



TRƯỜNG ĐẠI HỌC ĐIỆN LỰC

NHẬP MÔN XỬ LÝ ẢNH

TS. ĐOÀN THỊ HƯƠNG GIANG

Email: giangdth@epu.edu.vn

Mobil: 0372630593

Chương 2: Tăng cường ảnh qua xử lý lược đồ

- 2.1. Giới thiệu
- 2.2. Lược đồ của ảnh
- 2.3. Cân bằng lược đồ ảnh
- 2.4. Xử lý điểm
- 2.5. Biên đổi cường độ
- 2.6. Mặt phẳng bit

Cải thiện chất lượng ảnh

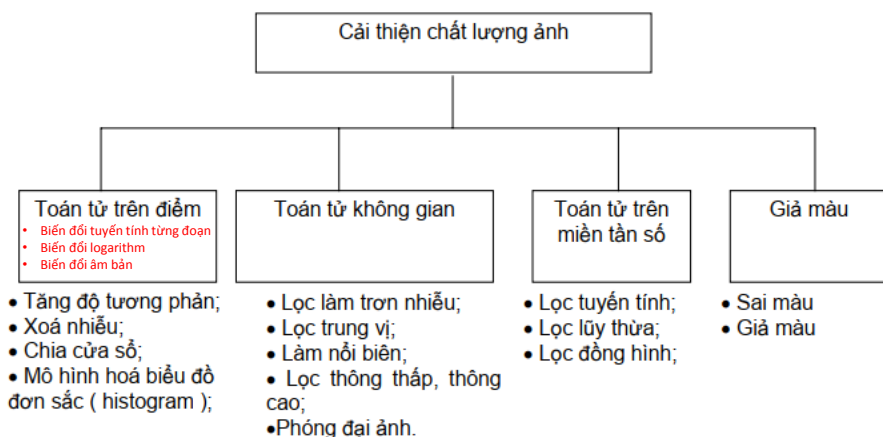
- Tăng cường các đặc điểm của ảnh về mặt cảm nhận hoặc cục bộ như các đường nét, đường biên, hay độ tương phản, màu sắc, ..., lọc nhiễu.
- Tăng cường có hiệu quả cho các bước xử lý tiếp theo như hiển thị ảnh hoặc phân tích ảnh.
- Xử lý cải thiện ảnh chất lượng ảnh không làm tăng thông tin vốn có chứa trong dữ liệu.
- Làm tăng dải động của các thuộc tính của ảnh. Những thuộc tính này giúp cho phân biệt dễ dàng các chi tiết trên ảnh.
- Xử lý ảnh để đầu ra tốt hơn đầu vào cho mục đích nhất định.
- Cải thiện ảnh rất phụ thuộc vào từng ứng dụng cụ thể.

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

3

Phương pháp cải thiện chất lượng ảnh



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

4

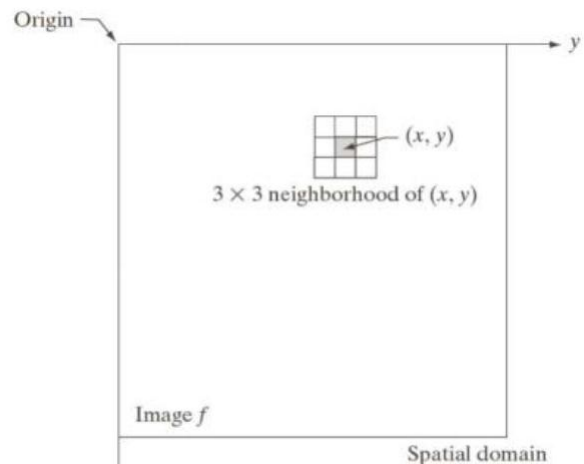
Xử lý trên miền không gian

- Spatial Domain process:

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

Trong đó:

- $f(x, y)$ ảnh gốc
- $g(x, y)$ ảnh sau biến đổi
- T : phép biến đổi ảnh



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

5

Xử lý trên miền không gian

- Nếu xét cửa sổ lân cận: 1×1
 - Phép xử lý trên điểm ảnh
 - Giá trị đầu ra tại một điểm ảnh chỉ phụ thuộc điểm đó, không phụ thuộc vào các điểm khác
- Nếu xét cửa sổ lân cận $w \times w$
 - Cửa sổ lân cận còn gọi là: mặt nạ (mask), nhân (kernel), Cửa sổ (window), bộ lọc (filter), template
 - Giá trị đầu ra tại một điểm phụ thuộc vào các điểm lân cận của nó

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

6

Các phép biến đổi ảnh dựa trên điểm ảnh

- Phép biến đổi âm bản ảnh
- Biến đổi dùng hàm logarit
- Biến đổi dùng hàm mũ
- Biến đổi dựa trên histogram
- Biến đổi dựa trên các phép số học/logic

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

7

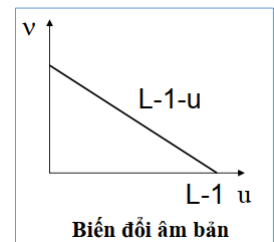
Phép biến đổi âm bản ảnh

- Chuyển đổi dương bản thành âm bản
- Dùng để tăng cường các chi tiết trắng hoặc xám trên nền tối
- Công thức:

$$v = L - 1 - u$$



```
import cv2
img_gray = cv2.imread("images\\1_gray.png")
img_neg = 255-img_gray
cv2.imshow('anh am ban',img_neg)
```



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

8

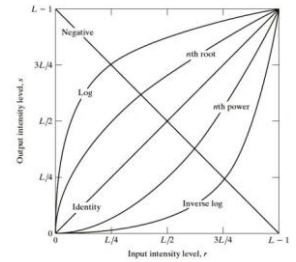
Phép biến đổi log

- Công thức:

$$img2 = c * \log(1 + img1)$$

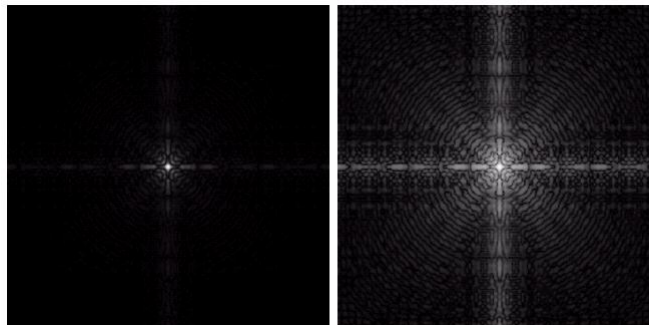
- Tác dụng:

Kéo giãn các giá trị ở vùng tối, thu hẹp các giá trị ở vùng sáng



```
1. import cv2
2. import numpy as np
3. img = cv2.imread("images\\1_gray.png")
4. c = 255 / np.log(1 + np.max(img))
5. img_log = c * (np.log(img + 1))
6. img_log = np.array(img_log, dtype = np.uint8)
7. cv2.imshow('anh bien doi log',img_log)
```

Phép biến đổi log

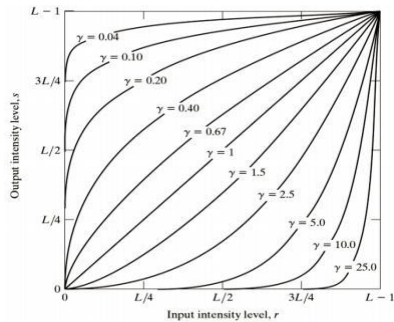


Kéo giãn các giá trị ở vùng tối, thu hẹp các giá trị ở vùng sáng

Phép biến đổi hàm mũ

- Công thức:

$$img2 = c * (img1 + \varepsilon)^\gamma$$



```
import cv2
import numpy as np
img = cv2.imread("images\\1_gray.png")
c, e, gamma = 0.5, 0.01, 0.2
img_exp = c*(255*(img/255+e)**gamma
img_exp = np.array(img_exp, dtype = np.uint8)
cv2.imshow('anh bien doi ham mu',img_exp)
```



Ảnh gốc

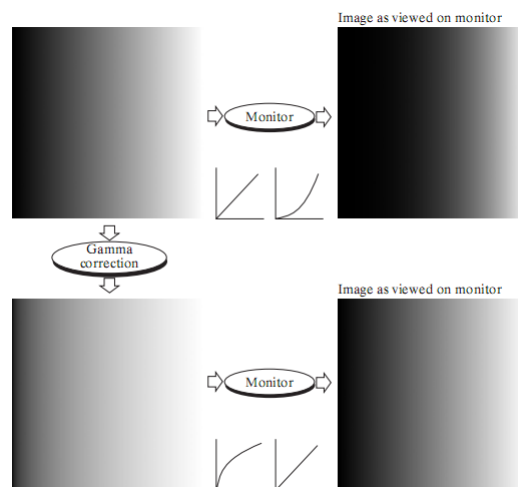
Ảnh biến đổi hàm mũ

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

11

Gama correction



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

12

Phép biến đổi hàm mũ

- $c = 1$; $\gamma_1 = 2$; $\gamma_2 = 4$; $\gamma_3 = 6$;

```
import cv2
import numpy as np
img = cv2.imread("images\\1_gray.png")
c, e, gamma = 1, 0, 3
img_exp = c*(255*(img/255+e)**gamma)
img_exp = np.array(img_exp, dtype = np.uint8)
cv2.imshow('anh bien doi ham mu',img_exp)
```



Ảnh gốc



Gamma=2



Gamma=4



Gamma=6

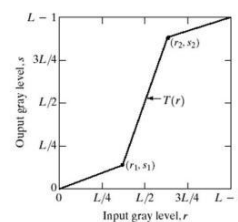
09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

13

Phép biến đổi tuyến tính từng khúc

$$\text{Contrast} = (I_{\max} - I_{\min}) / (I_{\max} + I_{\min})$$



Original

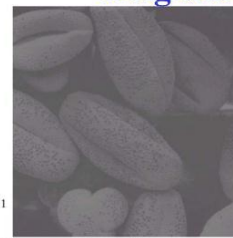


FIGURE 3.10
Contrast stretching. (a) Form of transformation function. (b) A low-contrast image. (c) Result of contrast stretching. (d) Result of thresholding. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)

C. S.



THR.



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

14

Phép biến đổi tuyến tính từng khúc - code

```
import cv2
import numpy as np
def changePixelVal(img, r1, s1, r2, s2):
    if (0 <= img and img <= r1):
        return (s1 / r1)*img
    elif (r1 < img and img <= r2):
        return ((s2 - s1)/(r2 - r1)) * (img - r1) + s1
    else:
        return ((255 - s2)/(255 - r2)) * (img - r2) + s2
img = cv2.imread('images\\1_gray.png')
r1, s1, r2, s2 = 70, 0, 140, 255
vec = np.vectorize(changePixelVal)
img1 = vec(img, r1, s1, r2, s2)
cv2.imshow('anh bien doi tung khuc',img1)
```



Ảnh gốc



Ảnh biến đổi tuyến tính từng khúc

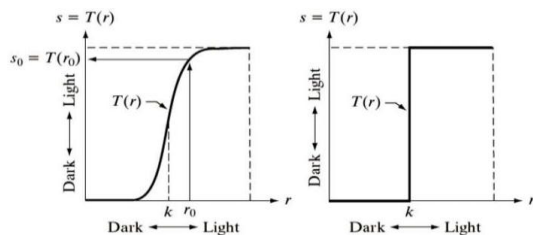
09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

15

Biến đổi tăng độ tương phản

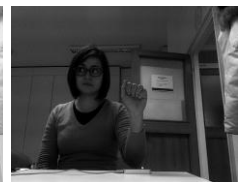
```
import cv2
import numpy as np
gamma = 0.5
img = cv2.imread("images\\1_gray.png")
img_constr = np.power(img, gamma)
max_val = np.max(img_constr.ravel())
img_constr = img_constr/max_val * 255
img_constr = img_constr.astype(np.uint8)
```



Ảnh gốc



Gama=0.5



Gama=1.5

09/02/2023

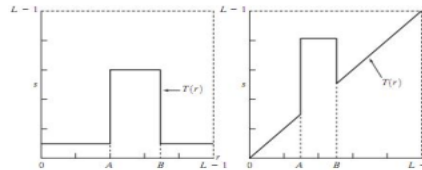
TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

16

Gray level slicing

- Tăng cường mức xám ở một dải cố định $[A, B]$

```
import cv2
import numpy as np
img=cv2.imread('images\\1_gray.png',0)
h,w=img.shape
img_gls=img.copy()
A, B =50, 150
for i in range(0,h):
    for j in range(0,w):
        if(img[i][j]>A and img[i][j]<B):
            img_gls[i][j]=255
cv2.imshow('gray level slicing',img_gls)
```



Ảnh gốc



Ảnh Gray level slicing

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

17

Bit-plan slicing

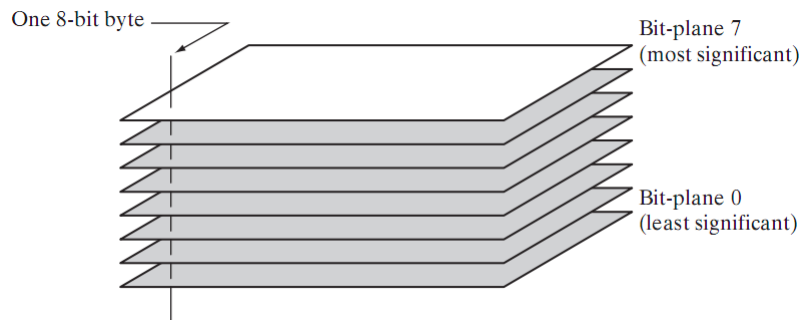
- Với ảnh 8bit: mỗi pixel được biểu diễn bằng 8 bit
- Tưởng tượng mỗi ảnh là tổng hợp của 8 mặt phẳng 1 bit (1bit - plan): từ plan 0 đến plan 7
 - Plan 0: chứa tất cả các bit thấp nhất trong các byte pixel trong ảnh
 - ...
 - Plan 7: chứa tất cả các bit cao nhất trong các byte pixel trong ảnh

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

18

Bit-plan slicing (tt)



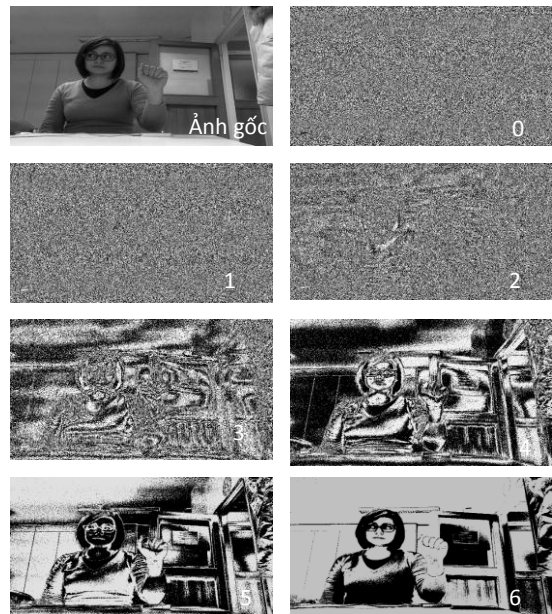
09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

19

Bit-plan slicing (tt)

```
import cv2
import numpy as np
img=cv2.imread('images\\1_gray.png',0)
img = np.array(img)
for k in range(7):
    plane = np.full((img.shape[0], img.shape[1]), 2 ** k, np.uint8)
    res = cv2.bitwise_and(plane, img)
    img_bit_slice = (res * 255).astype(np.uint8)
    cv2.imshow(f'anh bien doi bit plan slicing {k}',img_bit_slice)
cv2.imshow('anh goc',img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

20

Một số phép xử lý dựa trên điểm ảnh

- Bài tập: Cài đặt các phép biến đổi dựa trên điểm ảnh trên bằng Matlab/Python

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

21

Phép biến đổi dựa trên histogram

- Histogram là gì?
 - Phân bố các mức xám trong ảnh
 - Là đánh giá gần đúng hàm mật độ phân bố xác suất
 - Xác suất xuất hiện của một giá trị mức xám trên ảnh
- Histogram của ảnh đa mức xám: $[0, L-1]$ là hàm rời rạc:

$$P_u(x_i) = \frac{h(x_i)}{\sum_{i=1}^{L-1} h(x_i)}$$

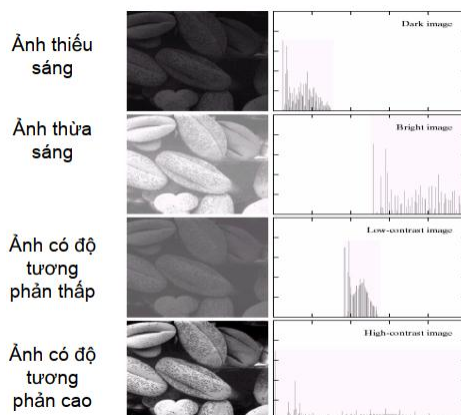
Trong đó $H(u)$ là số lượng điểm ảnh có mức xám bằng u

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

22

Ví dụ ảnh với các Histogram tương ứng



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

23

Các phương pháp cải thiện chất lượng ảnh dựa trên xử lý histogram

- Cân bằng histogram (histogram equalization)
- Biến đổi histogram (histogram modification)
- Đặc trưng hóa histogram (histogram specification)

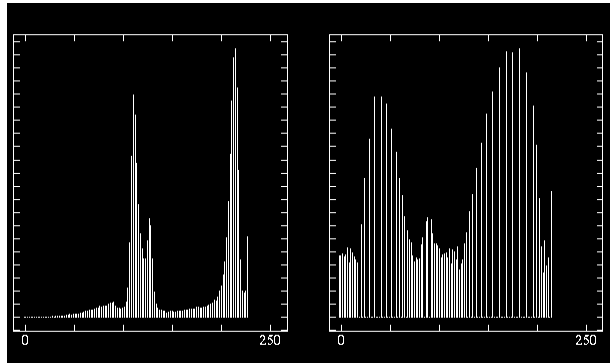
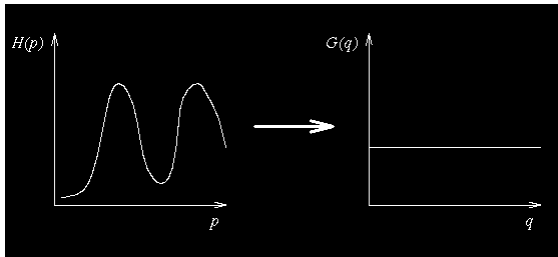
09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

24

Cân bằng histogram

- Ánh xạ độ chói của ảnh vào vùng giá trị mới sao cho histogram mới nhận được có dạng phân bố đều.



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

25

Hàm tính histogram trong python

- Cú pháp:

```
cv2.calcHist(images, channels, mask, histSize, ranges[, hist[, accumulate]])
```

- Trong đó:

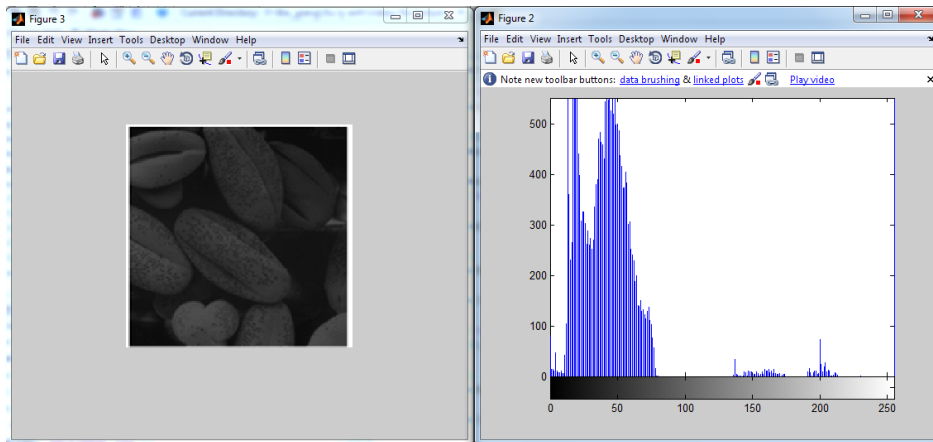
- **images** : ảnh đầu vào có định dạng uint8 hoặc float32.
- **channels** : Số kênh của ảnh. Ảnh xám (grayscale) thì đặt = [0]; Ảnh màu thì cần tính hist cho từng kênh màu [0], [1], [2] để tính cho kênh B, G, R
- **mask** : ảnh mặt nạ. Khi muốn tính hist cho cả ảnh thì đặt = "None".
- **histSize** : số BIN. Lớn nhất là 256.
- **ranges** : Giải giá trị [0,256]

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

26

Histogram của ảnh tối

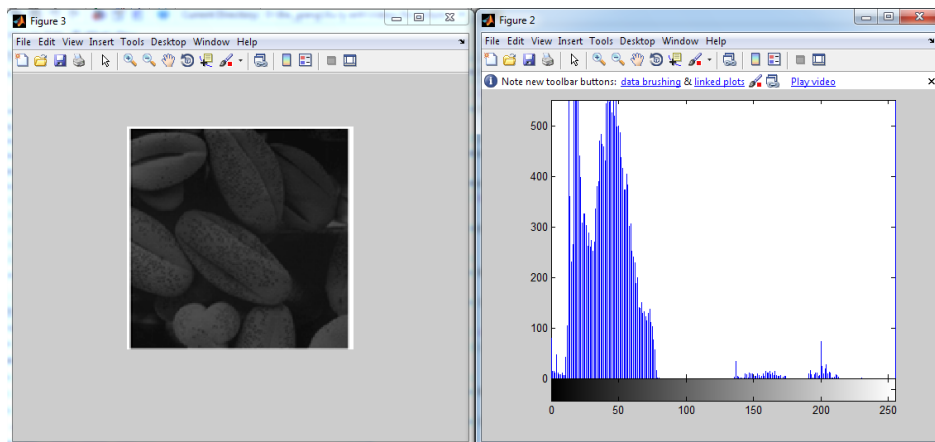


09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

27

Histogram của ảnh sáng

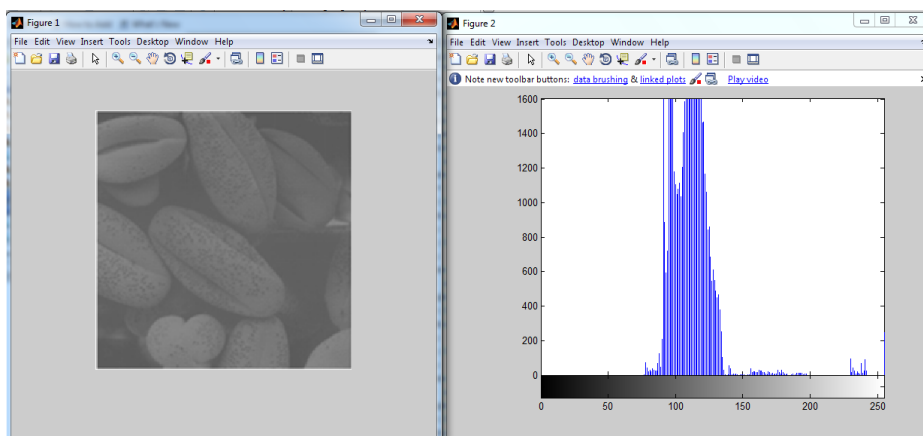


09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

28

Histogram của ảnh có độ tương phản thấp

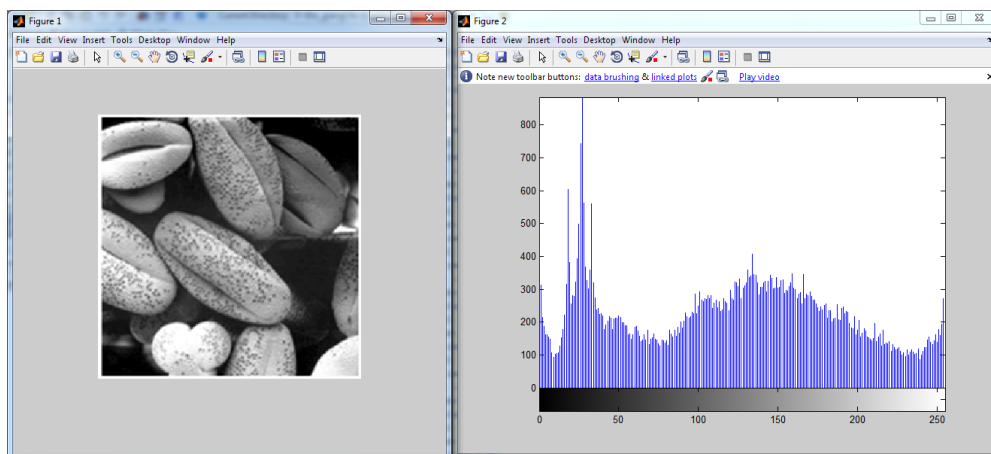


09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

29

Histogram của ảnh có độ tương phản cao



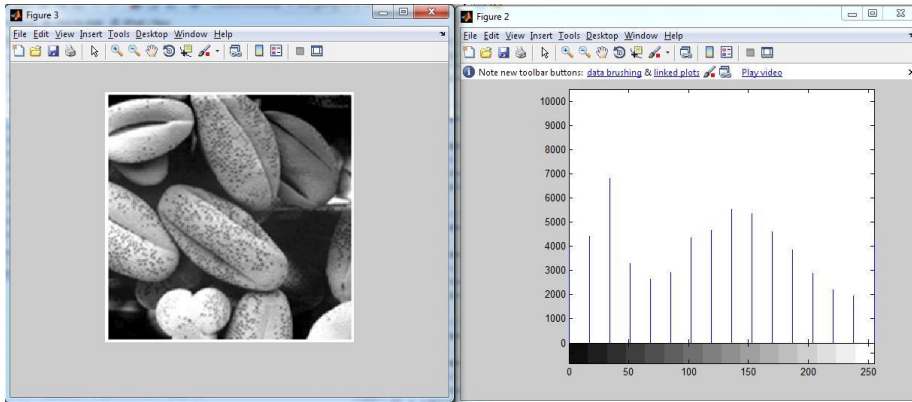
09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

30

Số bins?

- Mặc định trong ảnh đa mức xám: 256 bins



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

31

Phép cân bằng histogram

- Xét trên miền liên tục có:
 - r : biến ngẫu nhiên thể hiện các giá trị cấp xám trong ảnh ban đầu $0 \leq r \leq 1$
 - s : biến ngẫu nhiên thể hiện giá trị cấp xám trong ảnh biến đổi
 - Cần tìm phép biến đổi:

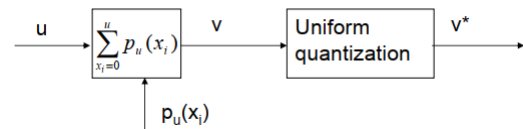
$$s = T[r]$$

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

32

Phép cân bằng histogram



- Ảnh ban đầu U có u nhận các giá trị mức xám x_i , $i = 0, 1, \dots, L - 1$;
- Xác suất xuất hiện các mức xám trong ảnh ban đầu:

$$P_u(x_i) = \frac{h(x_i)}{\sum_{i=1}^{L-1} h(x_i)}$$

- Biến ngẫu nhiên phân bố đều v:

$$v = f(u) = \sum_{x_i}^u P_u(x_i)$$

- Giá trị điểm ảnh mới:

$$v^* = \text{Int} \left[\frac{v - v_{\min}}{1 - v_{\min}} (L - 1) + 0.5 \right]$$

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

33

Hàm cân bằng ảnh bằng histogram trong python

cv2.equalizeHist()

```
import cv2
import numpy as np
img = cv2.imread("images\\1_gray.png",0)
img_equ = cv2.equalizeHist(img)
cv2.imwrite("images\\anh_equalizeHist.png",img_equ)
cv2.imshow("Anh can bang histogram", img_equ)
```



Ảnh gốc



Ảnh cân bằng histogram

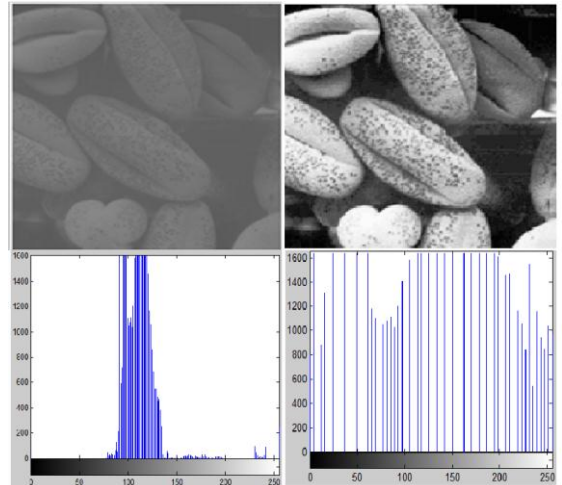
09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

34

Histogram equalization (8)

- So sánh:
 - Cân bằng histogram và phép biến đổi tăng độ tương phản bằng tuyến tính từng khúc
 - Cân bằng histogram trường hợp nào cũng tốt?

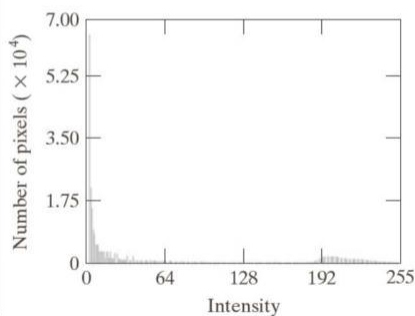


09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

35

Histogram equalization (9)



a b

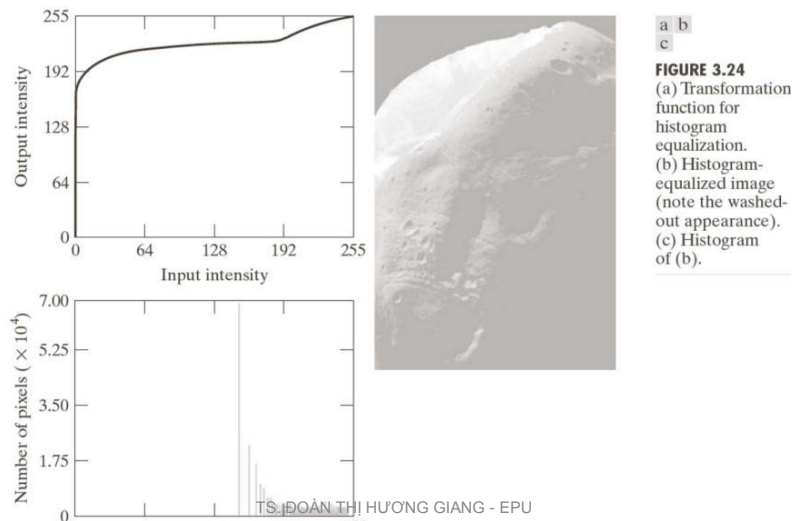
FIGURE 3.23
(a) Image of the Mars moon Phobos taken by NASA's *Mars Global Surveyor*. (b) Histogram. (Original image courtesy of NASA.)

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

36

Histogram equalization (10)



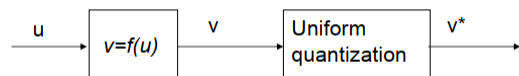
09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

37

Mở rộng phép cân bằng histogram

- Trong cân bằng histogram:



$$v = f(u) = \sum_{x_i}^u P_u(x_i)$$

- Trong các trường hợp khác: $v = f(u)$, trong đó hàm f có thể là:

$$v = f(u) = \log(1 + u), \quad u \geq 0$$

$$v = f(u) = u^{1/n}, \quad u = 2, 3, \dots$$

$$v = f(u) = \frac{\sum_{x_i=0}^u p_u^{1/n}(x_i)}{\sum_{x_i} p_u^{1/n}(x_i)}, \quad n = 2, 3, 4, \dots$$

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

38

Histogram matching/specification

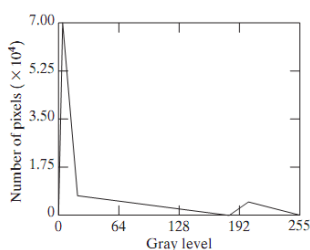
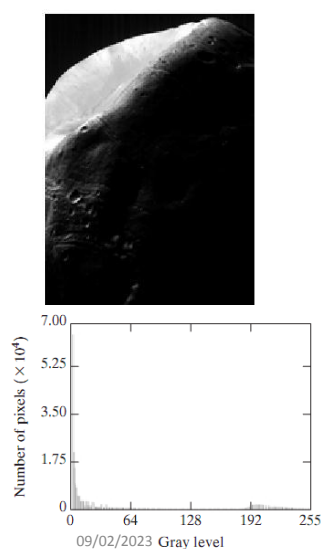
- Nhiều trường hợp: histogram phân bố đều không cho kết quả ảnh tốt nhất
 - Chỉ định histogram có hình dạng tốt
 - Biến đổi ảnh theo hình dạng histogram cho trước
- ➔ Bài toán histogram matching hay histogram specification

09/02/2023

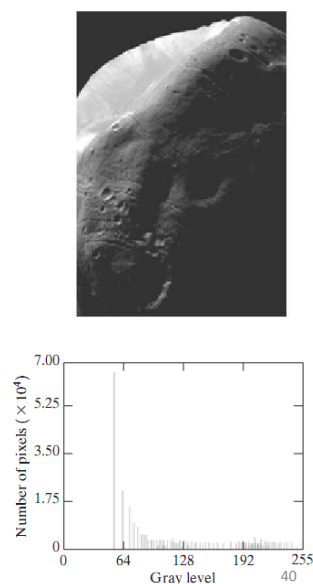
TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

39

Histogram matching



Ứng dụng: Hiệu chỉnh các ảnh chụp cùng một cảnh, nhưng được chụp bởi các camera, sensor khác nhau



TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

Histogram matching (1)

- Ý tưởng trên miền liên tục

$$\left. \begin{aligned} p_r(r) &\xrightarrow{?} p_z(z) \\ s = T(r) &= (L-1) \int_0^r p_r(w) dw \\ G(z) &= (L-1) \int_0^z p_z(t) dt = \mathbf{s} \end{aligned} \right\} \Rightarrow z = G^{-1}[T(r)] = G^{-1}[s]$$

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

41

Histogram matching (2)

- Áp dụng trên miền rời rạc (ảnh số)
- Từ r_k tính s_k :

$$\begin{aligned} s_k = T(r_k) &= \sum_{j=0}^k p_r(r_j) \\ &= \sum_{j=0}^k \frac{n_j}{n} \quad k = 0, 1, 2, \dots, L-1
 \end{aligned}$$

- Từ z_k tính v_k

$$v_k = G(z_k) = \sum_{i=0}^k p_z(z_i) = s_k \quad k = 0, 1, 2, \dots, L-1$$

- Từ đó tính z_k

$$z_k = G^{-1}[T(r_k)] \quad k = 0, 1, 2, \dots, L-1$$

$$z_k = G^{-1}(s_k) \quad k = 0, 1, 2, \dots, L-1.$$

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

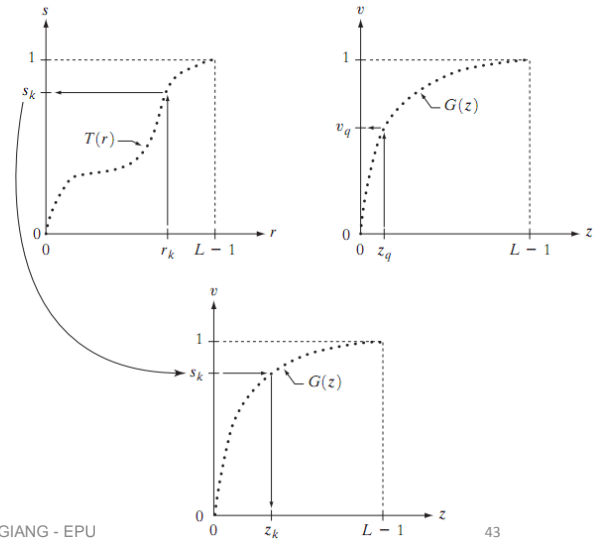
42

Histogram matching (3)

Python code cho histogram matching
???

a b
c

FIGURE 3.19
(a) Graphical interpretation of mapping from r_k to s_k via $T(r)$.
(b) Mapping of z_q to its corresponding value v_q via $G(z)$.
(c) Inverse mapping from s_k to its corresponding value of z_k .



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

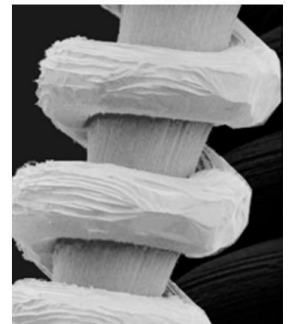
43

Cải thiện ảnh dựa trên các con số thống kê của histogram (histogram statistic)

- Làm nổi vùng ảnh bên phải?

Python code:
???

FIGURE 3.24 SEM image of a tungsten filament and support, magnified approximately 130 \times . (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene).



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

44

Histogram statistic

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

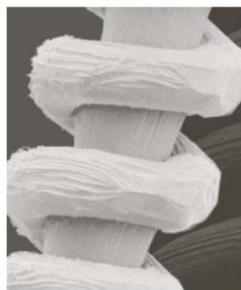
45

Histogram statistic (tt)

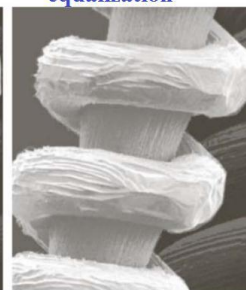
- Kết quả chưa tốt

Python code:
???

Ảnh gốc



Ảnh sau histogram equalization



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

46

Histogram statistic (tt)

- Tính toán các thống kê toàn cục

$$\mu_n(r) = \sum_{i=0}^{L-1} (r_i - m)^n p(r_i) \approx \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N [f(x, y) - m]^n$$

$$m = \sum_{i=0}^{L-1} r_i p(r_i) \approx \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N f(x, y)$$

- Các thống kê cục bộ: S_{xy} : mặt nạ tâm (x, y)

$$m_{S_{xy}} = \sum_{(s,t) \in S_{xy}} r_{s,t} p(r_{s,t})$$

$$\sigma_{S_{xy}}^2 = \sum_{(s,t) \in S_{xy}} [r_{s,t} - m_{S_{xy}}]^2 p(r_{s,t})$$

09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

47

Histogram statistic (tt)

- Vấn đề: Làm sao để tăng độ tương phản trong một vùng ảnh mà không làm ảnh hưởng đến các vùng khác
- ➔ Chỉ làm sáng, làm tối vùng cần thiết, các vùng còn lại giữ nguyên

$$g(x, y) = \begin{cases} E \cdot f(x, y) & m_s(x, y) \leq k_0 m_G \text{ and } k_1 \sigma_G \leq \sigma_s(x, y) \leq k_2 \sigma_G \\ f(x, y) & O.W \end{cases}$$

Với k_0, k_1, k_2 là các hằng số

09/02/2023

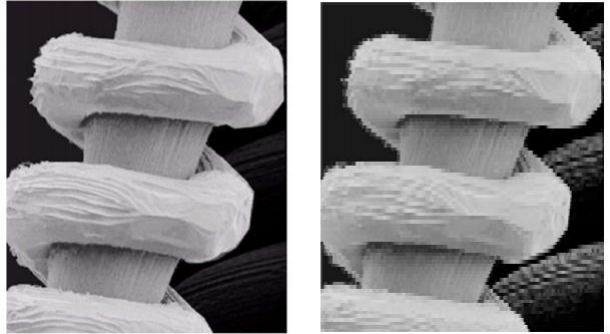
TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

48

Histogram statistic (tt)

Python code:
???

$$E = 4.0, k_0 = 0.4, k_1 = 0.02, k_2 = 0.4$$

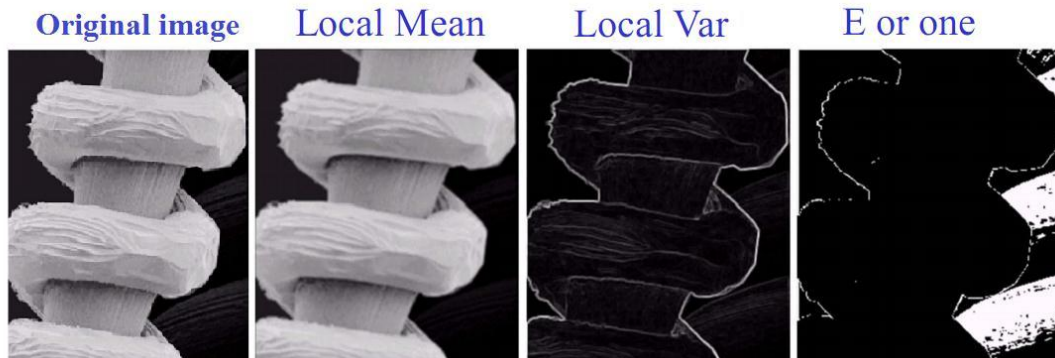


09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

49

Histogram statistic (tt)



09/02/2023

TS. ĐOÀN THỊ HƯƠNG GIANG - EPU

50

... To chapter 3