

Ảnh là gì?

- Ảnh là một biểu diễn thị giác dưới dạng một hàm $f(x,y)$ trong đó $f(x,y)$ là độ sáng hoặc màu tại điểm (x, y)
- Các ảnh thường là hình chữ nhật
- Liên tục trong biên độ và không gian

Xử Lý Ảnh

ẢNH SỐ LÀ GÌ?

19

20

Lấy mẫu

- Đối với máy tính xử lý, ảnh liên tục phải được rời rạc hóa theo không gian. Quá trình này được gọi là lấy mẫu.
- Một ảnh liên tục $f(x,y)$ có thể được xấp xỉ bởi mảng $M \times N$

$$f[x, y] \approx \begin{bmatrix} f[0,0] & \dots & f[0, N-1] \\ f[1,0] & \dots & f[1, N-1] \\ \vdots & \ddots & \vdots \\ f[M-1, 0] & \dots & f[M-1, N-1] \end{bmatrix}$$

21

Lấy mẫu

- Phần bên phải thường được xem như là một ảnh rác
- Quá trình lấy mẫu được xem như là việc phân chia mặt phẳng xy thành một lưới mà các đỉnh là các phần tử của tích Cartesian $Z \times Z$ với Z là tập các số nguyên
- Nếu Δx và Δy lần lượt chia cắt lưới điểm theo hướng x và y ta có $f[m, n] = f(m\Delta x, n\Delta y)$, với $m = 0 \dots M-1, n = 0 \dots N-1$
- Qui trình lấy mẫu yêu cầu chỉ rõ Δx và Δy hoặc M và N (tương ứng với các chiều của ảnh)

22

Lấy mẫu



FIGURE 2.19 A 1024×1024 , 8-bit image subsampled down to size 32×32 pixels. The number of allowable gray levels was kept at 256.

23

Lấy mẫu

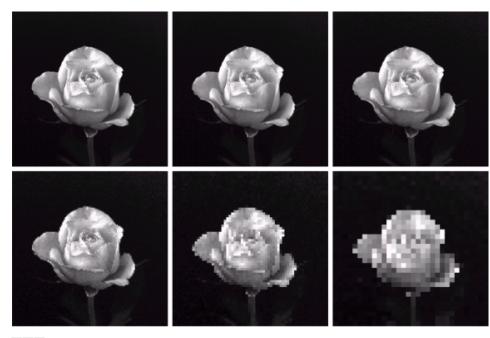
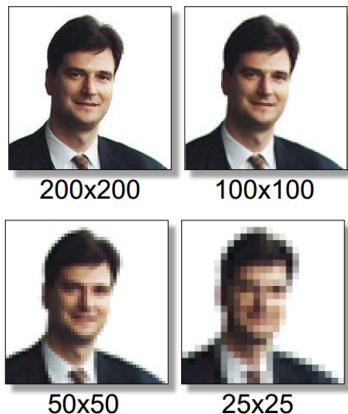


FIGURE 2.20 (a) 1024×1024 , 8-bit image. (b) 512×512 image resampled into 1024×1024 pixels by row and column duplication. (c) through (f) 256×256 , 128×128 , 64×64 , and 32×32 images resampled into 1024×1024 pixels.

24

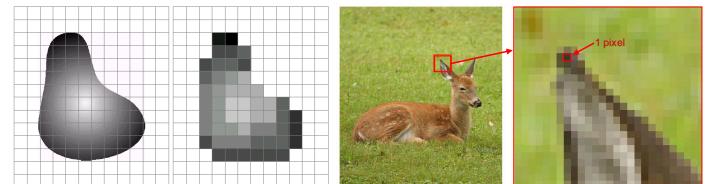
Lấy mẫu



25

Ảnh số và điểm ảnh

- **Ảnh số:** các mẫu rời rạc $f[x,y]$ trình bày cho ảnh liên tục $f(x,y)$
- Mỗi phần tử của mảng 2 chiều $f[x,y]$ được gọi là **điểm ảnh (pixel)**

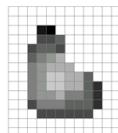


26

Hàm ảnh

• Ảnh xạ f

$$f: [0, M-1] \times [0, N-1] \rightarrow L \\ [x,y] \rightarrow f[x,y]$$

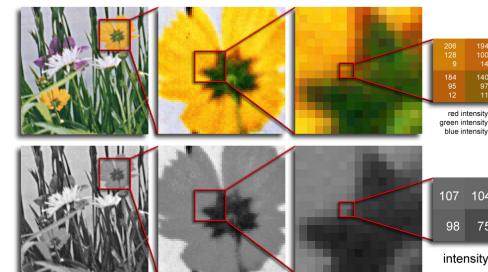


• Trong đó

- L là miền giá trị màu
- $f[x,y]$ cho biết giá trị màu tại điểm $[x,y]$ trên ảnh

27

Hàm ảnh



Ảnh màu có 3 kênh màu trên một điểm ảnh; ảnh thuần nhất có giá trị trên một điểm ảnh.

28

Mô hình màu

- Mô hình màu là cách thức tổ chức cụ thể để tạo ra màu sắc
- Một số mô hình màu thông dụng
 - RGB (hiển thị màn hình)
 - CMY (in ấn)
 - HSV
 - ...



Xử Lý Ảnh

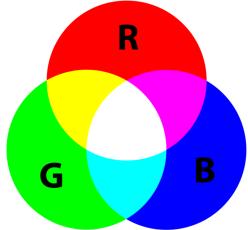
MỘT SỐ HỆ MÀU THÔNG DỤNG

29

30

Hệ màu RGB

- Mô hình màu được dùng trên màn hình CRT, LCD, ...
- Được cấu thành từ 3 kênh màu: Red, Green và Blue



31

Độ xám (Grayscale)

- Chỉ gồm một thành phần màu
 - Chỉ gồm 1 kênh màu: gray
 - Miền giá trị: $0 \leq \text{gray} \leq 255$
 - Lưu trữ: 8bits/điểm ảnh



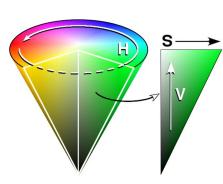
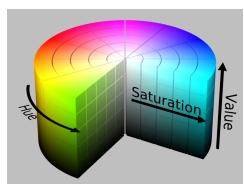
0

255

32

Hệ màu HSV

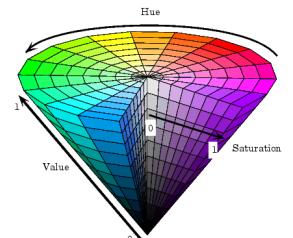
- Được cấu thành từ 3 kênh
 - Hue: sắc độ
 - Saturation: độ thuần khuyết (độ pha trắng)
 - Value: độ sáng màu



33

Hệ màu HSV

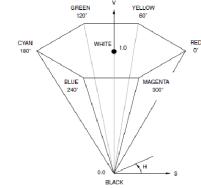
- Định lượng màu
 - Hue: gom các màu có sắc độ gần giống nhau lại thành nH bin
 - Saturation: gom các màu có độ thuần khuyết gần giống nhau lại thành nS bin
 - Value: gom các màu có cường độ sáng gần giống nhau lại thành nV bin



34

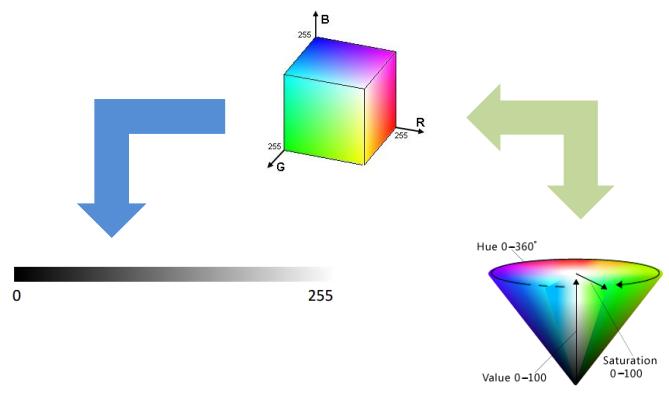
Hệ màu HSV

- Định lượng màu
 - Rời rạc hóa không gian màu
 - Các màu có cùng giá trị cảm nhận được gom lại
 - Dễ gọi tên màu
 - Dễ mô tả màu cho máy tính



35

Chuyển đổi hệ màu



36

RGB sang Grayscale

$$Gray = 0.30 \times R + 0.59 \times G + 0.11 \times B$$



RGB sang HSV

$$V = \max(R, G, B)$$

$$S = \begin{cases} \frac{V - \min(R, G, B)}{V} & \text{nếu } V \neq 0 \\ 0 & \text{ngược lại} \end{cases}$$

$$H = \begin{cases} 60(G - B)/S & \text{nếu } V = R \\ 120 + 60(B - R)/S & \text{nếu } V = G \\ 240 + 60(R - G)/S & \text{nếu } V = B \end{cases}$$

$$\text{nếu } H < 0, H = H + 360$$

37

38

HSV sang RGB



<http://www.had2know.com/technology/hsv-rgb-conversion-formula-calculator.html>

39

Xử Lý Ảnh

LƯU TRỮ VÀ NÉN DỮ LIỆU

40

Lưu trữ

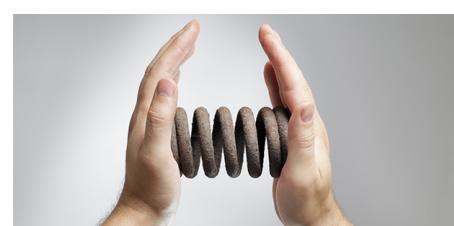
- Một ảnh màu có kích thước 720x480, tốn bao nhiêu bytes để lưu trữ?
- Một ảnh màu có kích thước 1920x1080, tốn bao nhiêu bytes lưu trữ?



41

Nén dữ liệu

- Các phương pháp nén
 - Nén bảo toàn thông tin
 - Nén mất mát thông tin

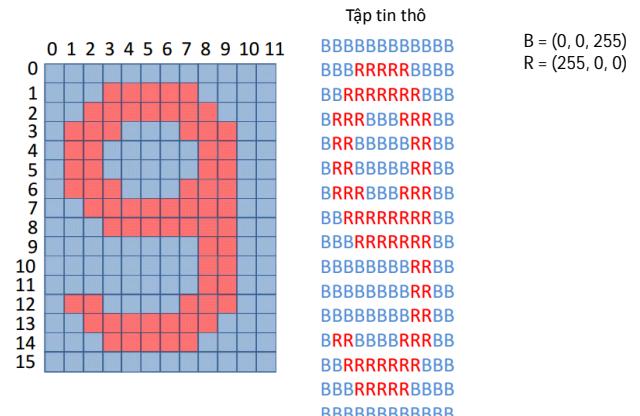


42

Nén bảo toàn thông tin

- Một số phương pháp
 - Run-length encoding.
 - Chain codes.
 - Entropy encoding.
 - Huffman coding
 - Arithmetic coding
 - Dictionary algorithm family:
 - Deflation.
 - Adaptive dictionary algorithm.

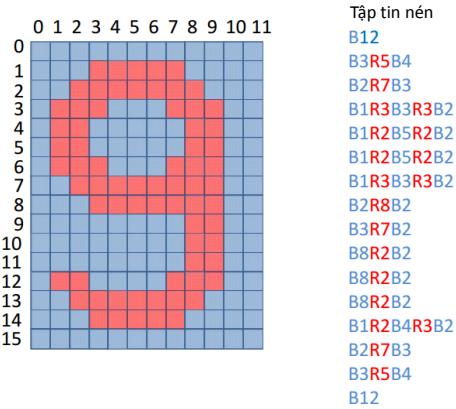
Run-length encoding



43

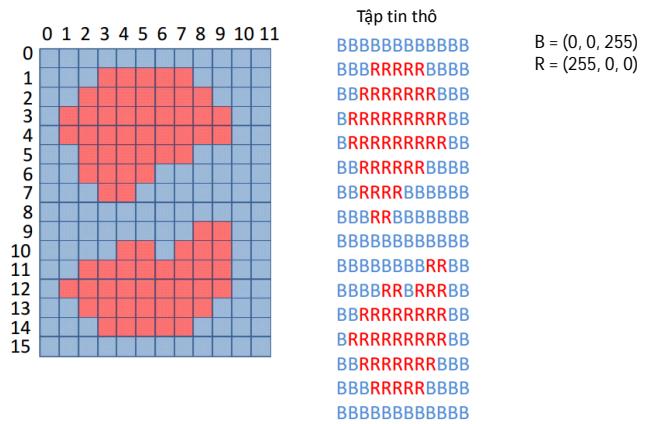
44

Run-length encoding



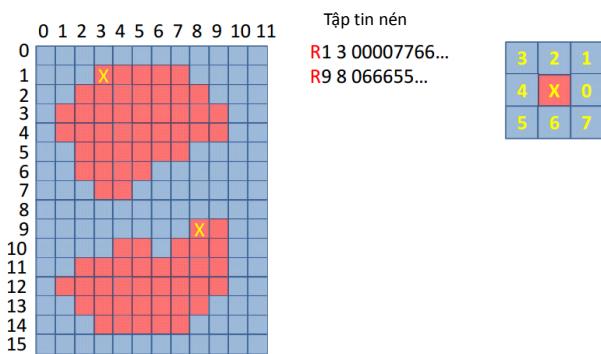
45

Chain codes



46

Chain codes



47

Huffman Coding

- Xét ký tự đại diện: A, B, C, D, E
- Giả sử có 1 dòng dữ liệu như sau:
ABCADAEABCA
- Thống kê tần suất xuất hiện
- Sắp xếp theo tần xuất giảm dần
- Xây dựng cây Huffman
- Dựa vào cây Huffman để phát sinh mã

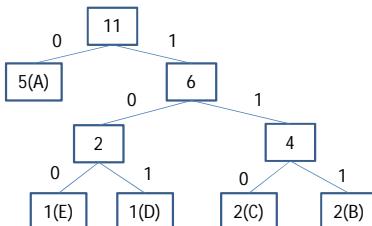
48

Huffman Coding

ABCADAEABCA

Ký tự	Tần suất
A	5
B	2
C	2
D	1
E	1

Ký tự	Mã
A	0
B	111
C	110
D	101
E	100



49

Huffman Coding

- Thay chuỗi thành bảng mã tương ứng

– ABCADAEABCA

– 0111110101100

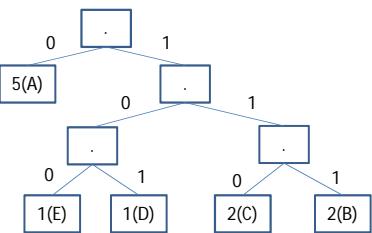
Ký tự	Mã
A	0
B	111
C	110
D	101
E	100

50

Huffman Coding

- Đọc bảng mã tái tạo cây Huffman

Ký tự	Mã
A	0
B	111
C	110
D	101
E	100



51

Nén mất mát thông tin

- Một số phương pháp

– Reducing the color space.

– Transform coding.

– Fractal compression.

- Duyệt chuỗi dữ liệu tìm đến nút là để giải mã

– 0111110101100

52

Reducing the color space

0	1	2	3	4	5
6	7	8	9	?	!
n	N	A	a	#	@
o	O	B	b	\$	%
p	P	C	c	&	I
q	Q	D	d	,	.
r	R	E	e	;	:
s	S	F	f	+	-
t	T	G	g	*	/
u	U	H	h	'	,
v	V	I	i	<	>
w	W	J	j	()
x	X	K	k	[]
y	Y	L	l	{	}
z	Z	M	m	^	=

Transform coding

- Là phương pháp được dùng phổ biến nhất hiện nay
- Cho dung lượng ảnh nhỏ và chất lượng ảnh tốt

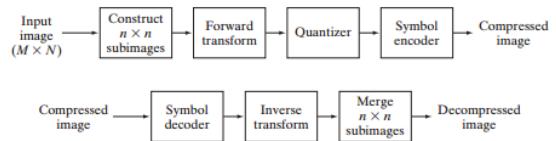
53

54

Transform coding

- Thuật toán nén
 - Chia ảnh thành từng ô lưới
 - Dùng phép biến đổi Cosine rời rạc (phép biến đổi Fourier khả nghịch) đưa từng ô lưới ảnh từ miền không gian sang miền tần số
 - Lượng hóa ảnh trong miền tần số
 - Dùng mã Huffman để mã hóa
- Thuật toán giải nén
 - Dùng mã Huffman để giải mã
 - Dùng phép biến đổi Cosine rời rạc nghịch đảo đưa dữ liệu từ miền tần số trở lại miền không gian
 - Ráp các ô lưới ảnh lại để được ảnh sau giải nén

Transform coding

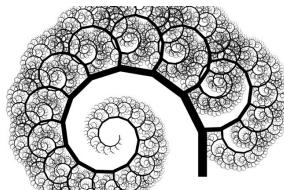


55

56

Nén Fractal

- Phù hợp nhất cho các ảnh thiên nhiên
- Dựa trên hiện tượng lặp lại của các thành phần chi tiết trên ảnh cần nén
- Rút ra tập luật sinh các chi tiết lặp lại từ tập các chi tiết hạt giống
- Lưu tập luật sinh và hạt giống



Nén Fractal



57

58

Nén Fractal



Nén Fractal



59

60

Một số dạng tập tin ảnh



Xử Lý Ảnh

MỘT SỐ ĐỊNH DẠNG ẢNH THÔNG DỤNG

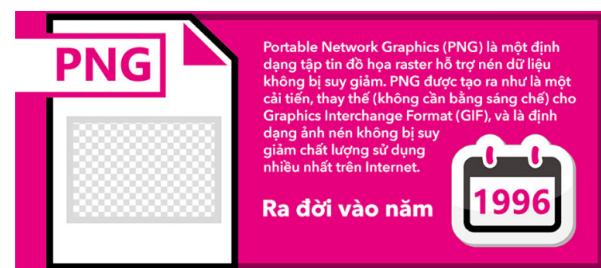
61

62

JPEG



PNG



63

64

GIF



TIFF



65

66

BMP



67

	JPG	TIF	PNG	GIF	BMP
Chuẩn nén Lossy	✓	✗	✗	✗	✗
Chuẩn nén Lossless	✗	✓	✓	✓	✓
Tùy chọn không nén	✓	✗	✗	✗	✓
Grayscale	✓	✓	✓	✓	✗
Màu RGB	✓	✓	✓	✗	✓
8 bit màu (24 bit data)	✓	✓	✓	✗	✓
Tùy chọn 16 bit màu	✗	✓	✓	✗	✓
CMYK hoặc màu LAB	✓	✓	✗	✗	✓
Tùy chọn màu Indexed	✓	✓	✓	✗	✗
Tùy chọn trong suốt	✗	✓	✓	✓	✓
Tùy chọn hình động	✗	✗	✗	✓	✗
Hỗ trợ nhiều Layer	✗	✓	✗	✓	✓
In ấn hoặc Web	Cả hai	In ấn	Web	Web	In ấn

68

JPEG	HÌNH ẢNH WEB
Có lẽ đây là định dạng phổ biến và thông dụng nhất cho hình ảnh web. Định dạng file JPEG là lựa chọn tốt nhất cho nhu cầu lưu hình ảnh lại với kích thước nhỏ gọn, chừng hạn như khi bạn cần phải upload nó online hoặc gửi email. Nếu bạn không cần quan tâm lắm đến chất lượng hình ảnh hoặc có thể hình ảnh sẽ bị giảm chất lượng đi một chút không đáng kể thì hãy nên dùng định dạng JPEG.	
GIF	HÌNH ẢNH WEB/ HÌNH ĐỘNG/ CLIP ART
Trong số ba định dạng (GIF, JPEG & PNG) thì GIF là lựa chọn tốt nhất đối với hình ảnh đồ họa web, mặc dù kích thước tệp tin của nó rất nhỏ, và nó được load rất nhanh. Thêm vào đó, nếu bạn muốn thêm các hiệu ứng hình ảnh động, thì nên sử dụng GIF. Nó cũng là định dạng tốt cho ảnh Clip Art.	
TIFF	HÌNH SỬ DỤNG ĐỂ IN ẤN
TIFF là sự lựa chọn tốt nhất và duy nhất cho các chuyên gia khi hình ảnh được dùng để in. Nó có khả năng đọc được màu sắc CMYK và YCbCr, cộng với khả năng có thể lưu được ảnh với mật độ pixel lớn làm cho nó trở thành sự lựa chọn duy nhất cho các nhà thiết kế, nhiếp ảnh gia và các nhà xuất bản.	
PNG	HÌNH ẢNH WEB/ LOGO & LINE ART
PNG là định dạng tốt cho hình ảnh đồ họa web. Nếu bạn muốn file ảnh có kích thước nhỏ, nhưng vẫn giữ được chất lượng, thì nên sử dụng PNG. Ngoài ra nếu bạn muốn sử dụng những ảnh có dạng trọng suất như logo hoặc những icon không có nền, để chèn lên bên trên những hình ảnh khác... PNG là định dạng dành cho bạn.	
BMP	HÌNH SỬ DỤNG ĐỂ IN ẤN
Những file với định dạng này có kích thước rất lớn và không hỗ trợ nén, những hình ảnh của nó mang lại rất tốt, màu sắc đậm đà, file chất lượng cao, đơn giản dễ sử dụng và có khả năng tương thích tốt nhất trong tất cả các phần mềm và các hệ điều hành Windows. File BMP cũng còn được gọi là ảnh lưới (raster) hoặc paint.	

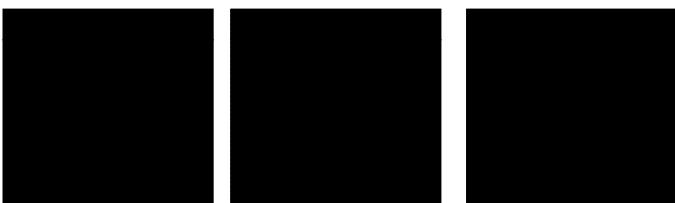
69

Xử Lý Ảnh

XỬ LÝ ẢNH SỐ LÀ GÌ?

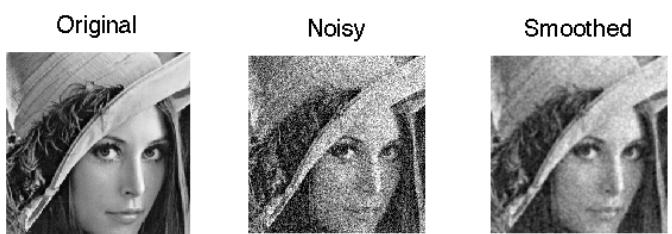
Xử lý ảnh số là gì?

- Xử lý ảnh số là quá trình thao tác trên ảnh đầu vào để tạo ra ảnh đầu ra theo một số điều kiện cho trước hoặc các thuộc tính liên quan đến ảnh bằng máy tính



71

Cấp thấp - Khử nhiễu ảnh

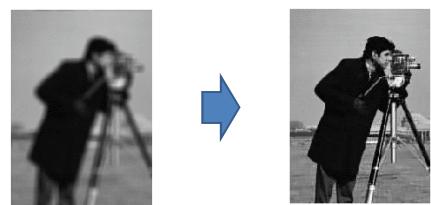


72

Cấp thấp - Sắc nét ảnh



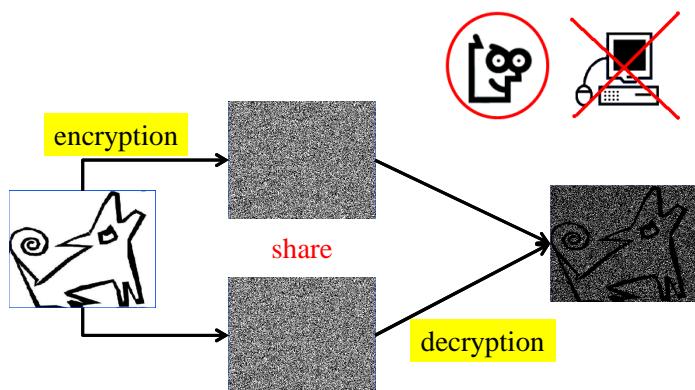
Cấp thấp - Phục hồi ảnh



73

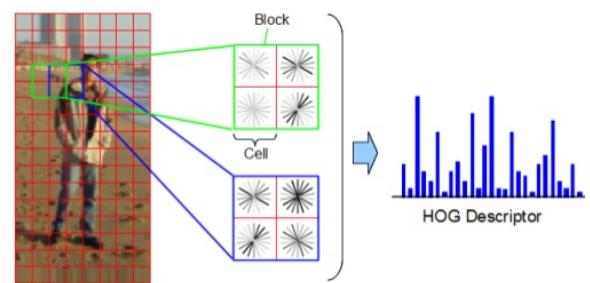
74

Cấp thấp - Mật mã thị giác



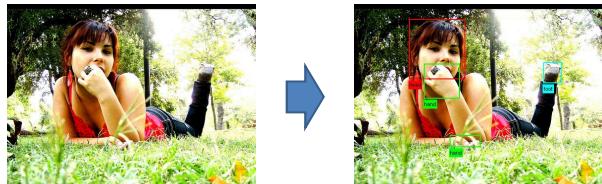
75

Cấp trung – Trích đặc trưng



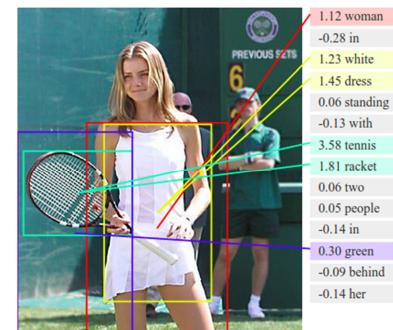
76

Cấp cao – Chú thích ảnh



77

Cấp cao – Mô tả ảnh



78

Cấp cao – Mô tả ảnh



two young girls are playing with lego toy

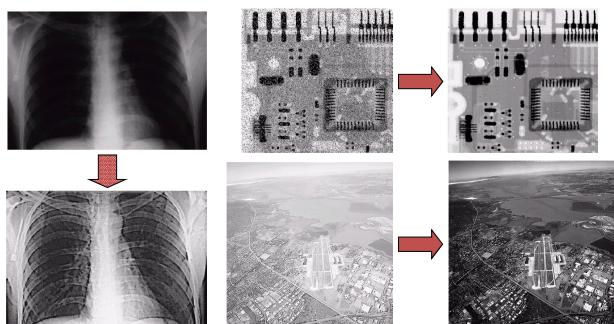
Xử Lý Ảnh

ỨNG DỤNG CỦA XỬ LÝ ẢNH

79

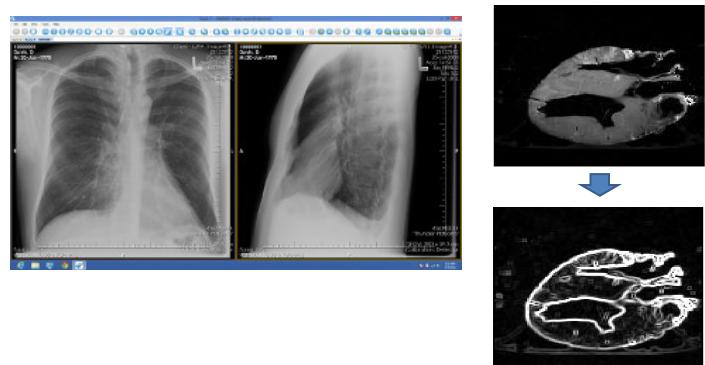
80

Tăng cường chất lượng ảnh



81

Chuẩn đoán ảnh y khoa



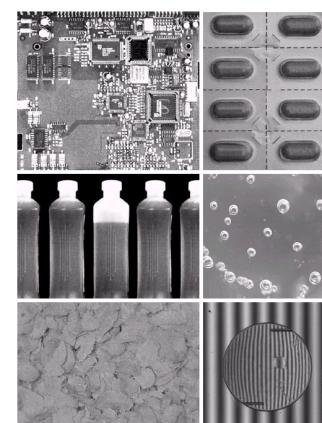
82

Ứng dụng giải trí Prisma



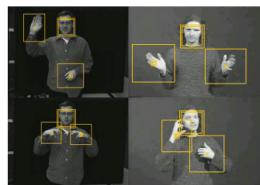
83

Kiểm tra công nghiệp



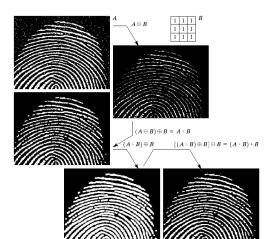
84

Tương tác người máy



85

Thực thi pháp luật



86

Bảo vệ quyền riêng tư



87

Các thành phần của hệ thống xử lý ảnh

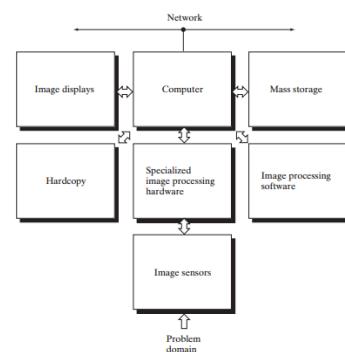
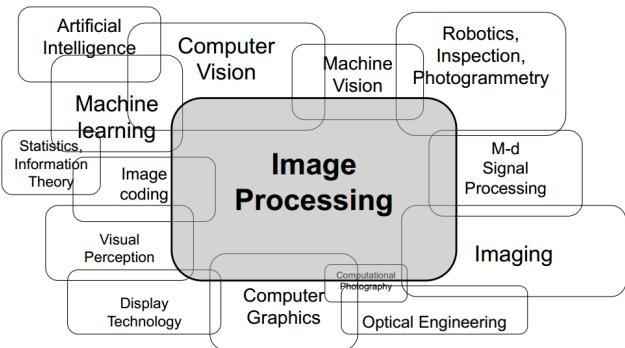


FIGURE 1.24
Components of a general-purpose image processing system.

88

Xử lý ảnh và lĩnh vực khác



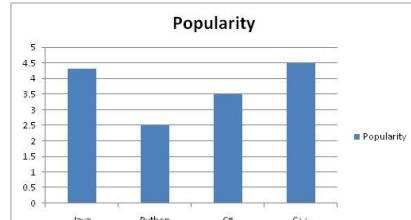
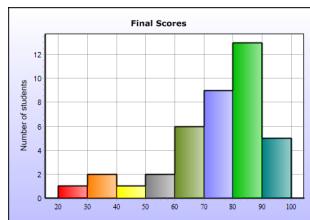
Bernd Girod, Digital Image Processing, EE368/CS232



90

Lược đồ (Histogram) là gì?

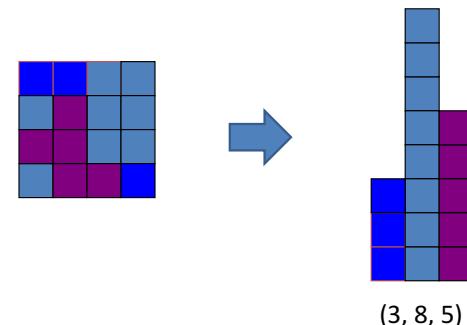
- Trong thống kê, lược đồ là đồ thị thể hiện thông tin về phân bố của dữ liệu



4

Lược đồ màu

- Phân bố của các điểm ảnh cùng mang một giá trị màu trong ảnh



(3, 8, 5)

5

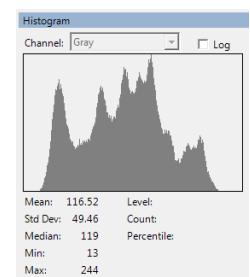
Lược đồ màu - Ảnh xám

- Lược đồ màu ảnh xám

$$H[i] = \frac{N[i]}{N}$$

- Trong đó

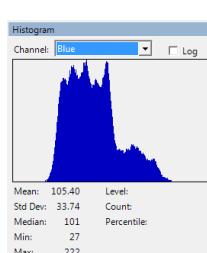
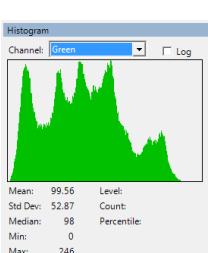
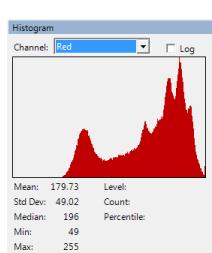
- $N[i]$ là số điểm ảnh mang màu i trong ảnh
- N là tổng số điểm ảnh có trong ảnh



6

7

Lược đồ màu



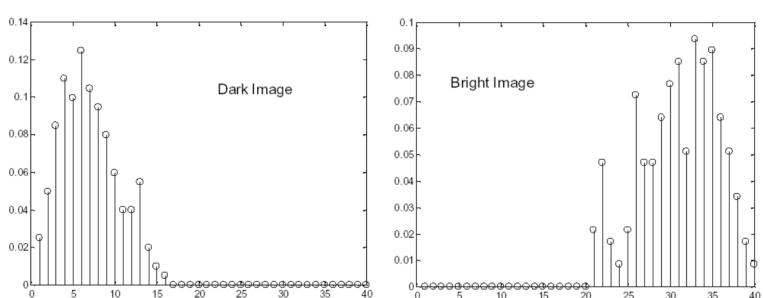
8

Lược đồ lượng hóa màu - Ảnh xám

254	143	200	178	105	229	177	220	192	0	329	142	198	64	0	0	0	28	0	0	0	0
27	68	231	75	141	107	149	210	13	239	141	35	68	242	110	208	244	0	33	88		
54	40	17	210	239	254	47	41	90	180	95	253	230	47	122	209	78	110	192	100		
9	180	192	71	101	193	88	17	37	233	19	147	174	1	143	211	170	188	192	68		
179	23	238	192	190	192	41	248	22	134	83	138	110	254	170	238	188	234	51	204		
232	25	0	183	174	159	01	30	110	189	0	173	197	183	133	43	22	87	68	118		
235	35	151	185	129	81	239	170	195	94	38	21	67	101	59	37	190	149	62	154		
155	240	244	0	104	109	189	47	194	254	225	196	91	181	181	15	128	35	232	203		
223	114	79	129	147	0	201	68	89	107	58	44	230	84	39	1	62	5	231	218		
55	188	237	189	80	101	131	241	09	133	181	151	111	29	190	6	240	78	117	145		
132	156	229	78	90	217	219	105	116	77	38	49	2	9	214	181	203	116	135	33		
182	94	178	190	20	149	07	223	232	110	32	45	177	19	31	170	100	119	209	81		
224	110	154	172	75	28	69	190	187	195	61	44	8	170	198	101	131	31	212			
238	83	38	7	83	69	173	185	98	237	67	227	18	218	248	237	75	192	201	148		
98	195	254	207	140	22	31	110	234	94	185	110	23	47	69	542	189	152	116	248		
140	37	101	230	246	145	122	64	27	38	229	1	225	145	91	110	196	90	49	195		
251	4	178	139	181	95	97	174	249	105	77	115	223	180	182	62	65	232	63	196		
179	109	223	230	87	102	149	76	170	19	17	4	184	170	193	102	83	81	132	206		
173	137	185	242	181	181	214	49	75	238	197	37	86	105	15	217	149	8	102	186		
85	9	17	222	16	210	70	21	76	241	104	216	93	93	206	102	153	212	119	47		

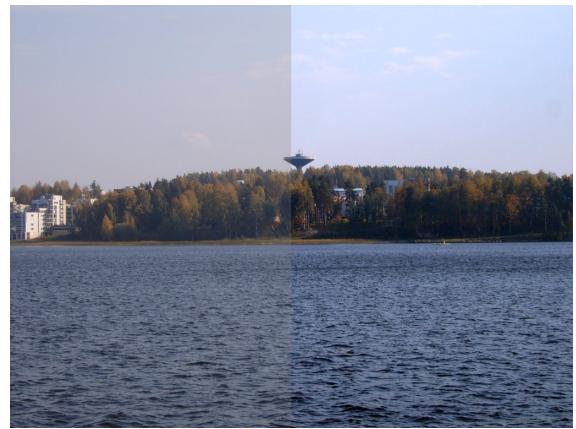
9

Một số dạng lược đồ



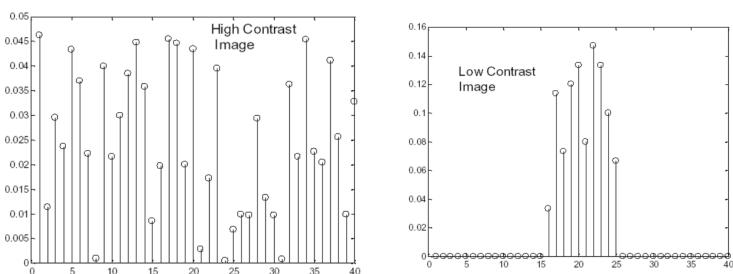
10

Một số dạng lược đồ



11

Một số dạng lược đồ



12

Lược đồ màu - ảnh xám

Bài tập

- Viết chương trình tính lược đồ màu xám trong một ảnh I và hiển thị ra màn hình
- Viết chương trình tính lược đồ lượng hóa màu xám thành 8 bin màu trong một ảnh I và hiển thị ra màn hình

13

Lược đồ màu – hệ RGB

Lược đồ màu RGB

$$H[r][g][b] = \frac{N[r][g][b]}{N}$$

Trong đó

- r, g và b là các kênh màu trong ảnh
- N[r][g][b] là số điểm ảnh mang màu (r,g,b) trong ảnh
- N là tổng số điểm ảnh có trong ảnh

Lược đồ màu – hệ RGB

Lược đồ từng kênh màu trong ảnh RGB

$$H_b[i] = \frac{N_b[i]}{N}$$

Trong đó

- b là các kênh màu {R, G, B}
- Nb[i] là số điểm ảnh mang giá trị thứ i của kênh màu b
- N là tổng số điểm ảnh

14

15

Lược đồ màu – hệ RGB

- Bài tập

- Viết chương trình tính lược đồ cho từng kênh màu trong hệ màu RGB của một ảnh I và hiển thị ra màn hình.
- Viết chương trình tính lược đồ màu lượng hóa RGB cho một ảnh I và hiển thị ra màn hình.

16

Lược đồ màu – hệ HSV

- Lược đồ màu RGB

$$H[h][s][v] = \frac{N[h][s][v]}{N}$$

- Trong đó

- h, s và v là các kênh màu trong ảnh
- N[h][s][v] là số điểm ảnh mang màu (h,s,v) trong ảnh
- N là tổng số điểm ảnh có trong ảnh

17

Lược đồ màu – hệ HSV

- Bài tập

- Viết chương trình tính lược đồ cho từng kênh màu trong hệ màu HSV của một ảnh I và hiển thị ra màn hình.
- Viết chương trình tính lược đồ màu lượng hóa HSV cho một ảnh I và hiển thị ra màn hình.

18

Xử Lý Điểm Ảnh

ĐỐI SÁNH ẢNH

19

Đối sánh ảnh

- Đánh giá sự tương đồng hoặc khác biệt của 2 ảnh



- Một số phương pháp thông dụng
 - Dựa trên điểm ảnh
 - Dựa trên lược đồ

So sánh điểm ảnh

- Ảnh xám

$$d(f, g) = \sum_{x=1}^w \sum_{y=1}^h |f(x, y) - g(x, y)|$$

$$d(f, g) = \sqrt{\sum_{x=1}^w \sum_{y=1}^h (f(x, y) - g(x, y))^2}$$

Trong đó:

- f, g là 2 ảnh cần đối sánh
- w, h lần lượt là chiều rộng và chiều cao của ảnh

20

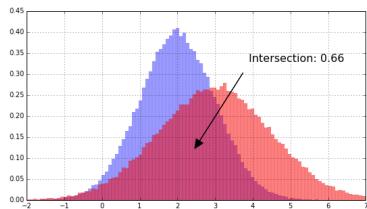
21

So sánh lược đồ màu

Độ đo Chi-square $d(H_1, H_2) = \sum_i \frac{(H_1[i] - H_2[i])^2}{H_1[i]}$

Độ đo Intersection $d(H_1, H_2) = \sum_i \min(H_1[i], H_2[i])$

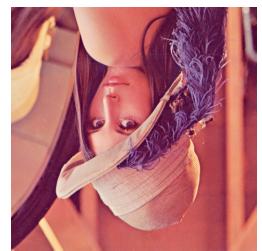
Trong đó $H[i]$ là giá trị bin màu thứ i của lược đồ H



22



Ảnh I



Ảnh J

23

Nhận xét

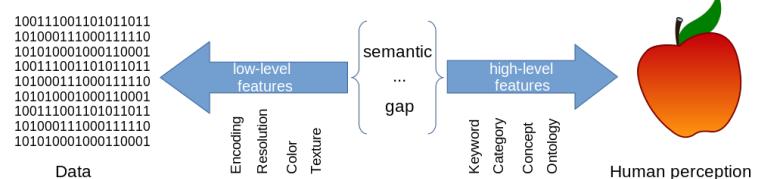


Ảnh I



Ảnh J

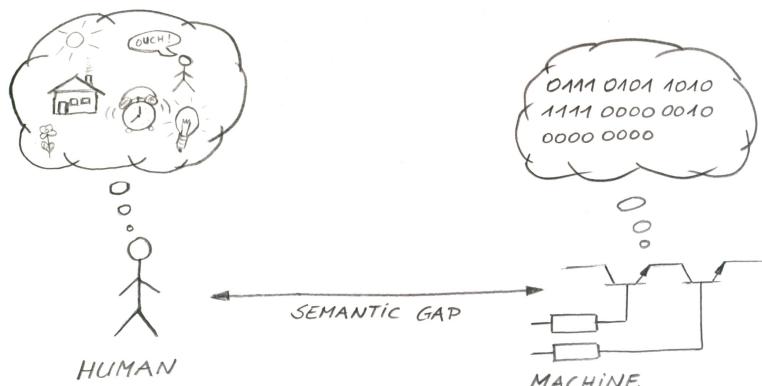
Nhận xét



24

25

Nhận xét



26

Đối sánh ảnh

Bài tập

- Viết chương trình đối sánh 2 ảnh dựa vào điểm ảnh và dựa vào lược đồ màu

27

Xử lý điểm ảnh là gì?

- Giá trị đầu ra của điểm ảnh $J(m,n)$ tại điểm ảnh (m,n) chỉ phụ thuộc vào giá trị đầu vào $I(m,n)$ (và không phụ thuộc vào giá trị của các điểm ảnh lân cận)

Xử Lý Ảnh

XỬ LÝ ĐIỂM ẢNH LÀ GÌ?

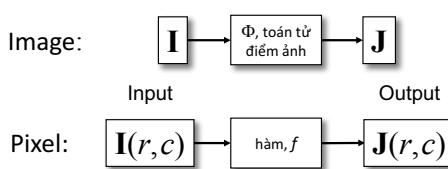
$$s = T(r)$$

Với s là giá trị đầu ra của điểm ảnh,
 r là giá trị đầu vào của điểm ảnh

28

29

Xử lý điểm ảnh qua hàm ánh xạ



Chuyển ảnh I thành ảnh J bằng cách thay giá trị đầu vào, g , bằng một giá trị cụ thể, k , tại mỗi vị trí (r,c,b) với $I(r,c,b) = g$.

$$J = F[I]$$

If $I(r,c,b) = g$
and $f(g) = k$
then $J(r,c,b) = k$.

Luật này liên quan giữa k và g thường được chỉ ra bằng một hàm cụ thể, f , sao cho $f(g) = k$.

30

Xử lý điểm ảnh qua hàm ánh xạ

Ảnh một kênh màu

$J(r,c) = f(I(r,c))$,
Cho tất cả các điểm ảnh tại vị trí, (r,c) .

Ảnh ba kênh màu

$J(r,c,b) = f(I(r,c,b))$, hoặc
 $J(r,c,b) = f_b(I(r,c,b))$, hoặc
Với $b = 1, 2, 3$, và tất cả (r,c) .

Xử lý điểm ảnh qua hàm ánh xạ

Một kênh màu
Mỗi kênh màu đều
được ảnh xạ thông
qua cùng một hàm, f ,
hoặc ...

$J(r,c) = f(I(r,c))$,
Cho tất cả các điểm ảnh tại vị trí, (r,c) .

Ba kênh màu

$J(r,c,b) = f(I(r,c,b))$, hoặc
 $J(r,c,b) = f_b(I(r,c,b))$, hoặc

Với $b = 1, 2, 3$, và tất cả (r,c)
... mỗi kênh màu
được ảnh xạ qua
một hàm riêng
biệt, f_b .

32

- Bảng tra là danh sách được chỉ mục các số có thể được dùng để thực hiện một hàm rời rạc, ánh xạ từ một tập số nguyên $\{g_{in,1}, g_{in,2}, \dots, g_{in,n}\}$ sang một tập các số $\{g_{out,1}, g_{out,2}, \dots, g_{out,n}\}$ (nguyên hoặc không). Một bảng tra có thể được định nghĩa như sau:

if $g_{out} = f(g_{in})$, where $g_{in} \in \{0, K, n-1\}$ and $g_{out} \in \{g_{out,k}\}_{k=1}^n$
then define $LUT(g_{in,k}+1) = LUT(k) = g_{out,k}$, for $k = 1, K, n$.

33

Xử lý điểm ảnh dùng bảng tra

Một bảng tra (LUT) có thể được sử dụng như một hàm ánh xạ.

Nếu $k = f(g)$
với $g = 0, \dots, 255$

Và nếu k có giá trị trong $\{0, \dots, 255\}, \dots$

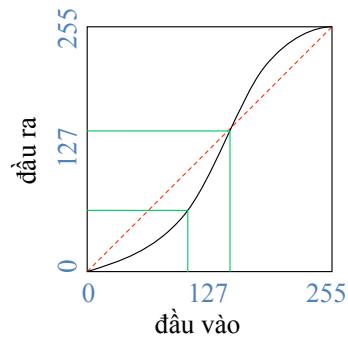
Tái ánh xạ
ảnh, I, to J :

... mà LUT được dùng như hàm f là một mảng 256×1 với giá trị $(g+1)^{\text{th}}$ là $k = f(g)$.

LUT là 256×1 .
Nhưng có thể là $R \times C$ hoặc $R \times C \times 3$.

$J = \text{LUT}(I+1)$

Xử lý điểm ảnh bằng bảng tra



VD.:	chỉ mục	giá trị
...
101	64	64
102	68	68
103	69	69
104	70	70
105	70	70
106	71	71
...

Tạo tác điểm ảnh màu dùng bảng tra

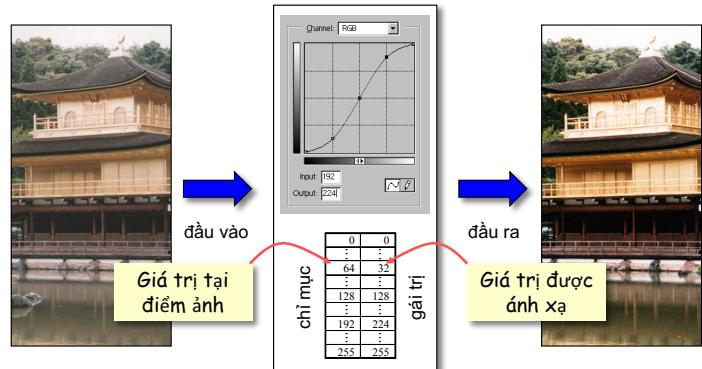
Nếu I là ảnh **3-kênh màu**,

- Mỗi kênh màu được ánh xạ tách biệt bằng cách sử dụng cùng LUT cho mỗi kênh màu *hoặc*
- Mỗi kênh màu được ánh xạ bằng nhiều LUTs - một hàm cho một kênh màu.

a) $J = \text{LUT}(I+1)$,

b) $J(:,:,b) = \text{LUT}_b(I(:,:,b)+1)$, với $b = 1, 2, 3$.

Một LUT cho 3 kênh màu



Một số thao tác biến đổi

- Điều chỉnh độ sáng ảnh
- Điều chỉnh độ tương phản của ảnh
- Lượng hóa màu
- Phân ngưỡng
- Cân bằng lược đồ ảnh
- Khớp lược đồ ảnh

Tăng độ sáng ảnh

$$J(r, c, b) = \begin{cases} I(r, c, b) + g, & \text{nếu } I(r, c, b) + g < 256 \\ 255, & \text{nếu } I(r, c, b) + g > 255 \end{cases}$$

$g \geq 0$ và $b \in \{1, 2, 3\}$ là các kênh màu



40

Giảm độ sáng ảnh

$$J(r, c, b) = \begin{cases} I(r, c, b) - g, & \text{nếu } I(r, c, b) - g \geq 0 \\ 0, & \text{nếu } I(r, c, b) - g < 0 \end{cases}$$

$g \geq 0$ và $b \in \{1, 2, 3\}$ là các kênh màu



41

Tăng độ tương phản

$$\mathbf{T}(r, c, b) = a[\mathbf{I}(r, c, b) - s] + s$$

$$\mathbf{J}(r, c, b) = \begin{cases} 0, & \text{if } \mathbf{T}(r, c, b) < 0, \\ \mathbf{T}(r, c, b), & \text{if } 0 \leq \mathbf{T}(r, c, b) \leq 255, \\ 255, & \text{if } \mathbf{T}(r, c, b) > 255. \end{cases}$$

$a > 1$, $s \in \{0, K, 255\}$, $b \in \{1, 2, 3\}$

s là trung tâm của hàm tương phản.



42

Giảm độ tương phản

$$\mathbf{T}(r, c, b) = a[\mathbf{I}(r, c, b) - s] + s,$$

where $0 \leq a < 1.0$,

$s \in \{0, 1, 2, K, 255\}$, and

$b \in \{1, 2, 3\}$.



43

Giãn độ tương phản ảnh

Let $m_i = \min[\mathbf{I}(r, c)]$, $M_i = \max[\mathbf{I}(r, c)]$,

$m_j = \min[\mathbf{J}(r, c)]$, $M_j = \max[\mathbf{J}(r, c)]$.

Then,

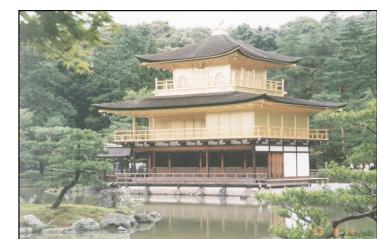
$$\mathbf{J}(r, c) = (M_j - m_j) \frac{\mathbf{I}(r, c) - m_i}{M_i - m_i} + m_j.$$



44

Tăng gamma

$$\mathbf{J}(r, c) = 255 \cdot \left[\frac{\mathbf{I}(r, c)}{255} \right]^{\frac{1}{\gamma}} \quad \text{for } \gamma > 1.0$$



45

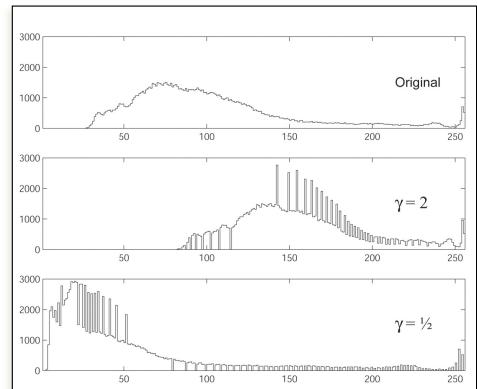
Giảm gama

$$J(r,c) = 255 \cdot \left[\frac{I(r,c)}{255} \right]^\gamma \quad \text{for } 0 < \gamma < 1.0$$



46

Tác động Gama lên lược đồ màu



47



Xử Lý Điểm Ảnh

LƯỢNG HÓA MÀU

Lượng hóa màu

- Một ảnh biểu diễn trong hệ màu RGB có 256 mức màu cho mỗi kênh hoặc $256^3 = 16777216$ màu
- Làm thế nào một ảnh có thể được hiển thị với ít hơn các màu mà nó chứa?

Lựa chọn một tập con của các màu và ánh xạ các màu còn lại vào chúng

48

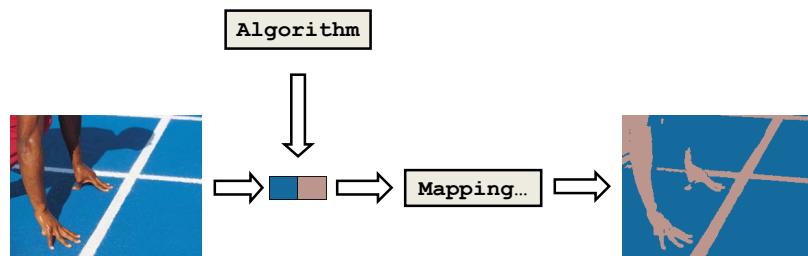
49

Lượng hóa màu



256 colors 50

Qui trình lượng hóa màu



4 colors

51

Qui trình lượng hóa màu



Xử Lý Điểm Ảnh

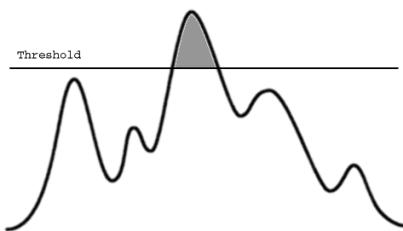
PHÂN NGƯỠNG

52

53

Phân ngưỡng

- Phân ngưỡng toàn cục
- Phân ngưỡng tự động



54

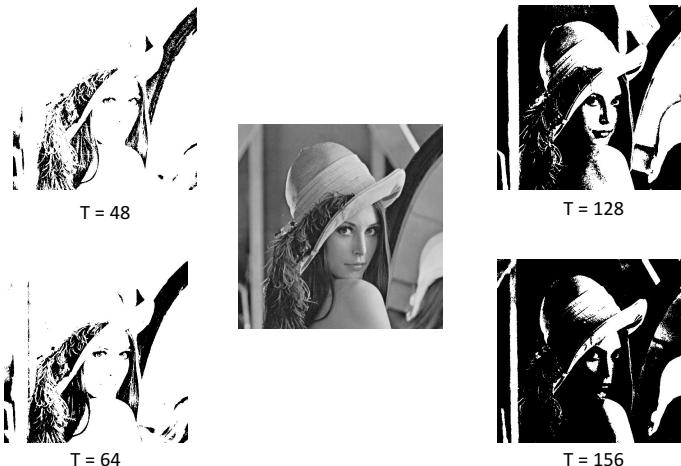
55

Phân ngưỡng toàn cục

$$J_b(i,j) = \begin{cases} 0, & \text{nếu } I_b(i,j) < T \\ 255, & \text{ngược lại} \end{cases}$$

- Trong đó
 - b là các kênh màu
 - T là ngưỡng toàn cục

Phân ngưỡng toàn cục



56

Phân ngưỡng tự động

- Dựa vào ảnh đầu vào để xác định ngưỡng
- Sử dụng các kỹ thuật thống kê hoặc heuristics

$$T = \text{mean}$$

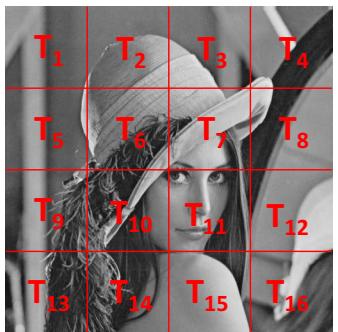
$$T = \text{median}$$

$$T = \frac{\max + \min}{2}$$

15	33	3
27	23	40
11	50	9

57

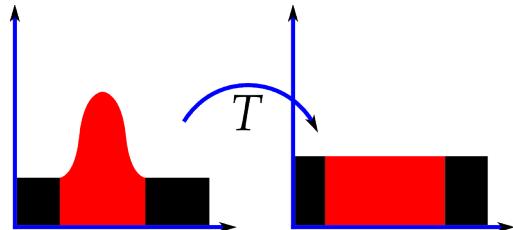
Phân ngưỡng thích nghi



58

Cân bằng lược đồ màu

- Điều chỉnh lại giá trị độ xám của các điểm ảnh trong ảnh I để tạo ra ảnh J sao cho lược đồ màu của J là lược đồ màu của I giãn ra càng nhiều càng tốt



59

Cân bằng lược đồ màu



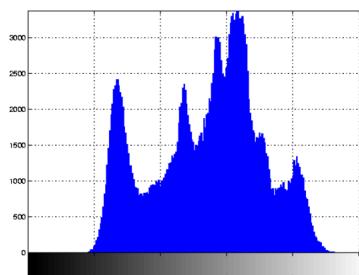
Ảnh gốc



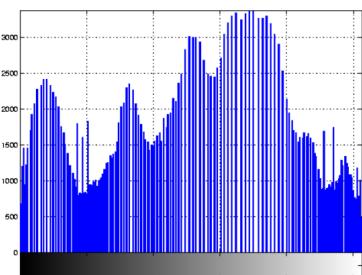
Ảnh sau khi cân bằng

60

Cân bằng lược đồ màu



Lược đồ ảnh gốc



Lược đồ ảnh gốc cân bằng

61

Cân bằng lược đồ màu



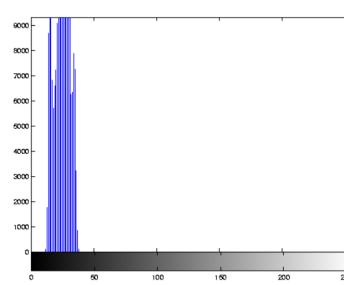
Ảnh gốc



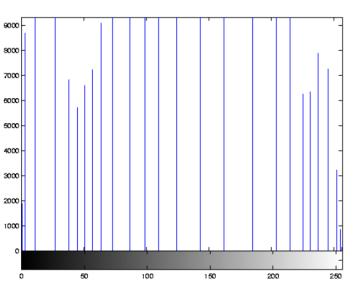
Ảnh sau khi cân bằng

62

Cân bằng lược đồ màu



Lược đồ ảnh gốc



Lược đồ ảnh gốc cân bằng

63

Cân bằng lược đồ màu

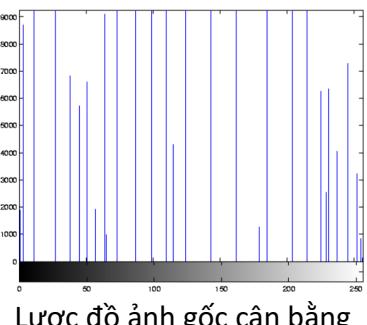
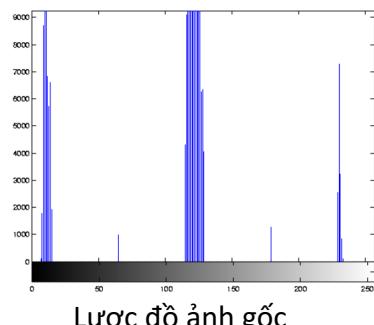


Ảnh gốc



Ảnh sau khi cân bằng

Cân bằng lược đồ màu



64

65

Cân bằng lược đồ màu

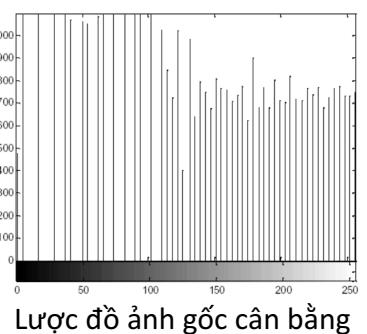
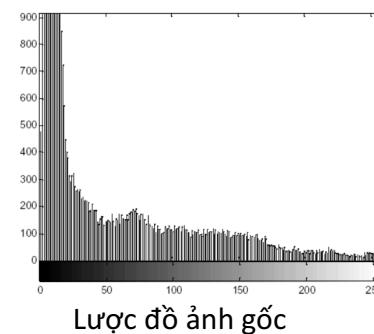


Ảnh gốc



Ảnh sau khi cân bằng

Cân bằng lược đồ màu

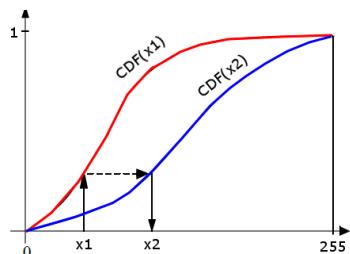


66

67

Khớp lược đồ ảnh

- Cho 2 ảnh I và J
- Điều chỉnh lược đồ màu của ảnh I sao cho: “Lược đồ của ảnh I càng giống lược đồ của ảnh J càng tốt”



68

Khớp lược đồ ảnh



Ảnh gốc



Ảnh đích



Ảnh khớp

69

Nội dung

- Lọc không gian là gì?
- Tích chập xoắn
- Lọc trung bình
- Lọc trung vị
- Lọc gauss

Lọc không gian là gì?

3	3	3
3	3	3
3	3	3

Giá trị của điểm ảnh trung tâm là bao nhiêu?

3	4	3
2	3	3
3	4	2

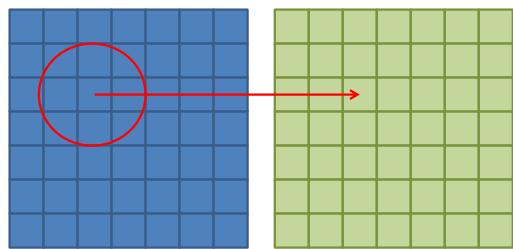
Giả định của bạn là gì để xác định giá trị của điểm ảnh trung tâm?

3

4

Lọc không gian là gì?

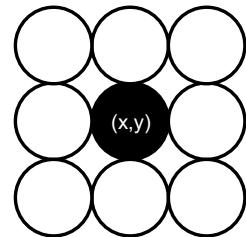
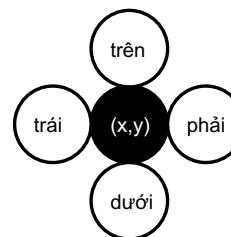
- Giá trị tại một điểm ảnh phụ thuộc vào giá trị của các điểm trong không gian lân cận của nó.



$J(x,y) = \Phi[I](x,y)$
 Φ là hàm thao tác
trên lân cận điểm
ảnh

- Có hai loại lân cận cơ bản : lân cận 4 và lân cận 8.

- **Lân cận 4:** $N_4(x, y) = \{(x-1, y), (x, y+1), (x+1, y), (x, y-1)\}$
- **Lân cận 8:** $N_8(x, y) = \{(x-1, y), (x-1, y+1), (x, y+1), (x+1, y+1), (x+1, y), (x+1, y-1), (x, y-1), (x-1, y-1)\}$



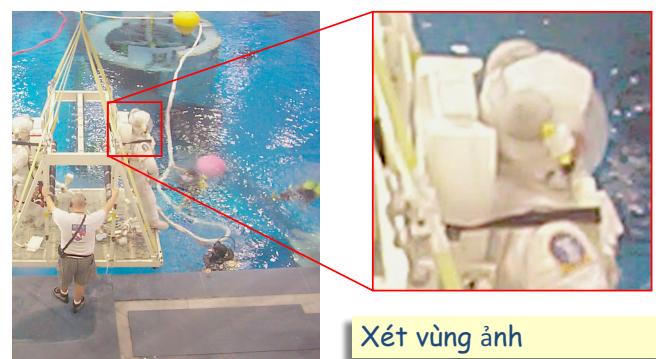
5

6

Lọc không gian là gì?

- Lấy I, J là sao ảnh sao cho $J = \Phi[I]$ và một cửa sổ hình vuông $M \times N$ (cả M và N đều lẻ)
- $J(r,c) = \Phi[I](r,c)$
 $= \Phi[\{I(\rho, \chi) | \rho \in \{r-m, \dots, r, \dots, r+m\}, \chi \in \{c-n, \dots, c, \dots, c+n\}\}].$
- Giá trị độ sáng mà J nhận được tại (r,c) là kết quả của hàm Φ tác động lên giá trị độ sáng của các điểm ảnh trên I nằm trong cửa sổ hình chữ nhật $M \times N$, $(2m+1) \times (2n+1)$, quanh vị trí (r,c) trên I

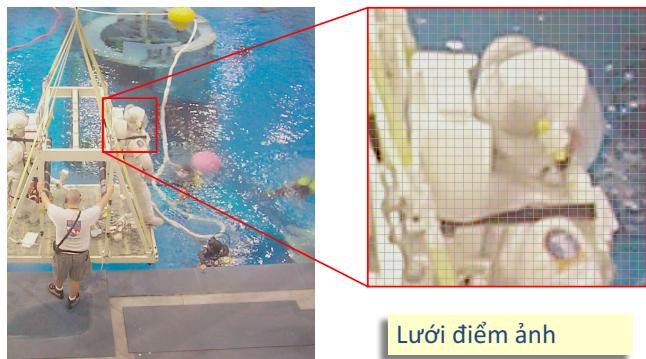
Dịch chuyển cửa sổ



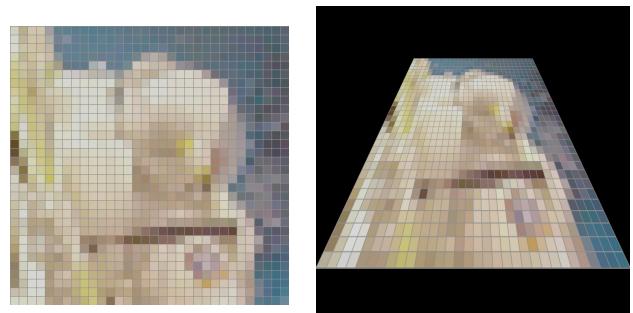
7

8

Dịch chuyển cửa sổ



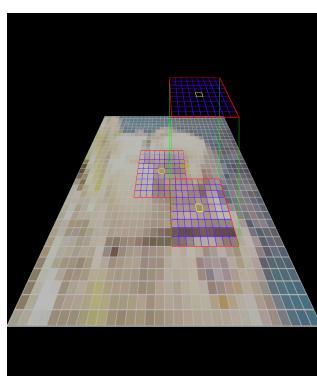
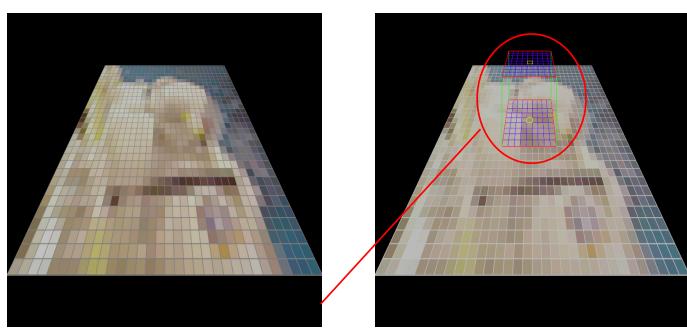
Dịch chuyển cửa sổ



9

10

Dịch chuyển cửa sổ



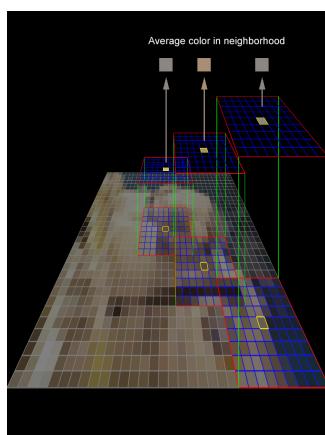
Dịch chuyển cửa sổ

11

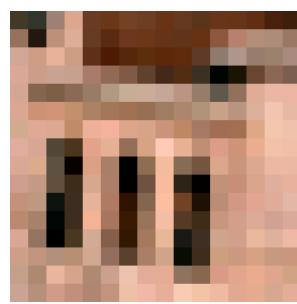
12

Dịch chuyển cửa sổ

Giá trị độ sáng tại r, c trên ảnh J bằng trung bình cộng giá trị độ sáng của những điểm quanh vị trí này trên ảnh I



Di chuyển cửa sổ



original

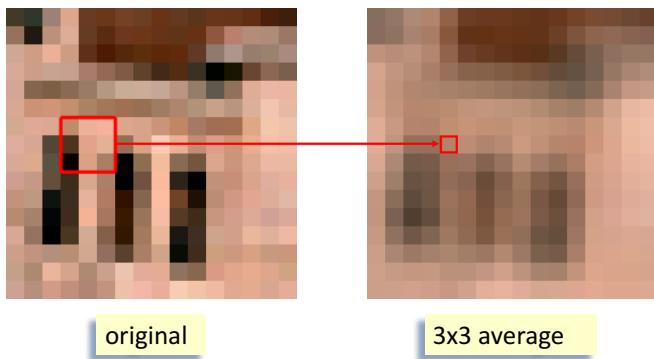


3x3 average

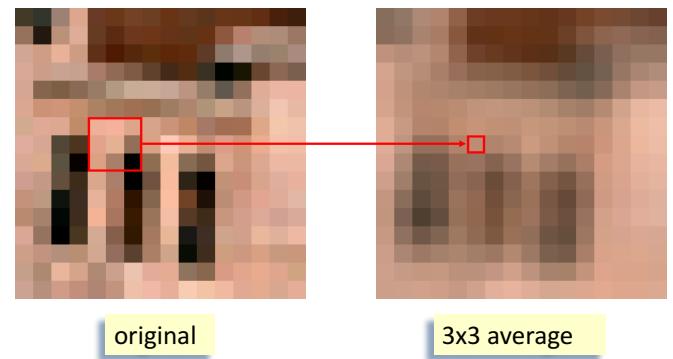
13

14

Di chuyển cửa sổ



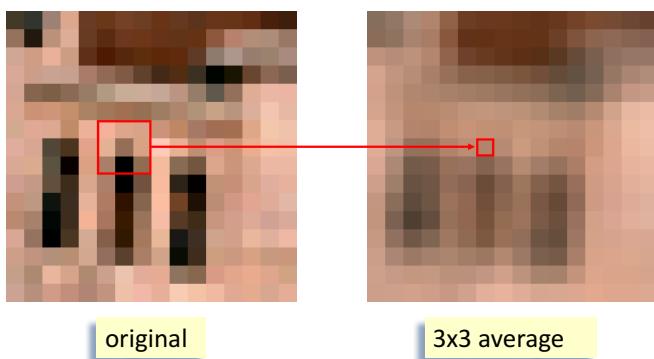
Di chuyển cửa sổ



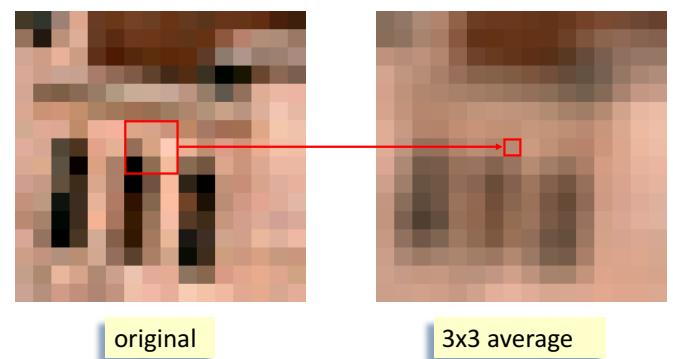
15

16

Di chuyển cửa sổ



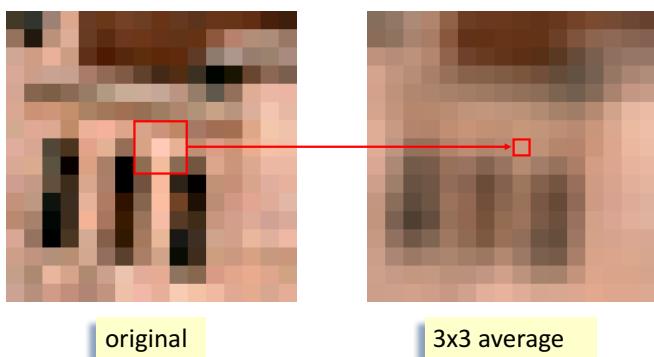
Di chuyển cửa sổ



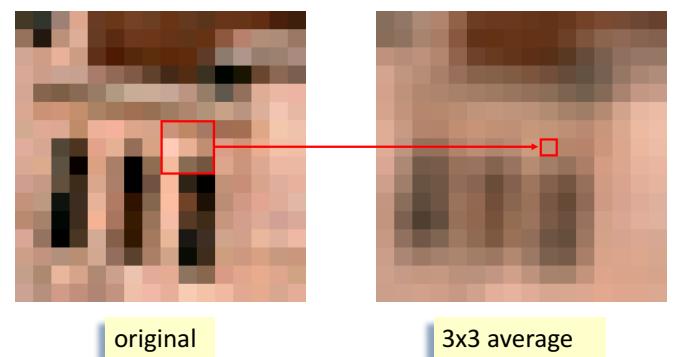
17

18

Di chuyển cửa sổ



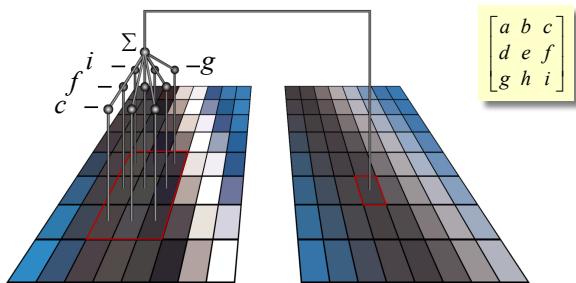
Di chuyển cửa sổ



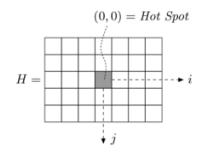
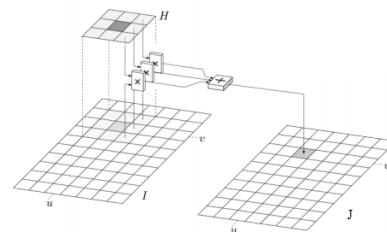
19

20

Tích chập xoắn



Tích chập xoắn



$$J(u, v) = \sum_{i=-1}^{i=1} \sum_{j=-1}^{j=1} I(u - i, v - j) \cdot H(i, j)$$

21

22

Tích chập xoắn

$$J(u, v) = \sum_{i=-m}^{i=m} \sum_{j=-n}^{j=n} I(u - i, v - j) \cdot H(i, j)$$

- H được gọi là mặt nạ (mask) hoặc nhân (kernel)
- H là ma trận số
- H là cửa sổ dịch chuyển trên ảnh gốc
- H là hàm đánh trọng số cho mỗi điểm trong cửa sổ dịch chuyển trên ảnh gốc
- Giá trị tại vị trí (i,j) trên ảnh kết quả là tổng theo trọng số giá trị của các điểm ảnh trong lân cận cửa sổ dịch chuyển quanh (i,j) do H trên ảnh gốc

1	2	1
-1	0	-1
1	-2	1

10	11	10	0	0	1
9	10	11	1	0	1
10	9	10	0	2	1
11	10	9	10	9	11
9	10	11	9	99	11
10	9	9	11	10	10

Tích chập xoắn

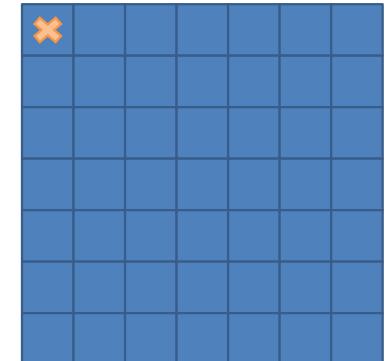
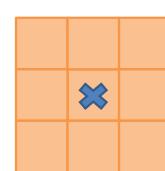
23

24

Tích chập xoắn

1	0	1
1	0	1
1	0	1

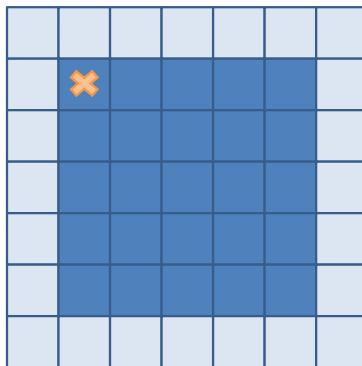
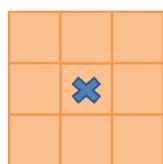
10	11	10	0	0	1
9	10	11	1	0	1
10	9	10	0	2	1
11	10	9	10	9	11
9	10	11	9	99	11
10	9	9	11	10	10



25

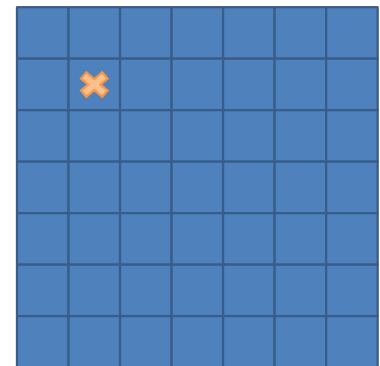
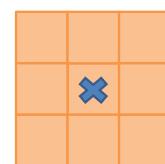
26

Tích chập xoắn



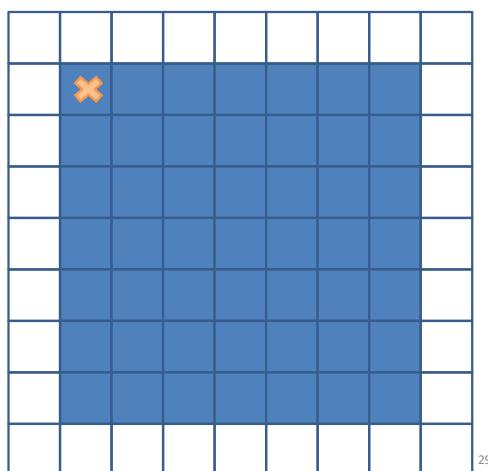
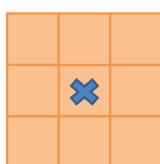
27

Tích chập xoắn



28

Tích chập xoắn



Zero padding

29

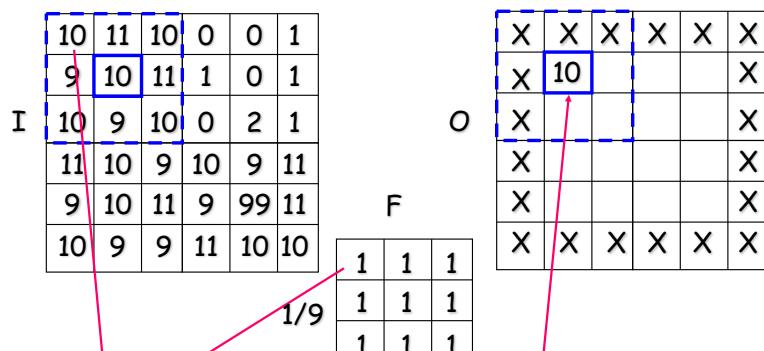
Lọc trung bình

- Sử dụng một mặt nạ trung bình
- Mặt nạ trung bình là mặt nạ có trọng số dương mà tổng là 1. Nó được tính dựa trên trung bình có trọng số giá trị điểm ảnh của các lân cận
- Một số mặt nạ trung bình

$$\frac{1}{5} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \frac{1}{32} \begin{bmatrix} 1 & 3 & 1 \\ 3 & 16 & 3 \\ 1 & 3 & 1 \end{bmatrix} \quad \frac{1}{8} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

30

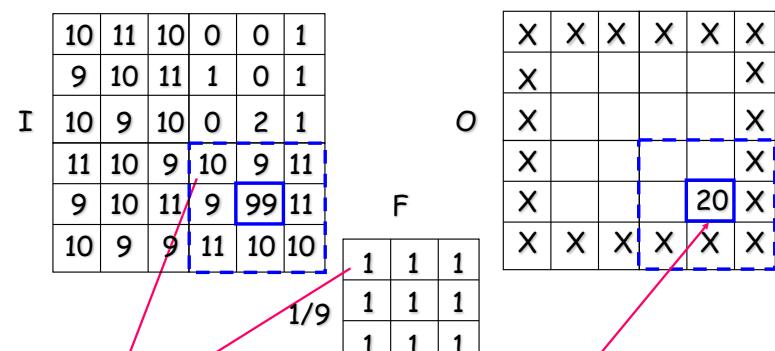
Lọc trung bình



$$\frac{1}{9} \cdot (10 \cdot 1 + 11 \cdot 1 + 10 \cdot 1 + 9 \cdot 1 + 10 \cdot 1 + 11 \cdot 1 + 10 \cdot 1 + 9 \cdot 1 + 10 \cdot 1) = \frac{1}{9} \cdot (90) = 10$$

31

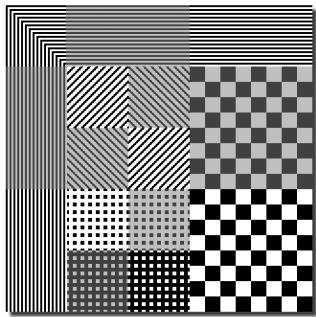
Lọc trung bình



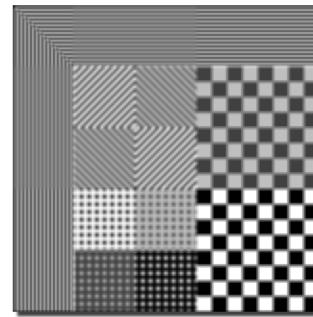
$$\frac{1}{9} \cdot (10 \cdot 1 + 9 \cdot 1 + 11 \cdot 1 + 9 \cdot 1 + 99 \cdot 1 + 11 \cdot 1 + 11 \cdot 1 + 10 \cdot 1 + 10 \cdot 1) = \frac{1}{9} \cdot (180) = 20$$

32

Lọc trung bình



Lọc trung bình

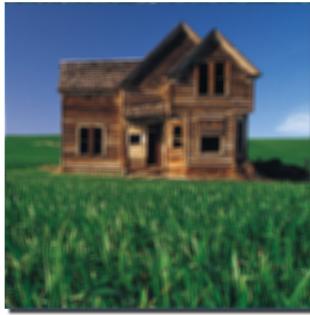
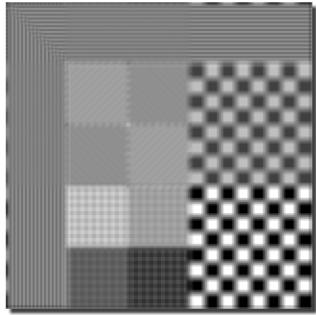


5×5

33

34

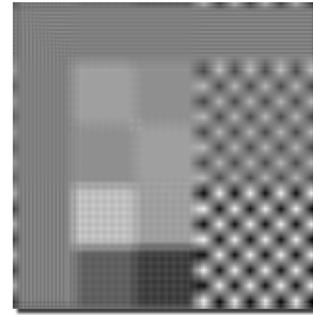
Lọc trung bình



9×9

35

Lọc trung bình



17×17

36

Lọc trung bình

- Bài tập (Cá nhân)
 - Viết chương trình lọc trung bình trên ảnh độ xám
 - Viết chương trình lọc trung bình trên ảnh màu
- Yêu cầu nộp bài
 - Báo cáo kết quả thực hiện chương trình (hình ảnh minh họa, nhận xét)

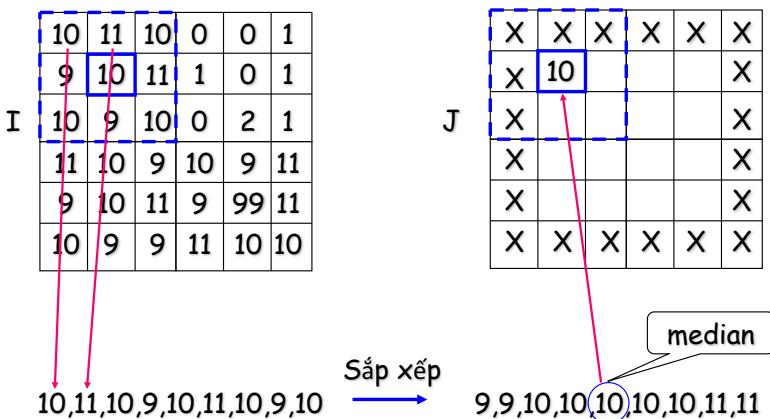
Lọc trung vị

- Lọc trung vị là bộ lọc phi tuyến
 - Thay mỗi điểm ảnh bởi giá trị trung vị của các điểm ảnh lân cận nó.
 - Giả sử $A = \{a_1, a_2, \dots, a_n\}$ là giá trị của các điểm ảnh trong lân cận của điểm ảnh cho trước với $a_1 \leq a_2 \leq \dots \leq a_n$
- $$\text{median}(A) = \begin{cases} a_{n/2} & \text{với } n \text{ chẵn} \\ a_{(n+1)/2} & \text{với } n \text{ lẻ} \end{cases}$$
- Ví dụ $A = \{0, 1, 2, 4, 6, 6, 10, 12, 15\}$ thì $\text{median}(A) = 6$

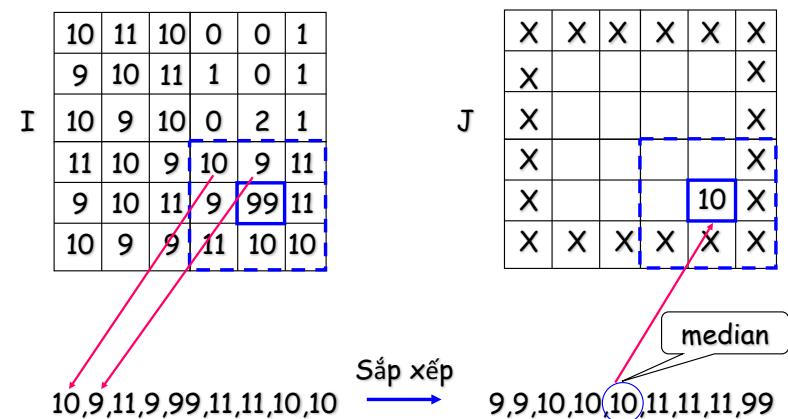
37

38

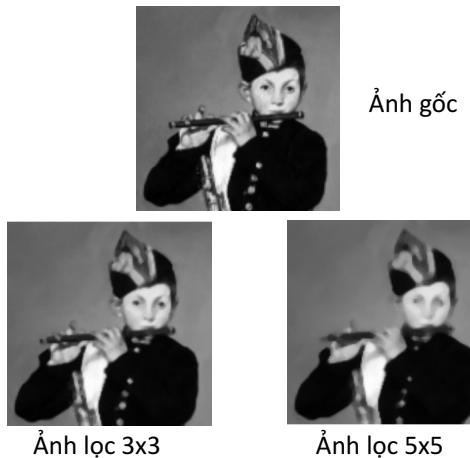
Lọc trung vị



Lọc trung vị



Lọc trung vị

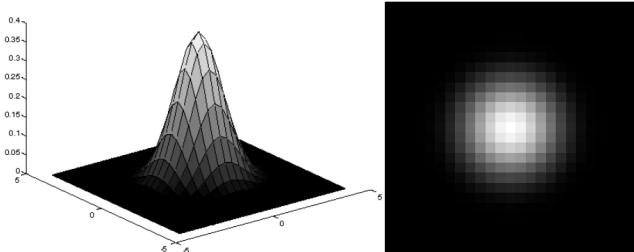


Lọc trung vị

- Bài tập cá nhân
 - Viết chương trình lọc trung vị trên ảnh độ xám
 - Viết chương trình lọc trung vị trên ảnh màu
- Yêu cầu nộp bài
 - Báo cáo kết quả thực hiện chương trình (hình ảnh minh họa, nhận xét)

Lọc Gauss

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$



- Ma trận gauss 5x5 với sigma = 1

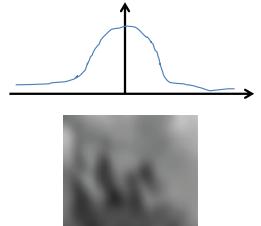
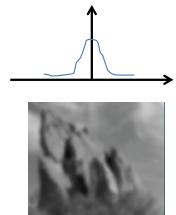
0.0001	0.0015	0.0067	0.0111	0.0067	0.0015	0.0001
0.0015	0.0183	0.0821	0.1353	0.0821	0.0183	0.0015
0.0067	0.0821	0.3679	0.6065	0.3679	0.0821	0.0067
0.0111	0.1353	0.6065	1.0000	0.6065	0.1353	0.0111
0.0067	0.0821	0.3679	0.6065	0.3679	0.0821	0.0067
0.0015	0.0183	0.0821	0.1353	0.0821	0.0183	0.0015
0.0001	0.0015	0.0067	0.0111	0.0067	0.0015	0.0001

Lọc Gauss

$\frac{1}{273}$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

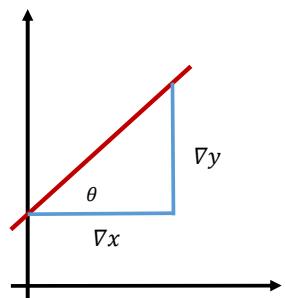
Lọc Gauss



45

46

Biểu diễn hàm



Hàm tuyến tính

$$f(x) = mx + b \quad m = \Delta y / \Delta x$$

Có thể được viết lại

$$f(x) = \tan(\theta) + b$$

Mô tả trực quan của các hàm tuyến tính

- Góc
- Dịch chuyển

Làm thế nào để biểu diễn hàm tổng quát?

Phân tích cơ sở

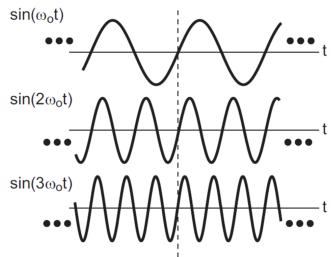
- Viết một hàm như là tổng có trọng số của các hàm cơ sở

$$f(x) = \sum w_i B_i(x)$$

- Tập hàm cơ sở tốt là gì?
- Xác định trọng số như thế nào?

Hàm sóng Sin

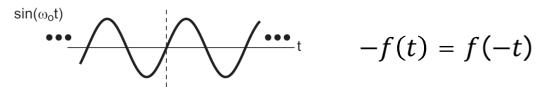
- Sử dụng hàm sóng Sin của các tần số khác nhau là các hàm cơ sở?



6

Giới hạn của hàm Sin

- Hàm Sines là hàm lẻ/ bất đối xứng:



$$-f(t) = f(-t)$$

- Hàm Sine không thể tạo các hàm chẵn:



$$f(t) = f(-t)$$

7

Giới hạn của hàm Cosines

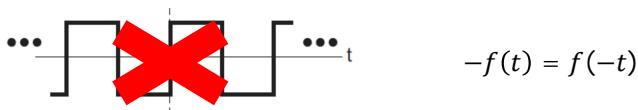
- Cosines là hàm chẵn/ hàm đối xứng:



6

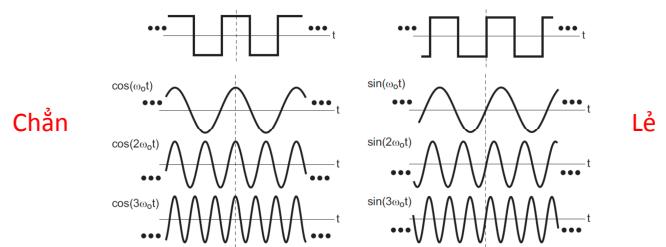
Kết hợp hàm Cosines và Sines

- Hàm cơ bản Cosine không thể tạo các hàm lẻ:



8

- Cho phép tạo cả hàm chẵn và lẻ với nhiều tổ hợp khác nhau:



9

Tại sao dùng Sines và Cosines?

- Biểu diễn hàm bởi tổ hợp của các hàm cơ bản theo các tần số khác nhau
- Mô tả trực quan của tín hiệu/ ảnh
 - Bao nhiêu nội dung tần số cao?
 - Nội dung tần số thấp như thế nào?
- “Ngôn ngữ” của xử lý ảnh:
 - Khử nhiễu bằng cách loại bỏ nội dung tần số cao
 - Giải thích hiện tượng lấy mẫu hoặc nhận thức

Hàm Euler

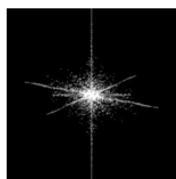
- Phép tính

$$e^{jx} = \cos(x) + j\sin(x)$$

- j là phần ảo của một số phức

10

11



Lý Thuyết Phép Biến Đổi Fourier

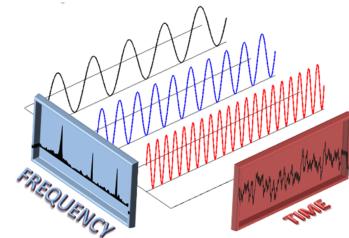
Phép biến đổi trong miền tần số

12

13

Phép biến đổi Fourier

- Phép biến đổi** là công cụ tách một dạng sóng (hàm hoặc tín hiệu) thành một biểu diễn thay thế được đặc trưng bởi hàm sines và cosines.



Phép biến đổi Fourier

- Hầu như mọi thứ trên thế giới đều có thể được mô tả thông qua dạng sóng - một hàm của thời gian, không gian hoặc một số biến khác
 - sóng âm, trường điện từ, độ cao của một ngọn đồi so với vị trí, giá cổ phiếu theo thời gian



- Phép biến đổi Fourier cho thấy rằng bất kỳ dạng sóng nào cũng có thể được viết lại dưới dạng tổng của các hàm hình sin

Phép biến đổi Fourier

- Joseph Fourier (1768 – 1830), nhà toán học và vật lý học người Pháp
- La Théorie Analytique de la Chaleur (The Analytic Theory of Heat)*, 1822
 - Ứng dụng chuỗi Fourier cho bài toán truyền nhiệt và chấn động



14

https://en.wikipedia.org/wiki/Joseph_Fourier

15

Chuỗi Fourier

- Biểu diễn **hàm tuần hoàn $f(t)$** bằng tổng các hàm sinusoidal có tần số và hệ số khác nhau

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{j \frac{2\pi n}{T} t}$$

- $c_n = \frac{1}{T} \int_0^T f(t) e^{-j \frac{2\pi n}{T} t} dt$ với $n = 0, \pm 1, \pm 2, \dots$
- T : chu kỳ của hàm f
- Công thức Euler: $e^{j\theta} = \cos\theta + j\sin\theta$, $j = \sqrt{-1}$

Chuỗi Fourier



16

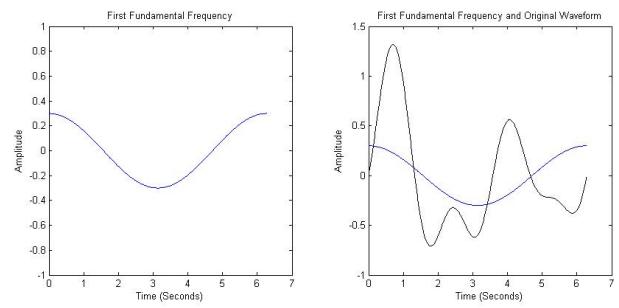
17

Chuỗi Fourier

- Mở rộng chuỗi Fourier cho **hàm $f(t)$ không tuần hoàn và có diện tích dưới đường cong xác định**

$$\mathcal{F}(\mu) = \int_{-\infty}^{\infty} f(t) e^{-j 2\pi \mu t} dt$$

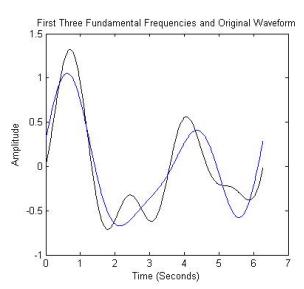
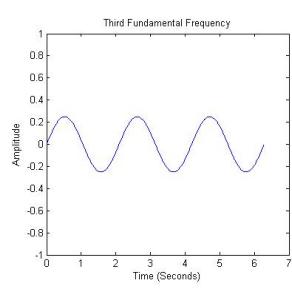
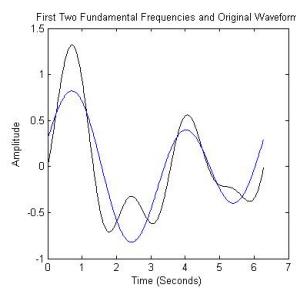
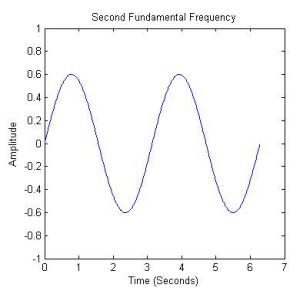
$$f(t) = \int_{-\infty}^{\infty} \mathcal{F}(\mu) e^{j 2\pi \mu t} d\mu$$



Thành phần đầu tiên là một sóng sinusoidal
có chu kỳ $T_1 = 2\pi$ và độ lớn 0.3

18

19

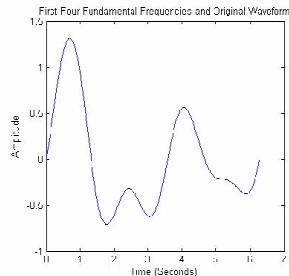
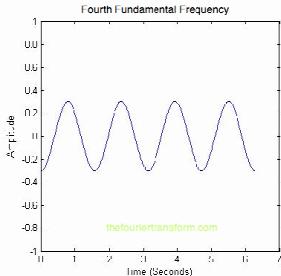


Thành phần thứ hai có chu kỳ $T_2 = T_1/2$

20

Thành phần thứ ba có chu kỳ $T_3 = T_1/3$

21



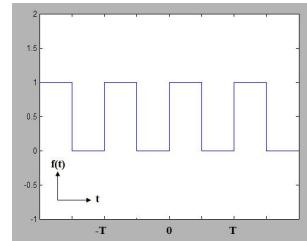
Thành phần thứ tư có chu kỳ $T_4 = T_1/4$

22

Hàm tuần hoàn

- Hàm f là tuần hoàn với chu kỳ cơ bản T nếu

$$f(t + T) = f(t), \quad \forall t$$



23

Chuỗi Fourier – Hệ số thực

- Chuỗi Fourier với chu kỳ T

$$\begin{aligned} g(t) &= a_0 + \sum_{m=1}^{\infty} a_m \cos\left(\frac{2\pi m t}{T}\right) + \sum_{n=1}^{\infty} b_n \sin\left(\frac{2\pi n t}{T}\right) \\ &= \sum_{m=0}^{\infty} a_m \cos\left(\frac{2\pi m t}{T}\right) + \sum_{n=1}^{\infty} b_n \sin\left(\frac{2\pi n t}{T}\right) \end{aligned}$$

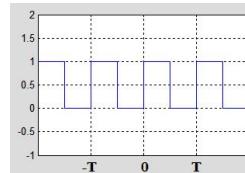
- a_m, b_n : hệ số của chuỗi Fourier

24

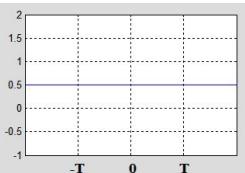
Chuỗi Fourier – Hệ số thực

- Ước lượng với 1 số hạng: $g(t) = a_0$

- $a_0 = \frac{1}{T} \int_0^T f(t) dt$: giá trị trung bình của hàm f



Dạng sóng vuông

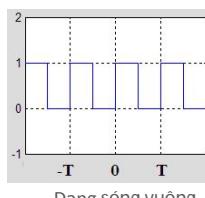


Mở rộng 1 số hạng (hằng)

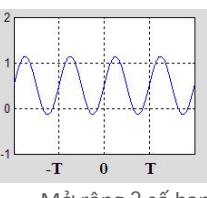
25

Chuỗi Fourier – Hệ số thực

- Ước lượng với 2 số hạng: $g(t) = a_0 + b_1 \sin\left(\frac{2\pi t}{T}\right)$
- $b_1 = \frac{2}{T} \int_0^T f(t) \sin(2\pi t/T) dt \quad (b_1 = \frac{2}{\pi})$



Dạng sóng vuông



Mở rộng 2 số hạng

26

Chuỗi Fourier – Hệ số thực

- Giá trị tối ưu cho a_m và b_n

- $a_0 = \frac{1}{T} \int_0^T f(t) dt$

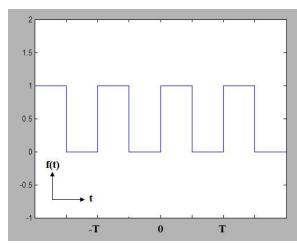
- $a_m = \frac{2}{T} \int_0^T f(t) \cos(2\pi m t/T) dt$

- $b_n = \frac{2}{T} \int_0^T f(t) \sin(2\pi n t/T) dt$

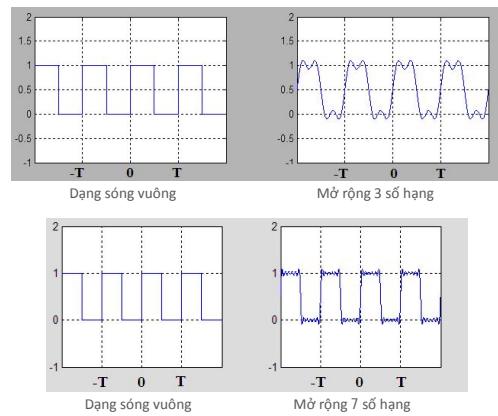
27

Chuỗi Fourier – Hệ số thực

- $a_0 = \frac{1}{2}$
- $a_m = 0, m = 1, 2, \dots$
- $b_n = \begin{cases} \frac{2}{\pi n}, & n lẻ \\ 0, & n chẵn \end{cases}$



28



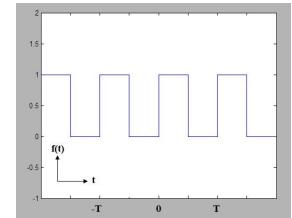
29

Chuỗi Fourier – Hệ số phức

- Chuỗi Fourier với chu kỳ T

$$g(t) = \sum_{n=-\infty}^{\infty} c_n e^{j \frac{2\pi n}{T} t}$$

- $c_n = \frac{1}{T} \int_0^T f(t) e^{-j \frac{2\pi n}{T} t} dt$ với $n = 0, \pm 1, \pm 2, \dots$
- Công thức Euler: $e^{j\theta} = \cos\theta + j\sin\theta, j = \sqrt{-1}$



30

31

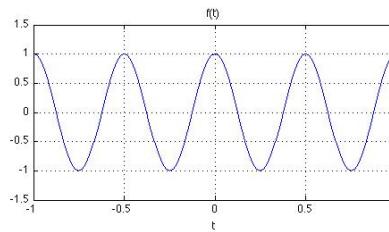
Chuỗi Fourier – Hệ số phức

- Mặc dù hệ số Fourier là số phức, hàm hợp $g(t)$ sẽ hoàn toàn là số thực nếu

$$c_n^* = c_{-n}$$

- Dấu * biểu diễn thành phần hợp phức (complex conjugate)

Chuỗi Fourier – Hệ số phức



$$f(t) = \cos(4\pi t)$$

$$T = 0.5$$

32

33

$$c_n = \frac{1}{T} \int_0^T f(t) e^{-j\frac{2\pi n t}{T}} dt = 2 \int_0^{0.5} \cos(4\pi t) e^{j4\pi n t} dt$$

Hàm cosine được viết lại theo công thức Euler: $\cos(\theta) = \frac{e^{j\theta} + e^{-j\theta}}{2}$

$$c_n = \int_0^{0.5} e^{j4\pi t(1-n)} dt + \int_0^{0.5} e^{-j4\pi t(n+1)} dt$$

Xét vế trái:

$$\int_0^{0.5} e^{j4\pi t(1-n)} dt = \frac{e^{j4\pi t(1-n)}}{j4\pi(1-n)} \Big|_0^{0.5} = \frac{e^{j2\pi(1-n)} - 1}{j4\pi(1-n)} = 0, \quad n \neq 1$$

$$c_1 = \int_0^{0.5} e^{j4\pi t(1-n)} dt = \int_0^{0.5} 1 dt = 0.5$$

Phân tích tương tự cho vế phải: $\int_0^{0.5} e^{-j4\pi t(n+1)} dt = 0, \quad n \neq -1$

$$c_{-1} = 0.5$$

Mọi hệ số bằng 0 trừ trường hợp $n = 1$ và $n = -1$. Như vậy, chuỗi Fourier là

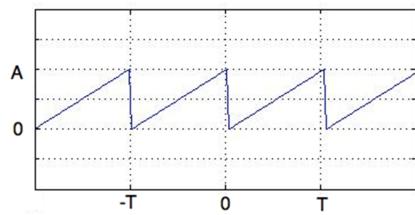
$$\begin{aligned} g(t) &= \sum_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi n t}{T}} = c_1 e^{j\frac{2\pi t}{T}} + c_{-1} e^{-j\frac{2\pi t}{T}} \\ &= 0.5(e^{j4\pi t} + e^{-j4\pi t}) \\ &= \cos(4\pi t) \\ &= f(t) \end{aligned}$$

Chuỗi Fourier $g(t)$ thể hiện chính xác hàm cần biểu diễn $f(t)$

34

35

Chuỗi Fourier – Hệ số phức



$$f(t) = A \frac{t}{T}, \quad 0 \leq t \leq T$$

$$\begin{aligned} c_n &= \frac{1}{T} \int_0^T f(t) e^{-j\frac{2\pi n t}{T}} dt = \frac{A}{T^2} \int_0^T t e^{-j\frac{2\pi n t}{T}} dt \\ &= \frac{A}{T^2} \left[\frac{tT}{-j2\pi n} e^{-j\frac{2\pi n t}{T}} + \frac{T^2}{(2\pi n)^2} e^{-j\frac{2\pi n t}{T}} \right]_0^T \end{aligned}$$

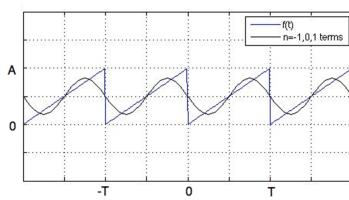
$$\text{Do } e^{-j2\pi n} = 1 \text{ nên } c_n = \frac{A}{T^2} \left[\frac{T^2}{-j2\pi n} \right] = \frac{jA}{2\pi n}, \quad n \neq 0$$

$$c_0 = \frac{A}{2}$$

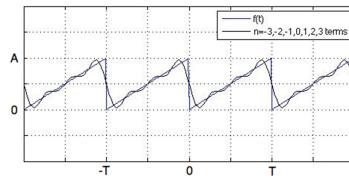
Thể các hệ số c_n vào công thức chuỗi Fourier: $g(t) = \sum_{n=-\infty}^{\infty} c_n e^{j\frac{2\pi n t}{T}}$

36

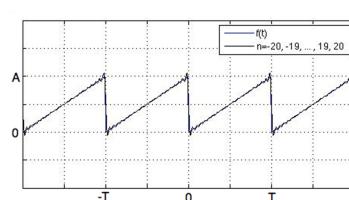
37



Hàm số với 3 hệ số Fourier
 $n = -1, 0, 1$ terms



Hàm số với 5 hệ số Fourier
 $n = -2, -1, 0, 1, 2$ terms

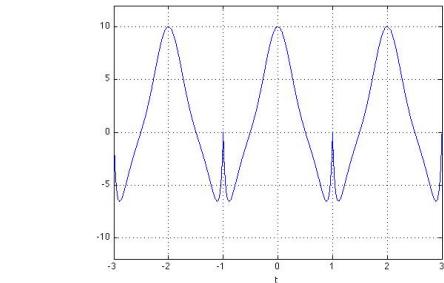


Hàm số với 20 hệ số Fourier
 $n = -20, -19, \dots, 19, 20$

38

39

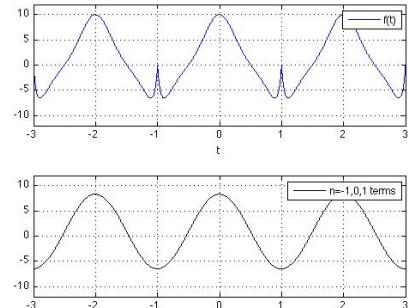
Chuỗi Fourier cho hàm phức



$$f(t) = \frac{(t-1)(t+1)(2t)^4}{\cos(1.4t)} + 10e^{-9t^2}, \quad -1 \leq t \leq 1$$

40

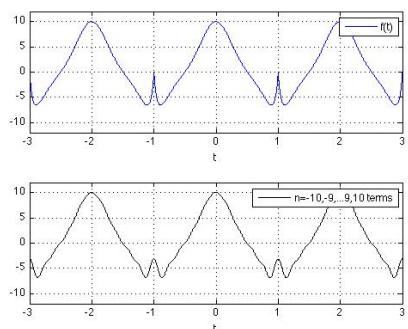
Chuỗi Fourier cho hàm phức



Hàm gốc (trên) và chuỗi Fourier với 3 hệ số (dưới)

41

Chuỗi Fourier cho hàm phức



Hàm gốc (trên) và chuỗi Fourier với 21 hệ số (dưới)

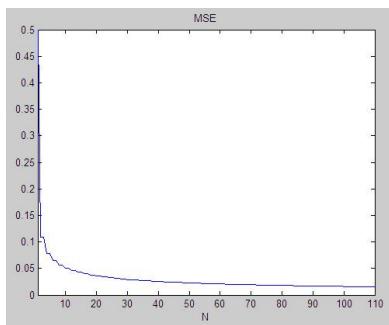
42

Mean Squared Error (MSE)

- Gọi $g_N(t) = \sum_{n=-N}^N c_n e^{j\frac{2\pi n t}{T}}$ là chuỗi Fourier có $2N + 1$ hệ số
 - $mse(N) = \|f(t) - g_N(t)\|$
- $$= \sqrt{\int_0^T |f(t) - g_N(t)|^2 dt}$$
- $$= \sqrt{\int_0^T \left| f(t) - \sum_{n=-N}^N c_n e^{j\frac{2\pi n t}{T}} \right|^2 dt}$$

43

Mean Squared Error (MSE)



44

Biến đổi Fourier liên tục

45

Biến đổi Fourier thuận

- Biến đổi Fourier thuận của hàm $g(t)$

$$\mathcal{F}\{g(t)\} = G(\mu) = \int_{-\infty}^{\infty} g(t) e^{-j2\pi\mu t} dt$$

- $G(\mu)$ cho biết năng lượng mà $g(t)$ chứa tại tần số μ , được gọi là **phổ của g**

Biến đổi Fourier nghịch

- Biến đổi Fourier nghịch phục hồi g từ G

$$\mathcal{F}^{-1}\{G(\mu)\} = \int_{-\infty}^{\infty} G(\mu) e^{j2\pi\mu t} d\mu = g(t)$$

- $g(t)$ và $G(f)$ tạo thành cặp biến đổi Fourier

$$g \xrightarrow{\mathcal{F}} G$$

46

47

Tính chất của biến đổi Fourier

- Tính tuyến tính (linearity)

$$\mathcal{F}\{c_1 g(t) + c_2 h(t)\} = c_1 G(\mu) + c_2 H(\mu)$$

- c_1, c_2 : hằng số bất kỳ (số thực hoặc số phức)

- Tính dịch chuyển (shift property)

Nếu dịch chuyển $g(t)$ theo thời gian một lượng hằng số a , hàm sẽ có cùng độ lớn phổ, $G(\mu)$

$$\mathcal{F}\{g(t-a)\} = e^{-j2\pi\mu a} G(\mu)$$

48

49

Tính chất của biến đổi Fourier

- Tính tỷ lệ (scaling property)

Nếu lấy tỷ lệ $g(t)$ theo thời gian bằng một hằng số c khác 0, ký hiệu $g(ct)$, biến đổi Fourier sẽ là

$$\mathcal{F}\{g(ct)\} = \frac{G\left(\frac{\mu}{c}\right)}{|c|}$$

- Tính đạo hàm (derivative property):

$$\mathcal{F}\left\{\frac{dg(t)}{dt}\right\} = j2\pi\mu \cdot G(\mu)$$

Tính chất của biến đổi Fourier

- Tính chất tích chập (convolution)

Gọi tích chập của $g(t)$ và $h(t)$ theo thời gian là

$$g(t) \star h(t) = \int_{-\infty}^{\infty} g(\tau) h(t - \tau) d\tau$$

Biến đổi Fourier của tích chập của $g(t)$ và $h(t)$ là

$$\mathcal{F}\{g(t) \star h(t)\} = G(\mu)H(\mu)$$

50

51

Tính chất của biến đổi Fourier

- Tính điều biến (modulation property)

Một hàm được gọi là "điều biến" bởi hàm khác nếu nhân chúng với nhau theo thời gian

$$\mathcal{F}\{g(t) h(t)\} = G(\mu) \star H(\mu)$$

- Tính đối ngẫu (Duality)

$$\mathcal{F}\{G(t)\} = g(-\mu)$$

Tính chất của biến đổi Fourier

- Định lý Parseval

$$\int_{-\infty}^{\infty} |g(t)|^2 dt = \int_{-\infty}^{\infty} |G(\mu)|^2 d\mu$$

- Tích phân của bình phương độ lớn của hàm là năng lượng của hàm
- Năng lượng của $g(t)$ bằng năng lượng chứa trong $G(\mu)$

52

53

Phép biến đổi Fourier rời rạc

Biến đổi Fourier rời rạc (DFT)

- Đây là biến đổi Fourier cho tín hiệu chỉ biết tại những điểm lấy mẫu cách nhau chu kỳ T
- Gọi $g(t)$ là tín hiệu gốc ở dạng liên tục. Biến đổi Fourier của $g(t)$ là

$$\mathcal{F}\{g(t)\} = G(\mu) = \int_{-\infty}^{\infty} g(t) e^{-j2\pi\mu t} dt$$

54

55

Biến đổi Fourier rời rạc

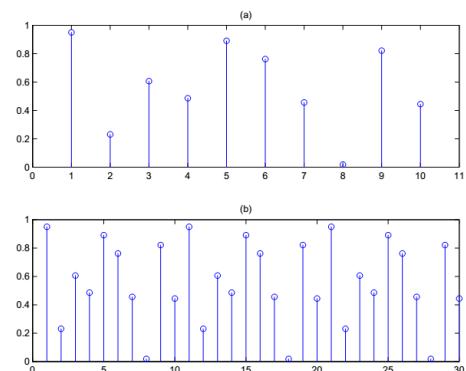
- Lấy N mẫu trên $g(t)$: $g_0, g_1, g_2, \dots, g_k, \dots, g_{N-1}$
 - Mỗi mẫu g_k là một xung có diện tích g_k
 - $G\{\mu\} = \int_0^{(N-1)T} g(t) e^{-j\omega t} dt$, với $\omega = 2\pi\mu$
- $$\begin{aligned} &= g_0 e^{-j0} + g_1 e^{-j\omega T} + \dots + g_k e^{-j\omega kT} + \dots \\ &\quad + g_{N-1} e^{-j\omega(N-1)T} \\ &= \sum_0^{N-1} g_k e^{-j\omega kT} \end{aligned}$$

Biến đổi Fourier rời rạc

- Do số lượng điểm dữ liệu đầu vào hữu hạn, DFT xem như dữ liệu có tính tuần hoàn
- Tức là g_N đến g_{2N-1} cũng giống như g_0 đến g_{N-1}

56

57



(a) Chuỗi gồm $N = 10$ mẫu. (b) Tính tuần hoàn ngầm định trong DFT

Biến đổi Fourier rời rạc

- Phương trình DFT được đánh giá tại tần số cơ bản và điểm điều hòa ($\omega = 0$)

$$\omega = 0, \frac{2\pi}{NT}, \frac{2\pi}{NT} \times 2, \dots, \frac{2\pi}{NT} \times n, \dots, \frac{2\pi}{NT} \times (N-1)$$

- Như vậy, DFT \mathcal{F}_n của chuỗi g_k là

$$\mathcal{F}_n\{g_k(t)\} = \sum_{k=0}^{N-1} g_k e^{-j\frac{2\pi}{N}nk}, n = 0, \dots, N-1$$

58

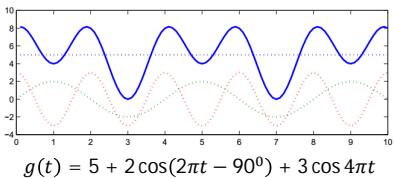
Biến đổi Fourier rời rạc

- Phương trình DFT viết theo dạng ma trận

$$\begin{pmatrix} \mathcal{F}_0 \\ \mathcal{F}_1 \\ \mathcal{F}_2 \\ \vdots \\ \mathcal{F}_{N-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & W & W^2 & W^3 & \cdots & W^{N-1} \\ 1 & W^2 & W^4 & W^6 & \cdots & W^{N-2} \\ 1 & W^3 & W^6 & W^9 & \cdots & W^{N-3} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W^{N-1} & W^{N-2} & W^{N-3} & \cdots & W \end{pmatrix} \begin{pmatrix} g_0 \\ g_1 \\ g_2 \\ \vdots \\ g_{N-1} \end{pmatrix}$$

$$W = e^{-j2\pi/N} \text{ và } W = W^{2N} \text{ etc.} = 1$$

59



$$g(t) = 5 + 2\cos(2\pi t - 90^\circ) + 3\cos 4\pi t$$

Lấy mẫu $g(t)$ 4 lần/giây từ $t = 0$ đến $t = 3/4$. Đặt $t = kTs = k/4$
Giá trị của các mẫu rời rạc là:

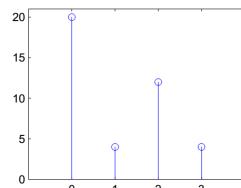
$$g_k = 5 + 2\cos\left(\frac{\pi}{2}k - 90^\circ\right) + 3\cos \pi k$$

Tức là $g_0 = 8, g_1 = 4, g_2 = 8, g_3 = 0, (N = 4)$

60

$$\text{Như vậy, } \mathcal{F}_n\{g(t)\} = \sum_0^3 g_k e^{-j\frac{\pi}{2}nk} = \sum_0^3 g_k (-j)^{nk}$$

$$\begin{pmatrix} \mathcal{F}_0 \\ \mathcal{F}_1 \\ \mathcal{F}_2 \\ \mathcal{F}_3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{pmatrix} \begin{pmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \end{pmatrix} = \begin{pmatrix} 20 \\ -i4 \\ 12 \\ i4 \end{pmatrix}$$



61

Biến đổi Fourier rời rạc

- Biến đổi DFT nghịch của $\mathcal{F}_n = \sum_{k=0}^{N-1} g_k e^{-j\frac{2\pi}{N}nk}$

$$\text{là } g_k = \frac{1}{N} \sum_{n=0}^{N-1} \mathcal{F}_n e^{j\frac{2\pi}{N}nk}$$

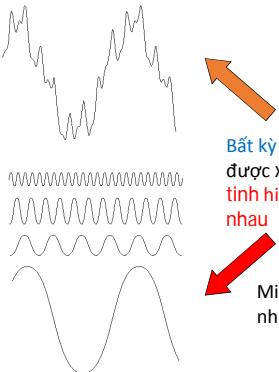
- Ma trận nghịch bằng $1/N$ lần phần bù phức của ma trận gốc (đối xứng)

Liên hệ phép biến đổi Fourier và
Ảnh Số

62

63

Chuỗi Fourier

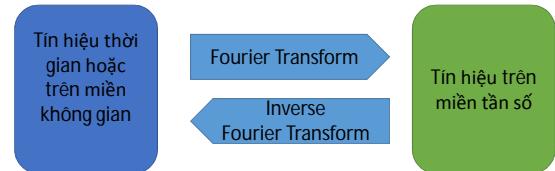


Bất kỳ một tín hiệu tuần hoàn có thể được xem như là **tổng trọng số** của các tín hiệu sinusoидal với các tần số khác nhau

Miền tần số: xem tần số như là một biến độc lập

Biến đổi Fourier

- Tín hiệu thời gian hoặc trên miền không gian được phân tích thành tần số và ngược lại



64

65

Biến đổi Fourier 1 chiều

- Trường hợp liên tục

$$\mathcal{F}(\mu) = \int_{-\infty}^{\infty} f(t) e^{-j2\pi\mu t} dt \quad f(t) = \int_{-\infty}^{\infty} \mathcal{F}(\mu) e^{j2\pi\mu x} d\mu$$

- Trường hợp rời rạc

$$\begin{aligned} \mathcal{F}_m &= \sum_{n=0}^{M-1} f_n e^{-j2\pi mn/M}, & m &= 0, 1, 2, \dots, M-1 \\ f_n &= \frac{1}{M} \sum_{m=0}^{M-1} \mathcal{F}_m e^{j2\pi mn/M}, & n &= 0, 1, 2, \dots, M-1 \end{aligned}$$

66

67

Biến đổi Fourier 1 chiều

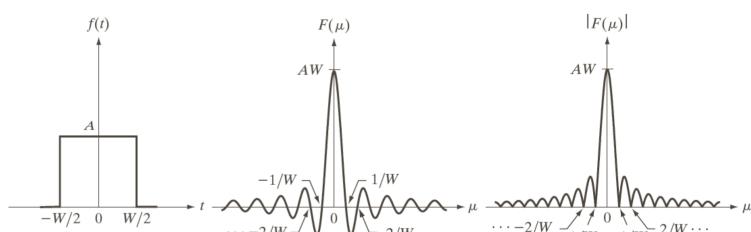
- $\mathcal{F}(\mu)$ có thể biểu diễn thành

$$c = R(\mu) + jI(\mu)$$

hoặc $\mathcal{F}(\mu) = |F(\mu)|e^{-j\phi(\mu)}$

- Độ lớn (Magnitude) (Fourier spectrum) $|F(\mu)| = \sqrt{R(\mu)^2 + I(\mu)^2}$
- Góc lệch pha $\phi(\mu) = \arctan\left(\frac{I(\mu)}{R(\mu)}\right)$

- a b c
 a. Hàm hộp có chiều rộng W
 b. Biến đổi Fourier của a.
 c. Phổ Fourier của a.



68

69

Quan hệ giữa Δt và $\Delta\mu$

- Cho $f(t)$ với M điểm lấy mẫu

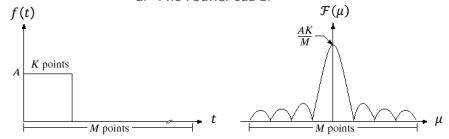
- Δt : độ phân giải không gian, là khoảng cách giữa các mẫu trong $f(t)$
- $\Delta\mu$: độ phân giải tần số, là khoảng cách giữa các thành phần tần số trong $\mathcal{F}(\mu)$

$$\bullet \text{Quan hệ giữa } \Delta t \text{ và } \Delta\mu \text{ là } \Delta\mu = \frac{1}{M\Delta t}$$

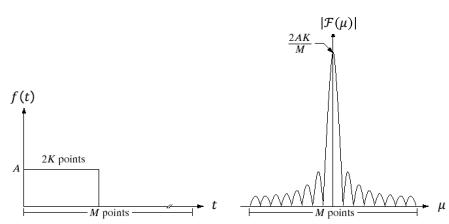
Quan hệ giữa Δt và $\Delta\mu$: Ví dụ

- Tín hiệu $f(t)$ có chu kỳ lấy mẫu là 0.5 giây, 100 điểm
- Độ phân giải tần số $\Delta\mu = \frac{1}{M\Delta t} = \frac{1}{0.5 \times 100} = 0.02 \text{ Hz}$
- Tức là trong $\mathcal{F}(\mu)$, ta có thể phân biệt hai tần số cách nhau 0.02 Hertz trở lên

- a. Hàm rời rạc M điểm (K điểm khác 0)
 b. Phổ Fourier của a.
 c. Hàm rời rạc có gấp đôi số điểm khác 0
 d. Phổ Fourier của b.



70



71

Biến đổi Fourier rời rạc 2 chiều

- Cho ảnh $f(x, y)$ có kích thước $M \times N$ pixels
- Biến đổi thuận

$$\mathcal{F}(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(\frac{ux}{M} + \frac{vy}{N})}$$

$$u = 0, 1, 2, \dots, M-1, \quad v = 0, 1, 2, \dots, N-1$$

Biến đổi Fourier rời rạc 2 chiều

- Cho ảnh $f(x, y)$ có kích thước $M \times N$ pixels
- Biến đổi nghịch

$$f(x, y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \mathcal{F}(u, v) e^{j2\pi(\frac{ux}{M} + \frac{vy}{N})}$$

$$x = 0, 1, 2, \dots, M-1, \quad y = 0, 1, 2, \dots, N-1$$

72

73

Biến đổi Fourier rời rạc 2 chiều

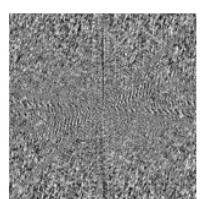
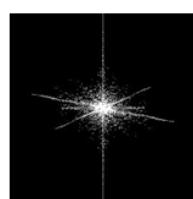
- $\mathcal{F}(u, v)$ có thể biểu diễn thành

$$\mathcal{F}(u, v) = R(u, v) + jI(u, v)$$

$$\text{hoặc } \mathcal{F}(u, v) = |\mathcal{F}(u, v)| e^{-j\phi(u, v)}$$

- Độ lớn (Magnitude) $|\mathcal{F}(u, v)| = \sqrt{R(u, v)^2 + I(u, v)^2}$
- Góc lệch pha (Phase angle) $\phi(u, v) = \arctan\left(\frac{I(u, v)}{R(u, v)}\right)$

- a. Ảnh gốc
 b. Phổ Fourier của ảnh a.
 c. Phase angle của ảnh a.

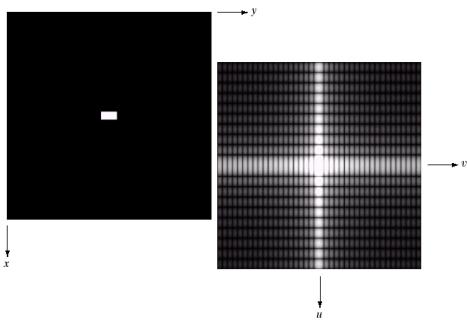


74

75

a

- a. Ảnh gồm hình chữ nhật trắng 20×40 trên nền đen 512×512
 b. Phó Fourier sau khi đã dịch chuyển và áp dụng log transformation



76

Property	Expression(s)
Fourier transform	$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi(ux/M + vy/N)}$
Inverse Fourier transform	$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{j2\pi(ux/M + vy/N)}$
Polar representation	$F(u, v) = F(u, v) e^{j\phi(u, v)}$
Spectrum	$ F(u, v) = [R^2(u, v) + I^2(u, v)]^{1/2}, \quad R = \text{Real}(F) \text{ and } I = \text{Imag}(F)$
Phase angle	$\phi(u, v) = \tan^{-1} \left[\frac{I(u, v)}{R(u, v)} \right]$
Power spectrum	$P(u, v) = F(u, v) ^2$
Average value	$\bar{f}(x, y) = F(0, 0) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)$
Translation	$f(x, y) e^{j2\pi(u_0x/M + v_0y/N)} \Leftrightarrow F(u - u_0, v - v_0)$ $f(x - x_0, y - y_0) \Leftrightarrow F(u, v) e^{-j2\pi(u_0x/M + v_0y/N)}$ When $x_0 = u_0 = M/2$ and $y_0 = v_0 = N/2$, then $f(x, y)(-1)^{x+y} \Leftrightarrow F(u - M/2, v - N/2)$ $f(x - M/2, y - N/2) \Leftrightarrow F(u, v)(-1)^{x+y}$

77

Conjugate symmetry	$F(u, v) = F^*(-u, -v)$ $ F(u, v) = F(-u, -v) $
Differentiation	$\frac{\partial f(x, y)}{\partial x^n} \Leftrightarrow (ju)^n F(u, v)$ $(-jx)^n f(x, y) \Leftrightarrow \frac{\partial^n F(u, v)}{\partial u^n}$
Laplacian	$\nabla^2 f(x, y) \Leftrightarrow -(u^2 + v^2) F(u, v)$
Distributivity	$\Im[f_1(x, y) + f_2(x, y)] = \Im[f_1(x, y)] + \Im[f_2(x, y)]$ $\Im[f_1(x, y) \cdot f_2(x, y)] \neq \Im[f_1(x, y)] \cdot \Im[f_2(x, y)]$
Scaling	$af(x, y) \Leftrightarrow aF(u, v), f(ax, by) \Leftrightarrow \frac{1}{ ab } F(u/a, v/b)$
Rotation	$x = r \cos \theta \quad y = r \sin \theta \quad u = \omega \cos \varphi \quad v = \omega \sin \varphi$ $f(r, \theta + \theta_0) \Leftrightarrow F(\omega, \varphi + \theta_0)$
Periodicity	$F(u, v) = F(u + M, v) = F(u, v + N) = F(u + M, v + N)$ $f(x, y) = f(x + M, y) = f(x, y + N) = f(x + M, y + N)$
Separability	See Eqs.(4.6-14) and (4.6-15). Separability implies that we can compute the 2-D transform of an image by first computing 1-D transforms along each row of the image, and then computing a 1-D transform along each column of this intermediate result. The reverse, columns and then rows, yields the same result.

78

Property	Expression(s)
Computation of the inverse Fourier transform using a forward transform algorithm	$\frac{1}{MN} f^*(x, y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F^*(u, v) e^{-j2\pi(ux/M + vy/N)}$ This equation indicates that inputting the function $F^*(u, v)$ into an algorithm designed to compute the forward transform (right side of the preceding equation) yields $f^*(x, y)/MN$. Taking the complex conjugate and multiplying this result by MN gives the desired inverse.
Convolution*	$f(x, y) * h(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) h(x - m, y - n)$
Correlation*	$f(x, y) \circ h(x, y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f^*(m, n) h(x + m, y + n)$
Convolution theorem*	$f(x, y) * h(x, y) \Leftrightarrow F(u, v) H(u, v);$ $f(x, y) h(x, y) \Leftrightarrow F(u, v) \circ H(u, v)$
Correlation theorem*	$f(x, y) \circ h(x, y) \Leftrightarrow F^*(u, v) H(u, v);$ $f^*(x, y) h(x, y) \Leftrightarrow F(u, v) \circ H(u, v)$

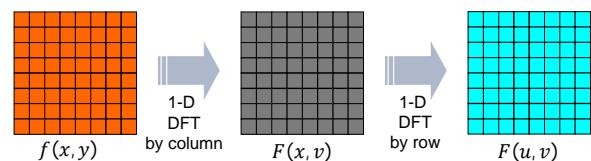
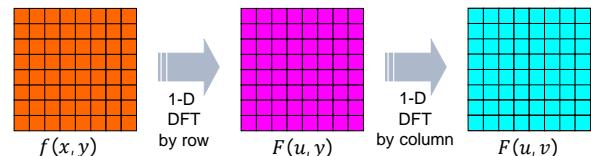
79

Some useful FT pairs:

Impulse	$\delta(x, y) \Leftrightarrow 1$
Gaussian	$A \sqrt{2\pi} \sigma e^{-2\pi^2\sigma^2(x^2 + y^2)} \Leftrightarrow A e^{-(u^2 + v^2)/2\sigma^2}$
Rectangle	$\text{rect}[a, b] \Leftrightarrow ab \frac{\sin(\pi ua)}{(\pi ua)} \frac{\sin(\pi vb)}{(\pi vb)} e^{-j\pi(ua + vb)}$
Cosine	$\cos(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow \frac{1}{2} [\delta(u + u_0, v + v_0) + \delta(u - u_0, v - v_0)]$
Sine	$\sin(2\pi u_0 x + 2\pi v_0 y) \Leftrightarrow j \frac{1}{2} [\delta(u + u_0, v + v_0) - \delta(u - u_0, v - v_0)]$

^{*} Assumes that functions have been extended by zero padding.

Thực hiện 2D-DFT với 1D-DFT

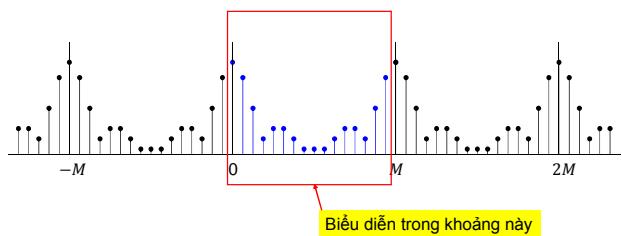


80

81

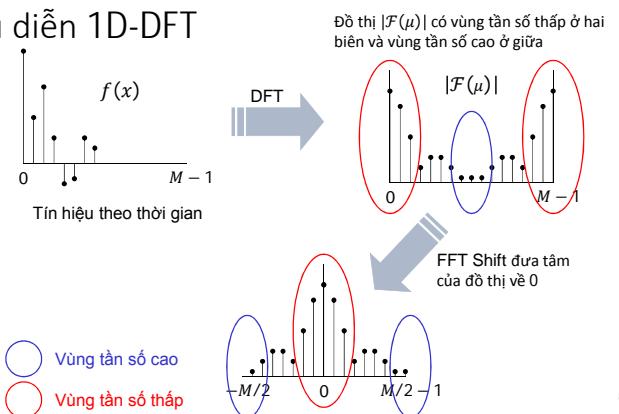
Tính tuần hoàn của 1D-DFT

- DFT lặp lại chính nó với chu kỳ M điểm nhưng ta thường chỉ biểu diễn DFT với $n = 0, \dots, M - 1$



82

Biểu diễn 1D-DFT

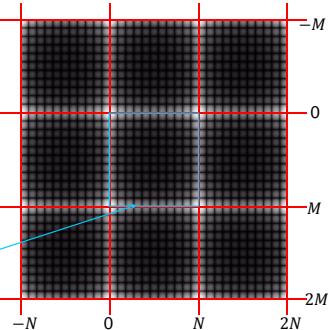


83

Tính tuần hoàn của 2D-DFT

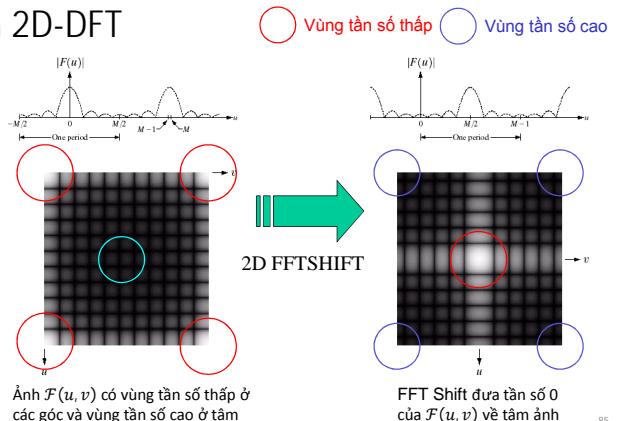
- Cho ảnh có kích thước $M \times N$ pixels
- 2D-DFT lặp lại với chu kỳ M điểm trên trục x và N điểm trên trục y

Biểu diễn trong
khoảng này



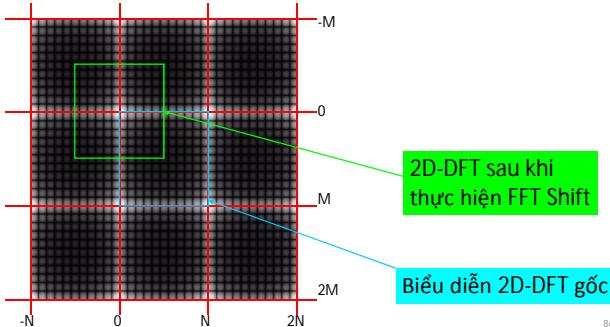
84

Biểu diễn 2D-DFT



85

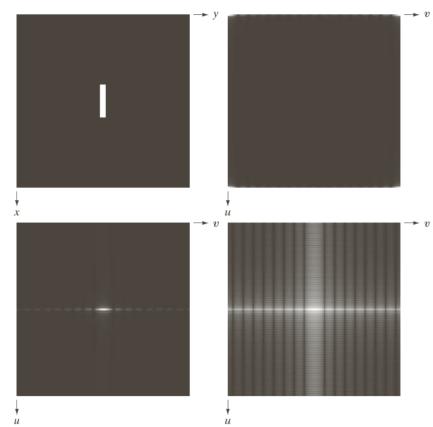
Biểu diễn 2D-DFT



86

a	b
c	d

- Ảnh gốc
- Phổ có 4 góc sáng
- Phổ đã dịch về trung tâm
- Tăng cường chi tiết trong phổ bằng log transformation



87

Tăng cường ảnh trên Miền tần số

Tính chất của miền tần số

- Tần số trong biến đổi Fourier liên hệ trực tiếp đến tỷ lệ biến đổi cường độ trong ảnh
 - Tần số tại trung tâm ($u = v = 0$) thay đổi chậm nhất, tỷ lệ với cường độ trung bình của ảnh
 - Càng đi xa tâm, tần số càng cao, tương ứng với sự thay đổi cường độ ngày càng tăng trong ảnh
 - Ví dụ: tường, sàn nhà → cạnh của đối tượng, nhiễu

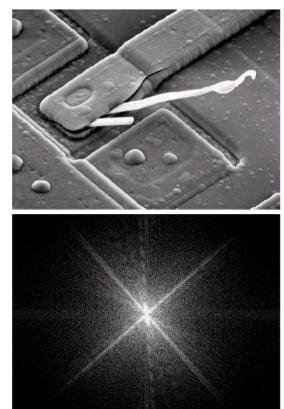
88

89

Tính chất của miền tần số

- Thành phần cơ bản của biến đổi Fourier đối với việc phân tích thị giác
 - **Magnitude (Fourier spectrum)**: cho biết tính chất tổng quan của ảnh gốc
 - **Phase angle**: thay đổi trên trường này gây biến đổi mạnh về cảm nhận thị giác

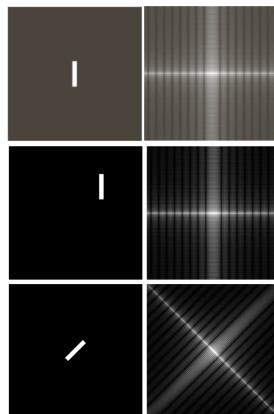
90



91

a	b
c	d
e	f

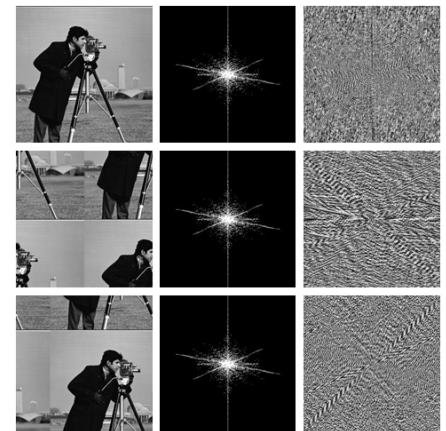
- a. Ảnh gốc
b. Phổ Fourier của a.
c. Hình chữ nhật trong a. dịch chuyển
d. Phổ Fourier của c.
e. Hình chữ nhật trong a. xoay
f. Phổ Fourier của e.



92

a	d	g
b	e	h
c	f	i

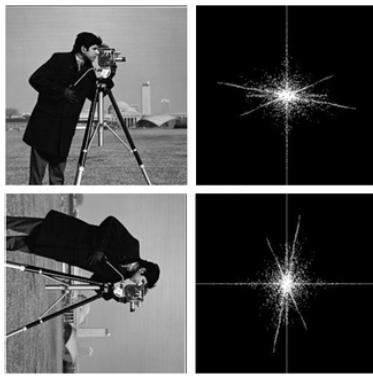
- a. Ảnh gốc
b. Ảnh a. dịch chuyển 128 dòng và 128 cột
c. Ảnh a. dịch chuyển 64 dòng và 64 cột
d – f Phổ Fourier của a – c
g – i Phase angle của a – c



93

a
b
c
d

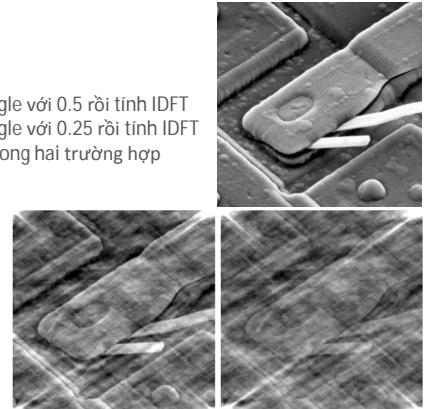
- a. Ảnh gốc
- b. Phổ Fourier của a.
- c. Ảnh gốc xoay 90°
- d. Phổ Fourier của b.



94

a
b
c

- a. Ảnh gốc
 - b. Kết quả khi nhân phase angle với 0.5 rồi tính IDFT
 - c. Kết quả khi nhân phase angle với 0.25 rồi tính IDFT
- Phổ Fourier không thay đổi trong hai trường hợp



95

Căn bản về lọc trên miền tần số

- Qui trình lọc trên miền tần số dựa trên **tích chất tích chập** (convolution) của biến đổi Fourier

$$g(x, y) = f(x, y) \star h(x, y)$$

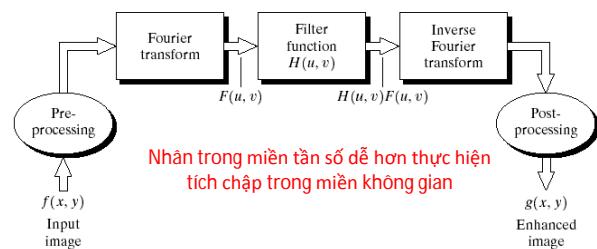
⇓

$$F(u, v)H(u, v) = G(u, v)$$

96

Căn bản về lọc trên miền tần số

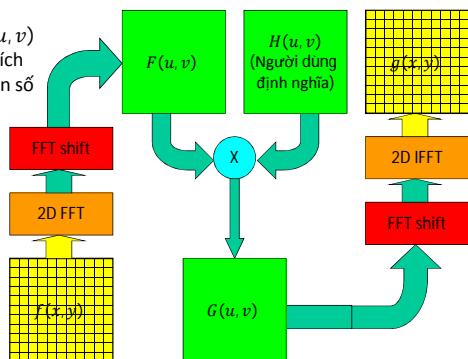
Frequency domain filtering operation



97

Lọc trên miền tần số với 2D-FFT

$F(u, v)$ và $H(u, v)$ phải có cùng kích thước và có tần số 0 tại tâm



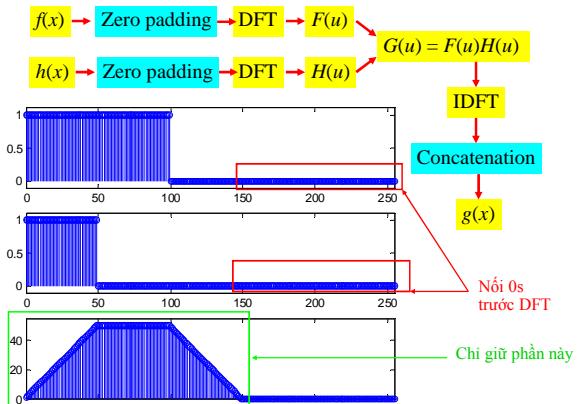
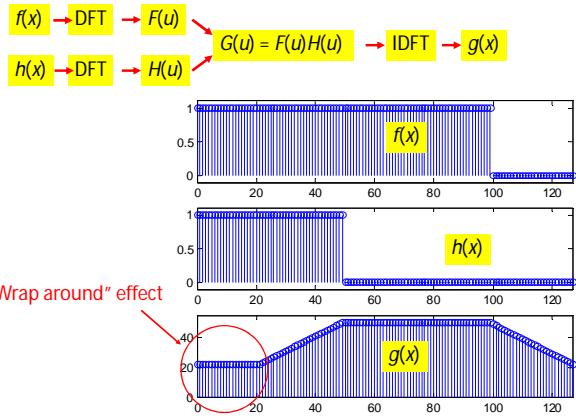
98

Phép nhân trên miền tần số

- Phép nhân hai DFT của hai tín hiệu tương đương với thực hiện **tích chập vòng**

- Khi tích chập hai hàm tuần hoàn thì phép tích chập cũng có tính tuần hoàn
- Hai chu kỳ gần nhau xen vào nhau gây ra hiện tượng **wraparound error**

99

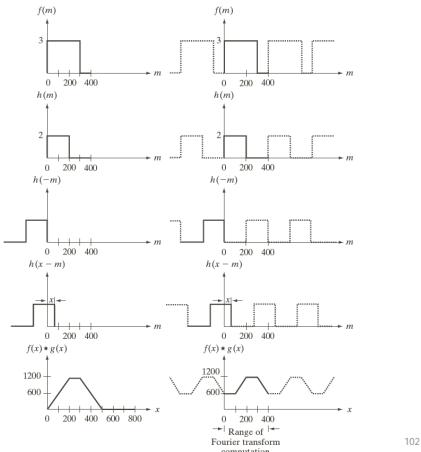


a [f]
b [g]
c [h]
d [i]
e [j]

Cột trái: tích chập của hai hàm rời rạc sử dụng cách trên miền không gian. Hình e. là kết quả đúng

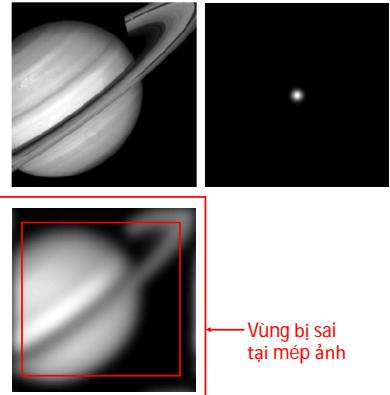
Cột phải: tích chập của hai hàm giống như ở cột trái, nhưng có xét tính tuần hoàn. Trong hình j, để ý cách dữ liệu thuộc chu kỳ liên kề nhau tạo wraparound error.

Để tạo kết quả đúng, hàm phải được nối 0s



102

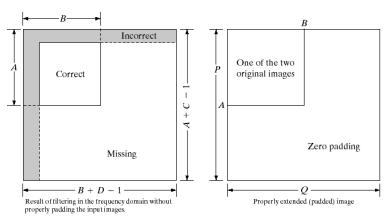
- a. Ảnh gốc
b. Bộ lọc Gaussian lowpass $H(u, v)$ với $D_0 = 5$
c. Ảnh sau khi lọc (sử dụng convolution vòng)



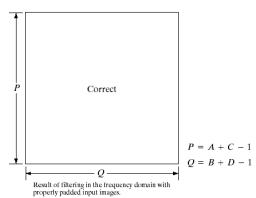
103

a [b]
c

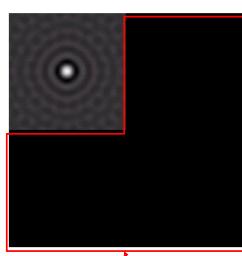
- a. Kết quả tích chập 2D không có nối 0s
b. Cách nối 0s đúng
c. Kết quả tích chập đúng



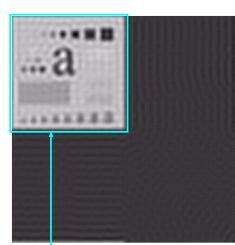
104



104



Vùng zero padding trong không gian của ảnh mặt nạ (bộ lọc ideal lowpass)



Ảnh sau khi lọc
Chi giữ phần này

105

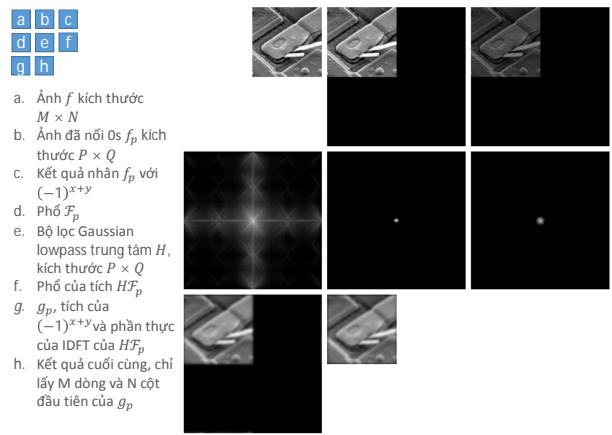
- Given an input image $f(x, y)$ of size $M \times N$, obtain the padding parameters P and Q from Eqs. (4.6-31) and (4.6-32). Typically, we select $P = 2M$ and $Q = 2N$.
- Form a padded image, $f_p(x, y)$, of size $P \times Q$ by appending the necessary number of zeros to $f(x, y)$.
- Multiply $f_p(x, y)$ by $(-1)^{x+y}$ to center its transform.
- Compute the DFT, $F(u, v)$, of the image from step 3.
- Generate a real, symmetric filter function, $H(u, v)$, of size $P \times Q$ with center at coordinates $(P/2, Q/2)$.⁷ Form the product $G(u, v) = H(u, v)F(u, v)$ using array multiplication; that is, $G(i, k) = H(i, k)F(i, k)$.
- Obtain the processed image:

$$g_p(x, y) = \{\text{real}[\Im^{-1}[G(u, v)]]\}(-1)^{x+y}$$

where the real part is selected in order to ignore parasitic complex components resulting from computational inaccuracies, and the subscript p indicates that we are dealing with padded arrays.

- Obtain the final processed result, $g(x, y)$, by extracting the $M \times N$ region from the top, left quadrant of $g_p(x, y)$.

106



107

Một vài dạng bộ lọc tiêu biểu

- Bộ lọc notch** đặt $F(0, 0) = 0$ để loại bỏ thành phần tần số zero

- Tần số zero = cường độ trung bình của ảnh

$$H(u, v) = \begin{cases} 0 & \text{if } (u, v) = (\frac{M}{2}, \frac{N}{2}) \\ 1 & \text{otherwise} \end{cases}$$

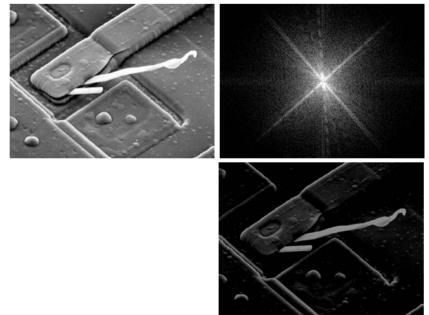
- Bộ lọc notch có thể làm việc trực tiếp với DFT không qua phép nối 0s

108

a
b
c

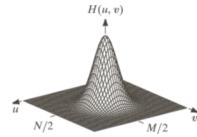
- a. Ảnh gốc
b. Phổ Fourier của a.
c. Ảnh sau khi lọc bằng bộ lọc notch

109

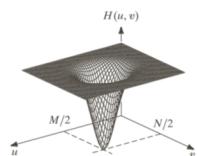


Một vài dạng bộ lọc tiêu biểu

- Bộ lọc lowpass làm yếu tần số cao** trong khi **cho qua tần số thấp**, tạo hiệu ứng ảnh nhòe



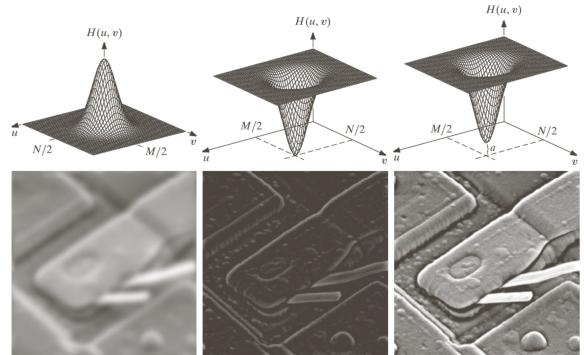
- Bộ lọc highpass làm yếu tần số thấp** trong khi **cho qua tần số cao**, tăng cường độ sắc nét cho chi tiết nhưng gây giảm độ tương phản



110

a
b
c
d
e
f

Dòng trên: các bộ lọc tần số
Dòng dưới: ảnh áp dụng các bộ lọc tương ứng
Sử dụng $\alpha = 0.85$ trong c. để thu được f.



111

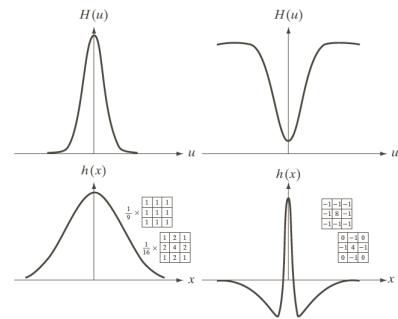
Lọc không gian và lọc tần số

- Mối liên hệ giữa lọc không gian và lọc tần số là [định lý tích chập](#)

$$h(x, y) \Leftrightarrow H(x, y)$$

- Bộ lọc tần số càng hẹp thì càng làm yếu tần số thấp, tăng hiệu ứng nhòe \Leftrightarrow Bộ lọc không gian lớn hơn

- | | |
|---|---|
| a | c |
| b | d |
- a. Bộ lọc Gaussian lowpass 1D trên miền tần số
 b. Bộ lọc lowpass trên miền không gian tương ứng với a.
 c. Bộ lọc Gaussian highpass trên miền tần số
 d. Bộ lọc highpass trên miền không gian tương ứng với b
 Mặt nạ 2D là bộ lọc trên miền không gian

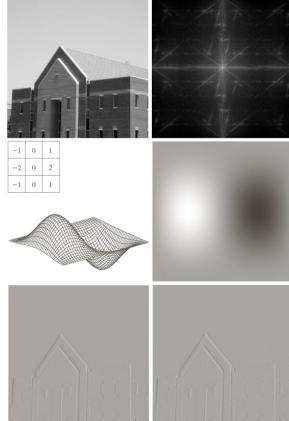


112

113

a	b
c	d
e	f

- a. Ảnh gốc
- b. Phổ Fourier của a.
- c. Mặt nạ không gian và bộ lọc tần số tương ứng
- d. Biểu diễn bộ lọc ở dạng ảnh
- e. Kết quả lọc ảnh a. trên miền tần số với bộ lọc d.
- f. Kết quả lọc ảnh a. bằng bộ lọc không gian tương ứng của d.



114

115

Bộ lọc tần số thấp (lowpass filters)

- Bộ lọc $H(u, v)$ trên miền tần số có tính chất sau
 - Là hàm rời rạc có kích thước $P \times Q$ (sau khi nối 0s)
 - $u = 0, 1, 2, \dots, P-1, v = 0, 1, 2, \dots, Q-1$
 - Tâm nằm tại **thành phần tần số zero** (zero-phase-shift) và **đối xứng xuyên tâm** (radially symmetric)
- Thao tác lọc với $H(u, v)$ tuân theo quy trình lọc trên miền tần số gồm 7 bước

Bộ lọc tần số thấp (lowpass filters)

Lọc ảnh trong miền tần số

- Given an input image $f(x, y)$ of size $M \times N$, obtain the padding parameters P and Q from Eqs. (4.6-31) and (4.6-32). Typically, we select $P = 2M$ and $Q = 2N$.
- Form a padded image, $f_p(x, y)$, of size $P \times Q$ by appending the necessary number of zeros to $f(x, y)$.
- Multiply $f_p(x, y)$ by $(-1)^{x+y}$ to center its transform.
- Compute the DFT, $F(u, v)$, of the image from step 3.
- Generate a real, symmetric filter function, $H(u, v)$, of size $P \times Q$ with center at coordinates $(P/2, Q/2)$.[†] Form the product $G(u, v) = H(u, v)F(u, v)$ using array multiplication; that is, $G(i, k) = H(i, k)F(i, k)$.
- Obtain the processed image:

$$g_p(x, y) = \{\text{real}[\mathfrak{J}^{-1}[G(u, v)]]\}(-1)^{x+y}$$

where the real part is selected in order to ignore parasitic complex components resulting from computational inaccuracies, and the subscript p indicates that we are dealing with padded arrays.

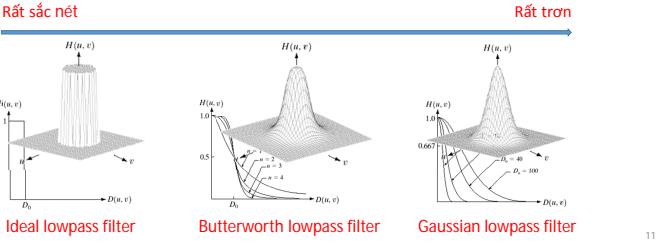
- Obtain the final processed result, $g(x, y)$, by extracting the $M \times N$ region from the top, left quadrant of $g_p(x, y)$.

116

117

Bộ lọc tần số thấp

- Tạo ứng tròn (nhòa) trên miền tần số bằng cách **làm yếu tần số cao**

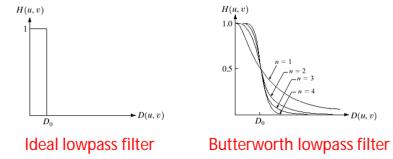


118

Cut-off frequency D_0

- Cut-off frequency D_0** là điểm đánh dấu sự thay đổi của $H(u, v)$ từ 0 đến 1 (hoặc ngược lại)

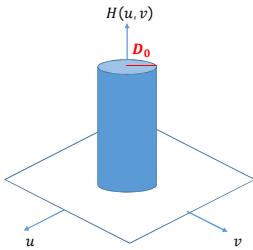
- Thay đổi có thể là đột ngột (ILPF) hoặc dần dần (BLPF và GLPF)



119

Cut-off frequency D_0

- D_0 là hằng số dương, xác định đường tròn có tâm tại thành phần tần số zero và bán kính D_0



120

121

Khoảng cách $D(u, v)$

- $D(u, v)$ là khoảng cách giữa một điểm (u, v) trong miền tần số và tâm của vùng tần số hình chữ nhật $P \times Q$

$$D(u, v) = \left[\left(u - \frac{P}{2} \right)^2 + \left(v - \frac{Q}{2} \right)^2 \right]^{1/2}$$

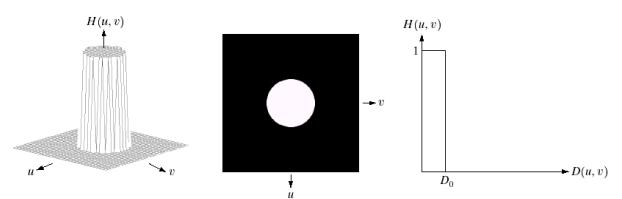
Ideal lowpass filter (ILPF)

- ILPF cho qua (không làm yếu) các tần số nằm trong đường tròn bán kính D_0 và đặt về 0 cho mọi tần số nằm ngoài đường tròn

$$H(u, v) = \begin{cases} 1 & \text{if } D(u, v) \leq D_0 \\ 0 & \text{if } D(u, v) > D_0 \end{cases}$$

- Mạch điện tử không thể nhận biết cách đặt ngưỡng đột ngột của ILPF, chỉ giả lập được trên máy tính

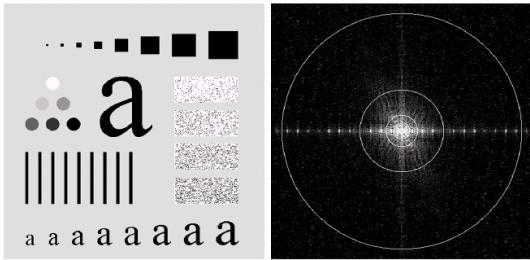
- a. Đồ thị phối cảnh của hàm chuyển ILPF
 b. Bộ lọc được biểu diễn thành ảnh
 c. Mặt cắt xuyên tâm của bộ lọc



122

123

- a | b**
- a. Ảnh có kích thước 500×500 pixels
 - b. Phổ Fourier của a.
- Các đường tròn đặt lén phổ có bán kính lần lượt là 5, 15, 30, 80, 230, chưa lần lượt là 92.0, 94.6, 96.4, 98.0 và 99.5% năng lượng ảnh



124



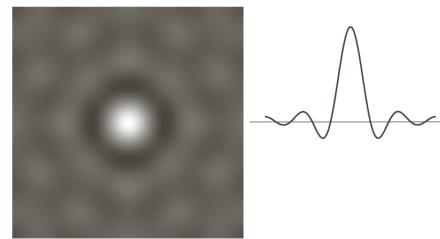
125

Hiện tượng “ringing”

- Mặt cắt của $H(u, v)$ trong miền tần số có dạng hàm hộp $\Rightarrow h(x, y)$ trong miền không gian có dạng hàm sinc
- Tích chập hàm sinc với mỗi điểm ảnh sẽ cho bản sao của sinc tại vị trí tương ứng
 - Thủy chính của sinc gây hiện ứng nhòe
 - Các thủy nhỏ hơn là nguyên nhân của “ringing”

a | b

- a. Biểu diễn trên miền không gian của một ILPF có bán kính 5 và kích thước 1000×1000
 b. Lát cắt cường độ của một đường thẳng nằm ngang đi qua tâm ảnh



126

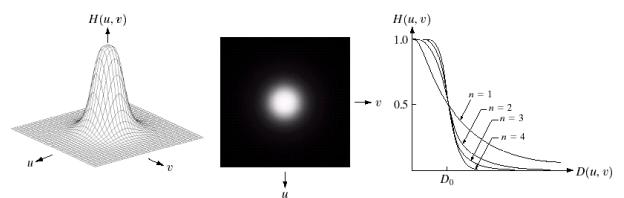
Butterworth lowpass filter (BLPF)

- Hàm chuyển của BLPF bậc n với cut-off frequency D_0

$$H(u, v) = \frac{1}{1 + [D(u, v)/D_0]^{2n}}$$

- Bậc 1: không xảy ra “ringing”
- Bậc 2: có “ringing” nhưng khó cảm nhận được
- Bậc >2: “Ringing” ngày càng nghiêm trọng hơn

- a | b | c**
- a. Đồ thị phối cảnh của hàm chuyển BLPF
 - b. Bộ lọc được biểu diễn thành ảnh
 - c. Mặt cắt xuyên tâm của bộ lọc với bậc 1 đến 4



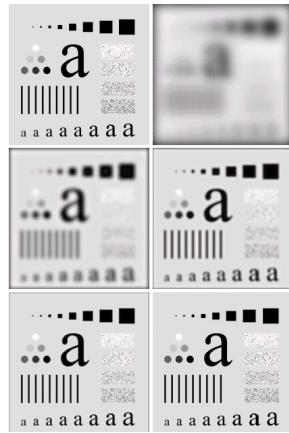
128

129

a b
c d
e f

a. Ảnh gốc

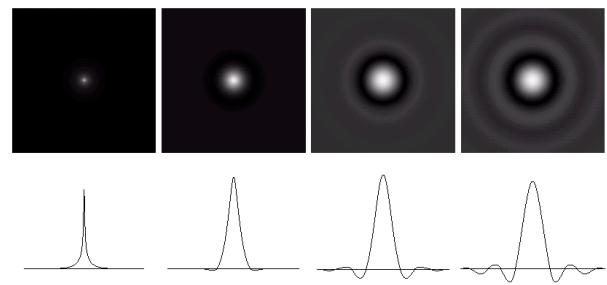
b – f Kết quả của BLPF ($n = 2$) với cut-off frequency tại bán kính 5, 15, 30, 80, và 230.



130

a b c d

a – d Biểu diễn không gian của BLPF bậc 1, 2, 5, và 20 và lát cắt cường độ đi qua tâm bộ lọc (kích thước 1000×1000 và cut-off frequency là 5)
Chú ý hiệu ứng ringing tăng dần theo bậc f



131

Gaussian lowpass filter (GLPF)

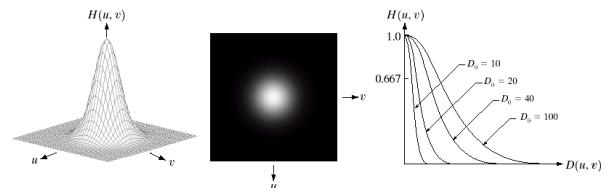
- Hàm chuyển của GLPF với cut-off frequency D_0

$$H(u, v) = e^{-\frac{D(u,v)^2}{2D_0^2}}$$

- Biến đổi Fourier nghịch của GLPF cũng là một hàm Gaussian
 \Rightarrow không xảy ra "ringing"

a b c

a. Đồ thị phối cảnh của hàm chuyển GLPF
 b. Bộ lọc được biểu diễn thành ảnh
 c. Mặt cắt xuyên tâm của bộ lọc với các D_0 khác nhau



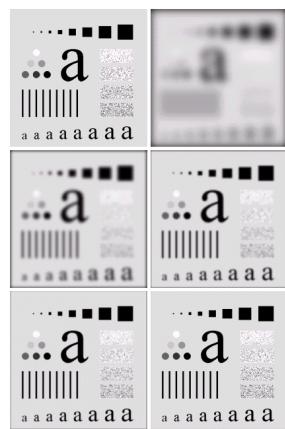
132

133

a b
c d
e f

a. Ảnh gốc

b – f Kết quả của GLPF với cut-off frequency tại bán kính 5, 15, 30, 80, và 230.



134

Tổng hợp các bộ lọc tần số thấp

Bộ lọc	Công thức
Ideal lowpass filter (ILPF)	$H(u, v) = \begin{cases} 1 & \text{if } D(u, v) \leq D_0 \\ 0 & \text{if } D(u, v) > D_0 \end{cases}$
Butterworth lowpass filter (BLPF)	$H(u, v) = \frac{1}{1 + [D(u, v)/D_0]^{2n}}$
Gaussian lowpass filter (GLPF)	$H(u, v) = e^{-D(u,v)^2/2D_0^2}$

* D_0 : cut-off frequency, n : bậc của bộ lọc Butterworth

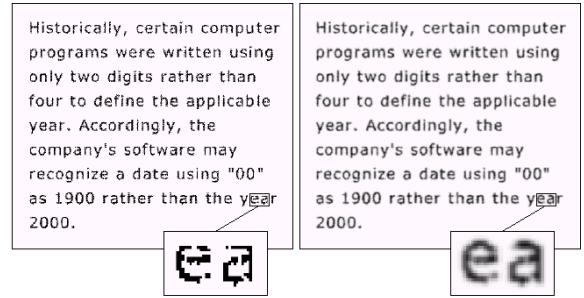
135

Ví dụ về lowpass filtering

- Lowpass filtering trên miền tần số được sử dụng trong nhiều ứng dụng thực tiễn

- Nhận thức của máy: nhận dạng chữ
- Công nghiệp in ấn và xuất bản
- Xử lý ảnh trên không và ảnh vệ tinh

- a | b**
- Văn bản có độ phân giải kém (chú ý các chữ cái bị đứt gãy trong khung phóng to)
 - Kết quả lọc bằng GLPF (các đoạn đứt gãy đã được nối liền)



136

137

- a | b | c**
- Ảnh gốc (1028×732 pixels)
 - Kết quả lọc bằng GLPF với $D_0 = 100$
 - Kết quả lọc bằng GLPF với $D_0 = 80$
 - Chú ý sự giảm các nếp nhăn trong ảnh b. và c. so với a.



138

- a | b | c**
- Ảnh gốc thấy rõ các đường quét
 - Kết quả lọc bằng GLPF với $D_0 = 50$
 - Kết quả lọc bằng GLPF với $D_0 = 20$



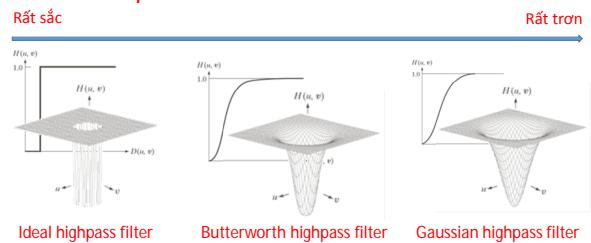
139

Bộ lọc tần số cao

Lọc ảnh trong miền tần số

Bộ lọc tần số cao

- Tạo hiệu ứng chi tiết sắc nét trên miền tần số bằng cách làm yếu tần số thấp



140

141

Liên hệ với bộ lọc tần số thấp

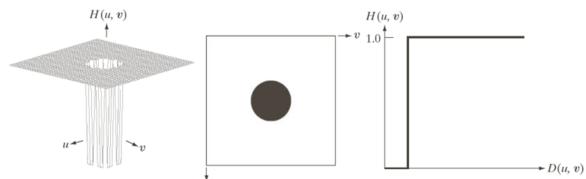
- Công thức liên hệ giữa bộ lọc tần số cao và bộ lọc tần số thấp là

$$H_{HP}(u, v) = 1 - H_{LP}(u, v)$$

• $H_{LP}(u, v)$: hàm chuyển của bộ lọc lowpass

- Những tần số nào mà bộ lọc tần số thấp làm yếu sẽ được bộ lọc tần số cao cho qua, và ngược lại

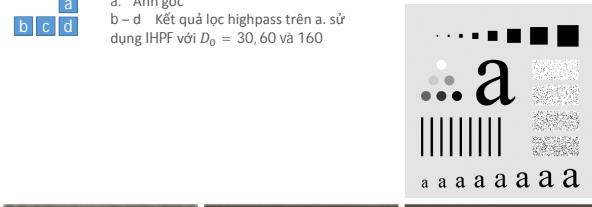
- a** a. Đồ thị phối cảnh của hàm chuyển IHPF
b. Bộ lọc được biểu diễn thành ảnh
c. Mặt cắt xuyên tâm của bộ lọc



142

143

a a. Ảnh gốc
b b – d. Kết quả lọc highpass trên a. sử dụng IHPF với $D_0 = 30, 60$ và 160



144

145

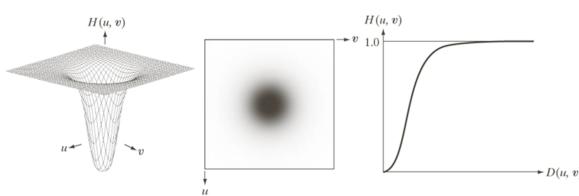
Butterworth highpass filter (BHPF)

- Hàm chuyển của BLPF bậc n với cut-off frequency D_0

$$H(u, v) = \frac{1}{1 + [D_0/D(u, v)]^{2n}}$$

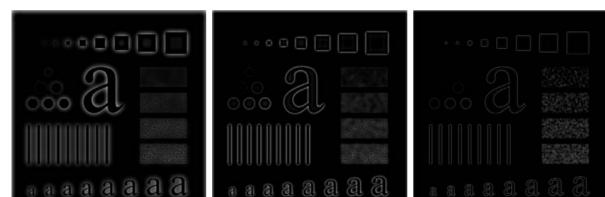
- a** a. Đồ thị phối cảnh của hàm chuyển BLPF
b. Bộ lọc được biểu diễn thành ảnh
c. Mặt cắt xuyên tâm của bộ lọc

Kết quả lọc highpass sử dụng BHPF ($n = 2$) với $D_0 = 30, 60$ và 160
 Ảnh đầu ra của BHPF mượt hơn so với ảnh đầu ra của IHPF



146

147

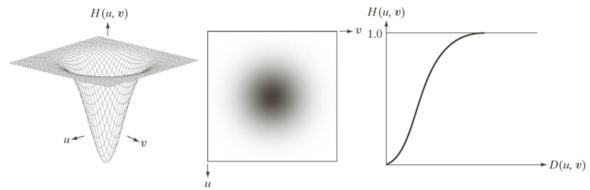


Gaussian highpass filter (GHPF)

- Hàm chuyển của GHPF với cut-off frequency D_0

$$H(u, v) = 1 - e^{-\frac{D(u,v)^2}{2D_0^2}}$$

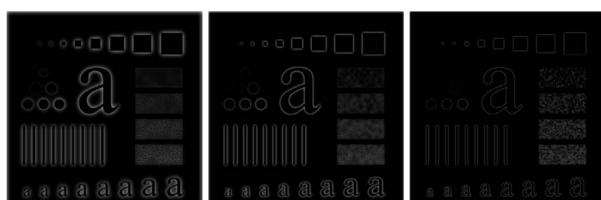
- a | b | c**
- a. Đồ thị phối cảnh của hàm chuyển GHPF
 - b. Bộ lọc được biểu diễn thành ảnh
 - c. Mặt cắt xuyên tâm của bộ lọc



148

149

a | b | c Kết quả lọc highpass sử dụng GHPF với $D_0 = 30, 60$ và 160



Tổng hợp các bộ lọc tầng số cao

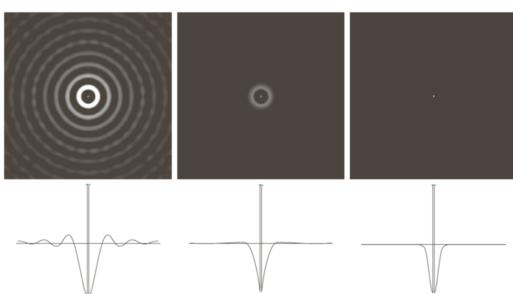
Bộ lọc	Công thức
Ideal high filter (IHPF)	$H(u, v) = \begin{cases} 0 & \text{if } D(u, v) \leq D_0 \\ 1 & \text{if } D(u, v) > D_0 \end{cases}$
Butterworth highpass filter (BHPF)	$H(u, v) = \frac{1}{1 + [D_0/D(u, v)]^{2n}}$
Gaussian highpass filter (GHPF)	$H(u, v) = 1 - e^{-\frac{D(u,v)^2}{2D_0^2}}$

* D_0 : cut-off frequency, n : bậc của bộ lọc Butterworth

150

151

a | b | c Biểu diễn không gian của bộ lọc (a) IHPF, (b) BHPF, và (c) GHPF, cùng với lát cắt cường độ qua tâm của mỗi bộ lọc



Laplacian trong miền tần số

152

153

Laplacian trong miền tần số

- Laplacian có thể được triển khai trong miền tần số bằng bộ lọc có dạng

$$H(u, v) = -4\pi^2 D^2(u, v)$$

- Ảnh Laplacian $\nabla^2 f(x, y) = \mathcal{F}^{-1}\{H(u, v)F(u, v)\}$

Laplacian trong miền tần số

- Tăng cường ảnh trong miền không gian

$$g(x, y) = f(x, y) + c\nabla^2 f(x, y)$$
 - Chọn $c = -1$ vì $H(u, v)$ âm
 - Miền giá trị của $\nabla^2 f(x, y)$ có thể gấp nhiều lần so với $f(x, y) \Rightarrow$ chuẩn hóa $f(x, y)$ về $[0, 1]$ trước khi tính DFT và $\nabla^2 f(x, y)$ về $[-1, 1]$

154

155

Laplacian trong miền tần số

- Tăng cường ảnh trong miền tần số

$$\begin{aligned} g(x, y) &= \mathcal{F}^{-1}\{F(u, v) - H(u, v)F(u, v)\} \\ &= \mathcal{F}^{-1}\{[1 - H(u, v)]F(u, v)\} \\ &= \mathcal{F}^{-1}\{[1 + 4\pi^2 D^2(u, v)]F(u, v)\} \end{aligned}$$

- Cũng gấp phải vấn đề miền giá trị
- Không được chọn phổ biến như thao tác trên miền không gian vì khó xác định hệ số chuẩn hóa

a b

a. Ảnh gốc bị nhòe

b. Ảnh tăng cường sử dụng Laplacian trong miền tần số



156

157

Qui trình unsharp masking

- Làm nhòe ảnh gốc $f(x, y)$ thành $f_{LP}(x, y)$

$$f_{LP}(x, y) = \mathcal{F}^{-1}\{H_{lp}(u, v)F(u, v)\}$$
 - $H_{lp}(u, v)$: bộ lọc lowpass trong miền tần số
- Trừ ảnh nhòe từ ảnh gốc (hiệu số gọi là mask)

$$g_{mask}(x, y) = f(x, y) - f_{LP}(x, y)$$
- Cộng mask và ảnh gốc

$$g(x, y) = f(x, y) + k * g_{mask}(x, y)$$
 - $k = 1$: unsharp masking, $k > 1$: highboost filtering

Qui trình unsharp masking

- Biểu diễn công thức tạo $g(x, y)$ theo khái niệm trong miền tần số

$$g(x, y) = \mathcal{F}^{-1}\{[1 + k * (1 - H_{LP}(u, v))]F(u, v)\}$$

158

159

High-frequency-emphasis filtering

- Công thức tính $g(x, y)$ sử dụng bộ lọc highpass $H_{HP}(x, y)$

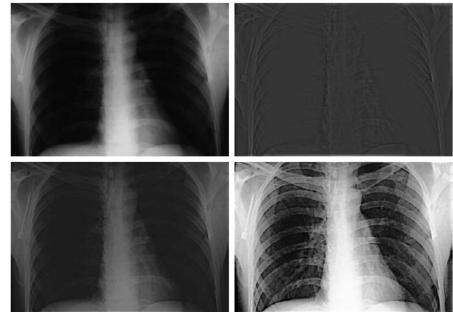
$$g(x, y) = \mathcal{F}^{-1}\{[1 + k * H_{LP}(u, v)]F(u, v)\}$$

- Công thức tổng quát

$$g(x, y) = \mathcal{F}^{-1}\{[k_1 + k_2 * H_{HP}(u, v)]F(u, v)\}$$

- k_1 là hằng số điều khiển giá trị offset tính từ gốc
- k_2 là hằng số điều khiển độ đóng góp của tần số cao

- | | |
|---|---|
| a | b |
| c | d |
- a. Ảnh X-quang ngực
 - b. Kết quả highpass filtering với bộ lọc Gaussian
 - c. Kết quả của high-frequency-emphasis filter với cùng bộ lọc
 - d. Kết quả áp dụng cân bằng histogram trên c.



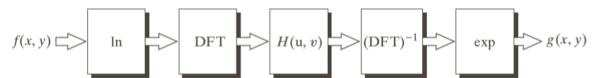
160

161

Lọc homomorphic

Lọc homomorphic

- Làm yếu sự đóng góp của tần số thấp **đồng thời** nhấn mạnh sự đóng góp của tần số cao



162

163

Lọc homomorphic

- Ảnh $f(x, y)$ được phân tích thành hai thành phần *illumination* $i(x, y)$ và *reflectance* $r(x, y)$

$$f(x, y) = i(x, y)r(x, y)$$
- Giả sử ta định nghĩa $z(x, y) = \ln f(x, y)$

Thì $\mathcal{F}\{z(x, y)\} = \mathcal{F}\{\ln f(x, y)\}$

- Đặt $F_i(x, y) = \mathcal{F}\{\ln i(x, y)\}$, $F_r(x, y) = \mathcal{F}\{\ln r(x, y)\}$
 $Z(u, v) = F_i(u, v) + F_r(u, v)$

Lọc homomorphic

- Lọc $Z(u, v)$ bằng bộ lọc $H(u, v)$ sao cho

$$\begin{aligned} S(u, v) &= H(u, v)Z(u, v) \\ &= H(u, v)F_i(u, v) + H(u, v)F_r(u, v) \end{aligned}$$

- Ảnh đã lọc trong miền không gian là

$$\begin{aligned} s(x, y) &= \mathcal{F}^{-1}\{S(u, v)\} \\ &= \mathcal{F}^{-1}\{H(u, v)F_i(u, v)\} + \mathcal{F}^{-1}\{H(u, v)F_r(u, v)\} \end{aligned}$$

- Đặt $i' = \mathcal{F}^{-1}\{H(u, v)F_i(u, v)\}$, $r' = \mathcal{F}^{-1}\{H(u, v)F_r(u, v)\}$
 $s(x, y) = i'(x, y) + r'(x, y)$

164

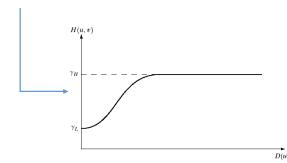
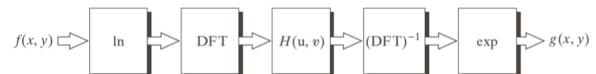
165

Lọc homomorphic

- Ảnh đầu ra

$$g(x, y) = e^{s(x, y)} = e^{i'(x, y)r'(x, y)} e^{r'(x, y)} = i_0(x, y)r_0(x, y)$$

- $i_0(x, y)$: thành phần illumination của ảnh đầu ra
- $r_0(x, y)$: thành phần reflectance của ảnh đầu ra



Đường biến thiên có thể được ước lượng bằng cách chỉnh sửa bộ lọc Gaussian highpass

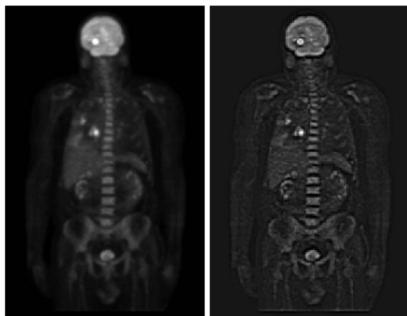
$$H(u, v) = (\gamma_H - \gamma_L) \left[1 - e^{-\frac{[D^2(u, v)]}{D_0^2}} \right] + \gamma_L$$

Khi $\gamma_L < 1$ và $\gamma_H > 1$: làm yếu sự đóng góp của tần số thấp (illumination) và tăng cường sự đóng góp của tần số cao (reflectance) \Rightarrow đồng thời nén dynamic range và tăng cường độ tương phản

166

167

- a. Ảnh PET của cơ thể người
b. Tăng cường ảnh bằng homomorphic filtering



168

169

Nâng cao chất lượng ảnh

Xử lý ảnh và Video số

Nâng cao chất lượng ảnh

• Xử lý ảnh đầu vào để ảnh đầu ra “nhìn tốt hơn” (visually better) ảnh đầu vào cho một ứng dụng cụ thể.

• Nâng cao chất lượng ảnh có thể thực hiện:

- Miền không gian: thao tác trên ảnh gốc
$$g(m, n) = T[f(m, n)]$$
- Miền tần số: thao tác trên DFT của ảnh gốc
$$G(u, v) = T[F(u, v)]$$

với $F(u, v) = F[f(m, n)]$ và $G(u, v) = F[g(m, n)]$

Nâng cao chất lượng ảnh



170

171

Vấn đề khó khăn của DFT

$$\mathcal{F}_n = \sum_{k=0}^{N-1} g_k e^{-j \frac{2\pi}{N} nk}$$

Vấn đề của DFT

Xử lý ảnh trong miền tần số

- Việc đánh giá cần N^2 phép nhân và $N(N - 1)$ phép cộng trên số phức
- Bài toán thực tế thường chọn $N \geq 256 \Rightarrow$ gấp phải vấn đề lớn về tốc độ tính toán

172

173

Fast Fourier Transform (FFT)

- 1805 – Gauss đề ra phương pháp nội suy quỹ đạo của tiểu hành tinh Pallas và Juno
- 1805 – 1965 Vài phiên bản FFT khác ra đời
 - 1932 Yates, Interaction algorithm cho Hadamard and Walsh transforms
 - 1942 Danielson và Lanczos, tính DFT cho dữ liệu tinh thể học, tận dụng tính chu kỳ thay vì tính đối xứng
- 1965 – Cooley and Turkey, tính DFT khi N đa hợp và thường là mũ của 2

Fast Fourier Transform (FFT)

- FFT tính DFT nhanh hơn với độ phức tạp $\Theta(N \log N)$ (so với $O(N^2)$)
- Nhiều thuật toán FFT còn cho độ lỗi thấp hơn đánh giá DFT trực tiếp
- Ví dụ:

	Đánh giá DFT trực tiếp	Cooley – Tukey (1965)
Phép nhân	N^2	$N/2\log_2(N)$
Phép cộng	$N(N - 1)$	$N\log_2(N)$

174

175

Phát hiện biên cạnh là gì?

- Phát hiện biên cạnh là quá trình xác định và định vị các điểm biên cạnh
- Là tác vụ nền tảng trong nhiều lĩnh vực: xử lý ảnh, thị giác máy tính.
- Thường được sử dụng để phát hiện và rút trích đặc trưng thành phần, hình dáng.

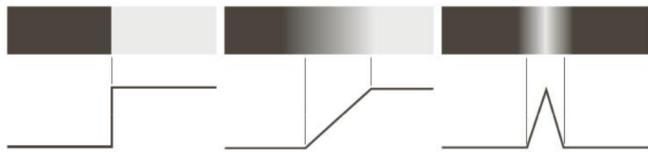
Biên cạnh là gì?



4

5

Đặt vấn đề



- Đại lượng nào biểu diễn cho sự thay đổi độ sáng ảnh?
- Đại lượng nào biểu diễn cho sự thay đổi độ sáng của một hàm số theo biến số vector?

6

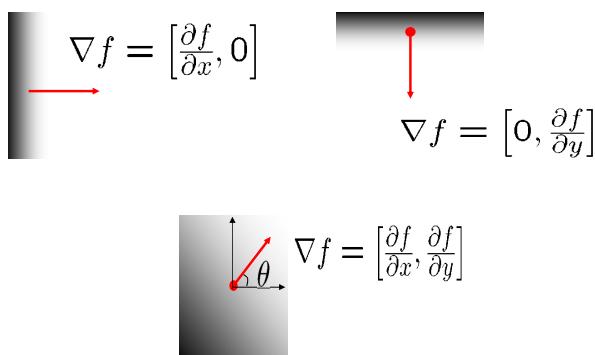
7

Gradient vector của một hàm ảnh

- Cho hàm ảnh f
- Gradient của hàm ảnh tại x, y là một vector được định nghĩa $\nabla f = \left[\frac{\partial f(x, y)}{\partial x}, \frac{\partial f(x, y)}{\partial y} \right]$
- Ta có hướng của đạo hàm là $\theta = \tan^{-1} \left(\frac{\partial f(x, y)}{\partial y} / \frac{\partial f(x, y)}{\partial x} \right)$
- Độ lớn của đạo hàm

$$|\nabla f| = \sqrt{\left(\frac{\partial f(x, y)}{\partial x} \right)^2 + \left(\frac{\partial f(x, y)}{\partial y} \right)^2}$$

Gradient vector của một hàm ảnh



8

9

Gradient vector của một ảnh

- Xét ảnh xám I
- Gradient của ảnh I tại x, y là một vector được định nghĩa như sau $\nabla I(x, y) = (I_x, I_y) = \left(\frac{\partial I(x, y)}{\partial x}, \frac{\partial I(x, y)}{\partial y} \right)$
- Trong đó $I_x = \frac{\partial I(x, y)}{\partial x} \approx \frac{I(x + dx, y) - I(x, y)}{dx}$
 $I_y = \frac{\partial I(x, y)}{\partial y} \approx \frac{I(x, y + dy) - I(x, y)}{dy}$

Gradient vector của một ảnh

- Độ lớn của gradient

$$|\nabla I(x, y)| = \left(\left(\frac{\partial I(x, y)}{\partial x} \right)^2 + \left(\frac{\partial I(x, y)}{\partial y} \right)^2 \right)^{1/2}$$

- Hướng của gradient

$$\angle = \tan^{-1} \left(\frac{\partial I(x, y)}{\partial y} / \frac{\partial I(x, y)}{\partial x} \right)$$

thể hiện góc lệch của gradient so với trục Ox



$$\nabla I = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right)$$

10

11

Một số phương pháp phát hiện biên cạnh

- Phát hiện biên cạnh sử dụng gradient
- Phát hiện biên cạnh sử dụng toán tử Laplace
- Thuật toán Canny

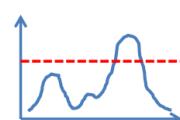
Phát Hiện Biên Cảnh

THUẬT TOÁN DỰA VÀO GRADIENT

Thuật toán sử dụng gradient



Tính độ lớn gradient tại từng vị trí (x,y) trong ảnh



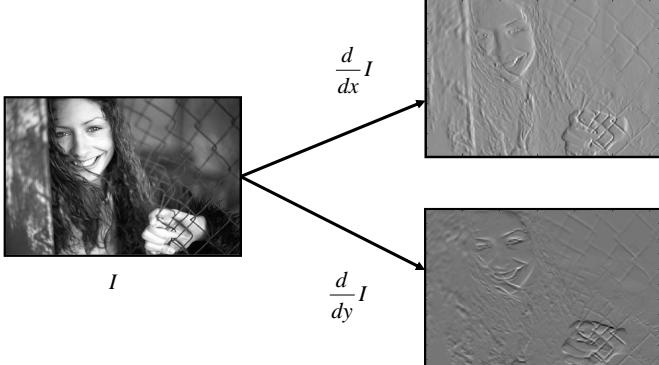
Nhi phân hóa ảnh dựa trên độ lớn của gradient



12

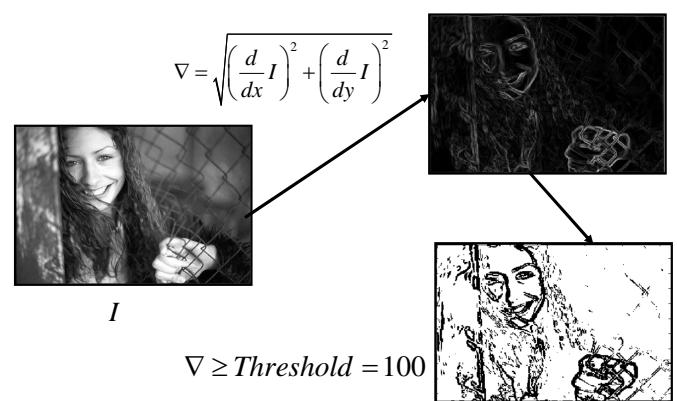
13

Ví dụ



14

Ví dụ



15

Một số mặt nạ thông dụng

- Mặt nạ Sobel

$$H_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$H_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

- Mặt nạ Prewitt

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

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

Một số mặt nạ thông dụng

- Mặt nạ Robert

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

H_x

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

H_y

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

H_x

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

H_y

16

17

Phát hiện biên cạnh sử dụng gradient

- Bài tập:

- Viết chương trình tính gradient theo hướng x và y của một ảnh bất kỳ.
- Viết chương trình phát hiện biên cảnh của một ảnh dựa trên gradient. Có thể cho phép chọn ngưỡng và chọn mặt nạ gradient (Sobel, Prewit, Robert)
- Chọn 3-5 ảnh chạy thực nghiệm và viết báo cáo kết quả trên từng bộ lọc.

Phát Hiện Biên Cảnh

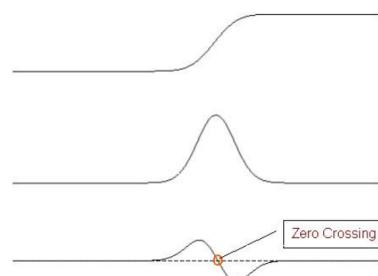
THUẬT TOÁN DỰA VÀO LAPLACE

Toán tử Laplace

- Phát hiện biên cạnh dụng gradient vector
 - Vùng độ sáng thay đổi độ ngột
 - Vùng độ sáng thay đổi tiệm tiến



Toán tử Laplace



20

21

Toán tử Laplace

- Xét ảnh xám I
- Laplace của ảnh I tại x, y là một vector được định nghĩa như sau

$$\nabla^2 I(x, y) = \frac{\partial^2 I(x, y)}{\partial x^2} + \frac{\partial^2 I(x, y)}{\partial y^2}$$

- Trong đó $\frac{\partial^2 I(x, y)}{\partial x^2} = \frac{\partial}{x} \left(\frac{\partial I(x, y)}{\partial x} \right)$

$$\frac{\partial^2 I(x, y)}{\partial y^2} = \frac{\partial}{y} \left(\frac{\partial I(x, y)}{\partial y} \right)$$

22

Toán tử Laplace

- Toán tử Laplace không dùng trực tiếp để phát hiện biên được bởi vì
 - Rất nhạy với nhiễu
 - Biên độ có thể phát sinh cạnh kép
 - Không thể dùng để xác định hướng cạnh
- Để định vị biên thì dùng zero-crossing

23

Toán tử Laplace

- Biểu diễn toán tử Laplace có thể thông qua qua tao tác lọc ảnh với mặt nạ

0	1	0
1	-4	1
0	1	0

hoặc

1	1	1
1	-8	1
1	1	1

hoặc

-1	2	-1
2	-4	2
-1	2	-1

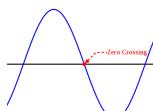
22

Thuật toán sử dụng gradient



Tính ảnh Laplace

$$\text{Laplace}(f) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

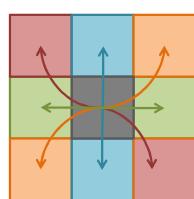


Xác định điểm Zero Crossing



25

Kiểm tra Zero Crossing



24

Bài tập

- Sử dụng hàm thư viện OpenCV chạy thuật toán Laplace.
- Chọn 3-5 ảnh thực nghiệm và báo cáo kết quả.

http://docs.opencv.org/3.2.0/de/db2/laplace_8cpp-example.html

Điều kiện Zero Crossing

- Hai giá trị đối ngẫu nghịch dấu nhau
- Độ lớn mỗi giá trị thỏa ngưỡng T

26

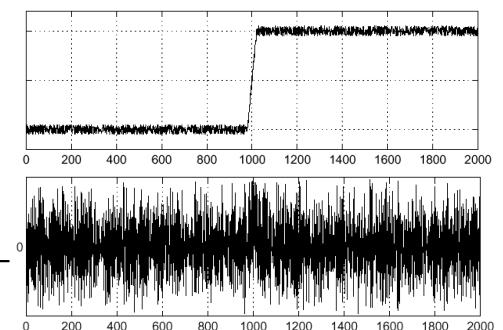
27

Tác động của nhiễu

Phát Phát Hiện Biên Cảnh

TÁC ĐỘNG CỦA NHIỄU

28

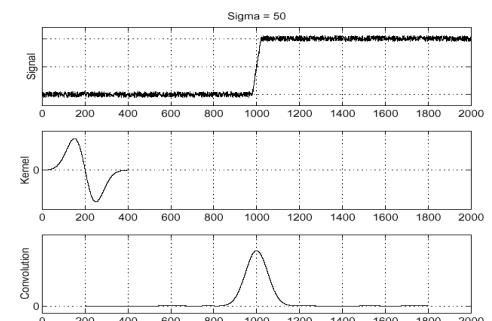


29

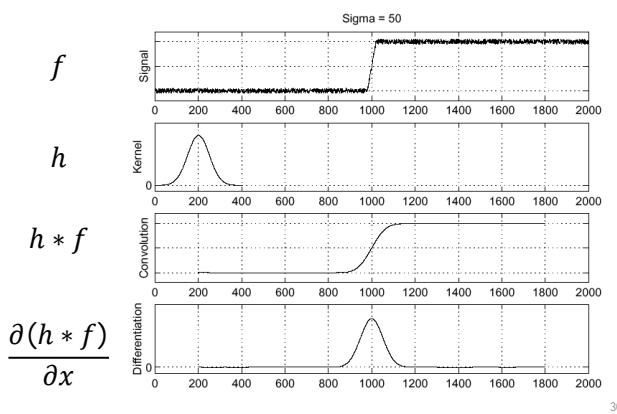
Tác động của nhiễu

Tác động của nhiễu

$$\frac{\partial(h * f)}{\partial x} = \frac{\partial h}{\partial x} * f$$



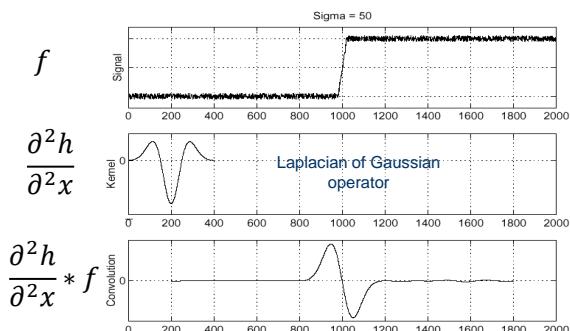
31



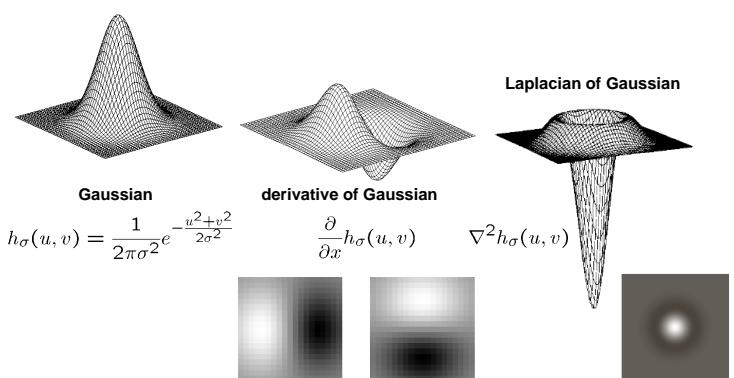
30

Tác động của nhiễu

Bộ lọc phát hiện biên cảnh



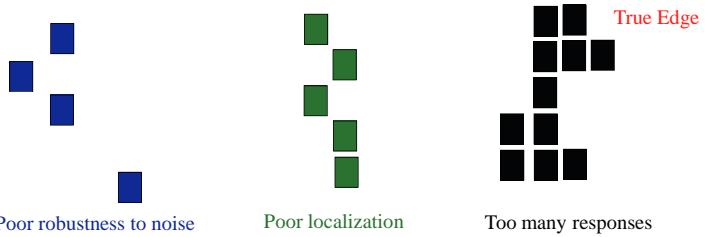
32



33

Chất lượng của thuật toán

- Bền vững với nhiễu
- Định vị chính xác
- Quá nhiều hoặc quá ít biên cạnh



Phát Hiện Biên Cạnh

THUẬT TOÁN CANNY

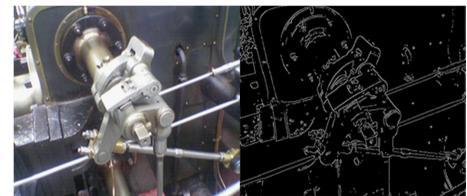
34

35

Thuật toán Canny

- Tiêu chuẩn 1: **Lỗi ít nhất**. Tất cả các biên phải được tìm thấy và không có biên giả.
- Tiêu chuẩn 2: **Định vị chính xác**. Biên cạnh được tìm thấy phải gần với biên thực tế nhất
- Tiêu chuẩn 3: **Đơn biên**. Chỉ rả về duy nhất một điểm cho các biên tìm được

Thuật toán Canny



36

37



Các bước của thuật toán



38

Các bước thuật toán

- Lọc ảnh gauss

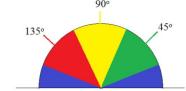
$$S = G_\sigma * I \quad G_\sigma = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- Tính đạo hàm theo x, y

$$\nabla S = \begin{bmatrix} \frac{\partial}{\partial x} S & \frac{\partial}{\partial y} S \end{bmatrix}^T = [S_x \quad S_y]^T$$

- Tính biên độ và hướng của đạo hàm

$$|\nabla S| = \sqrt{S_x^2 + S_y^2} \quad \theta = \tan^{-1} \frac{S_y}{S_x}$$



39

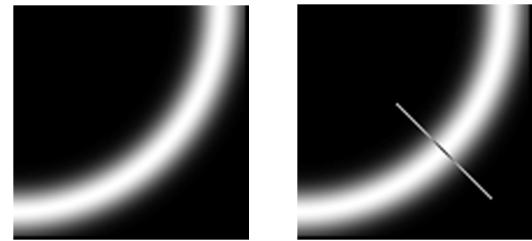
Khử phi cực đại



$$|\nabla S| \geq \text{Threshold} = 25$$



Khử phi cực đại

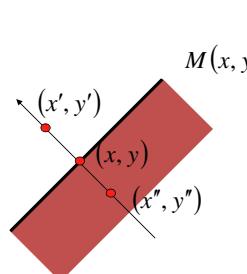


Chúng muốn đánh dấu những điểm dọc theo đường cong mà biên độ là lớn nhất. Chúng ta có thể làm điều này bằng cách tìm cực đại dọc theo một hướng của đường cong. Những điểm này tạo thành đường cong. Có 2 vấn đề trong thuật toán: đâu là điểm cực đại, và đâu là điểm tiếp theo?

41

Khử phi cực đại

- Khử các điểm ảnh trong ảnh biên độ gradient mà không cực đại cục bộ



$$M(x, y) = \begin{cases} |\nabla S|(x, y) & \text{if } |\nabla S|(x, y) > |\nabla S|(x', y') \\ & \& |\nabla S|(x, y) > |\nabla S|(x'', y'') \\ 0 & \text{otherwise} \end{cases}$$

(x', y') và (x'', y'') là các lân cận của (x, y) trong $|\nabla S|$ cùng chuẩn hóa hướng đến một cạnh

42

Khử phi cực đại



$$|\nabla S| = \sqrt{S_x^2 + S_y^2}$$

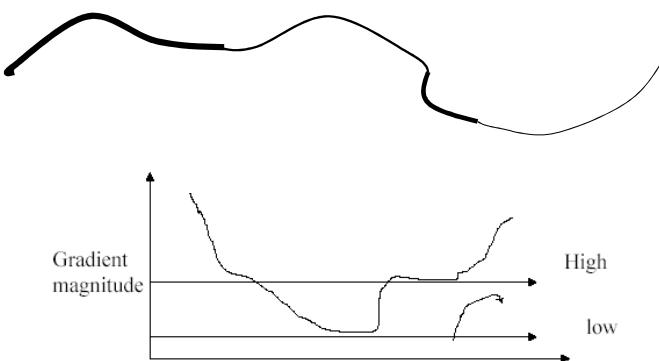


$$M \geq \text{Threshold} = 25$$



43

Liên kết cạnh



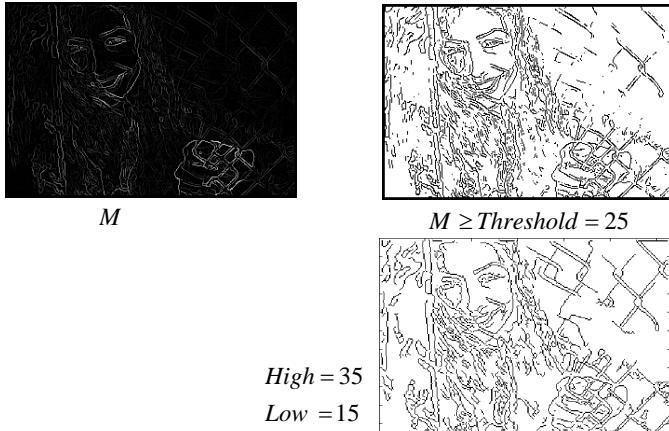
44

- Nếu biên độ tại điểm ảnh cao hơn ngưỡng trên thì nó là điểm biên cạnh
- Nếu biên độ tại điểm ảnh thấp hơn ngưỡng dưới thì nó không là điểm biên cạnh
- Nếu biên độ tại điểm ảnh nằm giữa ngưỡng thấp và cao là điểm biên cạnh nếu và chỉ nếu nó liên kết với một điểm biên cạnh trực tiếp hoặc qua những điểm ảnh nằm giữa ngưỡng thấp và cao.

45

Liên kết cạnh

Liên kết cạnh



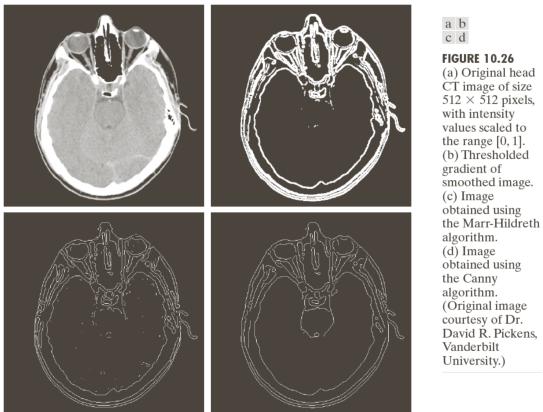
Thuật toán Canny



FIGURE 10.25
(a) Original image of size 834×1114 pixels, with intensity values scaled to the range $[0, 1]$.
(b) Thresholded gradient of smoothed image.
(c) Image obtained using the Marr-Hildreth algorithm.
(d) Image obtained using the Canny algorithm. Note the significant improvement of the Canny image compared to the other two.

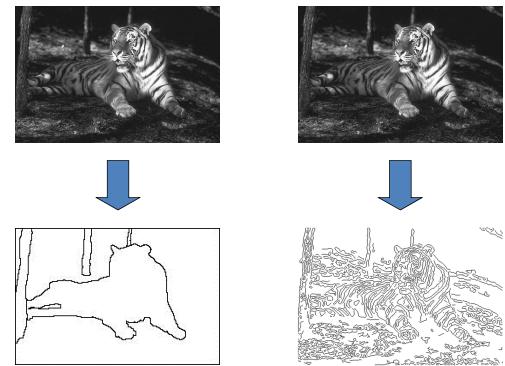
47

Thuật toán Canny



48

Nhận xét



49

Bài tập

- Sử dụng hàm thư viện OpenCV chạy thuật toán Canny.
- Chọn 3-5 ảnh thực nghiệm và báo cáo kết quả.

http://docs.opencv.org/3.0-beta/doc/tutorials/imgproc/imgtrans/canny_detector/canny_detector.html

Phát Hiện Biên Cảnh

LƯỢC ĐỒ HƯỚNG CẠNH

Lược đồ hướng cạnh

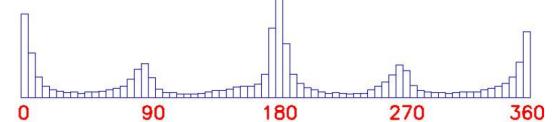
- Thể hiện phân bố của biên cạnh tìm được trong ảnh

$$H[i]_{ED} = \frac{N[i]}{N}$$

- Trong đó

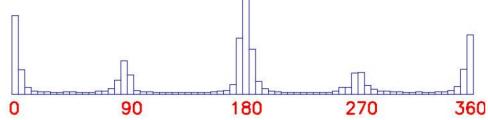
- N[i] là số điểm biên có hướng i trong ảnh
- N là tổng số điểm biên tìm được trong ảnh

52



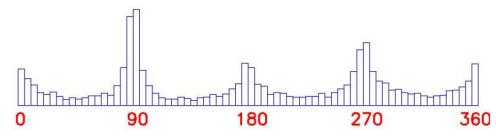
53

Lược đồ hướng cạnh



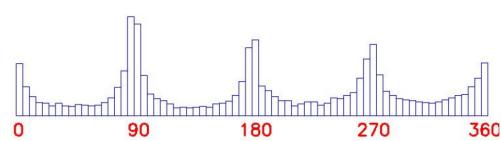
54

Lược đồ hướng cạnh



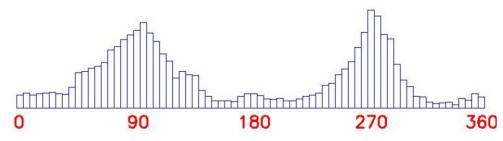
55

Lược đồ hướng cạnh



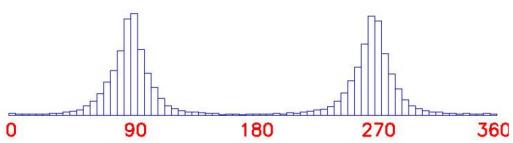
56

Lược đồ hướng cạnh



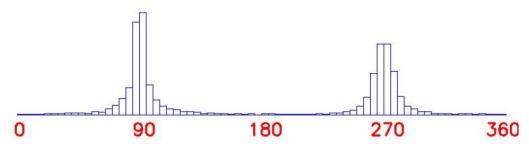
57

Lược đồ hướng cạnh



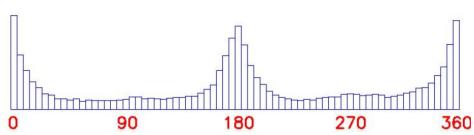
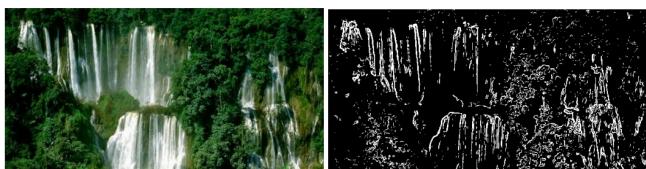
58

Lược đồ hướng cạnh



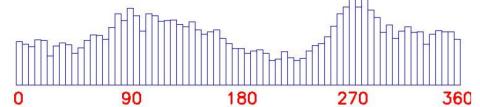
59

Lược đồ hướng cạnh



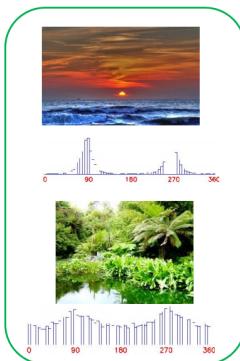
60

Lược đồ hướng cạnh



61

Phân lớp ảnh



Cảnh tự nhiên



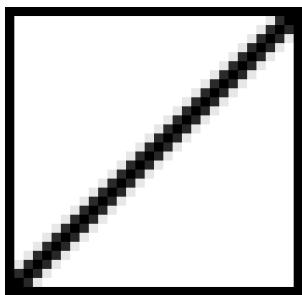
Cảnh nhân tạo

62

Đặt vấn đề

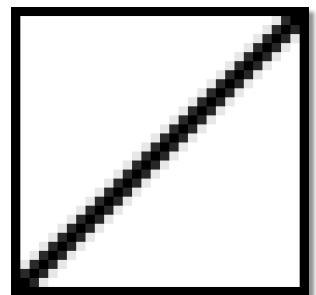


Phóng to

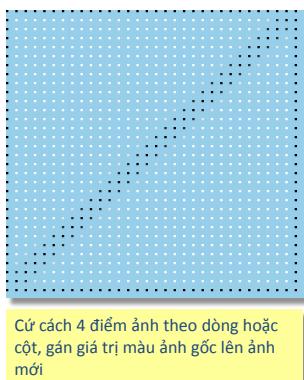
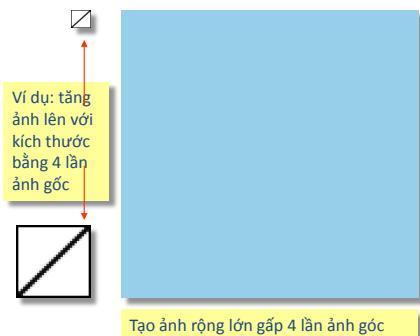


Đặt vấn đề

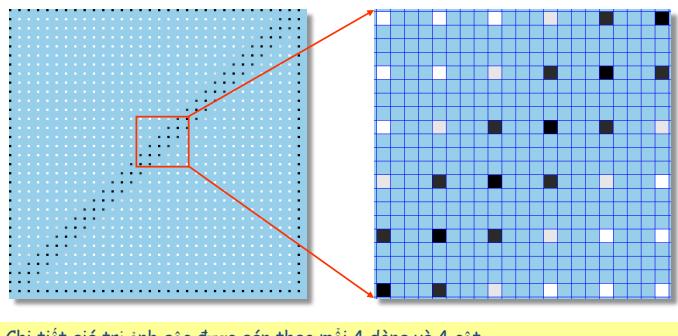
Thu nhỏ



Phóng to ảnh bằng nhân rộng điểm ảnh

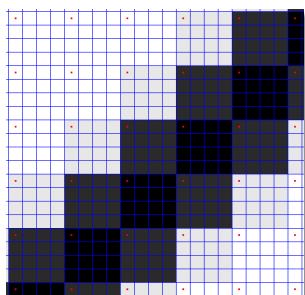
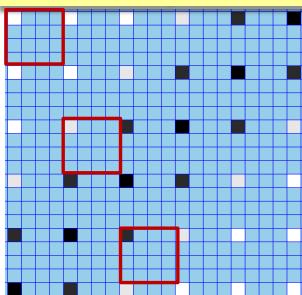


Phóng to ảnh bằng nhân rộng điểm ảnh

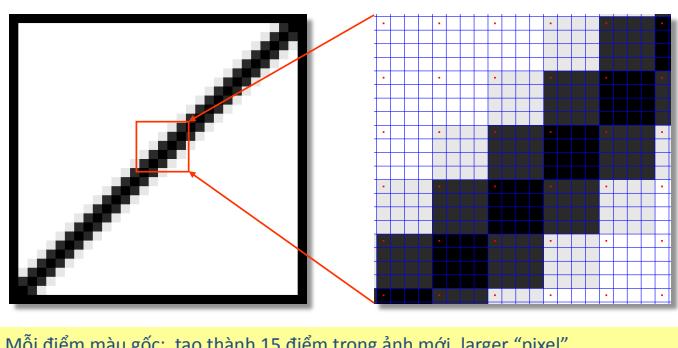


Phóng to ảnh bằng nhân rộng điểm ảnh

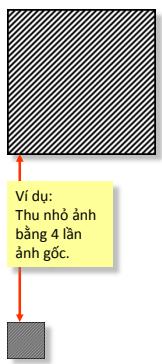
Điểm ảnh trống sẽ nhận giá trị màu của điểm ảnh nằm gốc bên trái



Phóng to ảnh bằng nhân rộng điểm ảnh



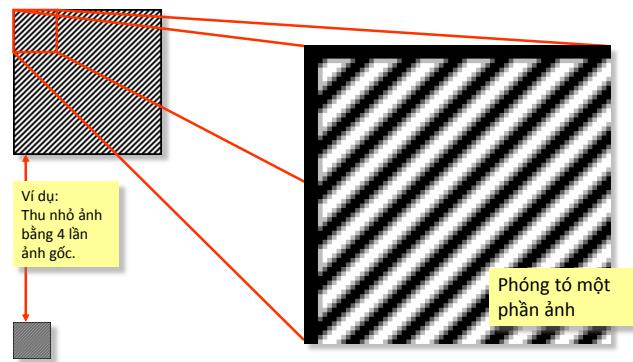
Thu nhỏ ảnh



Ví dụ:
Thu nhỏ ảnh
bằng 4 lần
ảnh gốc.

**Thu nhỏ theo
hệ số n :** giữ lại
điểm ảnh theo
mỗi n dòng và
n cột

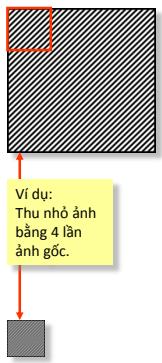
Thu nhỏ ảnh bằng chia thập phân điểm ảnh



Ví dụ:
Thu nhỏ ảnh
bằng 4 lần
ảnh gốc.

Phóng to một
phần ảnh

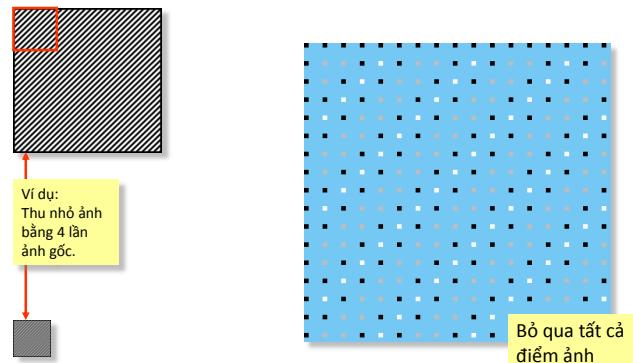
Thu nhỏ ảnh bằng chia thập phân điểm ảnh



Ví dụ:
Thu nhỏ ảnh
bằng 4 lần
ảnh gốc.

Giữ lại điểm ảnh
cứ mỗi 4 dòng và
4 cột

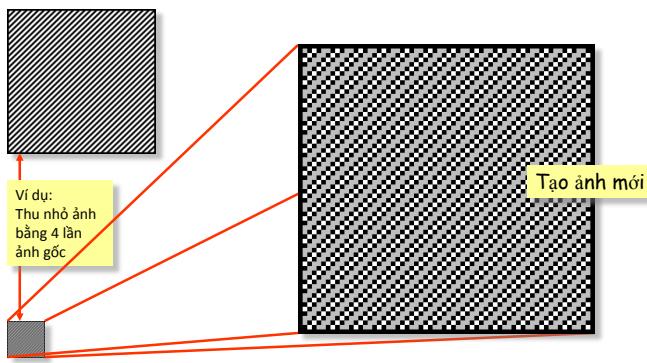
Thu nhỏ ảnh bằng chia thập phân điểm ảnh



Ví dụ:
Thu nhỏ ảnh
bằng 4 lần
ảnh gốc.

Bỏ qua tất cả
điểm ảnh khác

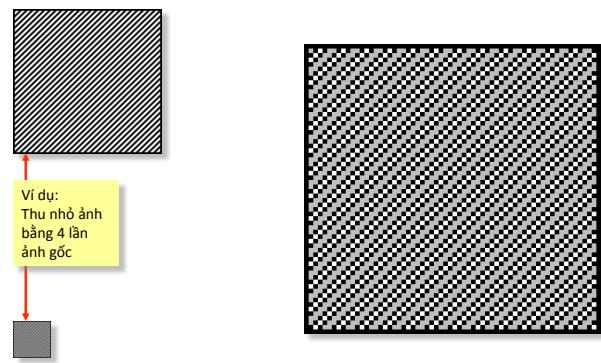
Thu nhỏ ảnh bằng chia thập phân điểm ảnh



Ví dụ:
Thu nhỏ ảnh
bằng 4 lần
ảnh gốc

Tạo ảnh mới

Thu nhỏ ảnh bằng chia thập phân điểm ảnh



Ví dụ:
Thu nhỏ ảnh
bằng 4 lần
ảnh gốc

Các hướng tiếp cận thông dụng

- Nội suy láng giềng gần nhất
- Nội suy song tuyến tính
- Nội suy bicubic

Tái lấy mẫu

CÁC HƯỚNG TIẾP CẬN THÔNG DỤNG

Lấy mẫu láng giềng gần nhất

Thuật toán láng giềng gần nhất là tổng quát hóa của việc nhân rộng điểm ảnh và lấy theo hệ số.

Nó bao gồm một tỷ lệ thay đổi kích thước, i.e. thay đổi kích thước ảnh có tỷ lệ p/q số điểm ảnh theo dòng và p/q số điểm ảnh theo cột trong ảnh gốc. (p và q là số nguyên.)



Lấy mẫu láng giềng gần nhất

thay đổi kích thước 3/7



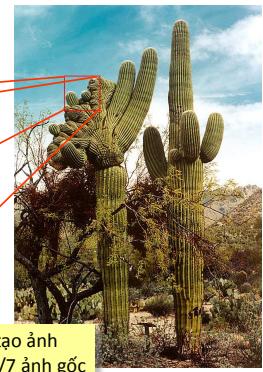
Phóng lớn một phần

Lấy mẫu láng giềng gần nhất

Phóng to một vùng ảnh

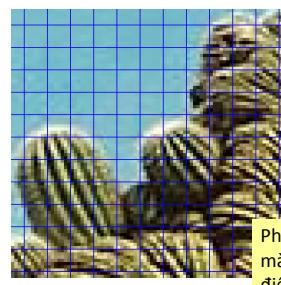


Ví dụ: tạo ảnh bằng 3/7 ảnh gốc

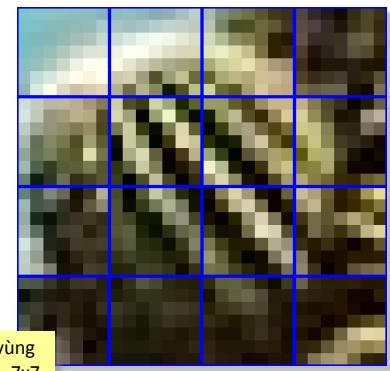


Lấy mẫu láng giềng gần nhất

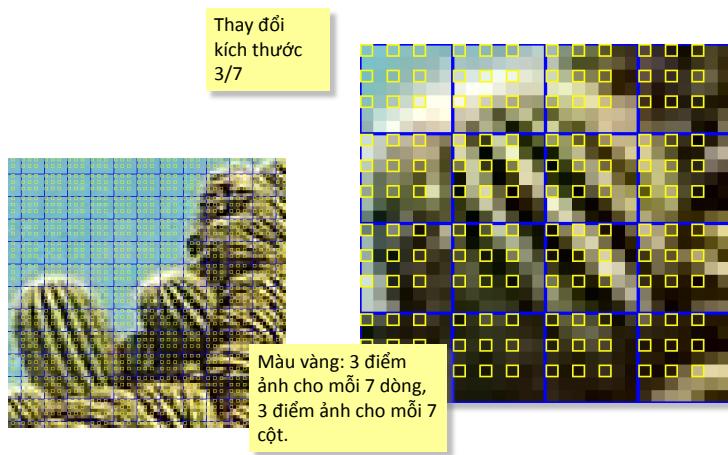
Thay đổi kích thước 3/7



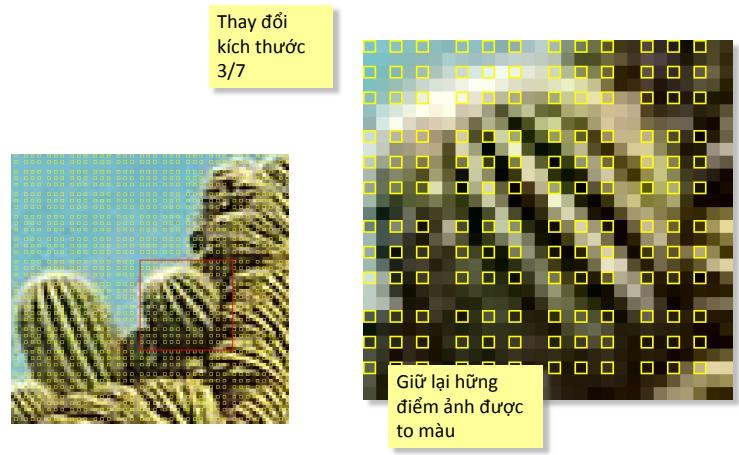
Phác họa vùng màu xanh: 7x7 điểm ảnh



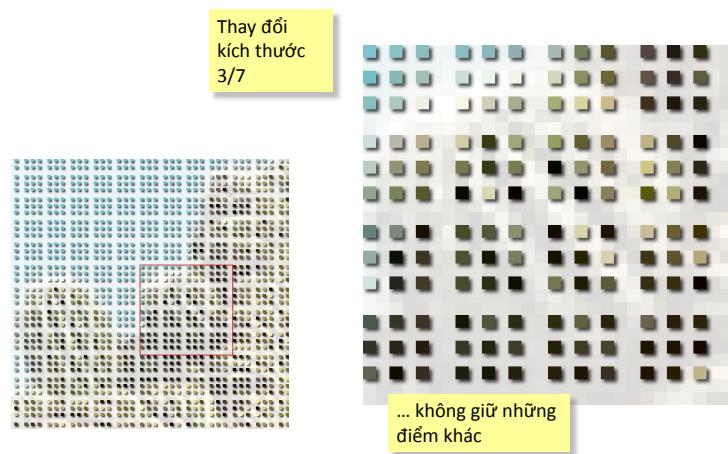
Lấy mẫu láng giềng gần nhất



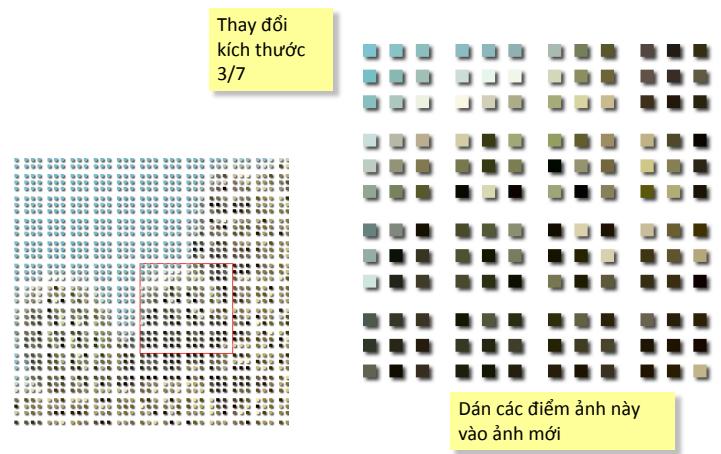
Lấy mẫu láng giềng gần nhất



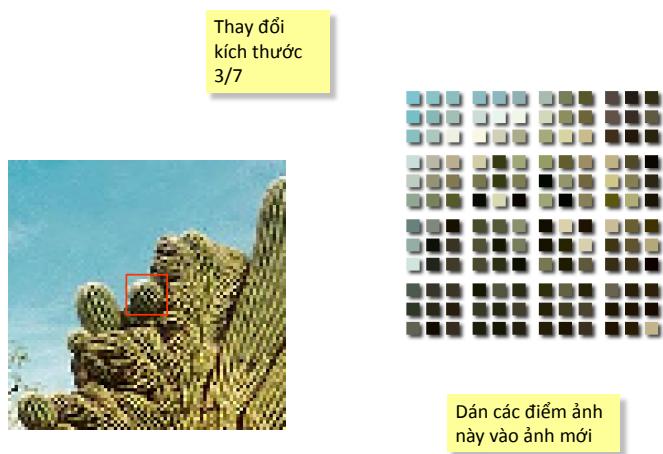
Lấy mẫu láng giềng gần nhất



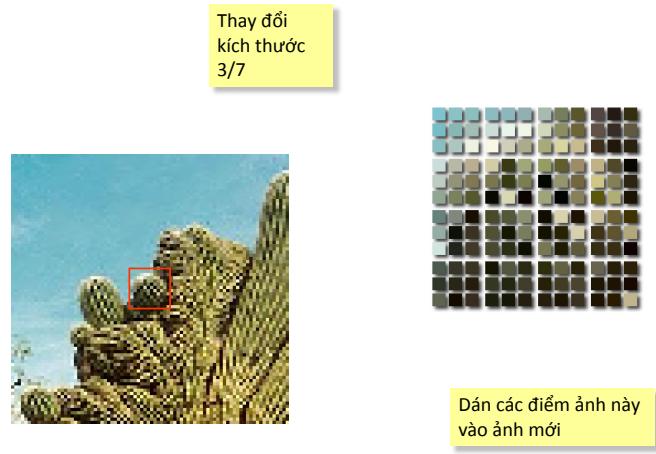
Lấy mẫu láng giềng gần nhất



Lấy mẫu láng giềng gần nhất



Lấy mẫu láng giềng gần nhất



Lấy mẫu láng giềng gần nhất



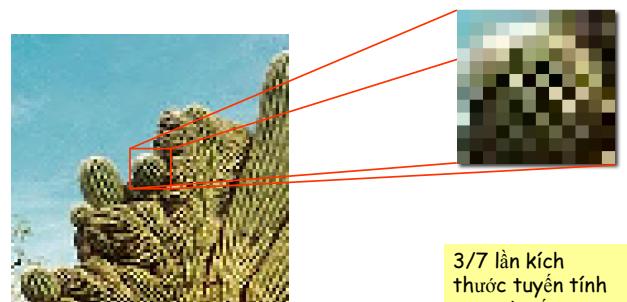
Lấy mẫu láng giềng gần nhất



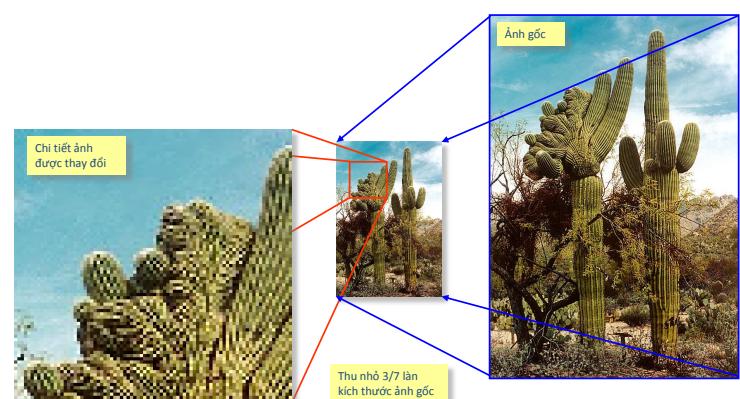
Lấy mẫu láng giềng gần nhất



Lấy mẫu láng giềng gần nhất



Lấy mẫu láng giềng gần nhất



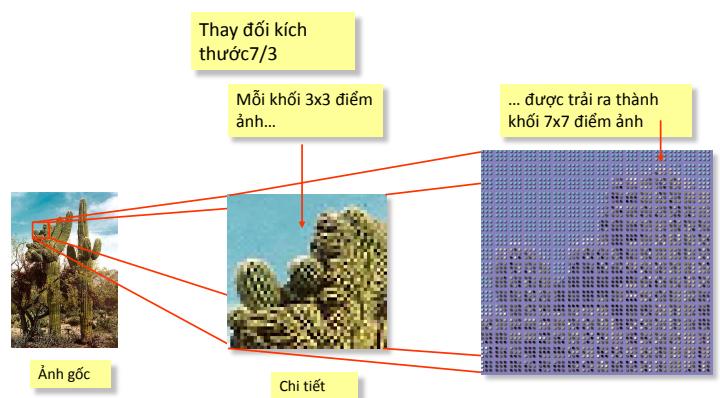
Lấy mẫu láng giềng gần nhất



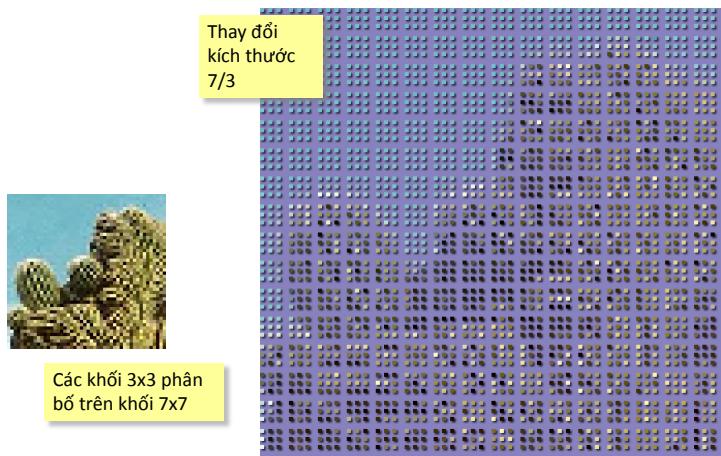
Lấy mẫu láng giềng gần nhất



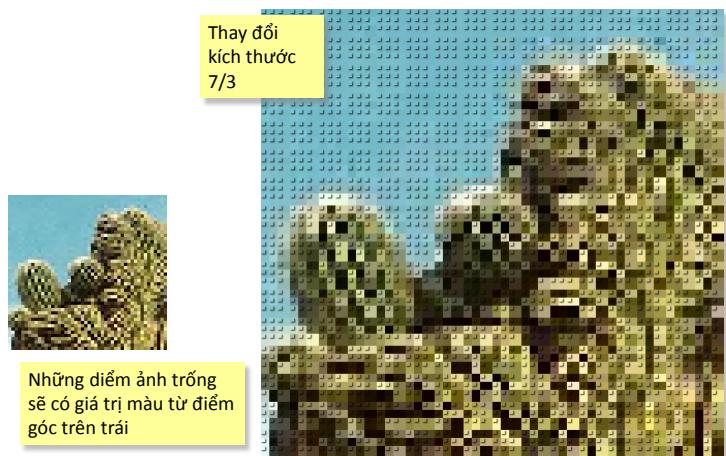
Lấy mẫu láng giềng gần nhất



Lấy mẫu láng giềng gần nhất



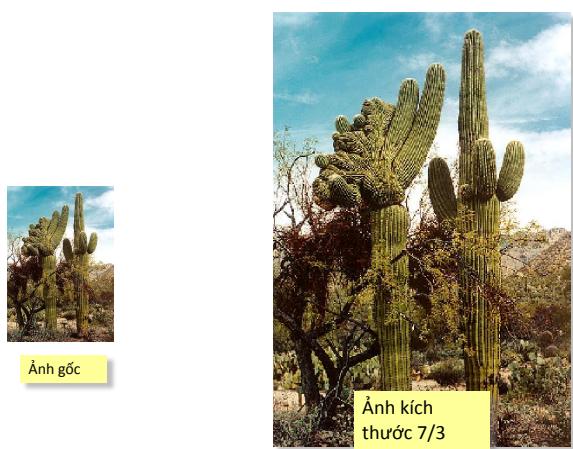
Lấy mẫu láng giềng gần nhất



Lấy mẫu láng giềng gần nhất



Lấy mẫu láng giềng gần nhất



Nội suy

Giả sử có một ảnh, I , được định nghĩa là một hình chữ nhật liên tục và có khoảng giá trị liên tục dương trong miền giá trị.

$$I: [r_l, r_h] \times [c_l, c_h] \rightarrow [g_{\min}, g_{\max}] \text{ where } g_{\min} \geq 0 \text{ and } g_{\max} < \infty.$$

Đồng thời giả sử rằng chúng ta biết giá trị của I chỉ trên một lưới điểm đều rời rạc trong miền giá trị.

$$I(r, c) \text{ for } r \in \{0, \dots, R-1\} \text{ and } c \in \{0, \dots, C-1\}.$$

Những giá trị không biết sẽ được ước lượng thông qua nội suy. Hầu hết các phương pháp nội suy thích hợp với một đa thức 2D của các điểm đã biết quanh các điểm chưa biết. Ví dụ, nếu chúng ta muốn biết giá trị I tại vị trí điểm anh con, $(4.25, 3.33)$, chúng ta tìm một đa thức, p , như sau:

$$p(4, 3) = I(4, 3), p(4, 4) = I(4, 4), p(5, 4) = I(5, 4), \text{ and } p(5, 3) = I(5, 3).$$

Đa thức được định nghĩa bất cứ đâu trong khoảng $[4, 5] \times [3, 4]$ để ước lượng $I(4.25, 3.33)$, chúng ta tính $p(4.25, 3.33)$.

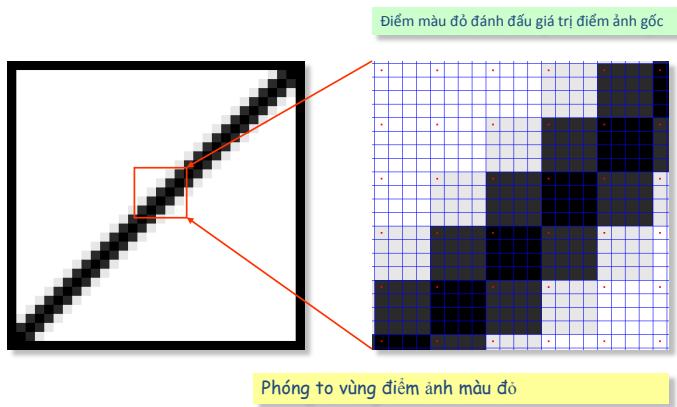
Phóng to ảnh



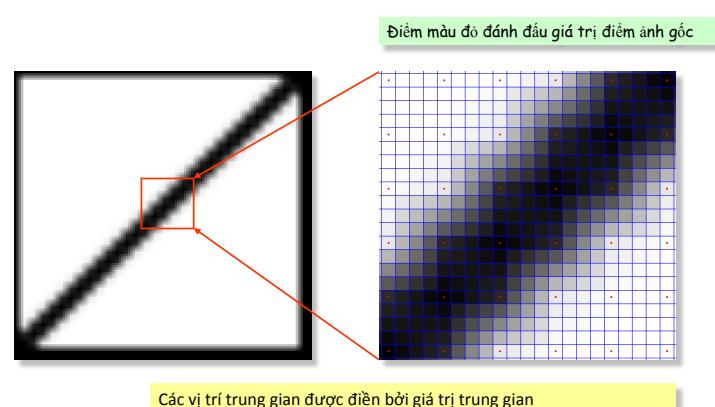
Nhân bản điểm ảnh
tạo kết quả rằng cửa vì mỗi điểm ảnh riêng biệt được làm lớn hơn.

Nội suy song tuyến tính
tạo những điểm ảnh mới có giá trị trung gian giữa ảnh gốc. Kết quả rõ ràng hơn nhưng mờ

Nhân bản điểm ảnh



Nội suy song tuyến tính

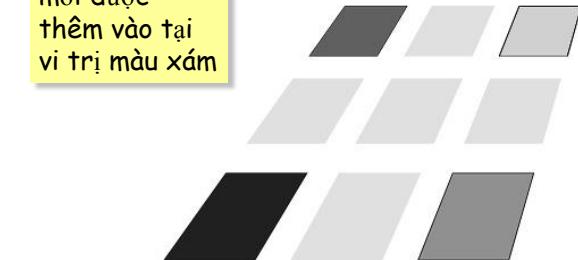


Tái lấy mẫu qua nội suy song tuyến tính

Tăng số mẫu
bởi hệ số 2

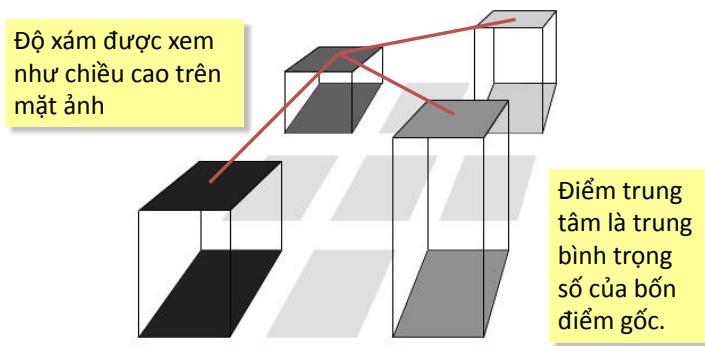


Những mẫu
mới được
thêm vào tại
vị trí màu xám

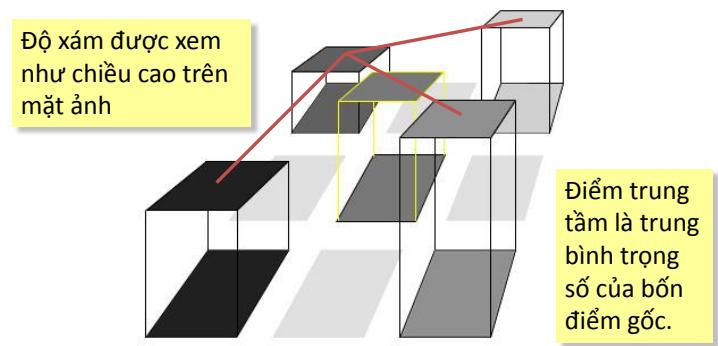


Nội suy song tuyến tính

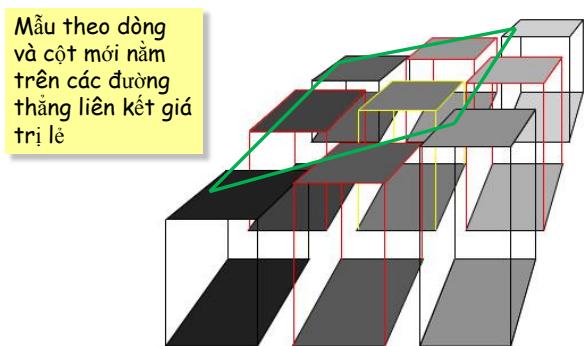
Nội suy song tuyến tính



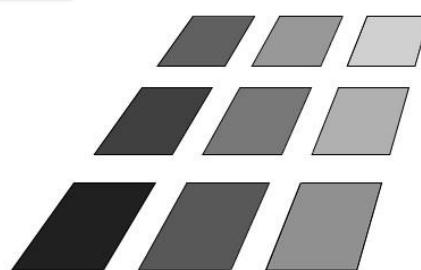
Nội suy song tuyến tính



Nội suy song tuyến tính



Nội suy song tuyến tính



Video số là gì?

- Là tập ảnh được ghi nhận theo chiều thời gian
- Thể hiện thay đổi nội dung theo thời gian



Video số là gì?



Thiết bị ghi nhận



Thiết bị ghi nhận



Thiết bị ghi nhận



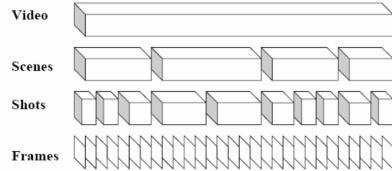
Thiết bị ghi nhận



Xử lý video số là ?

- Xử lý video số là quá trình thao tác trên video đầu vào để tạo ra video đầu ra, tập khung hình, tập video con, hoặc một số thuộc tính liên quan bằng máy tính.
- Thông thường dựa vào bài toán xử lý ảnh và thông tin biến đổi theo thời gian

Thuật ngữ trong video



- Khung hình (frame): vẫn là ảnh
 - 24 đến 30 khung hình một giây
- Đoạn cơ sở (shot): chuỗi khung hình được ghi nhận theo một thao tác camera
- Cảnh (Scene): tập hợp các đoạn hình thành một ngữ nghĩa thống nhất
 - Khái niệm, một thời gian, địa điểm

Một số bài toán cơ sở

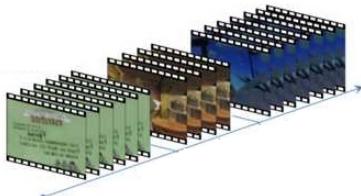
- Phát hiện đoạn cơ sở
- Rút trích khung hình chính
- Tóm tắt video

Xử Lý Video

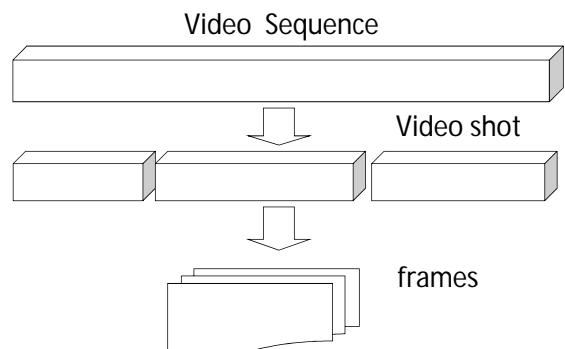
PHÁT HIỆN ĐOẠN CƠ SỞ'

Đoạn cơ sở là gì?

- Đoạn cơ sở là đơn vị cơ bản cấu thành nên video, gồm tập các khung hình liên tiếp mô tả một hoạt động diễn tiến liên tục theo không gian và thời gian.



Đoạn cơ sở là gì?



Đoạn cơ sở là gì?

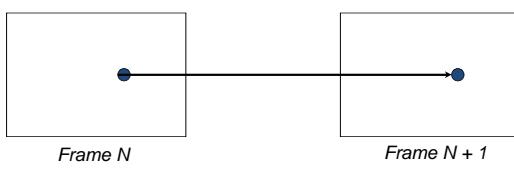
- Biên của đoạn cơ sở là chuyển đổi từ DCS này sang đoạn kế tiếp



Đoạn cơ sở là gì?

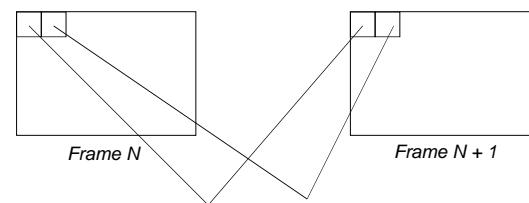
- Quá trình hình thành đoạn cơ sở
 - Con người chỉnh sửa
 - Nội dung thay đổi hoặc sự di chuyển của camera
- Phát hiện đoạn cơ sở là quá trình đi tìm biên chuyển đổi của các đoạn cơ sở

So sánh điểm ảnh



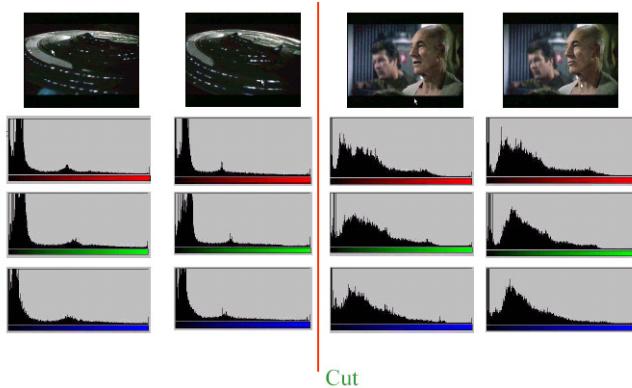
$$D(i,i+1) = \frac{\sum_{x=1}^X \sum_{y=1}^Y |P_i(x,y) - P_{i+1}(x,y)|}{XY}$$

So sánh khối điểm ảnh

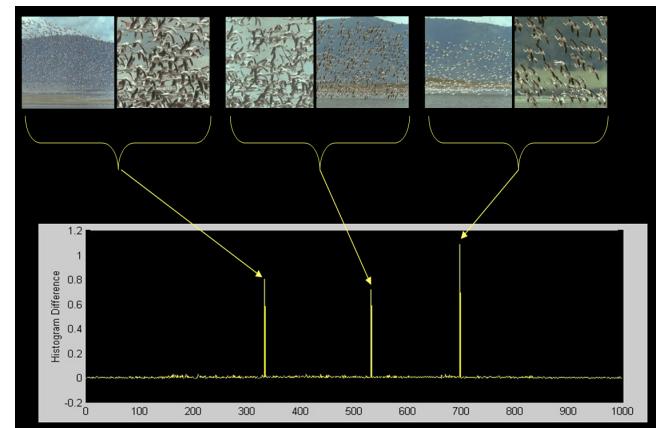


So sánh thống kê của các khối tương ứng
Đếm số lượng khối khác biệt có ý nghĩa

So sánh lược đồ màu



So sánh lược đồ màu



Phân loại chuyển cảnh

- Hard cut
 - Fade
 - Dissolve
 - Wipe
- } Chuyển cảnh đột ngột
} Chuyển cảnh tiệm tiến

Hard Cut

- Sự thay đổi đột ngột nội dung của video hoặc thao tác camera



Thay đổi đột ngột



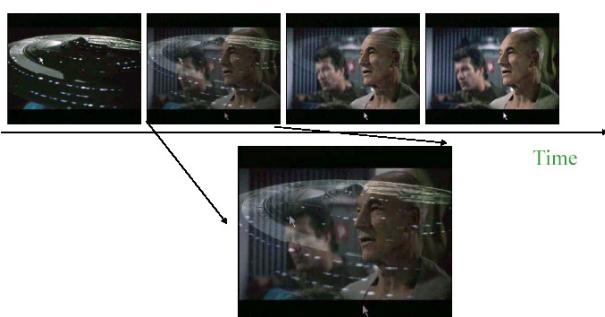
Fade

- Thay đổi dần dần giữa một màn hình và một ảnh cố định



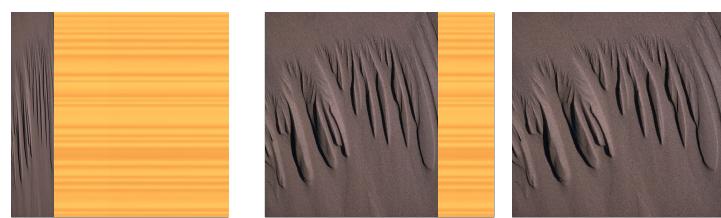
Dissolve

- Sự thay đổi dần dần từ một màn hình này tới màn hình khác

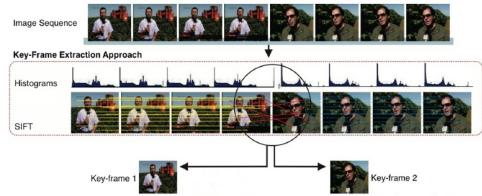
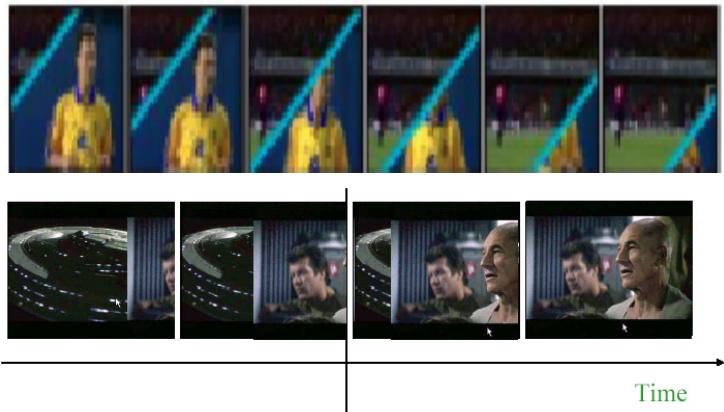


Wipe

- Một đường di chuyển qua màn hình và nội dung mới xuất hiện phía sau



Wipe



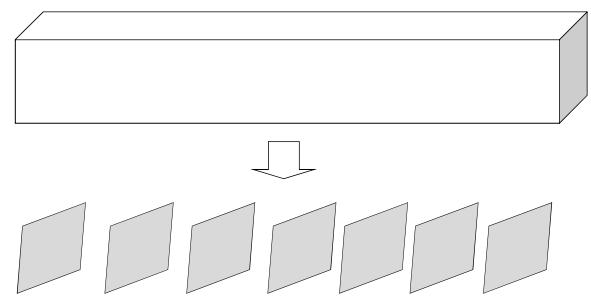
Xử Lý Video

TRÍCH CHỌN KHUNG HÌNH CHÍNH

Trích chọn khung hình chính

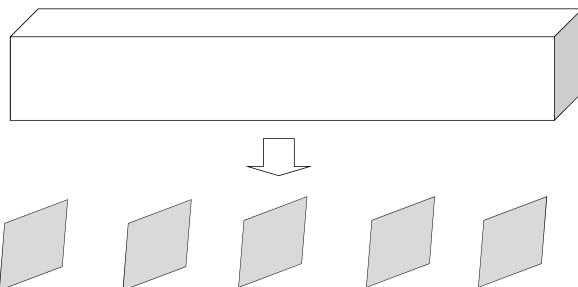


Trích chọn khung hình chính

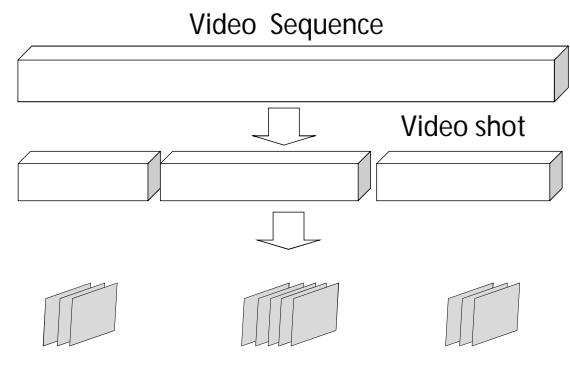


Trích chọn khung hình chính

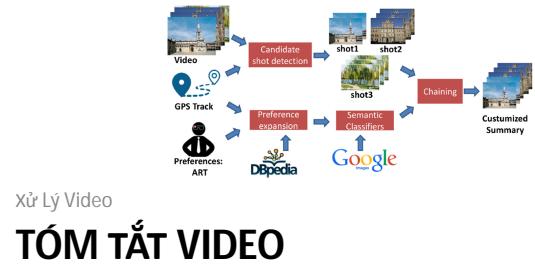
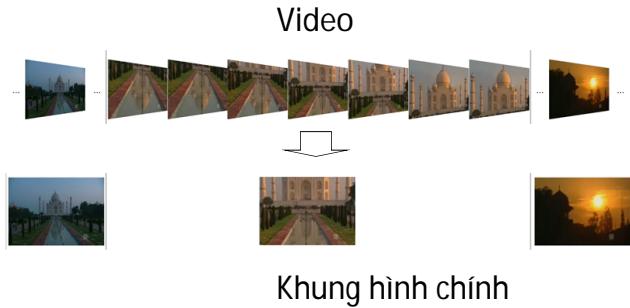
- Lấy mẫu đồng nhất (cứ k khung hình liên tiếp chọn một khung hình làm khung hình chính)



Trích chọn dựa vào phát hiện DCS



Phát hiện khung hình chính



Tóm tắt video

- Tóm tắt video là quá trình trừu tượng hóa các sự kiện chính, cảnh hoặc đối tượng trong một đoạn phim để cung cấp nội dung có thể hiểu được
- Video rất tốn thời gian để xem
- Nhiều video chất lượng thấp
- Sự giá tăng cực lớn nội dung video trong những năm gần đây

Các hướng tiếp cận

- Biểu diễn nội dung bằng khung hình
 - Sử dụng bài toán kích chọn khung hình chính
- Biểu diễn nội dung bằng đoạn cơ sở
 - Sử dụng bài toán phát hiện đoạn cơ sở