

3.1(C) a. Tính biểu đồ tần suất  $h(g)$  cho bức ảnh I sau:

$$I = \begin{bmatrix} 1 & 2 & 1 & 3 & 2 & 1 \\ 4 & 4 & 3 & 2 & 4 & 0 \\ 6 & 9 & 2 & 3 & 2 & 1 \\ 6 & 2 & 0 & 5 & 3 & 0 \\ 3 & 4 & 0 & 5 & 1 & 5 \\ 5 & 6 & 8 & 9 & 3 & 6 \end{bmatrix}$$

b.  $h'(g)$  là biểu đồ tần suất của ảnh I' biến đổi từ ảnh I bằng hàm  $f(g)$  sau. Hãy tính  $h'(g)$ .

$$f(g) = |g - 4|$$

a)

$g$  là các số xuất hiện trong ảnh I

$h(g)$ : tần suất xuất hiện

$g$	0	1	2	3	4	5	6	8	9
$h(g)$	4	5	6	6	4	4	4	1	2

b)  $h'(g)$  là biểu đồ tần suất của ảnh I' biến đổi từ ảnh I bằng hàm  $f(g)$  sau. Hãy tính  $h'(g)$

$$f(g) = |g-4|$$

$g$	0	1	2	3	4	5	6	8	9
$f(g)= g-4 $	4	3	2	1	0	1	2	4	5

Gọi  $q$  là các số xuất hiện trong ảnh I'

$q = 0$  (tương ứng với  $g = 4$ )

$q = 1$  (tương ứng với  $g = 3$  và  $g = 5$ )

...

$q$	0	1	2	3	4	5
$h'(g)$	$h(g=4)$	$h(g=3)+h(g=5)$	$h(g=2)+h(g=6)$	$h(g=1)$	$h(g=0)+h(g=8)$	$h(g=9)$
$h'(g)$	4	10	10	5	5	2

c) Thực hiện làm tròn biểu đồ tần suất tính được câu a với  $W = 3$

$g$	0	1	2	3	4	5	6	8	9
$h(g)$	4	5	6	6	4	4	4	1	2

- Trong hầu hết các trường hợp, ngưỡng được chọn một cách tự động từ lược đồ xám của ảnh.
- Để loại bỏ các dao động nhỏ về độ sáng, các thuật toán thường sử dụng kỹ thuật làm trơn lược đồ xám ban đầu nhưng không dịch chuyển vị trí đỉnh bằng công thức sau:

$$h_{smooth}(b) = \frac{1}{W} \sum_{w=-(W-1)/2}^{(W-1)/2} h_{raw}(b-w)$$

- W thường chọn là số lẻ (3 hoặc 5)

$$h_{smooth} = \frac{1}{w} \sum_{w=-\frac{w-1}{2}}^{\frac{w-1}{2}} h(g-w)$$

với  $W = 3$

$$\begin{aligned} h_{smooth}(0) &= \frac{1}{3} \sum_{w=-1}^1 h(0-w) = \frac{1}{3} [h(0-(-1))+h(0-0)+h(0-1)] \\ &= \frac{1}{3} [h(-1)+\mathbf{h(0)}+h(1)] = \frac{1}{3} (0+4+5) = 3 \end{aligned}$$

$$\begin{aligned} h_{smooth}(1) &= \frac{1}{3} \sum_{w=-1}^1 h(1-w) = \frac{1}{3} [h(1-(-1))+h(1-0)+h(1-1)] \\ &= \frac{1}{3} [h(0)+\mathbf{h(1)}+h(2)] = \frac{1}{3} (4+5+6) = 5 \end{aligned}$$

**Ví dụ  $W = 5$**

$$h_{smooth}(0) = 1/5 [h(-2)+h(-1)+\mathbf{h(0)}+h(1)+h(2)]$$

$$\begin{aligned} h_{smooth}(2) &= \frac{1}{3} \sum_{w=-1}^1 h(2-w) = \frac{1}{3} [h(2-(-1))+h(2-0)+h(2-1)] \\ &= \frac{1}{3} [h(1)+h(2)+h(3)] = \frac{1}{3} (5+6+6) \approx 5.7 = 6 \end{aligned}$$

$$\begin{aligned} h_{smooth}(3) &= \frac{1}{3} \sum_{w=-1}^1 h(3-w) = \frac{1}{3} [h(3-(-1))+h(3-0)+h(3-1)] \\ &= \frac{1}{3} [h(2)+h(3)+h(4)] = \frac{1}{3} (6+6+4) \approx 5.3 = 5 \end{aligned}$$

$$h_{smooth}(4) = \frac{1}{3} \sum_{w=-1}^1 h(4-w) = \frac{1}{3} [h(4-(-1))+h(4-0)+h(4-1)]$$

$$= \frac{1}{3} [h(3)+h(4)+h(5)] = \frac{1}{3} (6+4+4) \approx 4.7 = 5$$

$$hsmooth(5) = \frac{1}{3} \sum_{w=-1}^1 h(5-w) = \frac{1}{3} [h(5-(-1))+h(5-0)+h(5-1)]$$

$$= \frac{1}{3} [h(3)+h(4)+h(5)] = \frac{1}{3} (6+4+4) \approx 4.7 = 5$$

$$hsmooth(6) = \frac{1}{3} \sum_{w=-1}^1 h(6-w) = \frac{1}{3} [h(6-(-1))+h(6-0)+h(6-1)]$$

$$= \frac{1}{3} [h(5)+h(6)+h(7)] = \frac{1}{3} (4+4+0) \approx 2.7 = 3$$

$$hsmooth(8) = \frac{1}{3} \sum_{w=-1}^1 h(8-w) = \frac{1}{3} [h(8-(-1))+h(8-0)+h(8-1)]$$

$$= \frac{1}{3} [h(7)+h(8)+h(9)] = \frac{1}{3} (0+1+2) = 1$$

$$hsmooth(9) = \frac{1}{3} \sum_{w=-1}^1 h(9-w) = \frac{1}{3} [h(9-(-1))+h(9-0)+h(9-1)]$$

$$= \frac{1}{3} [h(8)+h(9)+h(10)] = \frac{1}{3} (1+2+0) = 1$$

g	0	1	2	3	4	5	6	8	9
hsmooth(g)	3	5	6	5	5	4	3	1	1

## NÂNG CAO CHẤT LƯỢNG

### I. Cân bằng tần suất

$l \sim$  kích thước  $m \times n$ ,  $new\_level \sim$  số mức xám của ảnh cân bằng

#### Cách làm

$TB = \frac{mxn}{new\_level}$  : số điểm ảnh TB của mỗi mức xám của ảnh cân bằng

$t(g) = \sum_{i=0}^g h(i)$  : số điểm ảnh có mức xám  $\leq g$

Xác định hàm  $f$ :  $g \mapsto f(g)$

Sao cho  $f(g) = \max \{0, \text{round}(\frac{t(g)}{TB})-1\}$

3.4(C) a. Thực hiện cân bằng tần suất cho ảnh I, được biết ảnh gốc và ảnh kết quả cùng là ảnh 6 cấp xám.

$$I = \begin{bmatrix} 3 & 1 & 4 & 4 & 2 & 3 \\ 3 & 2 & 2 & 0 & 5 & 1 \\ 1 & 2 & 3 & 1 & 0 & 1 \\ 4 & 1 & 0 & 2 & 4 & 4 \\ 3 & 1 & 0 & 1 & 0 & 1 \\ 2 & 1 & 3 & 2 & 1 & 3 \end{bmatrix}$$

b. Thế nào là ảnh cân bằng lý tưởng? Sau khi cân bằng thì ảnh đã là cân bằng lý tưởng chưa?

a)

- Tính biểu đồ tần suất của ảnh I

g	0	1	2	3	4	5
h(g)	5	11	7	7	5	1

**Bước 1: Số điểm ảnh TB của mỗi mức xám của ảnh cân bằng là:**

$$TB = (6*6)/6 = 6$$

**Bước 2: Số điểm ảnh có mức xám  $\leq g$  là:  $t(g) = \sum_{i=0}^g h(i)$**

$$t(0) = \sum_{i=0}^0 h(i) = h(0) = 5$$

$$t(1) = \sum_{i=0}^1 h(i) = h(0) + h(1) = 5 + 11 = 16$$

$$t(2) = \sum_{i=0}^2 h(i) = h(0) + h(1) + h(2) = 16 + 7 = 23$$

$$t(3) = \sum_{i=0}^3 h(i) = h(0) + h(1) + h(2) + h(3) = 23 + 7 = 30$$

$$t(4) = \sum_{i=0}^4 h(i) = h(0) + h(1) + h(2) + h(3) + h(4) = 30 + 5 = 35$$

$$t(5) = \sum_{i=0}^5 h(i) = h(0) + h(1) + h(2) + h(3) + h(4) + h(5) = 35 + 1 = 36$$

**Bước 3: Xác định  $f(g) = \max\{0, \text{round}(\frac{t(g)}{TB}) - 1\}$**

$$f(0) = \max\{0, \text{round}(\frac{5}{6}) - 1\} = \max\{0, 1 - 1\} = 0$$

$$f(1) = \max\{0, \text{round}(\frac{16}{6}) - 1\} = \max\{0, 3 - 1\} = 2$$

$$f(2) = \max\{0, \text{round}(\frac{23}{6}) - 1\} = \max\{0, 4 - 1\} = 3$$

$$f(3) = \max\{0, \text{round}(\frac{30}{6}) - 1\} = \max\{0, 5 - 1\} = 4$$

$$f(4) = \max\{0, \text{round}(\frac{35}{6}) - 1\} = \max\{0, 6 - 1\} = 5$$

$$f(5) = \max\{0, \text{round}(\frac{36}{6}) - 1\} = \max\{0, 6 - 1\} = 5$$

$$I = \begin{bmatrix} 3 & 1 & 4 & 4 & 2 & 3 \\ 3 & 2 & 2 & 0 & 5 & 1 \\ 1 & 2 & 3 & 1 & 0 & 1 \\ 4 & 1 & 0 & 2 & 4 & 4 \\ 3 & 1 & 0 & 1 & 0 & 1 \\ 2 & 1 & 3 & 2 & 1 & 3 \end{bmatrix}$$

I\_kq =

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

Ví dụ 2:

1	2	0	4
1	0	0	7
2	2	1	0
4	1	2	1
2	0	1	1

B1: TB =  $4 \cdot 5 / 5 = 4$

$r_k$	0	1	2	4	7
$p_r(r_k)$	5/20	7/20	5/20	2/20	1/20

**Bước 2: Số điểm ảnh có mức xám  $\leq g$  là:  $t(g) = \sum_{i=0}^g h(i)$**

$$t(0) = \sum_{i=0}^0 h(i) = h(0) = 5$$

$$t(1) = \sum_{i=0}^1 h(i) = h(0) + h(1) = 12$$

$$t(2) = \sum_{i=0}^2 h(i) = h(0) + h(1) + h(2) = 17$$

$$t(4) = \sum_{i=0}^4 h(i) = h(0) + h(1) + h(2) + h(3) + h(4) = 19$$

$$t(7) = \sum_{i=0}^7 h(i) = h(0)+h(1)+h(2)+h(4) +h(7)= 20$$

**Bước 3:Xác định  $f(g) = \max\{0, \text{round}(\frac{t(g)}{TB})-1\}$**

$$f(0)= \max\{0, \text{round}(\frac{5}{4})-1\} = \max\{0, 1-1\}= 0$$

$$f(1)= \max\{0, \text{round}(\frac{12}{4})-1\} = \max\{0, 3-1\} = 2$$

$$f(2)= \max\{0, \text{round}(\frac{17}{4})-1\} = \max\{0, 4-1\} = 3$$

$$f(4)= \max\{0, \text{round}(\frac{19}{4})-1\} = \max\{0, 5-1\} = 4$$

$$f(7)= \max\{0, \text{round}(\frac{20}{4})-1\} = \max\{0, 5-1\} = 4$$

1	2	0	4
1	0	0	7
2	2	1	0
4	1	2	1
2	0	1	1

I' =

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

**b) Ảnh I được gọi là cân bằng lý tưởng nếu mọi mức xám g, g' ta có  $h(g) = h(g')$**

Sau khi cân bằng tần suất

g	0	2	3	4	5
h'(g)	5	11	7	7	6

$h'(0)$  khác  $h'(2) \Rightarrow$  Ảnh I\_kq ko cân bằng lý tưởng

## II. Cân bằng lược đồ xám

$r_k$ : mức xám đầu vào

$s_k$ : mức xám đầu ra

k: dải cường độ xám

$n_j$ : tần suất cường độ sáng j

n: tổng số điểm ảnh

$L'$ : số mức xám ảnh đầu ra (thường bằng số mức xám đầu vào)

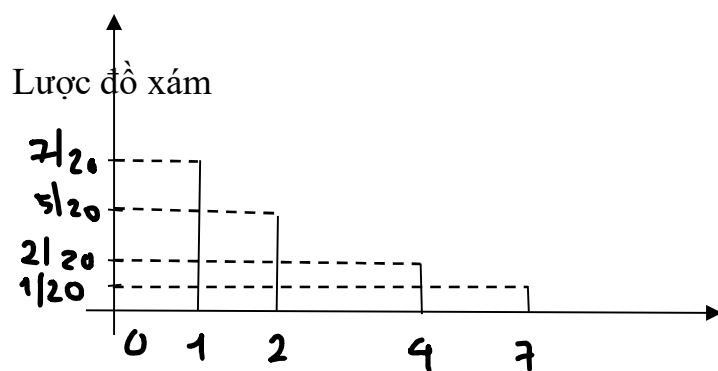
$$s_k = T(r_k) = (L' - 1) \sum_{j=0}^k \frac{n_j}{n} = (L' - 1) \sum_{j=0}^k p_r(r_j)$$

**Ví dụ: Cho bức ảnh I. Hãy vẽ lược đồ xám và thực hiện cân bằng lược đồ xám. Tìm ảnh I' sau khi cân bằng lược đồ xám**

I =

1	2	0	4
1	0	0	7
2	2	1	0
4	1	2	1
2	0	1	1

$r_k$	0	1	2	4	7
$p_r(r_k)$	5/20	7/20	5/20	2/20	1/20



Cân bằng lược đồ xám  $s_k = T(r_k) = (L' - 1) \sum_{j=1}^k p_r(r_j)$

$$s_1 = T(0) = (5-1) \sum_{j=1}^1 p_r(r_j) = 4 p(0) = 4 * 5/20 = 1$$

$$s_2 = T(1) = (5-1) \sum_{j=1}^2 p_r(r_j) = 4 [p(0) + p(1)]$$

$$= 4*[5/20+7/20] = 1+7/20*4 = 2.4 \Rightarrow 2$$

$$s_3 = T(2) = (5-1) \sum_{j=1}^3 p_r(r_j) = 4 [p(0) + p(1) + p(2)]$$

$$= 4*[5/20+7/20+5/20] = 4* 19/20 = 3.4 \Rightarrow 3$$

$$s_4 = T(4) = (5-1) \sum_{j=1}^4 p_r(r_j) = 4 [p(0) + p(1) + p(2)+ p(4)]$$

$$= 4*[5/20+7/20+5/20+2/20] = 4* 21/20 = 3.8 \Rightarrow 4$$

$$s_5 = T(7) = (5-1) \sum_{j=1}^5 p_r(r_j) = 4 [p(0) + p(1) + p(2)+ p(4) + p(7)]$$

$$= 4*[5/20+7/20+5/20+2/20+1/20] = 4* 22/20 = 4.4 \Rightarrow 4$$

I =

1	2	0	4
1	0	0	7
2	2	1	0
4	1	2	1
2	0	1	1

I\_kq =

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



## Nén ảnh

### RLC

## Run-length coding (RLC)

- Mã hoá chiều dài chạy (RLC) là kỹ thuật mã hoá đơn giản nhất và được sử dụng rộng rãi.
- Sử dụng mã này không cần biết tần suất xuất hiện các ký tự và rất hiệu quả trong trường hợp dữ liệu được hiển thị bằng các bit 0 và 1.
- Ta thay thế một chuỗi các ký tự lặp đi lặp lại nhiều lần bằng một cặp (*chiều dài chạy, ký tự*).
- Đối với ảnh, chiều dài chạy cực đại chính là kích thước của một hàng.

a. Original data

BBBBBBBBBAAAAAAAAAAAAAAAAANMMMMMMMMMM

b. Compressed data

B09A16N01M10

a. Original data

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

14

4

0

12

1110 0100 0000 1100

b. Compressed data

## Huffman

Phương pháp mã hóa Huffman là mã hóa các byte trong tệp dữ liệu nguồn bằng biến nhị phân. Nó tạo ra mã độ dài biến thiên là một tập hợp các bit. Đây thuộc dạng phương pháp nén kiểu thống kê và những ký tự xuất hiện nhiều hơn sẽ có mã ngắn hơn.

### Forward Pass

1. Sắp xếp xác suất các ký hiệu theo thứ tự
2. Gộp 2 xác suất thấp nhất
3. Lặp lại bước 2 cho đến khi tổng các xác suất bằng 1.
4. Ấn định từ mã cho ký tự bằng cách đọc đường dẫn từ gốc đến ký tự đó.

## Đặc điểm của mã Huffman

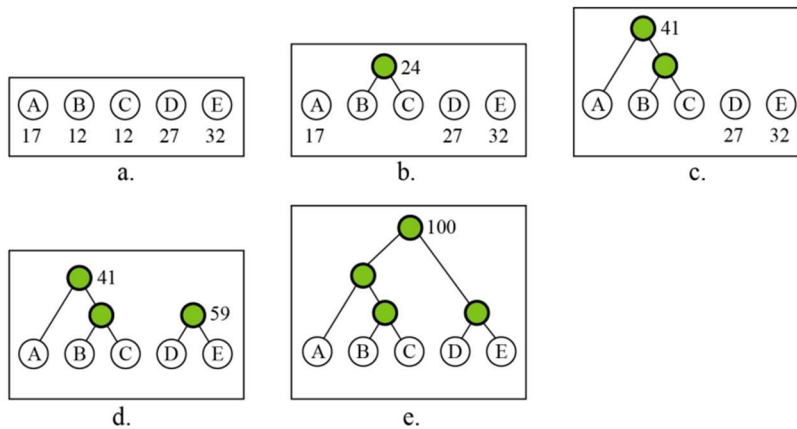
- Quá trình mã hóa các ký tự được thực hiện chỉ 1 lần.
- Mã Huffman được gọi là *mã khối* vì mỗi ký tự được biểu diễn bằng một từ mã cố định.
- Việc giải mã từng ký hiệu được thực hiện tức thời vì không phụ thuộc ký hiệu trước và sau.
- Nhược điểm của mã này là phải biết xác suất xuất hiện của các ký tự trước khi mã hoá.

## Ví dụ

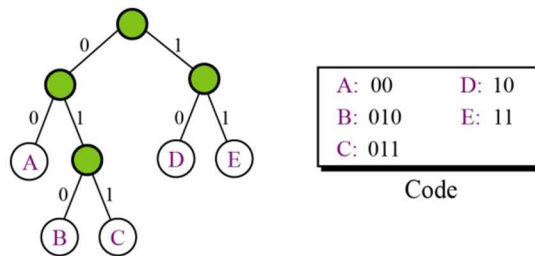
Cho tập 5 ký tự A, B, C, D, E với tần suất xuất hiện tương ứng trong bảng dưới đây. Thực hiện mã hoá Huffman cho tập ký tự này.

**Table 15.1** Frequency of characters

Character	A	B	C	D	E
Frequency	17	12	12	27	32



- Đọc từ mã tương ứng với ký tự theo đường dẫn từ gốc qua các nhánh đến ký tự đó.



3.8(D) Thực hiện mã hóa ảnh sau bằng thuật toán Huffman. Được biết ảnh được chia làm các khối kích thước 2x2 để làm đơn vị mã hóa (Mỗi khối này sẽ như là một chữ cái của bức ảnh).

$$I = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

tạo bảng tần suất (nếu đề chưa cho)

Đặt  $A = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$ ;  $B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ ;  $C = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ ;  $D = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}$ ;  $E = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}$

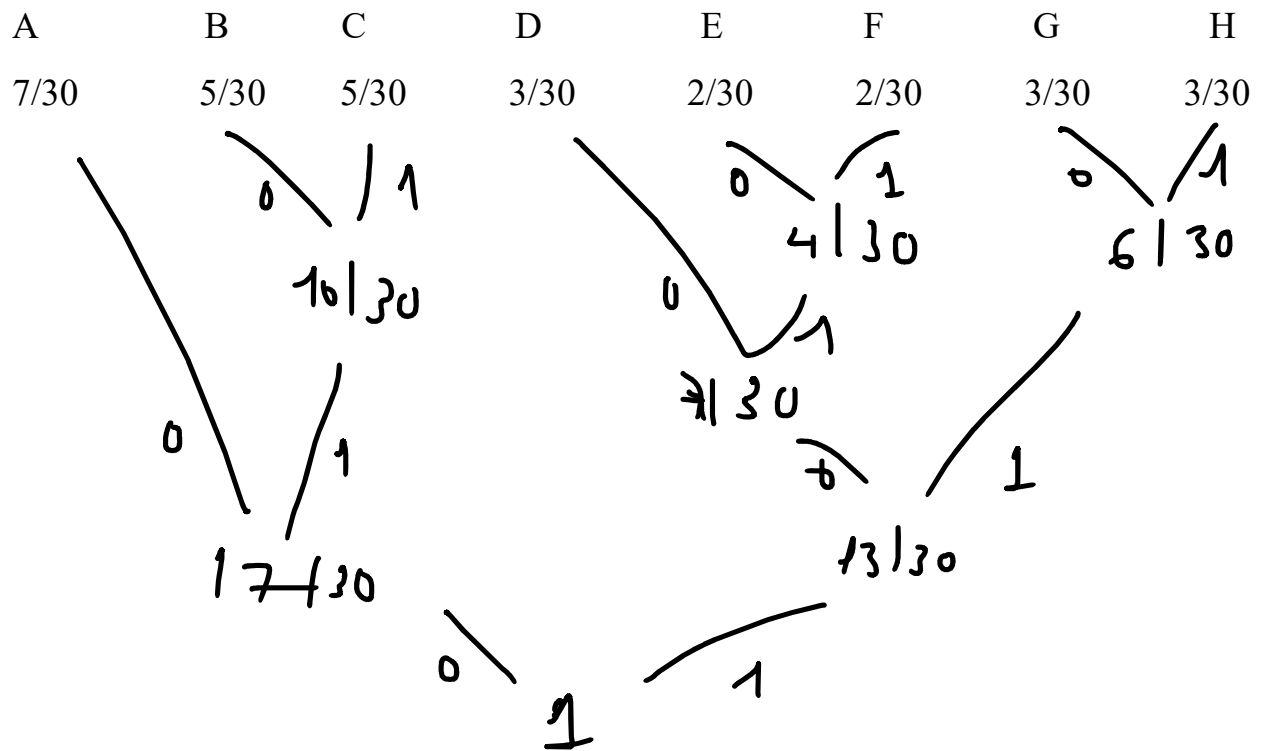
$F = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix}$ ;  $G = \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$ ;  $H = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$

I =

A	B	A	A	C	D	A	C	D	E
B	B	C	F	D	E	C	G	G	H
F	B	A	C	A	B	G	H	H	A

Bảng tần suất

Ký tự	A	B	C	D	E	F	G	H
n	7	5	5	3	2	2	3	3
p	7/30	5/30	5/30	3/30	2/30	2/30	3/30	3/30



A: 00; B: 010; C: 011;

D: 100; E: 1010; F: 1011; G: 110; H: 111

Ký tự	A	B	C	D	E	F	G	H
Mã hóa	00	010	011	100	1010	1011	110	111
n	7	5	5	3	2	2	3	3

I =

A	B	A	A	C	D	A	C	D	E
B	B	C	F	D	E	C	G	G	H
F	B	A	C	A	B	G	H	H	A

I\_kq =

00	010	00	00	011	100	00	011	100	1010
010	010	011	1011	100	1010	011	110	110	111
1011	010	00	011	00	010	110	111	111	00

+ Số bit trước khi mã hóa:  $N1 = 4 \cdot 10 \cdot 3 = 120$  bit

+ Số bit sau khi mã hóa:  $N2 = 2 \cdot 7 + 3 \cdot (5 + 5 + 3 + 3 + 3) + 4 \cdot (2 + 2) = 87$  bit

Tỷ lệ nén  $N1/N2 = 120/87 = 1.38$

## Mã hóa LZW

### Mã LZW – Lempel-Ziv-Welch

- Ý tưởng là tạo ra từ điển (1 bảng) các chuỗi được sử dụng trong phiên truyền thông.
- LZW sử dụng các từ mã chiều dài cố định để biểu diễn các chuỗi ký tự chiều dài thay đổi thường xảy ra cùng nhau, ví dụ các từ trong đoạn văn bản.
- Bộ mã hoá và giải mã LZW cùng xây dựng một bộ từ điển trong quá trình nhận dữ liệu.
- Nếu cả bên gửi và bên nhận đều có bản copy của cuốn từ điển (dictionary) thì các chuỗi đã gặp trước đó sẽ được thay thế bằng mục lục của chúng để làm giảm lượng thông tin cần truyền.

4.9(D) Thực hiện mã hóa ảnh sau bằng kỹ thuật LZW. Được biết ảnh được chia làm các khối kích thước 1x2 để làm đơn vị mã hóa. Và từ điển gốc bao gồm 4 đơn vị mã hóa sau 00, 01, 10, 11 tương đương với giá trị từ 0 đến 3, từ điển sẽ được xây dựng tiếp theo từ giá trị 4. Bức ảnh sẽ được đọc từ trái qua phải và từ trên xuống dưới.

- Thực hiện mã hóa và giải mã ảnh trên với LZW. Coi từ điển là đủ lớn để không thiếu chỗ.
- Ý tưởng cơ bản của mã hóa LZW là ở đâu? LZW có vấn đề gì và có cách nào để giải quyết nó không?

$$I = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

a. Mã hóa

Đặt

Từ điển gốc: 00 là 0; 01 là 1; 10 là 2; 11 là 3

output	Dictionary	
	Code word	String
2	4	10-00
0	5	00-11
3	6	11-00
5	7	00-11-10
2	8	10-01
1	9	01-11
3	10	11-11
6	11	11-00-11
3	12	11-01
1	13	01-01
9	14	01-11-10
4		

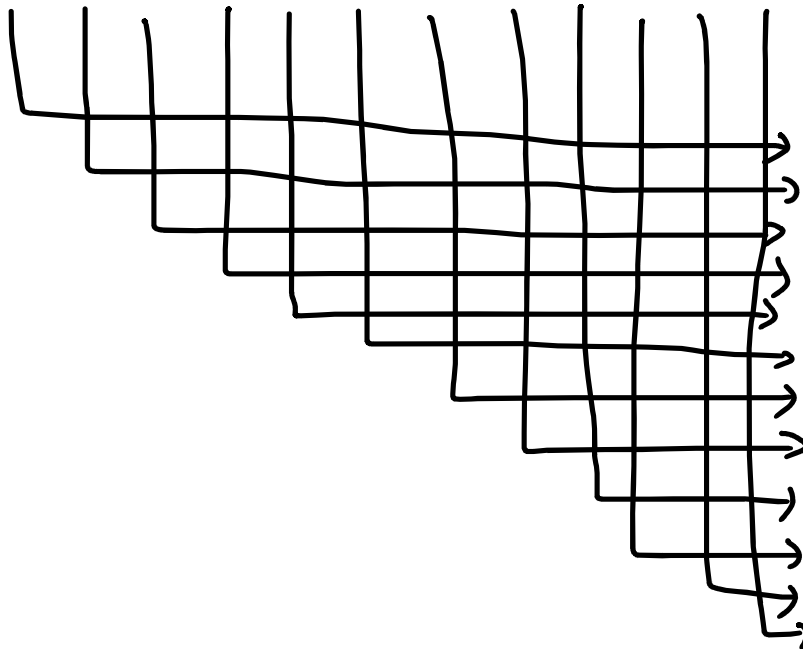
I\_kq = 2-0-3-5-2-1-3-6-3-1-9-4

**Giải mã**

Từ điển gốc: 00 là 0; 01 là 1; 10 là 2; 11 là 3

**Previous output1 + previous first output2**

<2> <0> <3> <5> <2> <1> <3> <6> <3> <1> <9> <4>



Output	Dictionary	
	Index	String
10		
00	4	10-00
11	5	00-11
00-11	6	11-00
10	7	00-11-10
01	8	10-01
11	9	01-11
11-00	10	11-11
11	11	11-00-11
01	12	11-01
01-11	13	01-01
10-00	14	01-11-10

I = 10-00-11-00-11-10-01-11-11-00-11-01-01-11-10-00

$$I = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

## PHÉP CO DẪN HÌNH

Phép co

❑ Phép co bức ảnh  $f$  bởi SE  $s$  được biểu diễn bởi phép toán:  $f \ominus s$

❑ Mô tả phép toán:

Giả sử SE  $s$  đang ở vị trí  $(x, y)$ . Pixel mới sau khi thực hiện phép toán có giá trị như sau:

$$g(x, y) = \begin{cases} 1 & \text{if } s \text{ fits } f \\ 0 & \text{otherwise} \end{cases}$$

Phép giãn

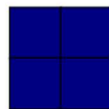
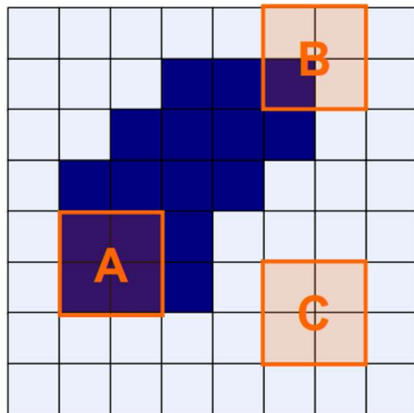


□ Phép giãn hình  $f$  bởi phần tử cấu trúc  $s$  được cho bởi công thức  $f \oplus s$

□ Mô tả phép toán:

Nếu phần tử cấu trúc  $s$  tại pixel có tọa độ  $(x, y)$  của ảnh gốc thì sau phép giãn hình, pixel đó có giá trị:

$$g(x, y) = \begin{cases} 1 & \text{if } s \text{ hits } f \\ 0 & \text{otherwise} \end{cases}$$



Structuring Element

**Fit:** Fit xảy ra khi tất cả pixel của SE trùng với các pixel của bức ảnh

**Hit:** Hit xảy ra khi bất kỳ pixel nào của SE trùng với pixel của ảnh

3.10(D) Thực hiện phép co và dẫn hình  $X$  với một phần tử cấu trúc  $B$  sau để được hai hình  $X_1$  và  $X_2$ .

$$X = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix} \quad \text{với } B = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$X_1 = X - B =$$

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

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

$$X2 = X+B =$$

1							0
			0				

### III.

3.6(B) a. Thực hiện khuếch tán lỗi một chiều với ảnh I sử dụng ngưỡng 127, được biết ảnh này là ảnh 256 mức xám với mức nhỏ nhất là 0 và lớn nhất là 255.

$$I = \begin{bmatrix} 1 & 23 & 156 & 22 & 45 \\ 133 & 13 & 12 & 12 & 212 \\ 12 & 232 & 127 & 32 & 21 \end{bmatrix}$$

b. Kỹ thuật khuếch tán lỗi được sử dụng để làm gì, ảnh thu được sau khuếch tán lỗi hơn gì ảnh tách ngưỡng thông thường ?

#### • Khuếch tán lỗi (Error Diffusion)

➤ Là một dạng nửa cường độ.

➤ Khuếch tán lỗi được thực hiện bằng cách so sánh màu sắc thực tế của 1 điểm ảnh với màu sắc gần nhất.

• **Khuếch tán lỗi 1 chiều**

- Cách đơn giản nhất để chuyển ảnh màu liên tục sang ảnh halftone (với chỉ 2 kênh từ ảnh đa cấp xám):
  - Quét ảnh lần lượt từng dòng và từng điểm ảnh một.
  - So sánh với giá trị xám trung bình.
    - Nếu giá trị xám lớn hơn thì thay bằng điểm ảnh trắng.
    - Nếu giá trị xám nhỏ hơn thì thay bằng điểm ảnh đen.
  - Vì điểm ảnh hoặc đen hoặc trắng nên giá trị bị làm tròn, phần thừa này được chuyển sang điểm tiếp theo.
  - Quá trình tiếp tục được lặp lại.

$$I = \begin{bmatrix} 1 & 23 & 156 & 22 & 45 \\ 133 & 13 & 12 & 12 & 212 \\ 12 & 232 & 127 & 32 & 21 \end{bmatrix}$$

a) Ngưỡng  $\theta = 127$ ,  $r_k = [0, 255]$

$a_{11} = I_{11} = 1 < \theta \Rightarrow b_{11} = 0$  và chuyển 1 sang  $a_{12}$

$a_{12} = 23 + 1 = 24 < \theta \Rightarrow b_{12} = 0$  và chuyển 24 sang  $a_{13}$

$a_{13} = 156 + 24 = 180 > \theta \Rightarrow b_{13} = 255$

$a_{14} = I_{14} = 22 < \theta \Rightarrow b_{14} = 0$  và chuyển 22 sang  $a_{15}$

$a_{15} = 45 + 22 = 67 < \theta \Rightarrow b_{15} = 0$

$a_{21} = 133 + 67 = 200 > \theta \Rightarrow b_{21} = 255$

$a_{22} = I_{22} = 13 < \theta \Rightarrow b_{22} = 0$

$a_{23} = 12 + 13 = 25 < \theta \Rightarrow b_{23} = 0$

$a_{24} = 12 + 25 = 37 < \theta \Rightarrow b_{24} = 0$

$a_{25} = 212 + 37 > \theta \Rightarrow b_{25} = 255$

Tương tự:  $b_{31} = 0$ ,  $b_{32} = 255$ ,  $b_{33} = 0$ ,  $b_{34} = 255$ ,  $b_{35} = 0$

$I_{kq} =$

0	0	255	0	0
255	0	0	0	255

0	255	0	255	0
---	-----	---	-----	---

b)

Khuếch tán lỗi cho phép giảm thiểu mức độ mất chi tiết của ảnh khi tách ngưỡng bằng cách phân tán lỗi do lượng tử hóa ra các điểm xung quanh (bên phải và bên dưới pixel hiện thời)

- Ảnh thu được sau khuếch tán lỗi hơn ảnh tách ngưỡng thông thường: tổng giá trị điểm ảnh của 1 vùng nhỏ được giữ tương đối gần với giá trị trên ảnh gốc

## PHÁT HIỆN BIÊN

### 1. Phát hiện biên sd đạo hàm bậc 1

- Phương pháp Gradient

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

$$\text{Góc } \theta = \tan^{-1}\left(\frac{G_y}{G_x}\right)$$

• Ta có thể thực hiện  $\frac{\partial f}{\partial x}$  và  $\frac{\partial f}{\partial y}$  sử dụng mặt nạ 1D:

$$\begin{array}{|c|} \hline -1 \\ \hline 1 \\ \hline \end{array} \quad \frac{\partial f}{\partial x} \quad \begin{array}{|c|c|} \hline -1 & 1 \\ \hline \end{array} \quad \frac{\partial f}{\partial y}$$

(toán tử Roberts):

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

Hằng số c nhấn mạnh rằng các pixel đó gần tâm cửa sổ.

- Toán tử Prewitt (c=1)

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

- Toán tử Sobel (c=2)

$$M_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad M_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

$$I1 = I.M_x + I.M_y$$

## **2. Phát hiện biên sử dụng đạo hàm bậc 2**

Toán tử Laplacian

Bộ lọc Laplacian:

0	1	0
1	-4	1
0	1	0

Sử dụng bộ lọc Laplacian ta được ảnh mới trong đó biên của các đối tượng hoặc các vùng không liên tục sẽ được làm rõ

Bản thân bộ lọc Laplacian không làm tăng cường chất lượng ảnh mà chỉ làm rõ nét các vị trí thay đổi trong ảnh.

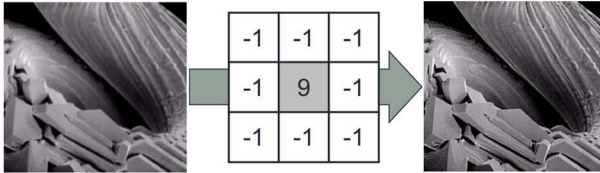
Để **tăng cường chất lượng ảnh**, ta phải thực hiện công việc sau

❖ Biến thể của bộ lọc Laplacian

0	1	0
1	-4	1
0	1	0

Simple Laplacian

1	1	1
1	-8	1
1	1	1

Variant of Laplacian


4.1(D) Cho ảnh I như sau:

$$I = \begin{bmatrix} 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 15 & 15 & 15 & 2 & 2 \\ 2 & 2 & 13 & 12 & 16 & 2 & 2 \\ 2 & 2 & 15 & 15 & 15 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 \end{bmatrix}$$

a. Thực hiện nhân chập ảnh I với các ma trận  $H_x$  và  $H_y$  rồi cộng với nhau để được ảnh  $I_1$

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

b. Thực hiện nhân chập ảnh I ở trên với  $H_z$  dưới đây để được  $I_2$

$$H_z = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

c. Ảnh  $I_1$ ,  $I_2$  đã là ảnh biên chưa? Cần phải làm thêm những gì để có ảnh biên? (nên sử dụng kết quả trên để minh họa)

Lưu ý nhân chập sử dụng tâm ở giữa.

Nhân chập sử dụng tâm ở giữa

Ảnh I =

2	2	2	2	2	2	2
2	2	2	2	2	2	2
2	2	15	15	15	2	2
2	2	13	12	16	2	2
2	2	15	15	15	2	2
2	2	2	2	2	2	2
2	2	2	2	2	2	2

Ma trận  $H_x =$

-1	0	1
-1	0	1
-1	0	1

$$I_{00} = 0*2+1*2+0*2+1*2 = 4$$

$$I_{01} = -1*2+0*2+1*2+(-1)*2+0*2+1*2 = 0$$

$$I_x H_x =$$

4	0	0	0	0	0	-4
6	13	13	0	-13	-13	-6
6	24	23	3	-23	-27	-6
6	37	36	3	-36	-40	-6
6	24	23	3	-23	-27	-6
6	13	13	0	-13	-13	-6
4	0	0	0	0	0	-4

$$I_x H_y =$$

4	6	6	6	6	6	4
0	13	26	39	26	-13	0
0	11	21	35	24	14	0
0	0	0	0	0	0	0
0	-11	-21	-35	-24	-14	0
0	-13	-26	-39	-26	-13	0
4	-6	-6	-6	-6	-6	-4

$$I_1 = I_x H_x + I_x H_y =$$

8	6	6	6	6	6	0
6	26	39	26	13	0	-6
6	35	44	38	1	-13	-6
6	37	36	3	-36	-40	-6
6	13	2	-32	-47	-41	-6
6	0	-13	-39	-39	-26	-6
0	-6	-6	-6	-6	-6	-8

sự thay đổi lớn về giá trị cường độ xám trong 1 bức ảnh để có ảnh biên

- Áp dụng pp Gradient, toán tử Roberts, Sobel, Prewitt
- PP phát hiện biên tối ưu (Canny)

Phân vùng theo ngưỡng biên độ

### **1. Tách ngưỡng toàn cục cơ bản**

♦ Ngưỡng là gì?

➡ Bất cứ điểm  $(x,y)$  nào thỏa mãn  $f(x,y) > T$  được gọi là *điểm đối tượng*, ngược lại nó được gọi là *điểm nền*.

➡ Ngưỡng đơn được cho theo công thức sau:

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

➡ Khi T không đổi trên toàn bộ ảnh, ta gọi đây là phân ngưỡng toàn cục (global thresholding).

### **2. Thuật toán OTSU**

Phương pháp này tối ưu theo nghĩa là tối đa hóa phương sai giữa các nhóm (Between class variance).

Ý tưởng cơ bản của phương pháp là các nhóm được phân ngưỡng tốt phải khác biệt với các giá trị cường độ của các pixel của chúng. Ngược lại, ngưỡng nào cho sự phân biệt tốt nhất giữa các nhóm (theo khía cạnh giá trị cường độ) sẽ được coi là ngưỡng tối ưu.

Phương pháp này dựa hoàn toàn vào việc tính toán trên lược đồ xám của ảnh.



1. Tính lược đồ xám chuẩn hóa của ảnh đầu vào. Tính các giá trị  $p_i = n_i / MN$  ( $i=0, 1, \dots, L-1$ ).
2. Tính tổng tích lũy  $P_1(k)$  với  $k=0, \dots, L-1$  theo công thức:  $P_1(k) = \sum_{i=0}^k p_i$
3. Tính trung bình tích lũy đến mức  $k$  là  $m(k)$ :  $m(k) = \sum_{i=0}^k i \cdot p_i$
4. Tính trung bình cường độ toàn cục  $m_G$ :  
(cường độ trung bình của toàn bộ ảnh).  $m_G = \sum_{i=0}^{L-1} i \cdot p_i$
5. Tính phương sai giữa các nhóm  $\sigma_B^2(k)$

$$\sigma_B^2(k) = \frac{[m_G P_1(k) - m(k)]^2}{P_1(k)[1 - P_1(k)]}$$

6. Ngưỡng Otsu  $k^*$  là giá trị  $k$  thỏa mãn điều kiện  $\sigma_B^2(k)$  đạt cực đại.  
Nếu cực đại không duy nhất thì  $k^*$  được tính là trung bình các giá trị  $k$  tương ứng với các cực đại tìm thấy được.

## Ví dụ

- Tìm ngưỡng tự động theo phương pháp Otsu của ảnh sau:

$$I = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 \\ 0 & 0 & 1 & 2 & 3 & 4 \\ 0 & 0 & 0 & 1 & 2 & 3 \\ 0 & 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$N \times m = 30$$

$$p_0 = 15/30$$

i	$p_i = n_i / MN$	$P_i(k) = \sum_{i=0}^k p_i$	$m(k) = \sum_{i=0}^k i \cdot p_i$	$m_g = \sum_{i=0}^{L-1} i \cdot p_i$	$\sigma_B^2(k) = \frac{[m_G P_1(k) - m(k)]^2}{P_1(k)[1 - P_1(k)]}$
0	15/30	15/30=0.5	15/30*0=0	7/6	$\frac{[7/6 \cdot 0.5 - 0]^2}{0.5(1-0.5)} = 1.36$
1	5/30	15/30+5/30= <b>20/30</b>	0+5/30*1= <b>5/30</b>	7/6	$\frac{[7/6 \cdot 20/30 - 5/30]^2}{20/30(1-20/30)} = \mathbf{1.68}$
2	<b>4/30</b>	20/30+4/30= <b>24/30</b>	5/30+4/30*2=13/30	7/6	1.56
3	<b>3/30</b>	24/30+3/30= <b>27/30</b>	13/30+3/30*3=11/15	7/6	1.11

4	2/30	27/30+2/30=29/30	11/15+2/30*4=1.0	7/6	0.51
5	1/30	29/30+1/30=1	1+1/30*5=7/6	7/6	0.00

Như vậy ngưỡng  $T = 1$  với  $\sigma_B^2(k) = 1.68$

4.6(C) a. Sử dụng thuật toán tìm ngưỡng tự động để tách ngưỡng ảnh I, được biết ảnh có 10 mức xám.

b. Có lúc nào chúng ta có thể tìm được nhiều ngưỡng không? Trong trường hợp như vậy thì ta nên xử lý như thế nào?

$$I = \begin{bmatrix} 1 & 4 & 1 & 1 & 2 & 3 & 2 & 3 \\ 3 & 2 & 3 & 2 & 5 & 2 & 6 & 2 \\ 2 & 1 & 8 & 2 & 3 & 2 & 5 & 6 \\ 2 & 5 & 2 & 4 & 2 & 9 & 1 & 4 \\ 2 & 2 & 3 & 2 & 0 & 1 & 2 & 1 \\ 1 & 5 & 7 & 1 & 2 & 4 & 5 & 6 \end{bmatrix}$$

a)

i	$p_i = n_i / MN$	$P_i(k) = \sum_{i=0}^k p_i$	$m(k) = \sum_{i=0}^k i \cdot p_i$	$m_g = \sum_{i=0}^{L-1} i \cdot p_i$	$\sigma_B^2(k) = \frac{[m_g P_1(k) - m(k)]^2}{P_1(k)[1 - P_1(k)]}$
0	1/48	1/48*1=1/48	1/48*0=0	144/48	$\frac{[144/48 \cdot 1/48 - 0]^2}{1/48(1-1/48)} = 0.19$
1	9/48	1/48+9/48 = 10/48	0+9/48*1=9/48	144/48	$\frac{[144/48 \cdot 10/48 - 9/48]^2}{10/48(1-10/48)} = 1.16$
2	17/48	27/48	9/48+2*17/48=43/48	144/48	2.55
3	6/48	33/48	43/48+3*6/48=61/48	144/48	2.92
4	4/48	37/48	77/48	144/48	2.84
5	5/48	42/48	102/48	144/48	2.29
6	3/48	45/48	120/48	144/48	1.67
7	1/48	46/48	127/48	144/48	1.32
8	1/48	47/48	135/48	144/48	0.77
9	1/48	48/48=1	144/48	144/48	0

Như vậy ngưỡng  $T = 3$  với  $\sigma_B^2(k) = 2.92$

### 3. Thuật toán đẳng liệu (ISODATA)

Ngưỡng toàn cục cơ bản T được tính như sau:

1. Chọn giá trị ước lượng ban đầu cho  $T=t_0$  (thông thường là mức xám trung bình trong bức ảnh).
2. Phân vùng bức ảnh sử dụng  $t_k$  để tạo ra tạo ra 2 nhóm pixel:  $G_1$  gồm các pixel với mức xám  $\leq t_k$  và  $G_2$  gồm các pixel có mức xám  $> t_k$
3. Tính mức xám trung bình của các pixel trong  $G_1$  là  $\mu_1(t_k)$  và trong  $G_2$  là  $\mu_2(t_k)$

4. Tính giá trị ngưỡng mới:

$$t_{k+1} = \frac{\mu_1(t_k) + \mu_2(t_k)}{2} = \frac{1}{2} \left( \frac{\sum_{z=0}^{t_k} z \cdot p(z)}{\sum_{z=0}^{t_k} p(z)} + \frac{\sum_{z=t_k+1}^{L-1} z \cdot p(z)}{\sum_{z=t_k+1}^{L-1} p(z)} \right)$$

5. Lặp lại các bước 2 – 4 cho đến khi sự khác nhau giữa các T trong các vòng lặp liên tiếp nhỏ hơn một giá trị giới hạn định trước  $\Delta T$  hay  $|t_{k+1} - t_k| < \Delta T$

*Thuật toán này làm việc hiệu quả trong việc tìm giá trị ngưỡng T khi ảnh có lược đồ xám thích hợp.*

Ví dụ:

4.7(C) a. Thực hiện tìm ngưỡng tự động với thuật toán đẳng liệu cho bức ảnh I có biểu đồ tần suất sau:

g	0	1	2	3	4	5	6	7	8	9
h(g)	27	45	33	22	22	36	45	34	23	13

Mô tả từng bước cho đến khi tìm được ngưỡng mong muốn. Được biết ảnh có 10 mức xám.

Chọn ngưỡng  $t_0=5$

$$t_{k+1} = \frac{\mu_1(t_k) + \mu_2(t_k)}{2} = \frac{1}{2} \left( \frac{\sum_{z=0}^{t_k} z \cdot p(z)}{\sum_{z=0}^{t_k} p(z)} + \frac{\sum_{z=t_k+1}^{L-1} z \cdot p(z)}{\sum_{z=t_k+1}^{L-1} p(z)} \right)$$

$$t1 = \frac{1}{2} * \left( \frac{0*27+1*45+2*33+3*22+4*22+ *36}{27+45+33+22+22+} + \frac{6*45+7*34+8*23+9*13}{45+34+23+13} \right) = 4.72$$

(ko làm tròn)

Xét < 4.72

$$t2 = \frac{1}{2} * \left( \frac{0*27+1*45+2*33+3*22+4*22}{27+45+33+22+22} + \frac{5*36+6*45+7*34+8*23+9*13}{36+45+34+23+1} \right) = 4.164$$

Xét < 4.164

$$t3 = \frac{1}{2} * \left( \frac{0*27+1*45+2*33+3*22+4*22}{27+45+33+22+2} + \frac{5*36+6*45+7*34+8*23+ *13}{36+45+34+23+} \right) = 4.164$$

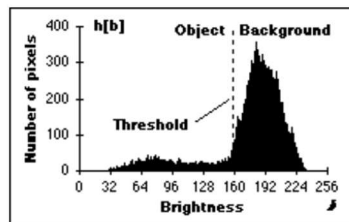
⇒ Lấy ngưỡng T = 4

#### 4. Thuật toán đối xứng nền

➤ Giả định trong lược đồ xám có đỉnh nằm ở phần nền và đối xứng quanh giá trị cực đại đó.

➤ Giá trị đỉnh (*maxp*) được xác định bằng cách tìm giá trị cực đại trong lược đồ xám.

➤ Thuật toán tìm kiếm phía các pixel không phải đối



tượng quanh giá trị cực đại để tìm điểm p% (Điểm có giá trị mức sáng a sao cho P(a)= p %).

- Trong hình trước, các pixel đối tượng nằm ở bên trái của đỉnh nền (đỉnh nền nằm ở mức sáng **maxp = 183**).
- Do đó việc tìm kiếm thực hiện ở bên phải của đỉnh. Ví dụ chọn p = 95%
- Điểm bên phải tương ứng với 95% mức sáng cực đại có độ sáng 216 (**p%**).
- Do giả định về tính đối xứng, ta tính được ngưỡng nằm ở bên trái của giá trị cực đại theo công thức

$$\theta = \text{max } p - (p\% - \text{max } p)$$

- Ngưỡng **θ=183 - (216 - 183) = 150**.

4.7

b. Thực hiện tìm ngưỡng tự động với thuật toán đối xứng nền cho bức ảnh I' có biểu đồ tần suất sau:

g	0	1	2	3	4	5	6	7	8	9
h(g)	2	3	4	5	7	8	12	47	10	2

Được biết độ chính xác cần tính là 88%.

Tìm mức xám có  $h(g)$  lớn nhất

$$\text{maxp} = 7$$

Tổng pixel = 100 (cộng các giá trị  $h(g)$  lại với nhau)

Số pixel từ 0  $\rightarrow$  p là  $88\% \cdot 100 = 88$  pixel

$$\Rightarrow p = 7 \quad (2+3+4+5+7+8+12+47 = 88)$$

$$\Rightarrow \text{Ngưỡng } \theta = \text{maxp} - (p - \text{maxp}) = 7 - (7 - 7) = 7$$

#### 4.8

b. Thực hiện tìm ngưỡng tự động với thuật toán đối xứng nền cho bức ảnh I' có biểu đồ tần suất sau:

g	0	1	2	3	4	5	6	7	8	9
h(g)	39	45	53	72	40	112	25	34	23	13

Được biết độ chính xác cần tính là 88%.

$$\text{maxp} = 5$$

Tổng pixel = 456

Số pixel từ 0  $\rightarrow$  p là  $88\% \cdot 456 \approx 401$  pixel

$$\Rightarrow p = 6 \quad (39+45+53+72+40+112+25 = 386)$$

$$\Rightarrow \text{Ngưỡng } \theta = \text{maxp} - (p - \text{maxp}) = 5 - (6 - 5) = 4$$