

# Thị giác máy tính

Bài 2: Tăng cường chất lượng ảnh – lọc ảnh



2

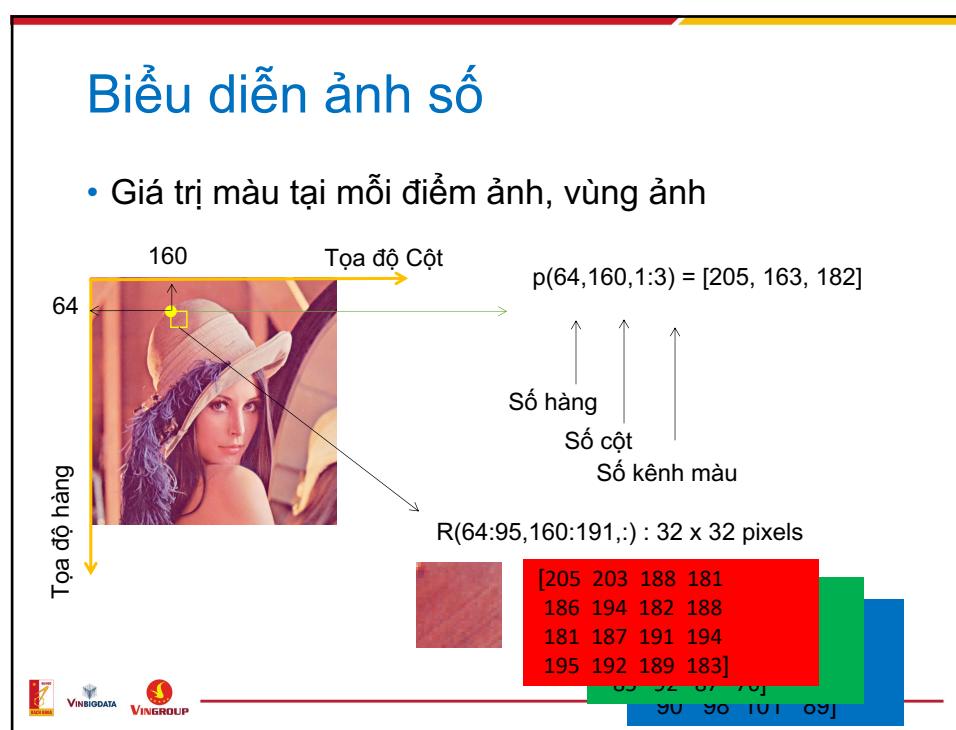
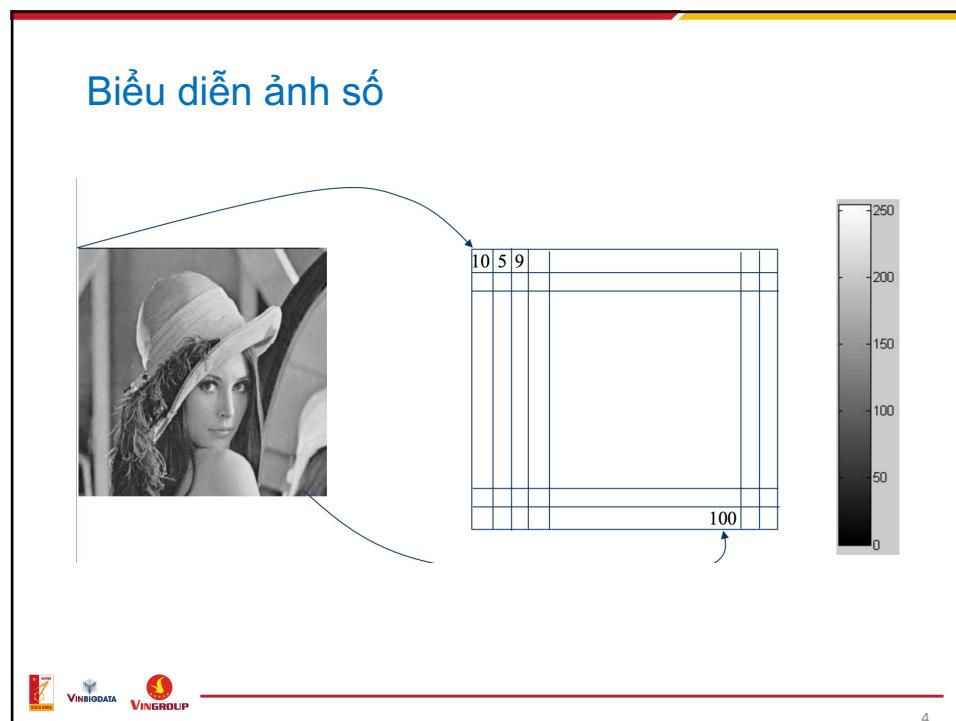
## Nội dung

- Nhắc lại: biểu diễn ảnh số
- Tăng cường độ tương phản trên ảnh
  - Kéo giãn histogram
  - Cân bằng histogram
  - Biến đổi phi tuyến
- Nhân chập và lọc ảnh
  - Nhân chập
  - Một số bộ lọc tuyến tính
- Một số bộ lọc khác



3

3

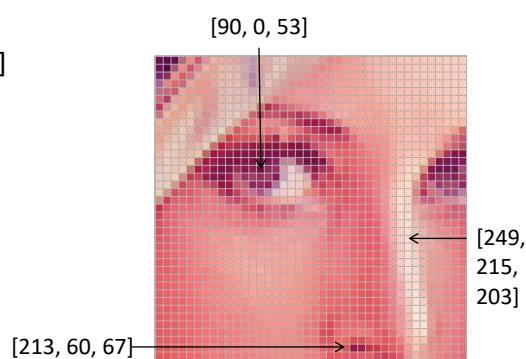


## Biểu diễn ảnh số

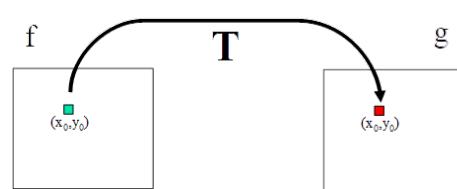
- Ảnh số chứa các điểm ảnh rời rạc
- Giá trị điểm ảnh:

— “grayscale”  
(or “intensity”): [0,255]

- “color”
- RGB: [R, G, B]
  - Lab: [L, a, b]
  - HSV: [H, S, V]



## Biến đổi giá trị điểm ảnh



**Isolated:**  $g(x_0, y_0) = T[f(x_0, y_0)]$



**Local:**  $g(x_0, y_0) = T[f(V)]$   
V: neighbors of  $(x_0, y_0)$



**Global:**  $g(x_0, y_0) = T[f(x, y)]$   
example: FFT

## Nội dung

- Nhắc lại: biểu diễn ảnh số
- Tăng cường độ tương phản trên ảnh
  - Kéo giãn histogram
  - Cân bằng histogram
  - Biến đổi phi tuyến
- Nhân chập và lọc ảnh
  - Nhân chập
  - Một số bộ lọc tuyến tính
- Một số bộ lọc khác

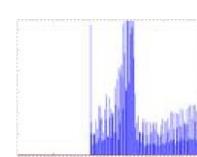
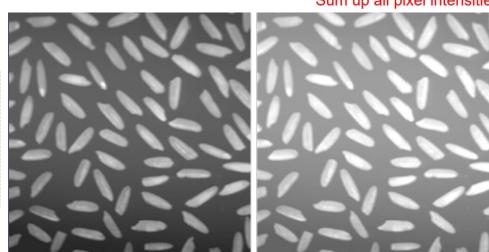
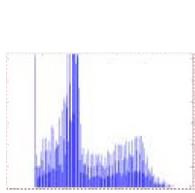
## Độ sáng của ảnh (Brightness)

- Trung bình cường độ sáng của tất cả các điểm ảnh
  - Thể hiện mức sáng, tối trên ảnh

$$B(I) = \frac{1}{wh} \sum_{v=1}^h \sum_{u=1}^w I(u, v)$$

Divide by total number of pixels

Sum up all pixel intensities



## Độ tương phản (contrast)

- Thể hiện sự phân biệt dễ dàng các đối tượng trong ảnh
- Một số cách tính

– Standard deviation of intensity values of pixels in the image

$$C = \sqrt{\frac{1}{wh} \sum_{u=1}^h \sum_{v=1}^w (I(u, v) - mean)^2}$$

– Difference between intensity value maximum et minimum

$$C = \frac{\max(I(u, v)) - \min(I(u, v))}{\max(I(u, v)) + \min(I(u, v))}$$

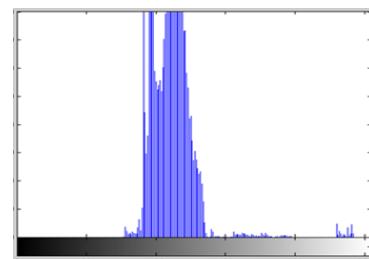
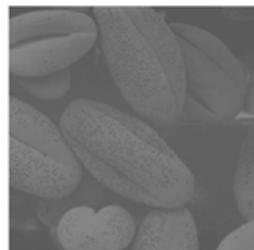


11

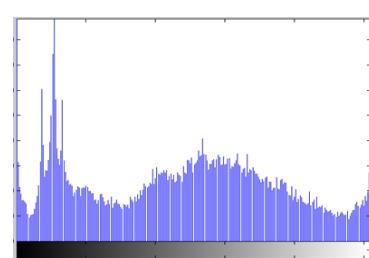
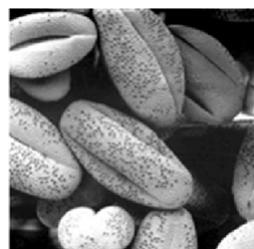
11

## Histogram & độ tương phản

Low- contrast  
image



High- contrast  
image

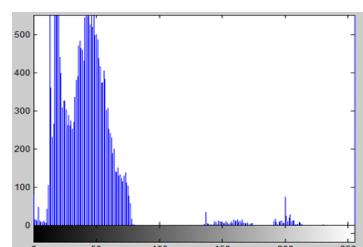
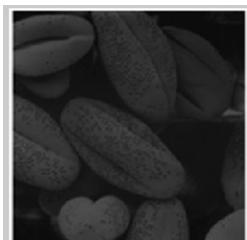


12

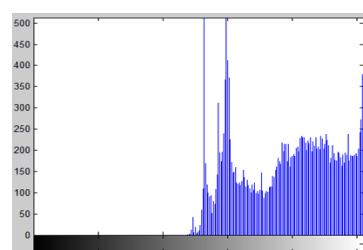
12

## Histogram & độ sáng

Dark image

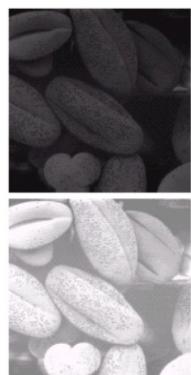


Light image

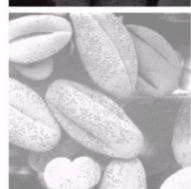
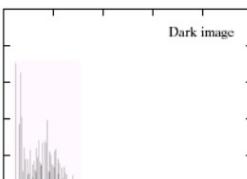


13

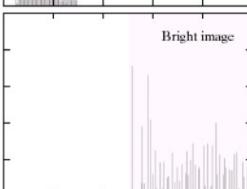
## Ví dụ



Dark image



Bright image



Low-contrast image



High-contrast image



14

## Tăng cường độ tương phản

- Thay đổi giá trị điểm ảnh để có độ tương phản cao hơn
- Một số phương pháp:
  - Kéo giãn dải động ảnh (Linear stretching of intensity range):
    - Linear stretching
    - Linear stretching with saturation
    - Piecewise linear transformation
  - Biến đổi phi tuyến. VD: log, gama correction
  - Cân bằng histogram



15

15

## Phép toán ở mức điểm ảnh

- Giá trị mới của điểm ảnh chỉ phụ thuộc vào giá trị hiện tại của điểm ảnh đó, không phụ thuộc vào các điểm lân cận

$$s = T(r)$$

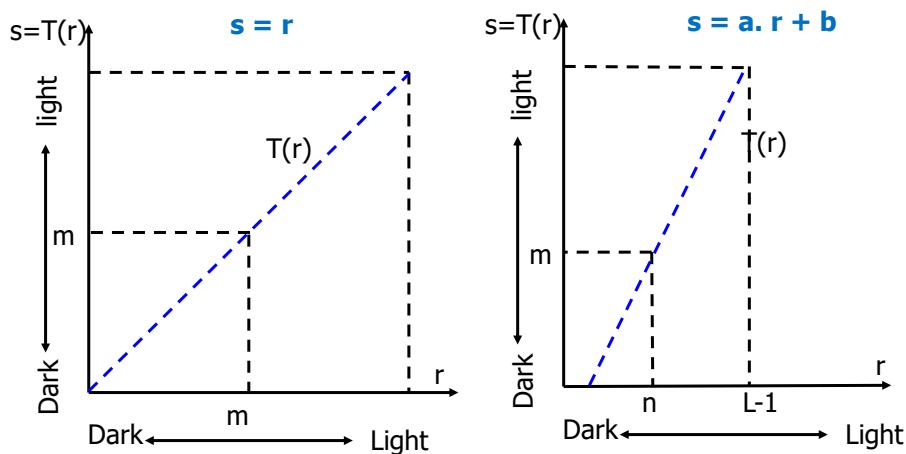
- $r = f(x, y)$
- $s = g(x, y)$
- $T$ : transformation function



16

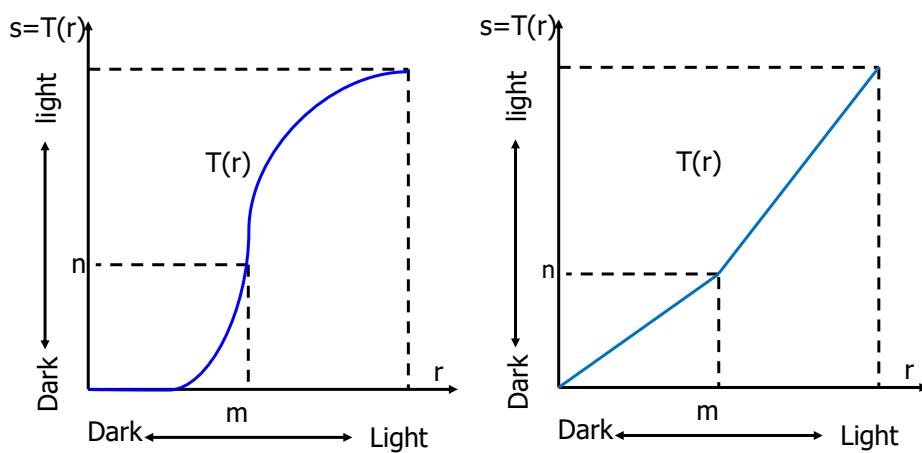
16

## Biến đổi đồng nhất / tuyến tính



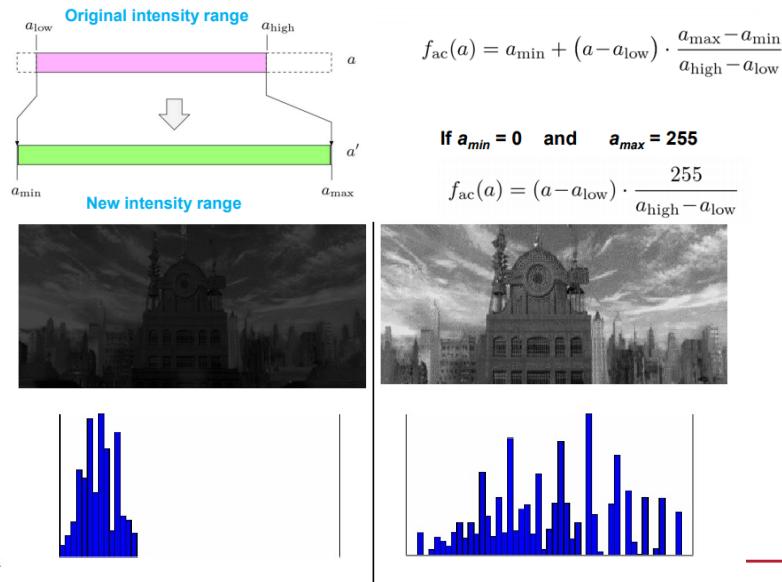
17

## Biến đổi tuyến tính / phi tuyến theo đoạn



18

## Linear stretching

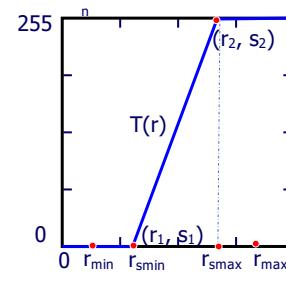
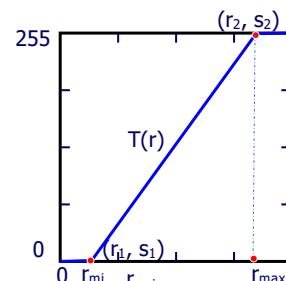


19

19

## Linear stretching

- If  $(r_1, s_1) = (r_{\min}, 0)$  ;  
 $(r_2, s_2) = (r_{\max}, 255)$   
— Linear stretching
- If  $(r_1, s_1) = (r_{s\min}, 0)$ ,  $r_{s\min} > r_{\min}$   
 $(r_2, s_2) = (r_{s\max}, 255)$   $r_{s\max} < r_{\max}$   
— Linear stretching with saturation



20

20

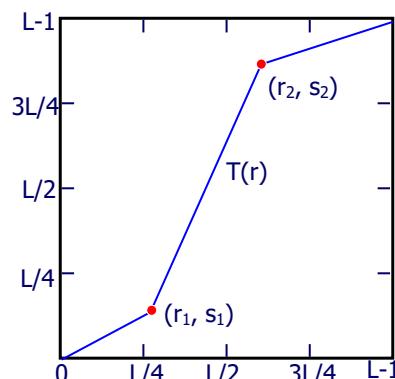
## Piecewise-Linear transformation

- Biến đổi tuyến tính theo đoạn:

- Tăng hoặc giảm độ tương phản ở các khoảng mức xám khác nhau
- Mức độ tăng, giảm khác nhau ở mỗi khoảng giá trị

- VD hàm bên:

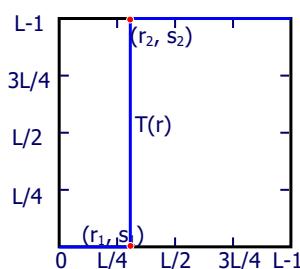
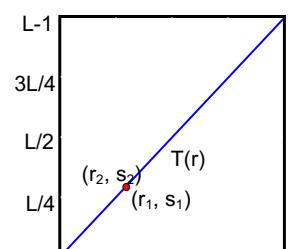
- Độ tương phản giảm trong khoảng
  - từ 0 đến  $r_1$  và
  - từ  $r_2$  đến  $L-1$
- Độ tương phản tăng trong khoảng từ  $r_1$  đến  $r_2$



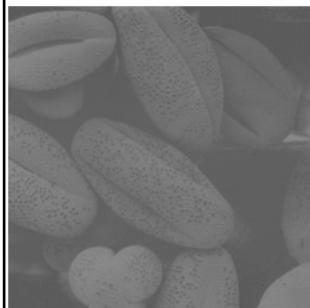
## Piecewise-Linear transformation

- Trường hợp đặc biệt

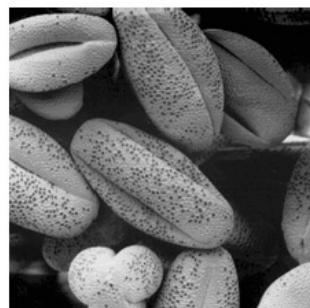
- Nếu  $r_1 = s_1$ ,  $r_2 = s_2$ ,
  - Biến đổi đồng nhất:  $I' = I$
- Nếu  $s_1 = 0$ ,  $s_2 = L-1$ ,
  - Trở thành hàm lấy ngưỡng
  - Đầu ra: ảnh nhị phân



## Piecewise-Linear transformation



Original image, low contrast

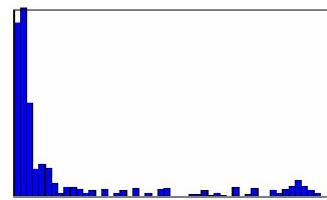
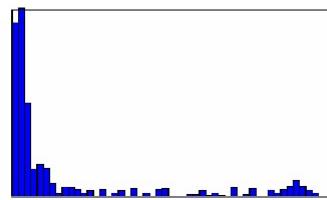


Contrast sketching  
 $(r_1, s_1) = (r_{\min}, 0)$   
 $(r_2, s_2) = (r_{\max}, L-1)$



Thresholding  
 $(r_1, s_1) = (m, 0)$   
 $(r_2, s_2) = (m, L-1)$   
 $m$ : mean of intensities

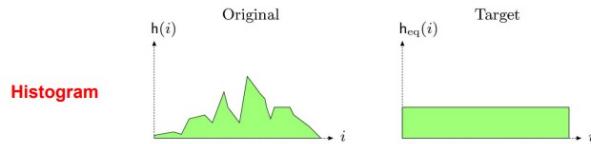
## Linear stretching



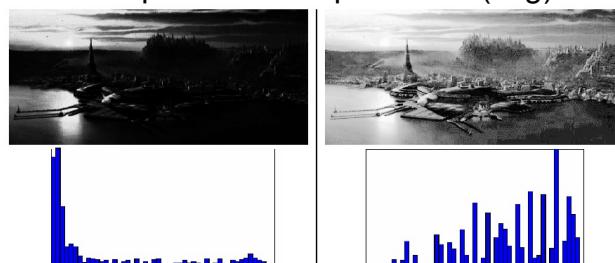
No efficace?

## Cân bằng histogram

- Histogram của ảnh sau thay đổi hướng tới phân phối đều



- Không tham số. OpenCV:cv2.equalizeHist(img)



25

25

## Cân bằng histogram (histogram equalization)

- s1. đếm số điểm ảnh có mức xám k trong ảnh:  $n_k \quad k = 0, 1, \dots, L-1$ .
- s2. tính histogram chuẩn hóa  $p(k)$

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

- s3. Tính histogram tích lũy  $T(k)$ :

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

- s4. Mức xám  $s_k$  của ảnh đầu ra J, tương ứng mức xám k của ảnh đầu vào:

$$s_k = round[(L-1)T(k)]$$

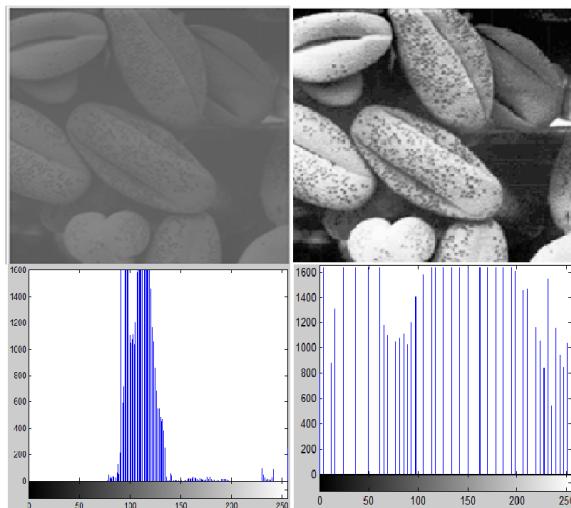
- s5. J được xác định bằng cách thay các giá trị mức xám k trong ảnh đầu vào bằng mức xám  $s_k$



26

26

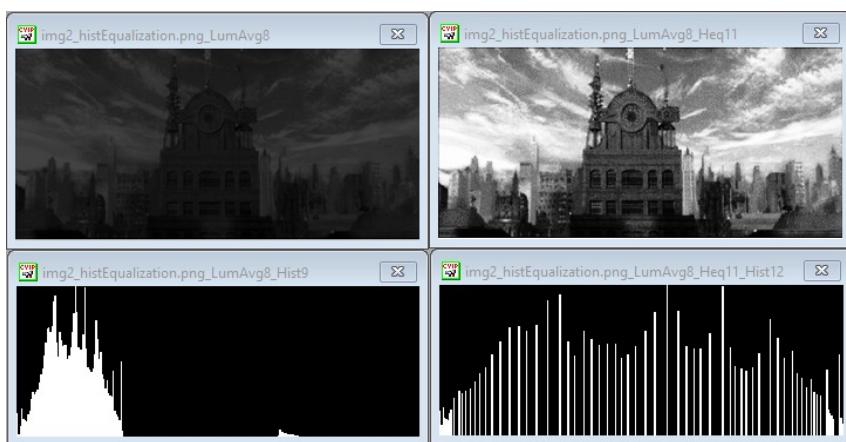
## Histogram equalization



28

28

## Cân bằng histogram



29

29

## Log transformations

- Biến đổi log:

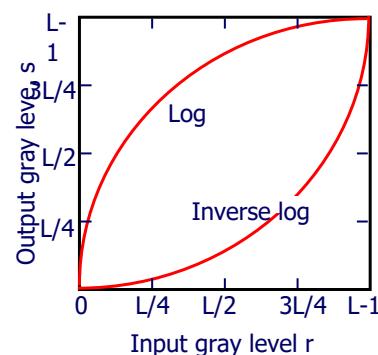
$$s = c \times \log(1 + r)$$

- c: constant
- $r \geq 0$

- Log

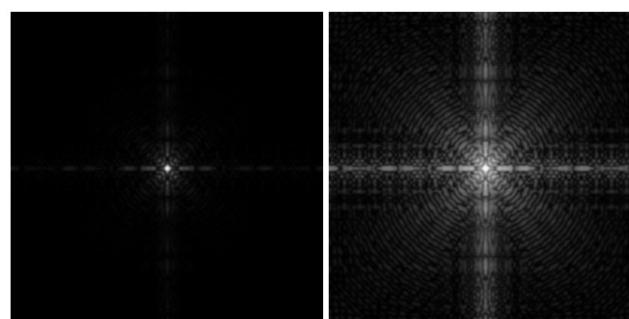
- Các điểm ảnh có **khoảng giá trị hẹp** ở **mức xám thấp** trên ảnh đầu vào thành **khoảng rộng hơn** ở ảnh đầu ra
- Ngược lại với các điểm ảnh có mức xám cao

- Inverse log: ngược lại



## Log transformations

- Làm sáng hơn các điểm có mức xám thấp trong khi  
nén các giá trị có mức xám cao

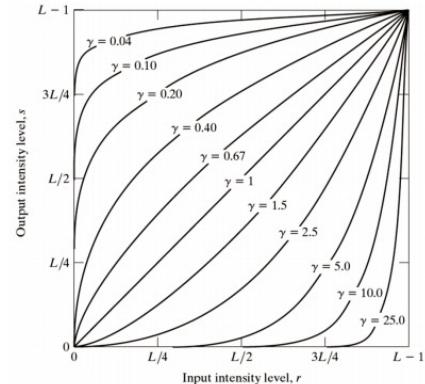


## Power-Law (Gamma) transformations

- The general form of power-law transformation is:

$$s = c \cdot r^\gamma$$

- $\gamma > 1$ : compress values in dark area, while expanding values in light area
- $\gamma < 1$ : expand values in dark area, while compressing values in light area
- $r$ : normalized values to  $[0, 1]$
- $c$ : scaling constant  $c$  corresponding to the bit size used



33

33

## Power-Law (Gamma) transformations

- $\gamma_1 = 1; \gamma_1 = 3; \gamma_2 = 4; \gamma_3 = 5;$



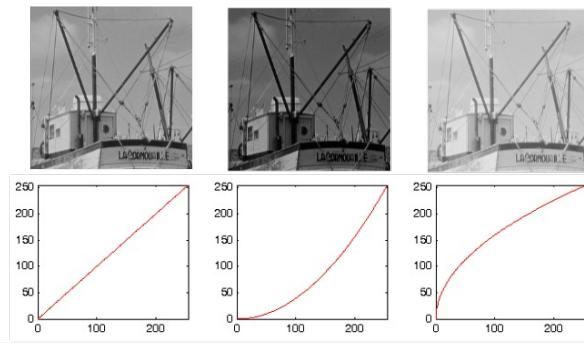
Increasing  
the dynamic  
ranges of  
high  
intensities



34

34

## Gama correction



Ex: Gamma correction (g)

$$\begin{aligned} I_{\text{emitted}} &= k \cdot n^{\gamma} \\ \Rightarrow n' &= n^{\frac{1}{\gamma}} \\ \gamma_{\text{screen}} &\in [1.3, 3.0] \\ \gamma_{\text{eye}} &\approx \frac{1}{2} \text{ to } \frac{1}{3} \end{aligned}$$

In Practice with grey level image :

$$\begin{aligned} k &= 255 \\ n &= r/255 \end{aligned}$$

## Gama correction

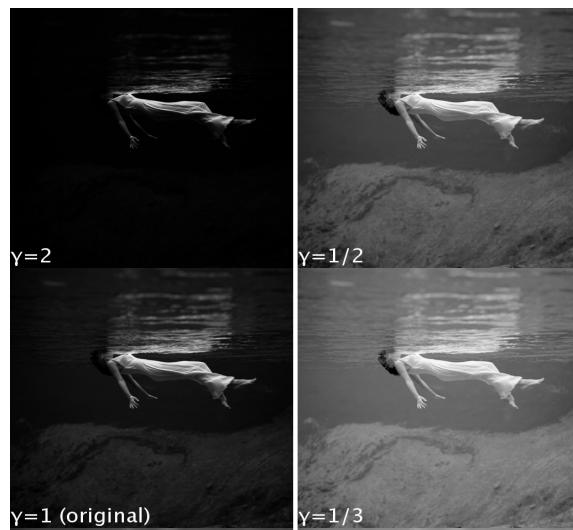


Image: Wikipedia

## Cân bằng histogram

If we take the **same image** with **different contrasts**, histogram equalization will give the **same results** for all images

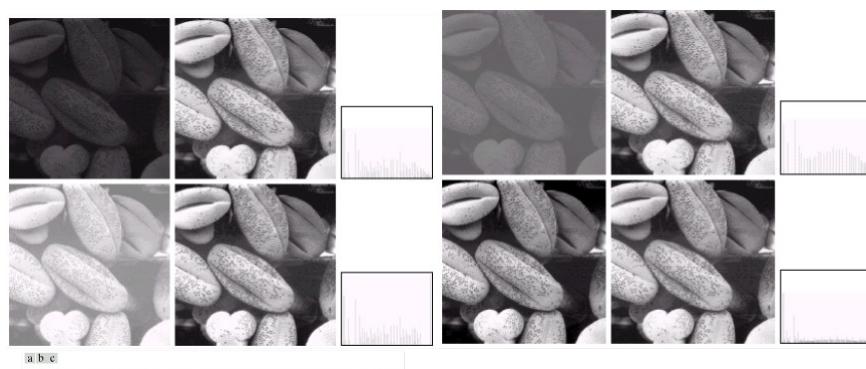


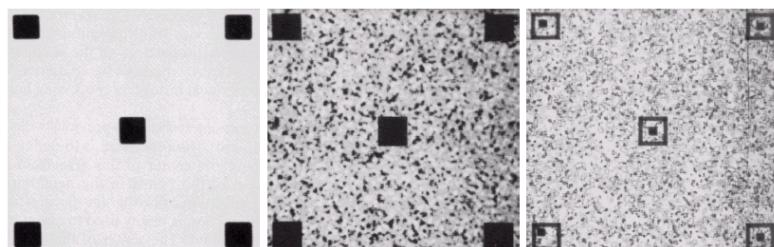
FIGURE 3.17 (a) Images from Fig. 3.15. (b) Results of histogram equalization. (c) Corresponding histograms

## Cân bằng histogram vs. pp khác

- Discussion

- Cân bằng histogram có luôn tốt?
- Cân bằng histo vs Piecewise-Linear transformation ?
- Cân bằng histo vs hiệu chỉnh gamma / log ?

## Cân bằng histogram cục bộ



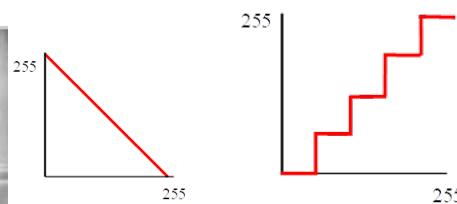
**FIGURE 3.23** (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization using a  $7 \times 7$  neighborhood about each pixel.

## Một số phép biến đổi ảnh khác

$$G(x,y) = F(x,y) + c$$

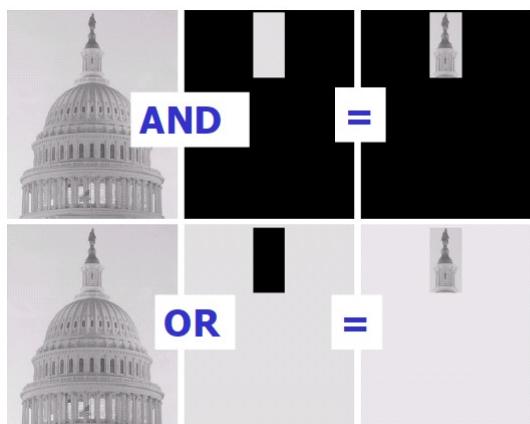


Ảnh gốc  $\rightarrow$  Ảnh sau biến đổi



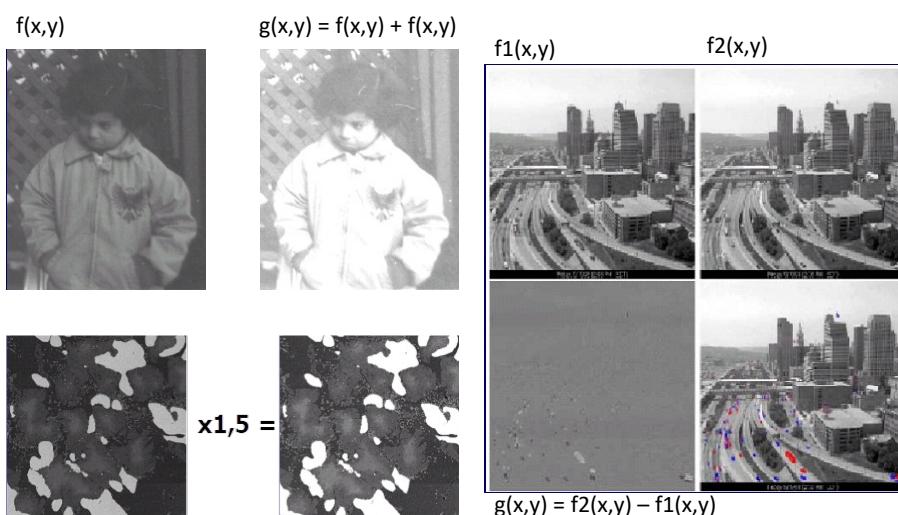
$$\text{Lấy âm bản: } s = (L - 1) - r$$

## Phép toán logic trên ảnh



**FIGURE 3.27**  
 (a) Original image. (b) AND image mask.  
 (c) Result of the AND operation on images (a) and (b). (d) Original image. (e) OR image mask.  
 (f) Result of operation OR on images (d) and (e).

## Phép toán +, -, x trên ảnh



## Tăng cường độ tương phản cho ảnh màu?

- Cân bằng histogram trên ảnh màu ?
  - Thực hiện **trên mỗi kênh màu**
    - Có thể tạo ra những thay đổi màu bất bình thường
    - **KHÔNG KHUYẾN NGHỊ** trừ khi có lý do cụ thể
  - Phương pháp gợi ý
    - Chuyển ảnh sang **không gian màu khác** như Lab, HSL/HSV
    - Thực hiện **cân bằng histogram trên kênh** thể hiện độ sáng tối trên ảnh như L hay V
    - Chuyển ngược lại sang **RGB**



43

43

## Nội dung

- Nhắc lại: biểu diễn ảnh số
- Tăng cường độ tương phản trên ảnh
  - Kéo giãn histogram
  - Cân bằng histogram
  - Biến đổi phi tuyến
- Nhân chập và lọc ảnh
  - Nhân chập
  - Một số bộ lọc tuyến tính
  - Một số bộ lọc khác



44

44

## Lọc ảnh (image filtering)

- Lọc ảnh: Với mỗi điểm ảnh, tính giá trị mới của điểm ảnh dựa trên 1 hàm theo các điểm trong lân cận của nó
  - Lọc trong **miền không gian**:
    - Với mỗi điểm ảnh, tính giá trị mới của điểm ảnh dựa trên **1 hàm theo các điểm trong lân cận** của nó
  - Lọc ảnh trong **miền tần số** ([bài sau](#))
    - Ảnh đầu vào và ra thường có **cùng kích thước**
- Nhân chập: phép lọc tuyến tính, hàm số là tổng có trọng số của các điểm ảnh trong lân cận của điểm ảnh xét.

$$I' = I * K$$

- Có vai trò quan trọng!
  - Tăng cường ảnh: giảm nhiễu, làm mờ, tăng độ tương phản, ...
  - Trích chọn thông tin từ ảnh: Texture, edges, distinctive points, etc.
  - Phát hiện mẫu (template matching)



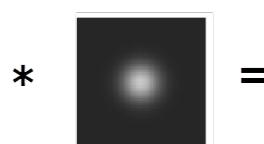
45

45

## Lọc ảnh



Original image



Mask (kernel)



Filtered image



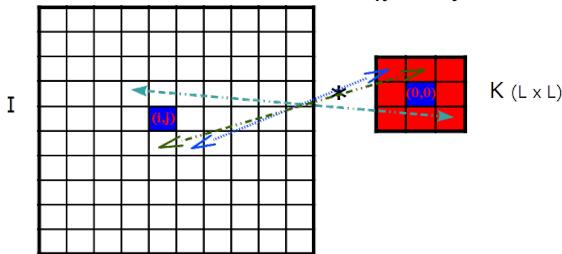
47

47

## Nhân chập (Convolution)

- Convolution: tổng có trọng của các điểm ảnh trong lân cận của điểm ảnh xét

$$f(n, m) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f[k, l] \times h[n - k, m - l]$$



$$I'(i, j) = \sum_{u=-(L-1)/2}^{(L-1)/2} \sum_{v=-(L-1)/2}^{(L-1)/2} I(i-u, j-v) K(u, v)$$



VINBODATA

VINGROUP

48

48

## Nhân chập

- New value of a pixel(i,j) is a weighted sum of its neighbors

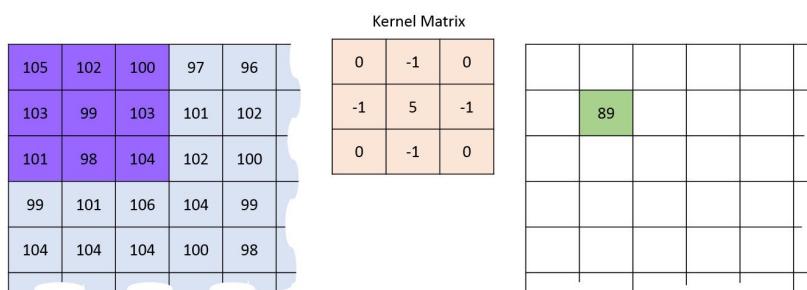


Image Matrix

$$\begin{aligned} & 105 * 0 + 102 * -1 + 100 * 0 \\ & + 103 * -1 + 99 * 5 + 103 * -1 \\ & + 101 * 0 + 98 * -1 + 104 * 0 = 89 \end{aligned}$$

Output Matrix

Source: <http://machinelearningguru.com>

VINBODATA

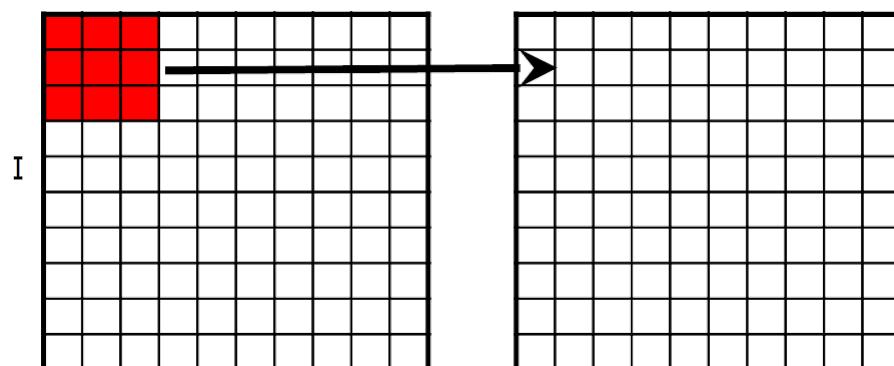
VINGROUP

49

49

## Nhân chập

$$I' = I * K$$

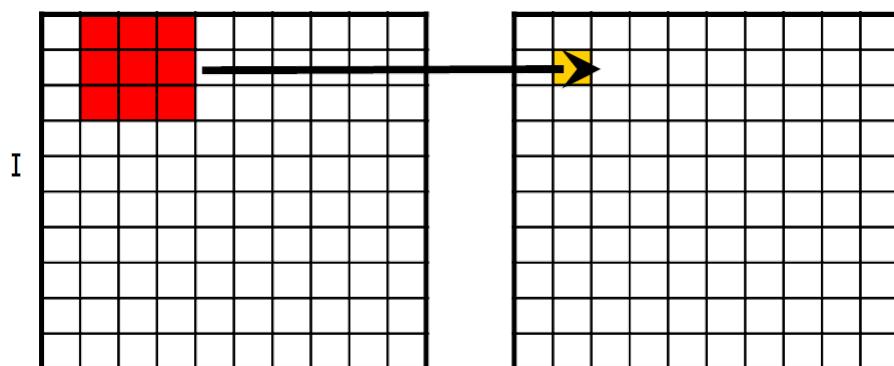


50

50

## Nhân chập

$$I' = I * K$$

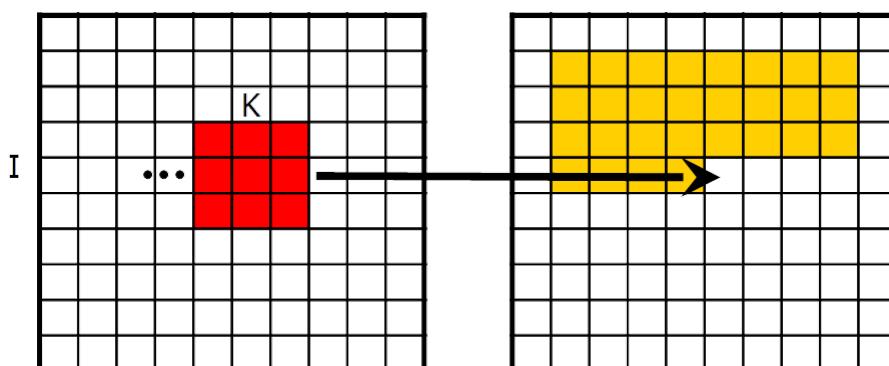


51

51

## Nhân chập

$$I' = I * K$$



52

52

## Nhân chập

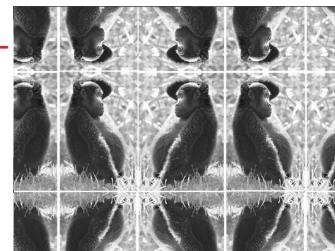
- Vấn đề ở cạnh ảnh?
  - Thêm dòng/cột 0 vào ma trận đầu vào
  - Đối xứng gương:
    - $f(-x,y) = f(x,y)$
    - $f(-x,-y) = f(x,y)$

?	?	?	?	?	?	?	?	?
?								?
?								?
?								?
?								?
?								?
?								?
?								?
?								?

0	0	0	0	0	0	0
0	105	102	100	97	96	0
0	103	99	103	101	102	0
0	101	98	104	102	100	0
0	99	101	106	104	99	0
0	104	104	104	100	98	0

105	105	102	100	97	96	0
105	105	102	100	97	96	0
103	103	99	103	101	102	0
101	101	98	104	102	100	0
99	99	101	106	104	99	0
104	104	104	104	100	98	0



53

53

## Nhân chập

Image Matrix

0	0	0	0	0	0
0	105	102	100	97	96
0	103	99	103	101	102
0	101	98	104	102	100
0	99	101	106	104	99
0	104	104	104	100	98

Kernel Matrix

0	-1	0
-1	5	-1
0	-1	0

Output Matrix

320			
210	89	111	

$$0 * 0 + 0 * -1 + 0 * 0 \\ + 0 * -1 + 105 * 5 + 102 * -1 \\ + 0 * 0 + 103 * -1 + 99 * 0 = 320$$

Image Matrix

105	105	102	100	97	96
105	105	102	100	97	96
103	103	99	103	101	102
101	101	98	104	102	100
99	99	101	106	104	99
104	104	104	104	100	98

Kernel Matrix

0	-1	0
-1	5	-1
0	-1	0

Output Matrix

110			
	89	111	

Source: <http://machinelearningguru.com>

54

54

## Spatial Correlation vs Convolution

Spatial Correlation ( $\star$ ) and Convolution ( $\star$ )

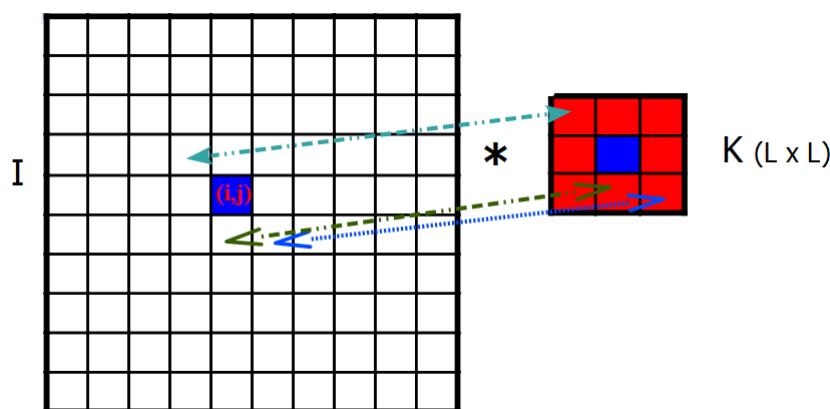
$$w(x, y) \star f(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x-s, y-t)$$

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

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \Rightarrow \begin{bmatrix} 9 & 8 & 7 \\ 6 & 5 & 4 \\ 3 & 2 & 1 \end{bmatrix}$$

## Spatial Correlation vs Convolution

- Correlation



56

56

## Spatial Correlation vs Convolution

- Nếu **mặt nạ đối xứng** thì 2 phép toán này là một
- Correlation:
  - Sử dụng để tìm một mẫu ("template") nào đó có xuất hiện trong ảnh không
  - Không có tính kết hợp → nếu thực hiện tìm mẫu (template matching), thì correlation là đủ
- Convolution:
  - Thường sử dụng lọc ảnh (noise removing, enhancement, ..)
  - Có tính chất kết hợp → hữu ích khi cần thực hiện nhiều bộ lọc liên tiếp:  

$$I * h * g = I * (h * g)$$
- In Matlab: correlation: `filter2`, convolution: `conv2`



57

57

## Một số bộ lọc (Some kernels)

Original image

$*$

0	0	0
0	1	0
0	0	0

Filtered image  
(no change)

Original image

$*$

0	0	0
1	0	0
0	0	0

Filtered image  
(shifted left by 1 pixel)

Source: David Lowe

58

## Một số bộ lọc (Some kernels)

- Nhân chập 2D
  - Chủ yếu được sử dụng để trích chọn đặc trưng trên ảnh
  - Được sử dụng như phép toán trong khái niệm cơ sở của mạng Neuron tích chập: Convolutional Neural Networks (CNNs)
- Mỗi bộ lọc có hiệu ứng riêng và hữu ích cho các nhiệm vụ cụ thể như:
  - Làm mờ (lọc nhiễu),
  - Làm nét biên,
  - Phát hiện cạnh,
  - ....

VINHOMES VINBIODATA VINGROUP

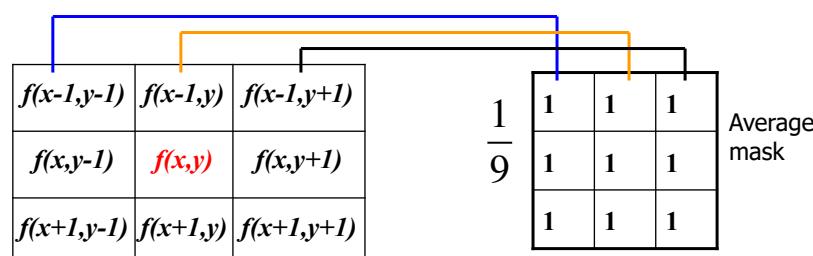
59

## Bộ lọc làm trơn ảnh

- Mục đích
  - Lọc nhiễu
  - Làm trơn ảnh
  - Còn gọi là bộ lọc thông thấp
- Một số bộ lọc thông thấp
  - Bộ lọc trung bình
  - Bộ lọc Gauss
- Để tránh thay đổi độ sáng của ảnh, **tổng các hệ số trong mặt nạ phải = 1**

## Bộ lọc làm trơn ảnh

- Lọc trung bình (**mean filter**):



$$g(x,y) = \frac{1}{9} [f(x-1,y-1) + f(x-1,y) + f(x-1,y+1) + f(x,y-1) + f(x,y) + f(x,y+1) + f(x+1,y-1) + f(x+1,y) + f(x+1,y+1)]$$

## Bộ lọc làm trơn ảnh

- Lọc trung bình (**mean filter**):

– Thay giá trị bởi giá trị trung bình của các hàng xóm

– Ảnh được làm trơn

$$1/9 \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$



Original image

Filtered image  
with box size 5x5Filtered image  
with box size 11x11

## Bộ lọc làm trơn ảnh

- Lọc trung bình có trọng số

– The pixel corresponding to the center of the mask is more important than the other ones.

$$\frac{1}{16} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

$$g(x, y) = \frac{1}{16} [f(x - 1, y - 1) + 2f(x - 1, y) + f(x - 1, y + 1) + 2f(x, y - 1) + 4f(x, y) + 2f(x, y + 1) + f(x + 1, y - 1) + 2f(x + 1, y) + f(x + 1, y + 1)]$$

## Bộ lọc làm trơn ảnh

- Gaussian filter

Gaussian function in 3D

Gaussian image

$G_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$

Rule for Gaussian filter:  
set filter half-width to about  $3\sigma$

0.003	0.013	0.022	0.013	0.003
0.013	0.059	0.097	0.059	0.013
0.022	0.097	0.159	0.097	0.022
0.013	0.059	0.097	0.059	0.013
0.003	0.013	0.022	0.013	0.003

Gaussian filter with size 5 x 5, sigma = 1

VINHOMES VINBIODATA VINGROUP

64

64

## Bộ lọc làm trơn ảnh

- Bộ lọc Gauss:
  - Bộ lọc thông thấp:** loại bỏ các thành phần tương ứng tần số cao trên ảnh
    - Ảnh trơn hơn
    - Tốt hơn bộ lọc trung bình
  - Nhập chập Gauss với chính nó ta được một hàm Gauss
    - Lặp nhận chập với bộ lọc có kích thước nhỏ => thu được kết quả như nhận chập với bộ lọc có kích thước lớn hơn.
    - Nhận chập 2 lần với bộ lọc Gauss có độ rộng  $\sigma$  giống như nhận chập 1 lần với bộ lọc có độ rộng  $\sigma\sqrt{2}$ :  $I^*G_\sigma^*G_\sigma = I^*G_{\sigma\sqrt{2}}^*$
  - Bộ lọc có thể phân tách được:** Hàm Gauss 2D có thể được biểu diễn như tích của 2 hàm 1D: 1 hàm theo x và 1 hàm theo y:
    - $G_\sigma(x,y) = G_\sigma(x).G_\sigma(y)$

VINHOMES VINBIODATA VINGROUP

65

65

## Bộ lọc làm trơn ảnh

- Bộ lọc Gauss



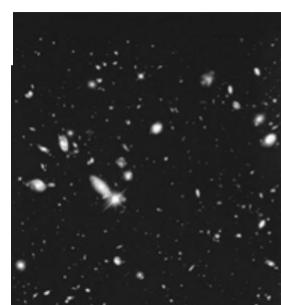
Original image

Filtered image  
with box size 5x5Filtered image  
with box size 11x11

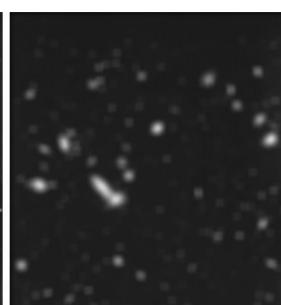
## Bộ lọc làm trơn ảnh

- VD: sử dụng bộ lọc thông thấp để loại bỏ các vùng nhỏ

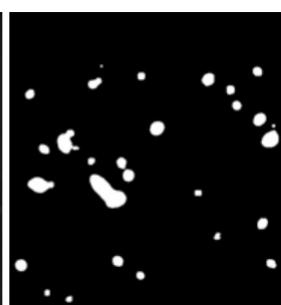
Original image



Average filtering: 15x15



Thresholding of blurring image



## Bộ lọc làm sắc nét cạnh

- Dựa trên ý tưởng của đạo hàm bậc 1 và đạo hàm bậc 2 trên ảnh

$$\frac{\partial f}{\partial x} \approx \begin{cases} f(x+1, y) - f(x, y) \\ f(x, y) - f(x-1, y) \\ 0.5(f(x+1, y) - f(x-1, y)) \end{cases}$$

$$\frac{\partial^2 f}{\partial x^2} \approx f(x+1, y) - 2f(x, y) + f(x-1, y)$$

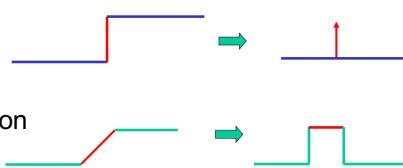


68

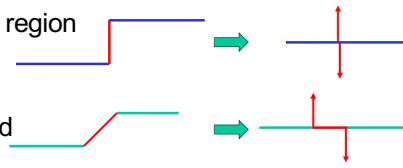
68

## Đạo hàm bậc 1 và bậc 2

- Đạo hàm bậc 1:
  - Zero in flat region
  - Non-zero at start of step/ramp region
  - Non-zero along ramp
  - Strong response for step changes

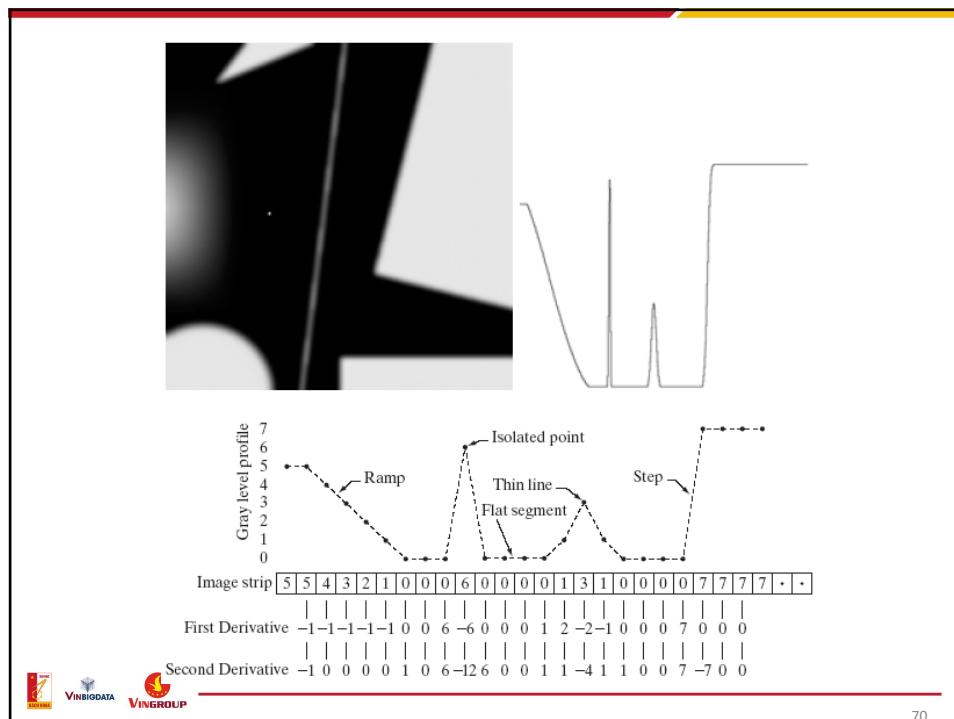


- Đạo hàm bậc 2:
  - Zero in flat region
  - Non-zero at start/end of step/ramp region
  - Zero along ramp
  - Double response at step changes
  - Strong response for fine details and isolated points;



69

69



70

## Đạo hàm bậc 1

- Một số bộ lọc để tính đạo hàm bậc 1 trên ảnh

- Robert

$$\begin{array}{|c|c|} \hline 1 & 0 \\ \hline 0 & -1 \\ \hline \end{array}$$

$$\begin{array}{|c|c|} \hline 0 & 1 \\ \hline -1 & 0 \\ \hline \end{array}$$

- Prewitt

- Ít nhạy cảm với nhiễu

$$\begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

$$\begin{array}{|c|c|c|} \hline -1 & 0 & 1 \\ \hline -1 & 0 & 1 \\ \hline -1 & 0 & 1 \\ \hline \end{array}$$

- Mean + đạo hàm bậc 1

- Sobel:

- Ít nhạy cảm với nhiễu

- Gauss + đạo hàm bậc 1

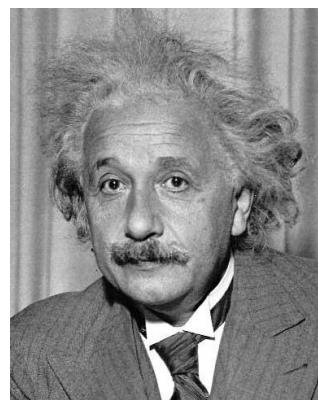
$$\begin{array}{|c|c|c|} \hline -1 & -2 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

$$\begin{array}{|c|c|c|} \hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array}$$

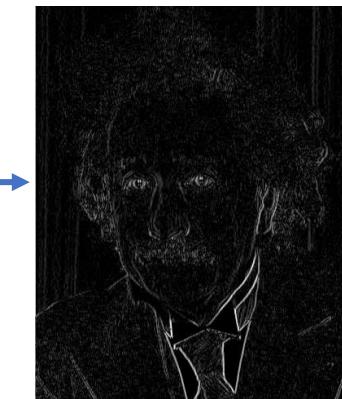
71

## Đạo hàm bậc 1

- Sobel



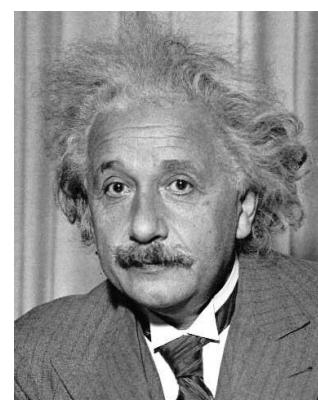
$$\begin{matrix} * & \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} \end{matrix}$$



Vertical Edge  
(absolute value)

## Đạo hàm bậc 1

- Sobel



$$\begin{matrix} * & \begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix} \end{matrix}$$



Horizontal Edge  
(absolute value)

## Đạo hàm bậc 2 - Laplacian filtering

- Bộ lọc Laplace:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y)$$

$$\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$$

## Đạo hàm bậc 2 - Laplacian filtering

- Có thể được tính như sau:

$$\nabla^2 f = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y)$$

- Hoặc

$$\nabla^2 f = 4f(x, y) - [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)]$$

- 90° isotropic filter

0	1	0
1	-4	1
0	1	0

0	-1	0
-1	4	-1
0	-1	0

## Đạo hàm bậc 2 - Laplacian filtering

- Có thể được tính với:

$$\nabla^2 f = [f(x+1, y+1) + f(x+1, y) + f(x+1, y-1) + f(x-1, y+1) + f(x-1, y) + f(x-1, y-1) + f(x, y+1) + f(x, y-1)] - 8f(x, y)$$

1	1	1
1	-8	1
1	1	1

- Hoặc

$$\nabla^2 f = 8f(x, y) - [f(x+1, y+1) + f(x+1, y) + f(x+1, y-1) + f(x-1, y+1) + f(x-1, y) + f(x-1, y-1) + f(x, y+1) + f(x, y-1)]$$

-1	-1	-1
-1	8	-1
-1	-1	-1

- 45° isotropic filter

## Làm sắc nét cạnh (sharpening filter) sử dụng bộ lọc laplacian

$$g(x, y) = \begin{cases} f(x, y) - \nabla^2 f(x, y) & -sign \\ f(x, y) + \nabla^2 f(x, y) & +sign \end{cases}$$

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & +5 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad 90^\circ \text{ isotropic} \quad \begin{bmatrix} -1 & -1 & -1 \\ -1 & +9 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad 45^\circ \text{ isotropic}$$

## Note

$$g(x, y) = \begin{cases} f(x, y) - \nabla^2 f(x, y) \\ f(x, y) + \nabla^2 f(x, y) \end{cases}$$

$$\begin{array}{|c|c|c|} \hline 0 & -1 & 0 \\ \hline -1 & 5 & -1 \\ \hline 0 & -1 & 0 \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} + \begin{array}{|c|c|c|} \hline 0 & -1 & 0 \\ \hline -1 & 4 & -1 \\ \hline 0 & -1 & 0 \\ \hline \end{array}$$
  

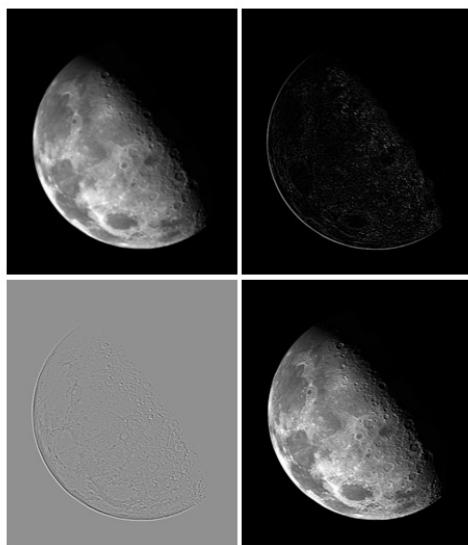
$$\begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline -1 & 9 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array} + \begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline -1 & 8 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array}$$

## Sharpening filter

- a) Original image
- b) Laplacian filtering
- c) Laplacian with scaling
- d) Adding original with Laplacian image

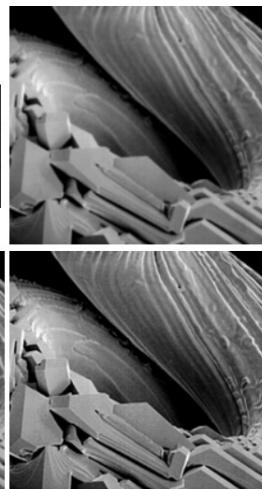
a b  
c d

**FIGURE 3.40**  
 (a) Image of the North Pole of the moon.  
 (b) Laplacian-filtered image.  
 (c) Laplacian image scaled for display purposes.  
 (d) Image enhanced by using Eq. (3.7-5). (Original image courtesy of NASA.)



## Sharpening filter

$$\begin{array}{|c|c|c|} \hline 0 & -1 & 0 \\ \hline -1 & 5 & -1 \\ \hline 0 & -1 & 0 \\ \hline \end{array} \quad \begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline -1 & 9 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array}$$



**FIGURE 3.41** (a) Composite Laplacian mask. (b) A second composite mask. (c) Scanning electron microscope image. (d) and (e) Results of filtering with the masks in (a) and (b), respectively. Note how much sharper (e) is than (d). (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)



80

80

## Nội dung

- Nhắc lại: biểu diễn ảnh số
- Tăng cường độ tương phản trên ảnh
  - Kéo giãn histogram
  - Cân bằng histogram
  - Biến đổi phi tuyến
- Nhận dạng và lọc ảnh
  - Nhận dạng
  - Một số bộ lọc tuyến tính
- Một số bộ lọc khác

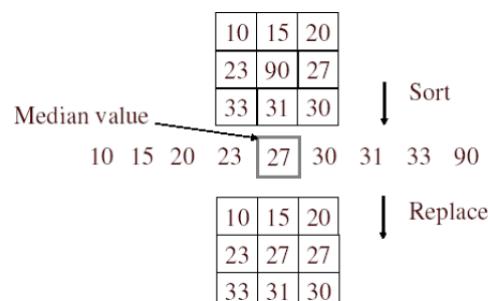


81

81

## Bộ lọc trung vị (median filter)

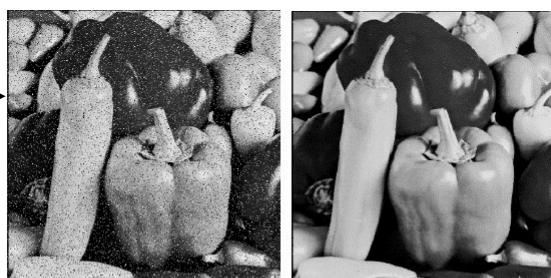
- Không có giá trị mới nào được tính
- Loại bỏ những điểm bất thường:
  - Phù hợp nhiễu xung và nhiễu muối tiêu (salt & pepper)
- Bộ lọc phi tuyến



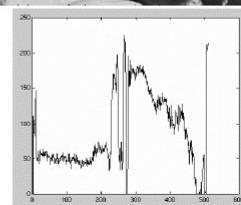
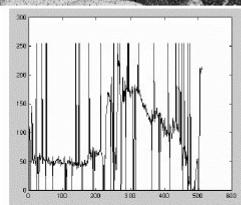
82

## Bộ lọc trung vị (median filter)

Salt and  
pepper  
noise



← Median  
filtered

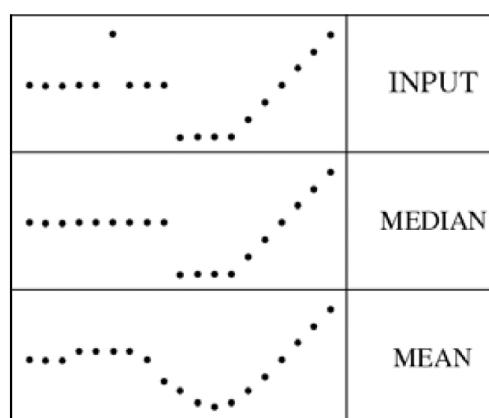


Plots of a row of the image

83

## Bộ lọc trung vị (median filter)

- Median filter is edge preserving



84

## Bộ lọc Max/ Min

- Bộ lọc min: thay giá trị ở giữa bằng **giá trị nhỏ nhất** trong cửa sổ xét



Original Image      with Minimum Filter

- Bộ lọc max: thay giá trị ở giữa bằng **giá trị lớn nhất** trong cửa sổ xét



Original Image      with Maximum Filter

85

## Bộ lọc Max/ Min

- Bộ lọc min: thay giá trị ở giữa bằng **giá trị nhỏ nhất** trong cửa sổ xét

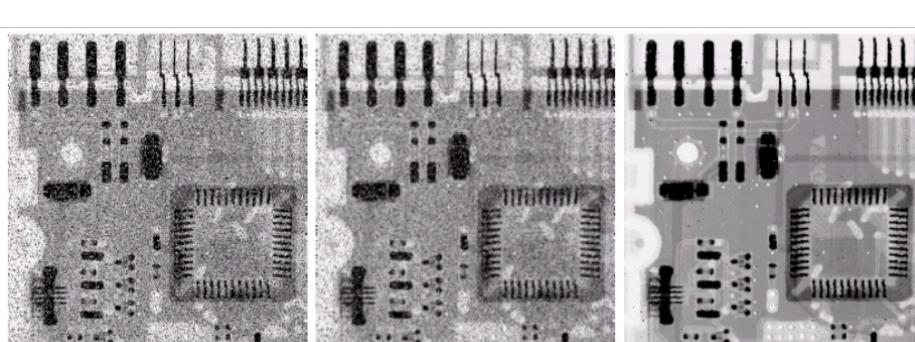


Original Image      with Minimum Filter

- Bộ lọc max: thay giá trị ở giữa bằng **giá trị lớn nhất** trong cửa sổ xét

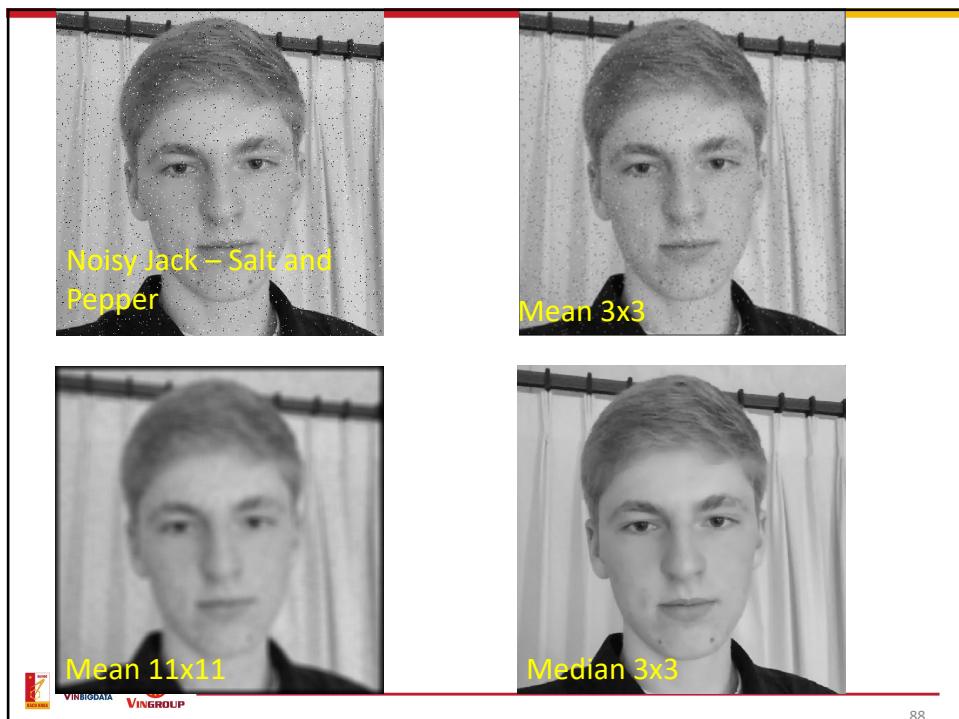


Original Image      with Maximum Filter



a b c

**FIGURE 3.37** (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a  $3 \times 3$  averaging mask. (c) Noise reduction with a  $3 \times 3$  median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)



88

88

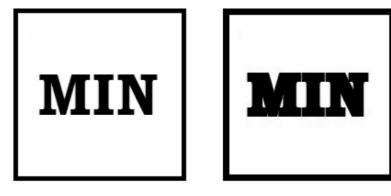


89

89

## Median/max/min: problem?

- Size filter !



90

90

The image features a red vertical banner on the left containing the university's name, logos, and website information. To the right is a photograph of a modern, multi-story white building with many windows and a curved facade.

**VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**  
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

DAI HOC  
BACH KHOA

25  
SOICT

VIETNAM  
UNIVERSITY

VIEN CONG NGHE THONG TIN VÀ TRUYEN THÔNG  
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

soict.hust.edu.vn/ fb.com/groups/soict

91