm một trong ảnh.
- Tại mỗi điểm (x, y) , đáp ứng của bộ lọc tại điểm đó được tính bằng cách sử dụng các mối quan hệ đã xác định trước.
- Với bộ lọc không gian tuyến tính, đáp ứng chính là tổng của các tích giữa hệ số của bộ lọc với cấp xám của điểm ảnh tương ứng trong vùng áp dụng lọc.

Cơ chế của lọc không gian (Cont.)

$$R = w(-1, -1)f(x - 1, y - 1) + w(-1, 0)f(x - 1, y) + \dots \\ + w(0, 0)f(x, y) + \dots + w(1, 0)f(x + 1, y) + w(1, 1)f(x + 1, y + 1)$$



Cơ chế của lọc không gian (Cont.)

- Thông thường hệ số $w(0, 0)$ trùng khớp với giá trị $f(x, y)$ của ảnh, nghĩa là mặt nạ có tâm tại điểm (x, y) trong quá trình tính toán.

$$R = w(-1, -1)f(x - 1, y - 1) + w(-1, 0)f(x - 1, y) + \dots \\ + w(0, 0)f(x, y) + \dots + w(1, 0)f(x + 1, y) + w(1, 1)f(x + 1, y + 1)$$

Lọc không gian tuyến tính (Cont.)

- Để có được ảnh lọc, chúng ta phải áp dụng công thức ở trên lên toàn bộ các điểm ảnh, tức là $x = 0, 1, \dots, M$ và $y = 0, 1, \dots, N$.
- Phương trình trên còn được gọi là tích chập của mặt nạ với ảnh.
- Mặt nạ bộ lọc nhiều lúc cũng còn được gọi là mặt nạ chập hay là nhân chập.

Lọc không gian tuyến tính

- Đối với mặt nạ có kích thước $m \times n$, với $m = 2 \times a + 1$ và $n = 2 \times b + 1$, với a, b là những số nguyên không âm.
- Với điều kiện trên, chúng ta sẽ có những mặt nạ với kích thước là những số lẻ.
- Khi đó, bộ lọc tuyến tính của ảnh f có kích thước $M \times N$ với mặt nạ lọc $m \times n$ áp dụng tại điểm (x, y) được cho bởi công thức sau:

$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t)f(x + s, y + t)$$

- Trong đó: $a = (m - 1)/2$ and $b = (n - 1)/2$

Lọc không gian tuyến tính (Cont.)

- Chúng ta cũng có thể ký hiệu đáp ứng R của ảnh tại điểm (x, y) với mặt nạ chập có kích thước $m \times n$ theo công thức:
$$R = w_1 z_1 + w_2 z_2 + \dots + w_{mn} z_{mn} = \sum_{i=1}^{mn} w_i z_i$$
- Trong đó, các giá trị w là các hệ số mặt nạ, các giá trị z là các giá trị cấp xám của ảnh tương ứng với các hệ số, mn là số lượng các hệ số trong mặt nạ.

Ví dụ lọc không gian tuyển tính

- Mặt nạ 3×3 trong hình dưới đáp ứng tại điểm (x, y) của ảnh được tính bởi công thức

$$R = w_1 z_1 + w_2 z_2 + \dots w_9 z_9 \\ = \sum_{i=1}^9 w_i z_i.$$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Ví dụ: lọc mảng một chiều

Ví dụ: lọc mảng một chiều

	Correlation	Convolution																																																																																												
(a)	<p style="text-align: center;">Origin f w</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	1	0	0	0	0	1	2	3	2	0	0	0	0	<p style="text-align: center;">Origin f w rotated 180°</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>2</td><td>3</td><td>2</td><td>1</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	1	0	0	0	0	0	2	3	2	1	0	0	0																																																												
0	0	0	1	0	0	0	0																																																																																							
1	2	3	2	0	0	0	0																																																																																							
0	0	0	1	0	0	0	0																																																																																							
0	2	3	2	1	0	0	0																																																																																							
(b)	<p style="text-align: center;">Starting position alignment</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	1	0	0	0	0	1	2	3	2	0	0	0	0	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>2</td><td>3</td><td>2</td><td>1</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	1	0	0	0	0	0	2	3	2	1	0	0	0																																																												
0	0	0	1	0	0	0	0																																																																																							
1	2	3	2	0	0	0	0																																																																																							
0	0	0	1	0	0	0	0																																																																																							
0	2	3	2	1	0	0	0																																																																																							
(c)	<p style="text-align: center;">Zero padding</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>2</td><td>3</td><td>2</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																									
1	2	3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
0	2	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
(d)	<p style="text-align: center;">Position after one shift</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>2</td><td>3</td><td>2</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
1	2	3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
0	2	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
(e)	<p style="text-align: center;">Position after four shifts</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>2</td><td>3</td><td>2</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
1	2	3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
0	2	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
(f)	<p style="text-align: center;">Final position</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>2</td><td>3</td><td>2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>2</td><td>3</td><td>2</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
1	2	3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
0	2	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																								
(g)	'full' correlation result 0 0 0 0 2 3 2 1 0 0 0 0	'full' convolution result 0 0 0 1 2 3 2 0 0 0 0 0																																																																																												
(h)	'same' correlation result 0 0 2 3 2 1 0 0	'same' convolution result 0 1 2 3 2 0 0 0																																																																																												

- **Correlation** là tiến trình trượt mặt nạ theo ánh f .

- **Convolution** thì cũng giống như *correlation*, nhưng mặt nạ w xoay 180^0 trước khi trượt theo ảnh f .

Xử lý trường hợp đặc biệt

- Trường hợp thực thi các phép toán lân cận khi tâm của bộ lọc nằm trên biên của ảnh?

$f(0,0)$	$f(0,1)$	$f(0,2)$	$f(0,3)$	$f(0,4)$
$f(1,0)$	$f(1,1)$	$f(1,2)$	$f(1,3)$	$f(1,4)$
$f(2,0)$	$f(2,1)$	$f(2,2)$	$f(2,3)$	$f(2,4)$
$f(3,0)$	$f(3,1)$	$f(3,2)$	$f(3,3)$	$f(3,4)$
$f(4,0)$	$f(4,1)$	$f(4,2)$	$f(4,3)$	$f(4,4)$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Măt na

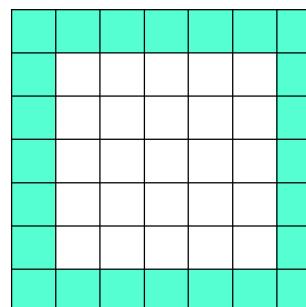
Xử lý trường hợp đặc biệt (Cont.)

- Khi tâm mặt nạ di chuyển gần đến biên của ảnh thì một hoặc một số dòng/cột của mặt nạ sẽ nằm ngoài ảnh.
- Ở ví dụ dưới các hệ số w_1, w_4, w_7 nằm ngoài ảnh.

w_1	w_2	w_3			
w_4	w_5	w_6			
w_7	w_8	w_9			

Xử lý trường hợp đặc biệt (Cont.)

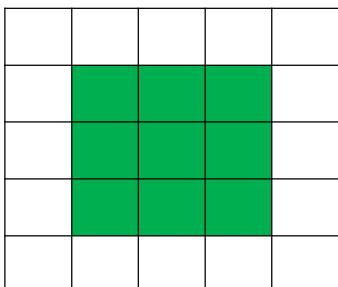
- Cách 2:** Yêu cầu ảnh kết quả có kích thước bằng ảnh gốc
 - Đưa thêm các dòng đệm và cột đệm mang giá trị 0 vào quanh biên của ảnh.



w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Xử lý trường hợp đặc biệt (Cont.)

- Cách 1:** Giả sử mặt nạ có kích thước $n \times n$.
 - Cho vị trí tâm của mặt nạ không được nhỏ hơn $(n-1)/2$ điểm ảnh kể từ biên \rightarrow Ảnh sau khi lọc có kích thước nhỏ hơn ảnh gốc, nhưng tất cả các điểm ảnh đều được xử lý.



w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Lọc không gian làm trơn (smoothing)

- Các bộ lọc làm trơn được sử dụng để khử đi vết mờ và khử nhiễu.
- Khử mờ thường được sử dụng trong các bước tiền xử lý ảnh.
 - Chẳng hạn xóa bỏ đi những chi tiết nhỏ khỏi ảnh trước khi trích chọn các đối tượng.
 - Lập lại các lỗ hổng nhỏ của các đường thẳng hay các cung
- Khử nhiễu dùng để khử đi những vết mờ của ảnh

Bộ lọc tuyển tính làm trơn

- Đầu ra (đáp ứng) của bộ lọc tuyển tính làm trơn đơn giản là trung bình các điểm ảnh chứa trong lân cận của mặt nạ lọc.
- Những bộ lọc này được gọi là lọc trung bình hay còn được gọi là **bộ lọc thông thấp**.

Bộ lọc tuyển tính làm trơn (Cont.)

- Bởi vì nhiều ngẫu nhiên có đặc trưng là chứa những điểm nhô trong các cấp xám → ứng dụng rõ ràng nhất của làm trơn là khử nhiễu.
- Tuy nhiên, các cạnh của đối tượng cũng được đặc trưng bởi các điểm nhô trong các cấp xám, vì vậy lọc trung bình sẽ có tác động phụ lên các cạnh của các đối tượng trong ảnh → nó sẽ làm nhòa (mờ) đi các cạnh.
- Ứng dụng chính của các lọc trung bình là khử đi các chi tiết không thích hợp trong ảnh.

Bộ lọc tuyển tính làm trơn (Cont.)

- Ý tưởng cơ bản của bộ lọc làm trơn là **thay thế giá trị của mỗi điểm ảnh trong ảnh bằng trung bình của các cấp xám trong lân cận được xác định bởi mặt nạ lọc**.
- Kết quả của quá trình lọc sẽ làm giảm đi những “**điểm nhô/điểm gờ**” trong ảnh.

Lọc trung bình lân cận

- Với lọc trung bình, mỗi điểm ảnh được thay thế bằng trung bình trọng số của các điểm lân cận.
- Lọc trung bình có trọng số chính là thực hiện chập ảnh đầu vào với nhân chập H. Nhân chập H trong trường hợp này có dạng:

$$H = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Lọc trung bình lân cận (Cont.)

- Trong lọc trung bình, thường người ta ưu tiên cho các hướng để bảo vệ biên của ảnh khỏi bị mờ khi làm trơn ảnh.
- Các kiểu mặt nạ được sử dụng tùy theo các trường hợp khác nhau.
- Bộ lọc tuyến tính theo nghĩa là điểm ảnh ở tâm cửa sổ sẽ được thay bởi tổ hợp các điểm lân cận chập với mặt nạ.

Lọc trung bình không gian (Cont.)

$f(x-1,y-1)$	$f(x-1,y)$	$f(x-1,y+1)$
$f(x,y-1)$	$f(x,y)$	$f(x,y+1)$
$f(x+1,y-1)$	$f(x+1,y)$	$f(x+1,y+1)$

1	1	1
1	1	1
1	1	1

Bộ lọc làm trơn

$g(x,y)$

$$\begin{aligned} &= \frac{1}{9}[f(x-1,y-1) + f(x-1,y) + f(x-1,y+1) \\ &+ f(x,y-1) + f(x,y) + f(x,y+1) + f(x+1,y-1) \\ &+ f(x+1,y+1)] \end{aligned}$$

Các mặt nạ lọc trung bình

- Các mặt nạ lọc trung bình trên vùng lân cận hay còn gọi lọc thông thấp

1	1	1
1	1	1
1	1	1

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1
1	1	1	1	1	1

$$\frac{1}{25} \times \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1

$$\frac{1}{49} \times$$

Lọc phi tuyến - lọc trung vị

- Lọc trung vị được sử dụng phổ biến bởi vì nó khử nhiễu rất tốt, nhất là đối với những loại nhiễu ngẫu nhiên
- Kết quả ảnh đạt được ít vết nhòe hơn so với lọc trơn tuyến tính cùng kích thước
- Lọc trung vị có hiệu quả đặc biệt khi có nhiều xung, bởi vì nhiều xung xuất hiện như những chấm trắng và đen ở trên ảnh

Lọc phi tuyển - lọc trung vị

- Trung vị θ của một dãy $\{x_n\}$ các giá trị là đại lượng mà ở đó có một nửa các giá trị trong $\{x_n\}$ nhỏ hơn hoặc bằng θ và một nửa các giá trị trong $\{x_n\}$ lớn hơn hoặc bằng θ

Cách thực hiện:

- Sắp xếp dãy $\{x_n\}$ theo thứ tự tăng dần các giá trị.
- Trung vị của $\{x_n\}$ chính là giá trị giữa dãy $\theta = \{x_{n/2}\}$

Lọc trung vị (Cont.)

- Kỹ thuật lọc trung vị dùng để lọc nhiễu bằng cách trượt trên mặt phẳng ảnh, mỗi lần di chuyển qua trên một điểm ảnh.

- Những phần tử trong cửa sổ được xem như là một chuỗi $\{x_n\}$ và điểm quan tâm được thay thế bởi giá trị trung vị θ của chuỗi.

- $\{x_n\} = \{1, 2, 9, 4, 5\}$

Ví dụ:
■ Sắp xếp $\{1, 2, 4, 5, 9\} \rightarrow \theta=4$

■ \rightarrow chuỗi sau khi lọc $\{1, 2, 4, 4, 5\}$

Lọc trung vị (Cont.)

- Kỹ thuật lọc trung vị thường dùng mặt nạ có kích thước là $3\times 3, 5\times 5$.
- Việc lọc sẽ dừng lại khi quá trình lọc không làm thay đổi kết quả của ảnh cần lọc.
- **Tóm lại:** Lọc trung vị với mặt nạ có kích thước $n\times n$ được tính như chuỗi một chiều. Ta tiến hành sắp xếp dãy đó rồi thay thế phần tử tâm bằng trung vị của dãy vừa tìm được.

Lọc trung vị (Cont.)

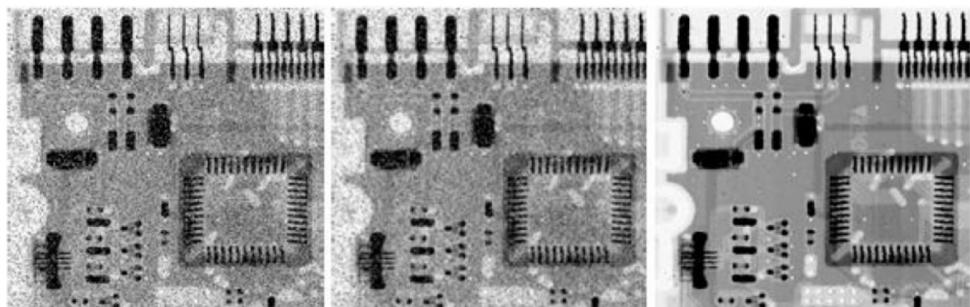
- Ví dụ: Lọc trung vị trên ảnh với cửa sổ lọc là 3×3

15	17	18	
16	78	17	
17	15	20	

15	17	18	
16	17	17	
17	15	20	

15	15	16	17	17	17	18	20	78
1	2	3	4	5	6	7	8	9

Lọc trung vị (Cont.)



a | b | c

- (a) Ảnh gốc X-quang chụp một bo mạch của một thiết bị (có nhiễu).
- (b) Ảnh sau khi lọc trung bình tuyến tính làm trơn với mặt nạ 3×3 .
- (c) Ảnh sau khi lọc trung vị với mặt nạ 3×3 .

Câu hỏi ôn tập

Chương 3

PHÁT HIỆN BIÊN EDGE DETECTORS

NỘI DUNG

- Khái niệm biên
- Các kỹ thuật phát hiện biên
 - Phương pháp tìm biên Roberts
 - Phương pháp tìm biên Sobel
 - Phương pháp tìm biên Prewitt
 - Phương pháp la bàn

Vùng và Biên của vùng

■ Vùng

- Cho R là một tập con của ảnh. R được gọi là vùng nếu R là một tập liên thông.

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

Vùng và Biên của vùng (Cont.)

■ Biên của vùng

- Biên của vùng R là tập hợp các điểm trong vùng R mà có một hoặc nhiều lân cận không thuộc R.
- Nếu R phủ toàn ảnh thì biên của nó là dòng đầu tiên, cột đầu tiên, dòng cuối cùng, cột cuối cùng của ảnh.

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

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1

Khái niệm biên

■ Biên

- Một điểm ảnh được gọi là biên nếu ở đó có sự thay đổi đột ngột về cấp xám.
- Tập hợp các điểm biên tạo thành một đường biên (đường bao) của ảnh.

Các kỹ thuật phát hiện biên (2/2)

■ Phát hiện biên gián tiếp:

- Phân chia ảnh thành các vùng → đường phân cách giữa các vùng đó chính là biên
- Kỹ thuật dò biên và phân vùng ảnh là hai bài toán đối ngẫu nhau, vì dò biên để thực hiện phân lớp đối tượng mà khi đã phân lớp xong nghĩa là đã phân vùng được ảnh và ngược lại

Các kỹ thuật phát hiện biên (1/2)

- ## ■ Phát hiện biên trực tiếp:
- phương pháp này chủ yếu dựa vào sự biến thiên độ sáng của điểm ảnh để làm nổi biên bằng kỹ thuật đạo hàm
- Nếu lấy đạo hàm bậc nhất của ảnh: ta có phương pháp Gradient
 - Nếu lấy đạo hàm bậc hai của ảnh: ta có phương pháp Laplace
 - Phương này gọi là phương pháp dò biên cục bộ

Phương pháp phát hiện biên cục bộ

- ## ■ Phương pháp Gradient:
- Gradient là một vec tơ $f(x, y)$ có các thành phần biểu thị tốc độ thay đổi mức xám của điểm ảnh, tức là:

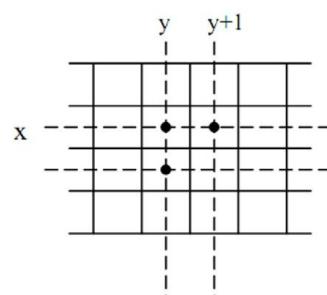
$$\frac{\partial f(x, y)}{\partial x} = f'_x \approx \frac{f(x + dx, y) - f(x, y)}{dx}$$
$$\frac{\partial f(x, y)}{\partial y} = f'_y \approx \frac{f(x, y + dy) - f(x, y)}{dy}$$

Trong đó dx, dy là khoảng cách (tính bằng số điểm) theo hướng x và y . Và $dx=dy=1$

Phương pháp Roberts

- Toán tử Roberts do Roberts đề xuất vào năm 1965
- Với mỗi điểm ảnh $I(x,y)$ đạo hàm theo x, y được ký hiệu tương ứng g_x, g_y

$$\begin{cases} g_x = I(x+1, y) - I(x, y) \\ g_y = I(x, y+1) - I(x, y) \end{cases}$$



Phương pháp Roberts (Cont.)

- PP Roberts có tốc độ tính toán nhanh
- Sử dụng 4 điểm ảnh để tính giá trị cấp xám đầu ra
- Chỉ có phép toán cộng và trừ được thực hiện trong ảnh

Phương pháp Roberts (Cont.)

- Công thức g_x, g_y được cụ thể hóa bằng các mặt nạ theo chiều x và y tương ứng như sau:

$$H_x = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

Hướng ngang (x)

$$H_y = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

Hướng dọc (y)

Mặt nạ Roberts

$$\begin{aligned} G_x &= a_2 - a_3 \\ G_y &= a_4 - a_1 \end{aligned}$$

Độ lớn của Gradient:

$$|G| = \sqrt{G_x^2 + G_y^2} = |G_x| + |G_y|$$

a_1	a_2
a_3	a_4

Điểm ảnh

Phương pháp Roberts (Cont.)



a) Ảnh gốc



b) Ảnh sau khi áp dụng toán tử Roberts



c) Ảnh sau khi phân ngược ảnh (b)

Phương pháp Roberts (Cont.)

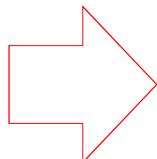
Ví dụ:

1	1	2	6	7	7
1	2	2	7	7	6
2	2	2	6	7	6
2	2	3	7	7	6
2	3	1	7	7	6
3	1	4	6	7	5

Ảnh gốc

$$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

Mặt nạ Roberts



1	1	9	1	1	13
1	0	9	1	2	12
0	1	8	1	2	12
1	1	10	0	2	12
1	1	8	1	3	11
4	5	10	13	12	5

Ảnh sau khi áp dụng toán tử Roberts

Phương pháp Prewitt (Cont.)

$$H_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad H_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Mặt nạ Prewitt

a_1	a_2	a_3
a_4	a_5	a_6
a_7	a_8	a_9

Điểm ảnh

$$G_x = a_3 + a_6 + a_9 - (a_1 + a_4 + a_7)$$

$$G_y = a_7 + a_8 + a_9 - (a_1 + a_2 + a_3)$$

Độ lớn của Gradient:

$$|G| = \sqrt{G_x^2 + G_y^2} = |G_x| + |G_y|$$

Phương pháp Prewitt

■ Phương pháp Prewitt được đề xuất năm 1970 sử dụng 2 mặt nạ nhân chập xấp xỉ đạo hàm theo 2 hướng x và y là:

$$H_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

Hướng ngang (x)

$$H_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

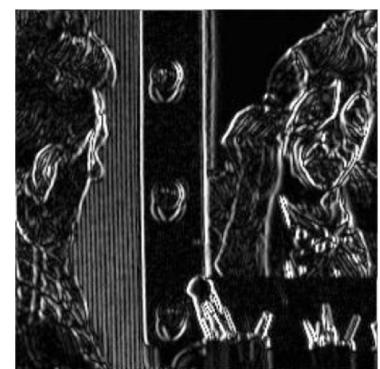
Hướng dọc (y)

■ Mặt nạ này có thể nhận từ mặt nạ kia bằng cách quay một góc 90°

Phương pháp Prewitt (Cont.)



Ảnh gốc

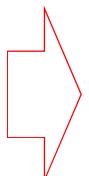


Ảnh sau khi áp dụng toán tử Prewitt

Phương pháp Prewitt (Cont.)

- Ví dụ: Cho ảnh I, sử dụng PP Prewitt để tìm biên với mặt nạ như sau:

1	1	2	6	7	7
1	2	2	7	7	6
2	2	2	6	7	6
2	2	3	7	7	6
2	3	1	7	7	6
3	1	4	7	7	5
-1	0	1	1	1	1
-1	0	1	0	0	0
-1	0	1	-1	-1	-1



6	7	21	26	20	27
7	4	15	15	1	22
7	4	15	15	2	21
8	0	14	15	3	21
6	2	15	13	2	22
9	6	20	24	22	27

Trường Đại học Trà Vinh (TVU)

17

Phương pháp Sobel (Cont.)

$$H_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \quad H_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}$$

Mặt nạ Sobel

Điểm ảnh

$$\begin{aligned} G_x &= a_3 + 2a_6 + a_9 - (a_1 + 2a_4 + a_7) \\ G_y &= a_7 + 2a_8 + a_9 - (a_1 + 2a_2 + a_3) \end{aligned}$$

Độ lớn của Gradient:

$$|G| = \sqrt{G_x^2 + G_y^2} = |G_x| + |G_y|$$

Trường Đại học Trà Vinh (TVU)

19

Phương pháp Sobel

- Tương tự như phương pháp Prewitt, phương pháp Sobel sử dụng 2 mặt nạ nhân chập theo 2 hướng x và y là:

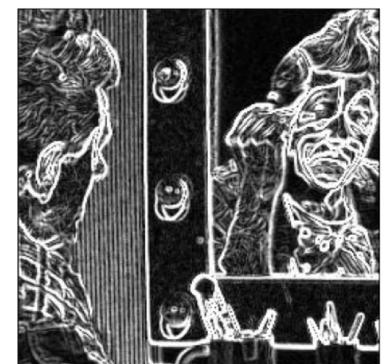
$$H_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \quad H_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}$$

- Mặt nạ này có thể nhận từ mặt nạ kia bằng cách quay một góc 90°

Phương pháp Sobel (Cont.)



Ảnh gốc



Ảnh sau khi áp dụng toán tử Sobel

Trường Đại học Trà Vinh (TVU)

18

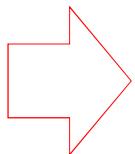
Trường Đại học Trà Vinh (TVU)

20

Phương pháp Sobel (Cont.)

- Ví dụ: Cho ảnh I, sử dụng PP Sobel để tìm biên với mặt nạ như sau:

1	1	2	6	7	7
1	2	2	7	7	6
2	2	2	6	7	6
2	2	3	7	7	6
2	3	1	7	7	6
3	1	4	7	7	5
-1	0	1	1	2	1
-2	0	2	0	0	0
-1	0	1	-1	-2	-1



4	6	16	24	16	40
8	4	30	34	14	30
10	6	30	32	14	28
14	2	32	32	14	28
10	8	30	34	16	30
12	10	26	34	30	40

Bài tập ôn tập

- Xây dựng thuật toán phát hiện biên ảnh dùng toán tử Roberts
- Xây dựng thuật toán phát hiện biên ảnh dùng toán tử Sobel
- Xây dựng thuật toán phát hiện biên ảnh dùng toán tử Prewitt

Câu hỏi ôn tập

Cho ảnh số I:

$$I = \begin{bmatrix} 1 & 2 & 1 & 1 & 2 \\ 1 & 5 & 5 & 7 & 3 \\ 11 & 5 & 5 & 7 & 3 \\ 1 & 5 & 7 & 7 & 6 \\ 1 & 8 & 7 & 6 & 8 \end{bmatrix}$$

- Hãy tính $G=|G_x|+|G_y|$ với $G_x=H_x \otimes I$, $G_y=H_y \otimes I$, H_x và H_y là mặt nạ nhân chập Prewitt
- Tương tự như câu 1, nhưng H_x và H_y là mặt nạ nhân chập Sobel

Chương 4

PHÂN VÙNG ẢNH IMAGE SEGMENTATION

NỘI DUNG

- Mục tiêu
- Khái niệm
- Các kỹ thuật phân vùng ảnh

Mục tiêu

- Phân vùng ảnh là phân chia ảnh thành những vùng ảnh
- Trong chương 3, ta đã tiến hành xác định biên cho các đối tượng bằng các phương pháp khác nhau.
- Từ biên của các đối tượng, ta có thể tìm ra các vùng của ảnh

Các khái niệm cơ bản

- Giả sử R biểu diễn vùng của toàn ảnh, chúng ta có thể chia R ra thành nhiều vùng con khác nhau R_1, R_2, \dots, R_n thỏa điều kiện:
 - (a) $\bigcup_{i=1}^n R_i = R$
 - (b) R_i là một vùng liên thông với mọi $i = 1, \dots, n$
 - (c) $R_i \cap R_j = \emptyset, \forall i \neq j$
 - (d) $P(R_i) = \text{TRUE}, \forall i = 1, 2, \dots, n$
 - (e) $P(R_i \cup R_j) = \text{FALSE}, \forall i \neq j$

Các khái niệm cơ bản (Cont.)

■ Trong đó:

- $P(R_i)$ là một hàm được định nghĩa trước trên các điểm ảnh trong tập R_i và \emptyset là tập rỗng.
- Điều kiện (a) để đảm bảo việc phân vùng là hoàn toàn, có nghĩa mỗi điểm ảnh phải thuộc vào một vùng nào đó
- Điều kiện (b) R là là một vùng liên thông

Phân vùng ảnh

- Phát triển vùng là một thủ tục nhắm nhóm các điểm ảnh hoặc các vùng lại với nhau thành một vùng lớn hơn dựa trên một tiêu chuẩn cho trước nào đó.
- Hướng tiếp cận cơ bản là dựa trên tập những điểm “hạt giống”.
- Bắt đầu với tập các điểm “hạt giống”
- Phát triển vùng bằng cách thêm vào tập các điểm “hạt giống” những điểm lân cận thỏa mãn một tính chất cho trước (như cấp xám, màu sắc, kết cấu).

Các khái niệm cơ bản (Cont.)

- Điều kiện (c) để đảm bảo các vùng phải rời nhau
- Điều kiện (d) để đảm bảo các điểm ảnh trong vùng phải thỏa một tính chất p nào đó.
- Điều kiện (e) để đảm bảo hai vùng khác nhau về tính chất P được định nghĩa trước.

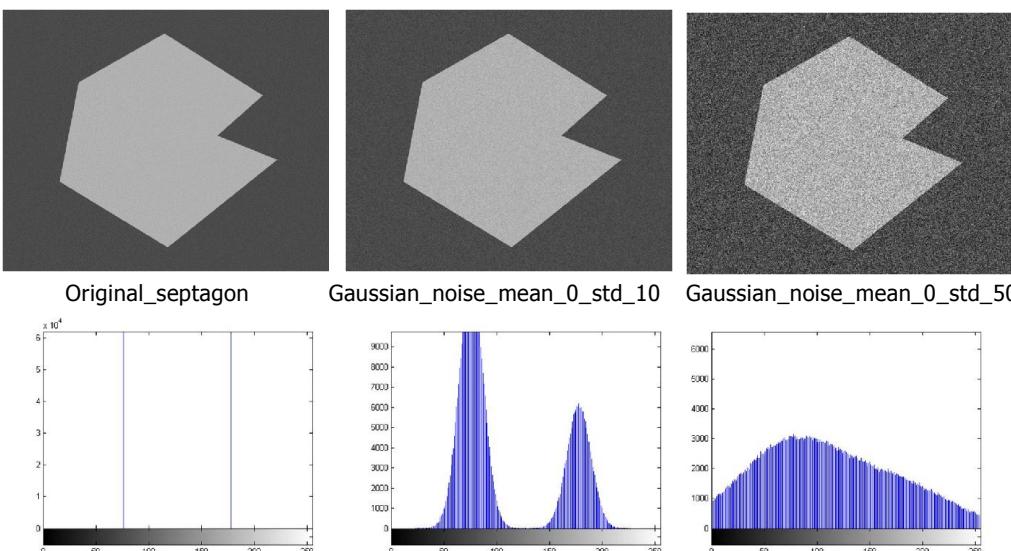
Các kỹ thuật phân vùng ảnh

- **Ngưỡng (thresholding)**
 - Đơn giản và nhanh
- Phân vùng dựa trên cách tiếp cận biên ảnh
 - Làm thế nào để tìm biên đúng nhất
- Các phương pháp phân vùng dựa trên ảnh
 - Phát triển vùng
 - Tách và kết hợp
 - Nhóm các điểm ảnh

Ngưỡng (Thresholding)

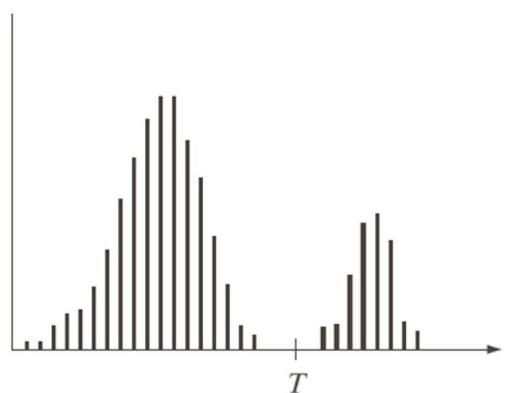
- Phân đoạn ảnh theo tiêu chuẩn dựa vào độ đồng nhất – tương tự về giá trị mức xám, màu, ...
- Phân đoạn theo giá trị mức xám – chặng hạn dựa vào phân đoạn histogram
- Làm thế nào xác định ngưỡng?

Ngưỡng (Thresholding) (Cont.)



Ngưỡng (Thresholding) (Cont.)

- Biểu đồ tần số ảnh $f(x,y)$, gồm những đỗi tượng ảnh sáng trên nền tối.
- Rõ ràng, điểm ảnh đối tượng và nền phân thành 2 nhóm
- Như vậy, trích xuất đỗi tượng ảnh khỏi nền là chọn ngưỡng T phân biệt giữa 2 nhóm này

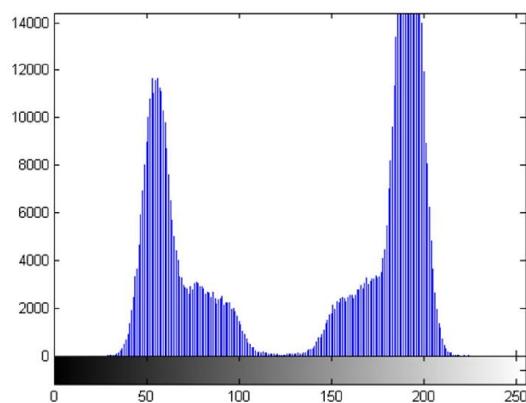


$$g(x, y) = \begin{cases} 1 & \text{if } f(x, y) > T \\ 0 & \text{if } f(x, y) \leq T \end{cases}$$

Thuật toán xác định ngưỡng T

- Ước lượng giá trị ngưỡng ban đầu T
- Phân vùng ảnh theo ngưỡng $T \rightarrow$ Tạo ra 2 nhóm pixels:
 - G_1 gồm những pixels với giá trị mức xám $> T$
 - G_2 gồm những pixels với giá trị mức xám $\leq T$
- Tính giá trị trung bình μ_1 và μ_2 cho những pixels trong vùng G_1 và G_2 .
- Tính giá trị ngưỡng T mới : $T = \frac{1}{2}(\mu_1 + \mu_2)$.

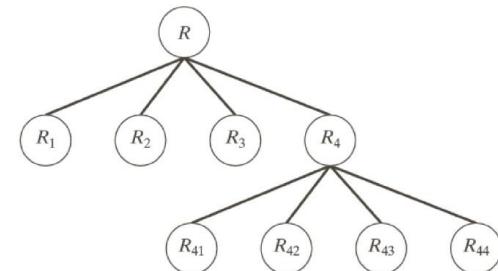
Ví dụ



Tách vùng và hợp vùng (Cont.)

- Chia R_i thành 4 vùng không giao nhau nếu $P(R_i)=\text{FALSE}$.
- Hợp các vùng R_j và R_k nếu $P(R_j \cup R_k)=\text{TRUE}$
- Dừng khi không thể tách hoặc hợp các vùng được nữa.

R_1	R_2
R_3	R_{41} R_{42}
R_{43}	R_{44}



Tách vùng và hợp vùng

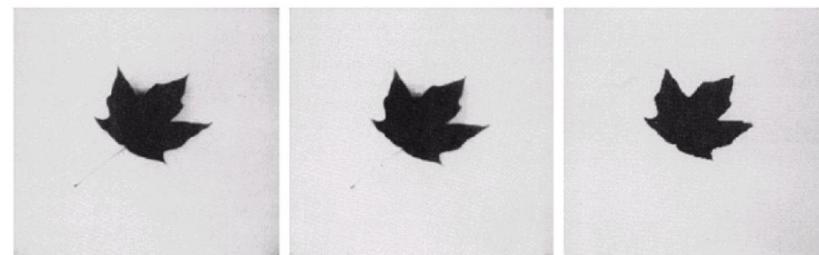
- Một kỹ thuật nữa để phân vùng đó là tách vùng và hợp vùng
- Giả sử R biểu diễn một vùng cho toàn ảnh và P là một tiêu chuẩn dùng để tách vùng và hợp vùng.
- Có nhiều kỹ thuật tách và hợp vùng. Trong phần này ta tìm hiểu về kỹ thuật tách vùng và hợp vùng theo cây tứ phân

Tách vùng và hợp vùng (Cont.)

a b c

FIGURE 10.43

(a) Original image. (b) Result of split and merge procedure. (c) Result of thresholding (a).



- $P(R_i)=\text{TRUE}$ nếu có ít nhất 80% các điểm trong R_i có tính chất $|z_j - m| > 2\partial_i$
- Trong đó:
 - z_j là cấp xám của điểm ảnh thứ j trong vùng R_i
 - m là giá trị trung bình của vùng R_i
 - ∂_i là độ lệch chuẩn của các cấp xám trong R_i

Tính độ lệch chuẩn

■ Phương sai:

$$\sigma^2 = \frac{\sum_{j=1}^n (z_j - \bar{z})^2}{n-1}$$

■ Độ lệch chuẩn:

$$\sigma = \sqrt{\frac{\sum_{j=1}^n (z_j - \bar{z})^2}{n-1}}$$

Chương 5

NÉN ẢNH

IMAGE COMPRESSION

NỘI DUNG

- Giới thiệu
- Một số khái niệm
- Sự dư thừa dữ liệu
- Các phương pháp nén ảnh
 - Mã hóa RLC
 - Mã hóa Huffman
 - Mã hóa LZW

Giới thiệu

■ Lý do cần nén dữ liệu

- Dữ liệu được lưu trữ, trao đổi trên các phương tiện xử lý và truyền tải thông tin
- Quá trình xử lý dữ liệu đặt ra cho chúng ta những yêu cầu sau:
 - Yêu cầu về lưu trữ là phải chính xác và tiết kiệm
 - Yêu cầu về truyền tải phải nhanh chóng, hiệu quả.
- Nén dữ liệu là một yêu cầu tất yếu được đặt trong thời đại bùng nổ thông tin

Một số khái niệm

■ Nén dữ liệu (Data compression)

- Là quá trình làm giảm lượng thông tin “dư thừa” trong dữ liệu gốc
- Do vậy, lượng thông tin thu được sau khi nén nhỏ hơn rất nhiều so với dữ liệu gốc.
- Còn có tên gọi khác: “giảm độ dư thừa”, “mã hóa ảnh gốc”

Một số khái niệm (Cont.)

■ Tỷ lệ nén (Compression rate)

- Tỷ lệ nén là một trong những đặc trưng quan trọng nhất của mọi phương pháp nén.
- Giả sử n_1 là kích thước của ảnh sau khi nén, n_2 là kích thước của ảnh gốc. Khi đó, tỷ lệ nén được tính theo công thức $r=n_2/n_1$
- Độ dư thừa dữ liệu được tính theo công thức:

$$d = 1 - \frac{1}{r}$$

Một số khái niệm (Cont.)

■ Tỷ lệ nén

- Tỷ lệ nén >< tốc độ nén
- Tỷ lệ nén >< chất lượng nén

Một số khái niệm (Cont.)

■ Ví dụ:

- Ảnh gốc có kích thước 1000byte, ảnh sau khi nén có kích thước 100 byte. Khi đó tỷ lệ nén

$$r = 1000/100 = 10$$

- Độ dư thừa dữ liệu là:

$$d = 1 - 1/10 = 0.9 = 90\%$$

Dư thừa dữ liệu

- Nén nhằm mục đích giảm kích thước dữ liệu, bằng cách loại bỏ dư thừa dữ liệu.
- Việc xác định bản chất các kiểu dư thừa dữ liệu rất có ích cho việc xây dựng các phương pháp nén dữ liệu khác nhau

■ Dư thừa dữ liệu

- Dư thừa mã
- Dư thừa nội tại điểm ảnh
- Dư thừa do mắt nhìn

Dư thừa mã

- Giả sử các cấp xám là các đại lượng ngẫu nhiên, n_k là số lượng điểm ảnh có giá trị cấp xám là k , n là tổng số điểm ảnh trong ảnh. Khi đó xác suất để xảy ra cấp xám thứ k là:

$$p(k) = \frac{n_k}{n}, k = 0, 1, 2, \dots, L - 1$$

- Trong đó: L là mức cấp xám của điểm ảnh

Dư thừa mã (Cont.)

- Giả sử mỗi điểm ảnh được biểu diễn bởi m bit cố định. Khi đó $L_{avg} = m$
- Để nén ảnh, ta sẽ tìm cách giảm số bit trung bình biểu diễn một điểm ảnh
- L_{avg} càng nhỏ $\rightarrow M \times N \times L_{avg}$ càng nhỏ \rightarrow tỷ lệ nén càng cao.

Dư thừa mã (Cont.)

- Gọi $l(k)$ là số bit biểu diễn điểm ảnh có cấp xám bằng k . Khi đó, số bit trung bình để biểu diễn một điểm ảnh là:

$$L_{avg} = \sum_{k=0}^{L-1} l(k)p(k)$$

- Vậy, tổng số bit cần để biểu diễn ảnh có kích thước $M \times N$ là:

$$M \times N \times L_{avg}$$

Ví dụ về dư thừa mã

- Giả sử có một ảnh 8 cấp xám

k	p(k)	Code 1	l₁(k)	Code 2	l₂(k)
0	0.19	000	3	11	2
1	0.25	001	3	01	2
2	0.21	010	3	10	2
3	0.16	011	3	001	3
4	0.08	100	3	0001	4
5	0.06	101	3	00001	5
6	0.03	110	3	000001	6
7	0.02	111	3	000000	6

Ví dụ về dư thừa mã

- Số bit trung bình để biểu diễn cho một điểm ảnh theo cách mã hóa 1 (code 1) là 3
- Số bit trung bình để biểu diễn cho một điểm ảnh theo cách mã hóa 1 (code 1) là:

$$L_{avg} = \sum_{k=0}^7 L_2(k)p(k)$$
$$= 2(0.19) + 2(0.25) + 2(0.21) + 3(0.16) + 4(0.08) + 5(0.06) + 6(0.03) + 6(0.02) = 2.7$$

k	p(k)	Code 1	L ₁ (k)	Code 2	L ₂ (k)
0	0.19	000	3	11	2
1	0.25	001	3	01	2
2	0.21	010	3	10	2
3	0.16	011	3	001	3
4	0.08	100	3	0001	4
5	0.06	101	3	00001	5
6	0.03	110	3	000001	6
7	0.02	111	3	000000	6

Dư thừa do mắt nhìn

- Mắt người không thể đáp ứng được với tất cả độ nhạy sáng
- Thông tin đơn giản có tầm quan trọng ít hơn những thông tin khác trong vùng nhìn thấy
- Những thông tin không chủ yếu có thể loại bỏ được mà không ảnh hưởng đến hình ảnh nhận thức của con người.

Dư thừa nội tại điểm ảnh

- Một điểm ảnh nào đó có thể được dự báo từ các điểm ảnh xung quanh.
- Để giảm độ dư thừa trong điểm ảnh, người ta biến đổi dạng 2 chiều dùng cho việc biểu diễn ảnh thành một dạng khác có hiệu quả hơn
- Chẳng hạn, có thể sử dụng sự khác nhau giữa 2 điểm ảnh kề nhau để biểu diễn

Các phương pháp nén/mã hóa ảnh

- Mã hóa loạt dài RLC (Run Length Coding)
- Mã hóa Huffman
- Mã hóa LZW (Lempel Ziv-Wenck)

Mã hóa loạt dài RLC (Run Length Coding)

- Được áp dụng cho ảnh trắng-đen (có 2 mức xám, màu đen (0), màu trắng (1), chẵng hạn như văn bản trên nền trắng, trang in, các bản vẽ kỹ thuật)
 - Ý tưởng của phương pháp này là phát hiện các bit lặp, chẵng hạn như một loạt bit 0 nằm giữa hai bit 1, hay ngược lại, một loạt bit 1 nằm giữa hai bit 0 (dãy này gọi là mạch)
 - Thay thế chuỗi đó bởi một chuỗi mới gồm hai thông số: chiều dài của chuỗi và bit lặp.

Mã hóa loạt dài (Cont.)

- Cần lưu ý rằng đối với ảnh, chiều dài của chuỗi có thể lớn hơn 255. nếu dùng 1 byte để mã hóa thì không đủ. Giải pháp được dùng là:
 - Tách chuỗi lặp ra thành nhiều chuỗi
 - Các chuỗi đầu có độ dài là 255
 - Chuỗi cuối cùng có độ dài là số bit còn lại

Mã hóa loạt dài (Cont.)

- #### ■ Ví dụ:

21 bit 0 12 bit 1

00010101 00001010

- Thay thế chuỗi bit 0 bằng 00010101
 - Thay thế chuỗi bit 1 bằng 00001010
 - Kết quả ta có: 00010101000001010

Mã hóa loạt dài (Cont.)

- Đối với ảnh không phải là ảnh nhị phân, chúng ta cũng thực hiện tương tự
 - Thay loạt các giá trị cấp xám giống nhau bằng hai thông tin:
 - Số điểm ảnh có giá trị bằng nhau (Count)
 - Giá trị cấp xám (Value)

Mã hóa loạt dài (Cont.)

■ Ví dụ:

1	1	1	3	3	8	8	8
8	9	9	7	7	7	7	7
6	6	6	5	5	5	3	3
1	1	1	1	1	2	2	2

(3,1) (2,3) (4,8)
(2,9) (5,7)
(3,6) (3,5) (2,3)
(5,1) (3,2)

- Mỗi điểm ảnh sử dụng 4 bit để biểu diễn → số bit là: $32 \times 4 = 128$
- Count ≤ 5 → Count sử dụng 3 bit để biểu diễn
- Value ≤ 9 → Value sử dụng 4 bit để biểu diễn
- Số lượng sau mã hóa là 10 → số bit là: $10 \times 7 = 70$ bit

Số bit cho một
loại là: $3+4=7$

Mã hóa Huffman

■ Ý tưởng của phương pháp mã hóa Huffman

- Là phương pháp dựa vào mô hình thống kê để tính tần suất xuất hiện của các tín hiệu
- Gán cho những tín hiệu có tần suất xuất hiện cao một từ mã có độ dài bé
- Gán cho những tín hiệu có tần suất xuất hiện thấp một từ mã có độ dài lớn

→ **giảm độ dài trung bình của từ mã**

Mã hóa Huffman

- Huffman là một trong những phương pháp mã hóa thông tin được sử dụng nhiều trong việc giảm kích thước dữ liệu và truyền tin
- Được đề nghị và năm 1952
- Phương pháp này có thể áp dụng trong việc nén dữ liệu tổng quát

Thuật toán mã hóa Huffman

Thuật toán gồm 2 bước:

■ **Bước 1:**

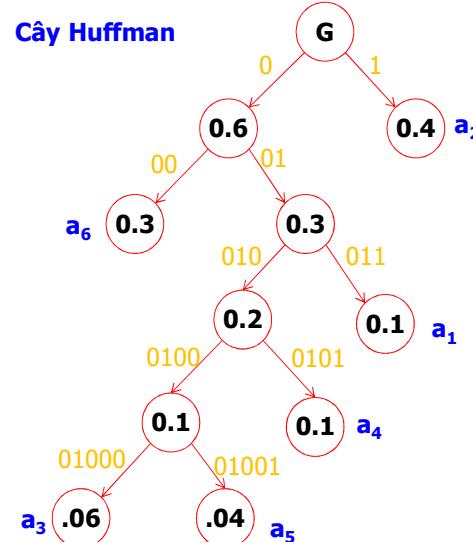
- Duyệt và sắp xếp các tín hiệu theo thứ tự giảm dần của xác suất xuất hiện.
- Duyệt bảng xác suất từ cuối lên để thực hiện ghép hai phần tử có xác suất thấp thành một phần tử mới
- Phần tử này có xác suất bằng tổng xác suất hai phần tử thành phần và loại bỏ hai thành phần này.
- Lặp lại quá trình cho đến khi chỉ còn hai phần tử

Thuật toán mã hóa Huffman (Cont.)

Bước 2:

- Xây dựng cây nhị phân, với gốc là phần tử giả. Các cây con trái phải tương ứng là hai phần tử có xá suất lớn nhất. Tiến trình tiếp tục cho các phần tử tiếp theo cho đến khi hoàn chỉnh.
- Tiến hành mã hóa tín hiệu: Bắt đầu GỐC, nếu đi xuống cây con trái thì thêm vào bit 0, cây con phải thì thêm vào bit 1. Tiến trình kết thúc khi tất cả các nút lá đều được gán mã.

Ví dụ (Cont.)



$a_2 \rightarrow 1$
$a_6 \rightarrow 00$
$a_1 \rightarrow 011$
$a_4 \rightarrow 0101$
$a_3 \rightarrow 01000$
$a_5 \rightarrow 01001$

Ví dụ

- Cho ảnh 8 cấp xám (3 bit) kích thước 100×100

Nguồn		Giảm nguồn			
Tín hiệu	X/S	1	2	3	4
a_2	0.4	0.4	0.4	0.4	→ 0.6
a_6	0.3	0.3	0.3	0.3	→ 0.4
a_1	0.1	0.1	→ 0.2	→ 0.3	
a_4	0.1	0.1	→ 0.1	→ 0.1	
a_3	0.06	→ 0.1			
a_5	0.04				

Ví dụ (Cont.)

$$L_{\text{avg}} = (0.4) \times 1 + (0.3) \times 2 + (0.1) \times 3 + (0.1) \times 4 + (0.04) \times 5 + (0.06) \times 5 = 2.2$$

- Kích thước ảnh gốc: $100 \times 100 \times 3 = 30000$ bits
- Kích thước ảnh sau khi nén: $100 \times 100 \times 2.2 = 22000$ bits
- Tỷ lệ nén $30000/22000 \sim 1.36$
- Độ dư thừa dữ liệu: $1 - (22000/30000) \sim 27\%$

Phương pháp LZW

- Phương pháp LZW xây dựng một từ điển để lưu các mẫu có tần suất xuất hiện cao trong ảnh
- Sự có mặt của một chuỗi con trong từ điển khẳng định chuỗi đó có xuất hiện trong ảnh
- Thuật toán liên tục tra cứu và cập nhật từ điển sau mỗi lần đọc một ký tự ở dữ liệu đầu vào
- Kích thước từ điển giới hạn ở 4096 phần tử (12 bit)

Cấu trúc từ điển (Cont.)

- Các từ mã từ 0..255 được biểu diễn bằng 8 bit
- Các từ mã từ 256..511 được biểu diễn bằng 9 bit
- Các từ mã từ 512 đến 1023 biểu diễn bởi 10 bit
- Các từ mã từ 1024 đến 2047 được biểu diễn bởi 11 bit
- Các từ mã từ 2048 đến 4095 được biểu diễn bởi 12 bit.

0	0
1	1
...	...
255	255
256	256
257	257
258	Chuỗi
...	...
4095	Chuỗi

Cấu trúc từ điển

- 256 từ mã đầu tiên theo thứ tự 0..255 chứa các số nguyên từ 0 đến 255. Đây là mã của 256 ký tự ASCII cơ bản
- Từ mã 256 chứa mã đặc biệt gọi là mã xóa (CC-Clear Code)
- Từ mã 257 chứa mã kết thúc (EOI – End Of Information)
- Các từ mã từ 258 trở đi chứa các mẫu thường lặp lại trong ảnh

0	0
1	1
...	...
255	255
256	256
257	257
258	Chuỗi
...	...
4095	Chuỗi

Ví dụ mã nén LZW

- Cho chuỗi đầu: "**ABCBCABCABCD**"
- Mã của A là 65, B là 66, C là 67, D là 68
- Chuỗi có dạng: 65-66-67-66-67-65-66-67-68
- Từ điển ban đầu đã gồm 256 ký tự cơ bản

Ví dụ mã nén LZW (Cont.)

Đầu vào	Đệm	Đầu ra	Thực hiện
A(65)	A		A đã có trong từ điển → đọc tiếp
B(66)	AB	65	Thêm từ vào từ điển mã 258 đại diện cho chuỗi AB
C(67)	BC	66	Thêm vào từ điển mã 259 đại diện cho chuỗi BC
B	CB	67	Thêm vào từ điển mã 260 đại diện cho chuỗi CB
C			BC đã có trong từ điển → đọc tiếp
A	BCA	259	Thêm vào từ điển mã 261 đại diện cho chuỗi BCA
B	AB		AB đã có trong từ điển → đọc tiếp
C	ABC	258	Thêm vào từ điển mã 262 đại diện cho chuỗi ABC
A	CA	67	Thêm vào từ điển mã 263 đại diện cho chuỗi CA
B	AB		AB đã có trong từ điển → đọc tiếp
C	ABC		ABC đã có trong từ điển → đọc tiếp
D	ABCD	262	Thêm vào từ điển mã 264 đại diện cho chuỗi ABCD

"**ABCBCABCABCD**" → 65-66-67-259-258-67-262

33

Ví dụ mã nén LZW (Cont.)

- Với chuỗi đầu vào "ABCBCABCABCD"
- Mã đầu vào: 65-66-67-66-67-65-66-67-65-66-67-68
- Mã đầu ra: 65-66-67-259-258-67-262
- Kích thước dữ liệu đầu vào: $12 \times 8 = 96$ bit
- Kích thước dữ liệu đầu ra: $4 \times 8 + 3 \times 9 = 59$ bit
- Tỷ lệ nén: $96/59 \sim 1.63$
- Độ dư thừa dữ liệu: $1-(59/96) \sim 39\%$