

# 第6章

## **JPEG**影像編碼

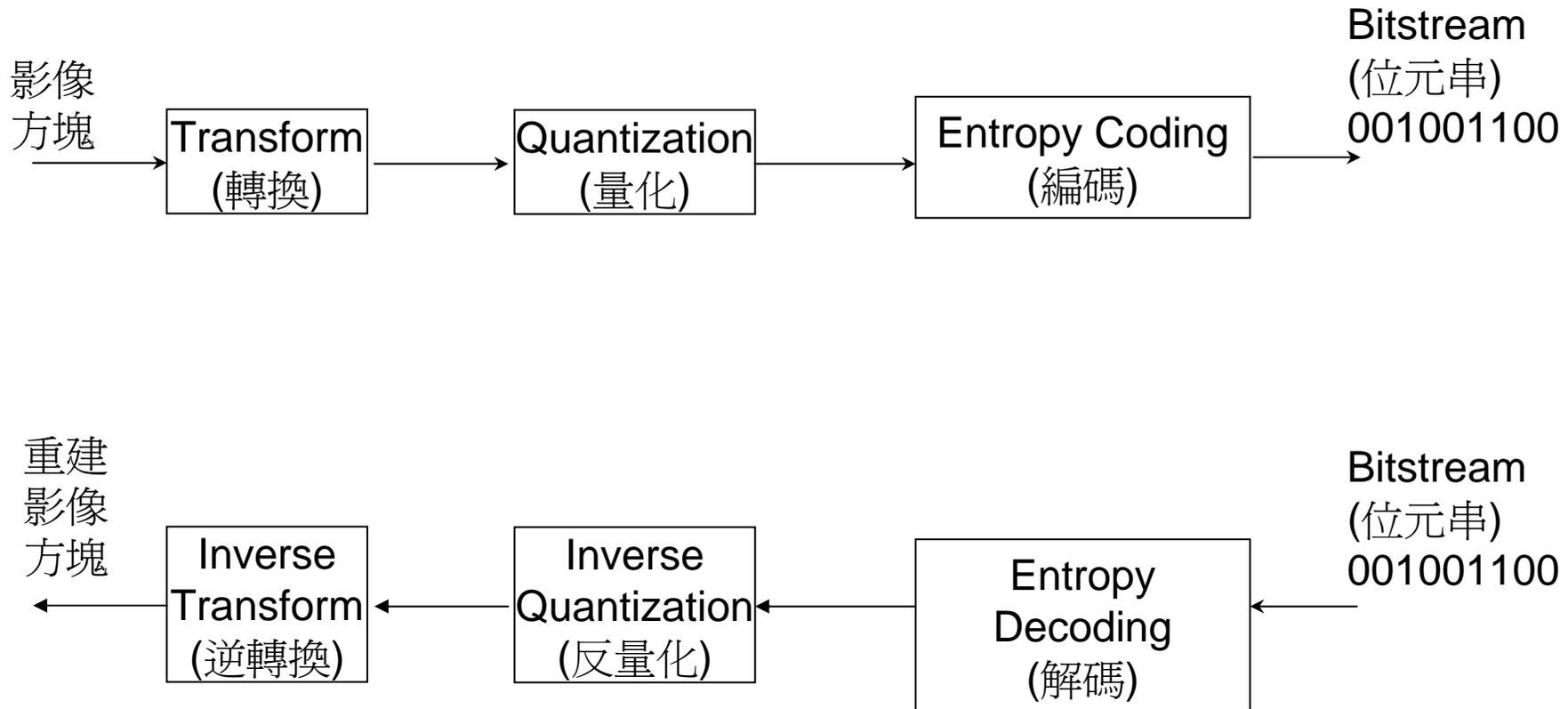
# 課程內容

- JPEG 簡介
  - 壓縮系統架構
  - 壓縮流程
- 色彩空間轉換
  - 縮減取樣
  - 重建
- JPEG壓縮過程
  - DPCM預測編碼
    - 量化器
    - 增量調變(DM)
    - 差值脈碼調變(DPCM)
    - DPCM預測器
  - 交流係數可變長度編碼
- JPEG 流程範例

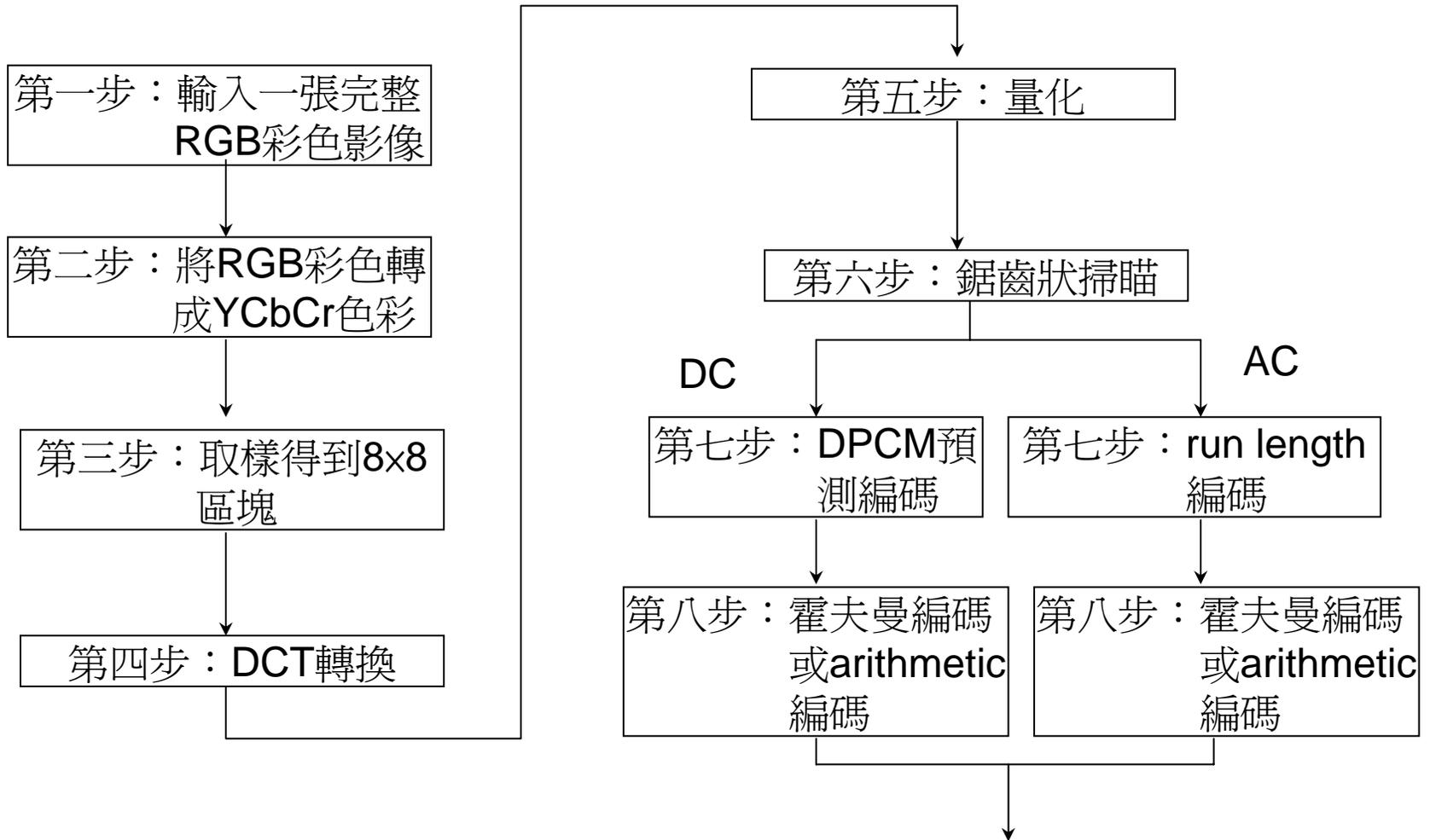
# 6.1 JPEG 簡介

- JPEG ( Joint Photographic Experts Group )是一群由全球各地的影像專業科學家於1987年組合起來，共同為黑白或彩色靜態影像的壓縮建立一套標準，此套標準列ISO 10918 建議書，現已廣泛流行。
- JPEG 以2-D cosine transform 做為映射方式，將映射的結果經過適當量化後，以Huffman code或Arithmetic 編碼。
- JPEG 規範每一階段的規格如大小範圍，能階數和Huffman codeTable 等。

# 影像壓縮(解壓縮)主要過程



# JPEG壓縮系統架構



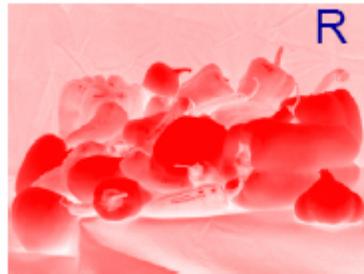
# JPEG壓縮流程

- 主要步驟
  - DCT (Discrete Cosine Transformation)
  - Quantization
  - Zigzag Scan
  - DPCM on DC component
  - RLC (run length coding) on AC Components
  - Entropy Coding (Huffman or arithmetic coding)

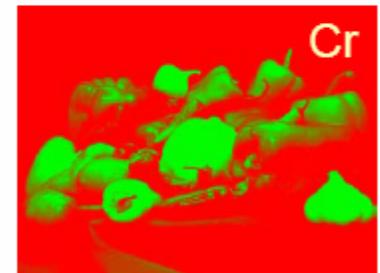
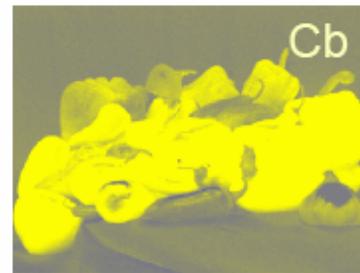
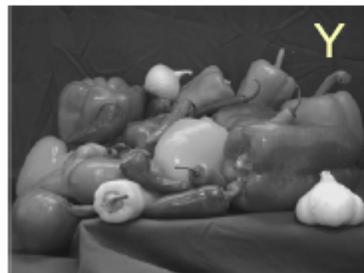
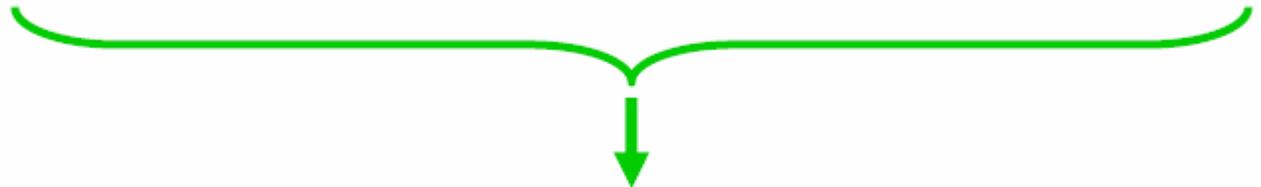
## 6.2 色彩空間轉換

- **JPEG**可適用的影像種類包括灰階與全採色的影像。RGB是最簡易的色彩模式，適合於監視器，但對眼睛而言，未必是最自然的表現方式。與眼睛最有直接關係的是色彩的亮度與鮮艷度。
- **JPEG**的色彩模式是採Y(亮度),  $C_b$ ,  $C_r$ 模式。
- 眼睛對於亮度Y的感覺會比兩色度 $C_b$ ,  $C_r$ 來的敏銳，也就是視覺上三種成份的重要性並不相同。
- 由於 $C_b$ ,  $C_r$ 的重要性降低，可以容忍的誤差較大，為獲取較大的壓縮率，對於 $C_b$ ,  $C_r$ 的資料可以每兩點(或四點)只取一點方式做處理。

- Example:



=



Remember: all JPEG process is operating on **YCbCr** color space !

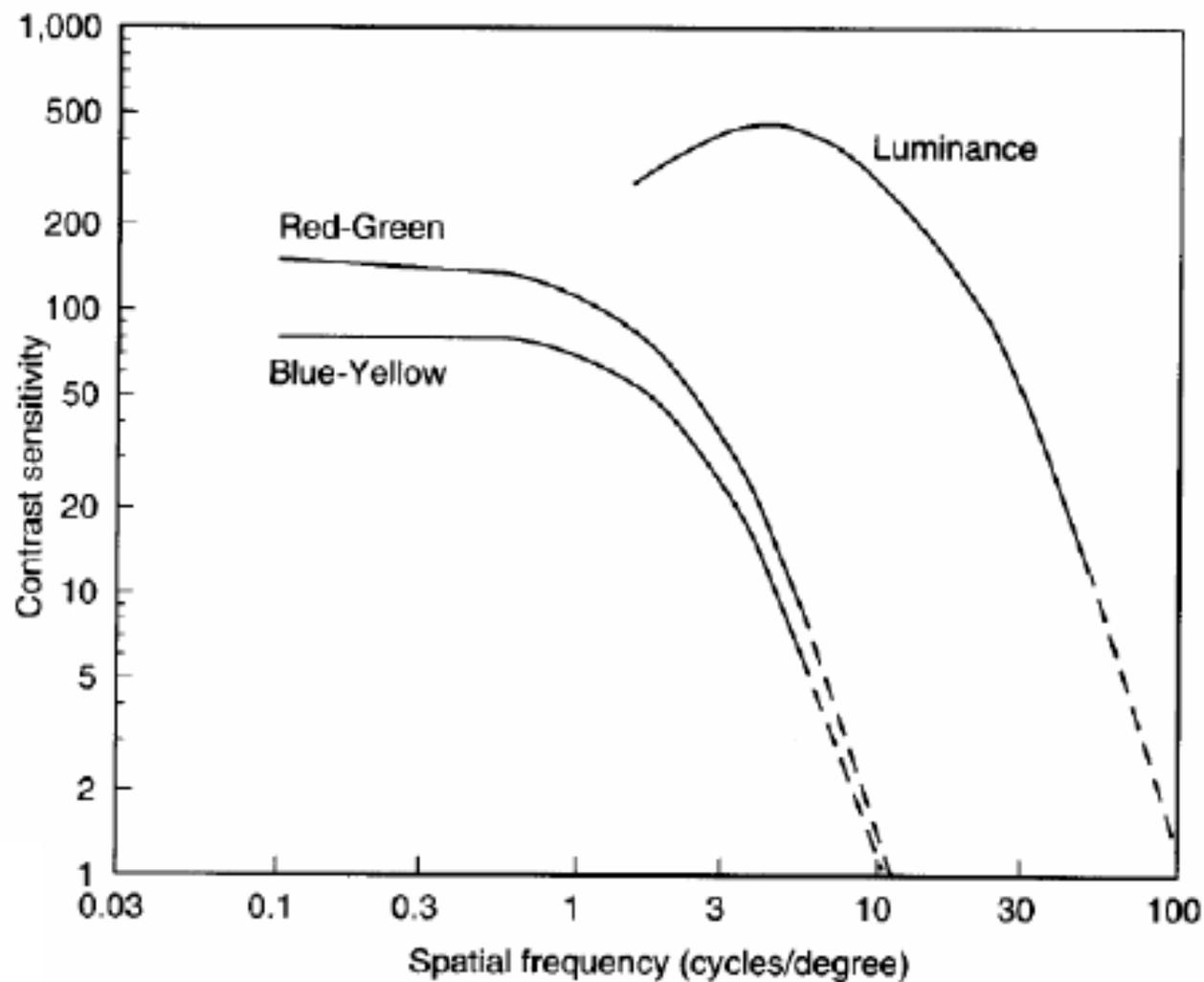
- RGB與 $YC_bC_r$ 存有以下的關係

$$Y = 0.299R + 0.587G + 0.114B$$

$$C_b = -0.168R - 0.331G + 0.499B$$

$$C_r = 0.5R - 0.419G - 0.081B$$

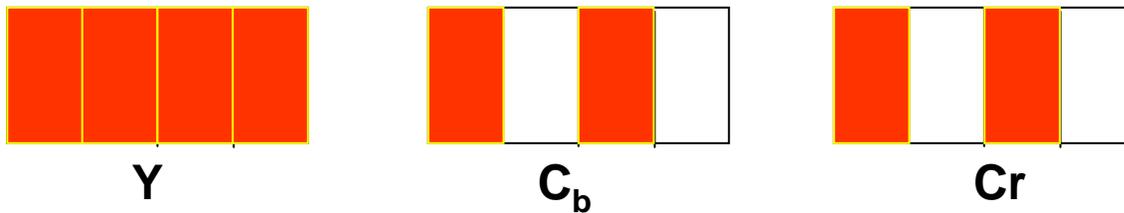
# Eye Sensitivity



# 縮減取樣

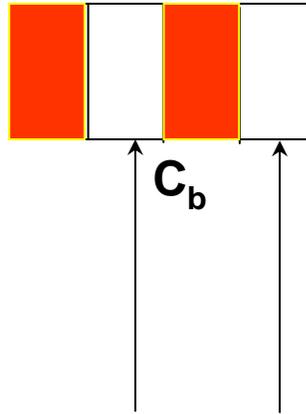
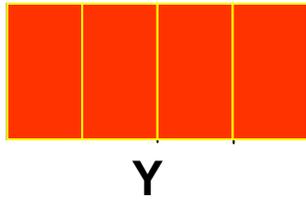
- YUV422:

對於16x16的區塊而言，Y取四個區塊(每個區塊為8x8)， $C_b$ ， $C_r$ 則各取兩個區塊(以每二點取一點方式進行)，做隨後的壓縮處理。

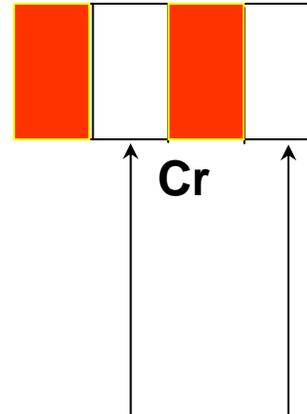


- 壓縮比為  $8/12 = 2/3$

# 重建



利用已知之兩點  
加以重建(中間值)

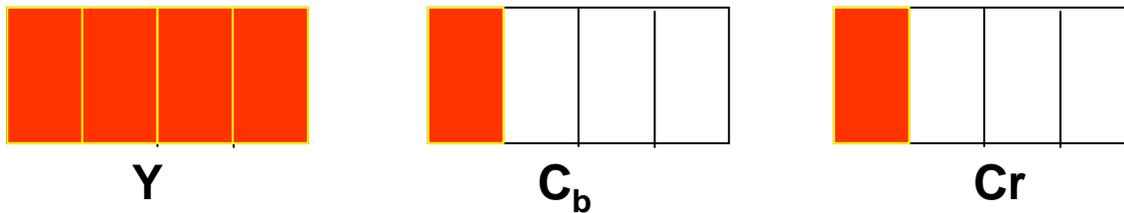


利用已知之兩點  
加以重建(中間值)

# 縮減取樣(續)

- YUV411:

對於16x16的區塊而言，Y取四個區塊(每個區塊為8x8)， $C_b$ ， $C_r$ 則各取一個區塊(以每四點取一點方式進行)，做隨後的壓縮處理。



- 壓縮比為  $6/12 = 1/2$

- **JPEG** 壓縮過程中，必須經過取樣(sampling)及量化(quantization)等兩個步驟。
- 因此部分資料遺失，這是**JPEG**的特性；所以經**JPEG**壓縮後的數位影像，在解壓縮後無法正確無誤的回復成原始的影像。
- 因為**JPEG**的壓縮比高，所以成爲網路上最常使用的影像壓縮方法之一。除了令人滿意的壓縮率外，經**JPEG**壓縮後再還原的影像以肉眼觀察，不容易發現其和原始影像之間的差異。

## 6.3 JPEG壓縮過程

JPEG 基本運作之簡單描述如下：

1. 原影像被切割成不重疊的 $8 \times 8$  方塊，然後每個方塊亮度值減去128，各自使用DCT轉換。
2. 使用量化矩陣(或正規化矩陣)（通用於所有的方塊），將所有的DCT轉換後之係數，進行量化(正規化)。
3. 直流係數（(0,0)位置上的係數）使用DPCM預測編碼方式，其中DPCM的預測器是利用前一個方塊的直流係數做預測。

$$F(i, j) =$$

498.46	35.54	-4.14	1.91	1.33	-12.95	-0.92	1.48
-224.33	-9.45	-5.66	14.24	7.21	-0.53	-2.57	-3.26
13.42	4.47	3.42	-4.42	-3.26	3.79	-0.49	-3.23
4.95	-4.53	6.43	0.88	-1.23	6.71	3.6	2.48
5.72	2.51	-1.47	-0.93	-2.57	-0.67	0.44	8.61
1.32	2.61	0.47	2.23	-0.89	2.54	-2.73	-2.21
-2.45	-1.66	-4.63	2.14	2.57	-4.6	-1.2	1.89
4.36	3.21	4.82	-1.31	0.54	1.36	0.87	-2.54

(a) 8×8DCT係數矩陣

$$Q(i, j) =$$

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

(b) 8×8量化矩陣

$$F^*(i, j) =$$

31	3	0	0	0	0	0	0
-19	-1	0	1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

(c) 量化後DCT係數矩陣

量化方式 
$$F^*(i, j) = \text{round}\left(\frac{F(i, j)}{Q(i, j)}\right)$$



## 5. 交流係數可變長度編碼(run length編碼)

每一個非0的係數，其交流係數之編碼，以下列方式為之

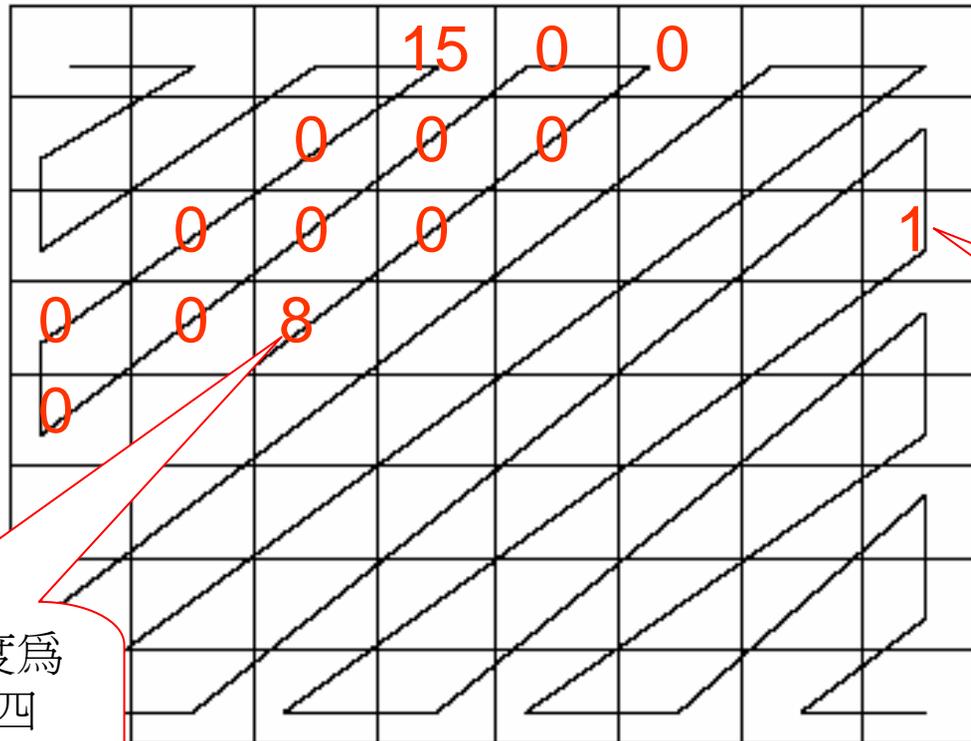
$$(N, S)$$

其中， $N$ 是目前非0係數相對於前一個非0係數的距離(或長度)，即兩個非0係數之間的0的長度。 $N$ 值的變化為  $0 \sim F$ ，共計16種距離(或長度)。

“**S**”用來定義係數大小的類別。在第**k**類的交流值裡，其值範圍為 $(2^{k-1}, 2^k - 1)$ 與 $(-2^k + 1, -2^{k-1})$ 之間，其中**k**滿足 $1 \leq k \leq 14$ （考量到全白的圖像， $255 \times 64 = 16320$ ），共**14**個類別。表**6.1**所示為每一類的交直流係數值，共計**16**個類別。

如果**0**的串長大於**15**，必須將其分成幾次編碼。符號**(F,0)**，表示一個串長為**15**的連續**0**。另一個特殊符號(代表方塊結束)**EOB = “1010”**，表示所有後面剩下的係數都是**0**的時候（送出**EOB**，就不需要再編碼）。

- 例如



與15相距0的長度為11，且屬於為第四類，它的編碼為(B,4)

與8相距0的長度為24，且屬於為第一類，它的編碼為(F,0)與(9,1)

6. Huffman 編碼：每一個方塊的輸出符號經過 Huffman 編碼後送出。在 JPEG 系統中，使用兩個 Huffman 表，其中一個為直流係數編碼，另一個為交流係數編碼。

# 表6.1: JPEG 係數編碼的類別

Range	DC Difference Category	AC Category
0	0	N/A
-1, 1	1	1
-3, -2, 2, 3	2	2
-7, ..., -4, 4, ...	3	3
-15, ..., -8, 8, ..., 15	4	4
-31, ..., -16, 16, ..., 31	5	5
-63, ..., -32, 32, ..., 63	6	6
-127, ..., -64, 64, ..., 127	7	7
-255, ..., -128, 128, ..., 255	8	8
-511, ..., -256, 256, ..., 511	9	9
-1023, ..., -512, 512, ..., 1023	A	A
-2047, ..., -1024, 1024, ..., 2047	B	B
-4095, ..., -2048, 2048, ..., 4095	C	C
-8191, ..., -4096, 4096, ..., 8191	D	D
-16383, ..., -8192, 8192, ..., 16383	E	E
-32767, ..., -16384, 16384, ..., 32767	F	N/A

-3->00  
-2->01  
2->10  
3->11

-15->0000  
-14->0001  
-13->0010  
...  
15->1111

# 表6.2: JPEG 預設亮度直流值編碼

Category	Base Code	Length	Category	Base Code	Length
0	010	3	6	1110	10
1	011	4	7	11110	12
2	100	5	8	111110	14
3	00	5	9	1111110	16
4	101	7	A	11111110	18
5	110	8	B	111111110	20

例如：直流值14，屬於第4類別，其編碼為1011110，碼長為7。

# 表6.3-1: JPEG 預設亮度交流值(N,S)編碼

Run / Category	Base Code	Length
0/0	1010 (=EOB)	4
0/1	00	3
0/2	01	4
0/3	100	6
0/4	1011	8
0/5	11010	10
0/6	111000	12
0/7	1111000	14
0/8	111110110	18
0/9	111111110000010	25
0/A	111111110000011	26
1/1	1100	5
1/2	111001	8
1/3	1111001	10
1/4	111110110	13
1/5	11111110110	16

Run / Category	Base Code	Length
8/1	11111010	9
8/2	11111111000000	17
8/3	111111110110111	19
8/4	111111110111000	20
8/5	111111110111001	21
8/6	111111110111010	22
8/7	111111110111011	23
8/8	111111110111100	24
8/9	111111110111101	25
8/A	111111110111110	26
9/1	111111000	10
9/2	111111110111111	18
9/3	111111111000000	19
9/4	111111111000001	20
9/5	111111111000010	21

# 表6.3-2: JPEG 預設亮度交流值(N,S)編碼

1/6	1111111110000100	22
1/7	1111111110000101	23
1/8	1111111110000110	24
1/9	1111111110000111	25
1/A	1111111110001000	26
2/1	11011	6
2/2	11111000	10
2/3	1111110111	13
2/4	111111110001001	20
2/5	111111110001010	21
2/6	111111110001011	22
2/7	111111110001100	23
2/8	111111110001101	24
2/9	111111110001110	25
2/A	111111110001111	26
3/1	111010	7

9/6	1111111111000011	22
9/7	1111111111000100	23
9/8	1111111111000101	24
9/9	1111111111000110	25
9/A	1111111111000111	26
A/1	111111001	10
A/2	1111111111001000	18
A/3	1111111111001001	19
A/4	1111111111001010	20
A/5	1111111111001011	21
A/6	1111111111001100	22
A/7	1111111111001101	23
A/8	1111111111001110	24
A/9	1111111111001111	25
A/A	1111111111010000	26
B/1	111111010	10

# 表6.3-3: JPEG 預設亮度交流值(N,S)編碼

3/2	111110111	11
3/3	1111110111	14
3/4	111111110010000	20
3/5	111111110010001	21
3/6	111111110010010	22
3/7	111111110010011	23
3/8	111111110010100	24
3/9	111111110010101	25
3/A	111111110010110	26
4/1	111011	7
4/2	1111111000	12
4/3	1111111110010111	19
4/4	1111111110011000	20
4/5	1111111110011001	21
4/6	1111111110011010	22
4/7	1111111110011011	23
4/8	1111111110011100	24
4/9	1111111110011101	25

B/2	11111111111010001	18
B/3	11111111111010010	19
B/4	11111111111010011	20
B/5	11111111111010100	21
B/6	11111111111010101	22
B/7	11111111111010110	23
B/8	11111111111010111	24
B/9	11111111111011000	25
B/A	11111111111011001	26
C/1	1111111010	11
C/2	11111111111011010	18
C/3	11111111111011011	19
C/4	11111111111011100	20
C/5	11111111111011101	21
C/6	11111111111011110	22
C/7	11111111111011111	23
C/8	11111111111100000	24
C/9	11111111111100001	25

# 表6.3-4: JPEG 預設亮度交流值(N,S)編碼

4/A	1111111110011110	26
5/1	1111010	8
5/2	111111001	12
5/3	111111110011111	19
5/4	111111110100000	20
5/5	111111110100001	21
5/6	111111110100010	22
5/7	111111110100011	23
5/8	111111110100100	24
5/9	111111110100101	25
5/A	111111110100110	26
6/1	1111011	8
6/2	1111111000	13
6/3	111111110100111	19
6/4	111111110101000	20
6/5	111111110101001	21
6/6	111111110101010	22
6/7	111111110101011	23

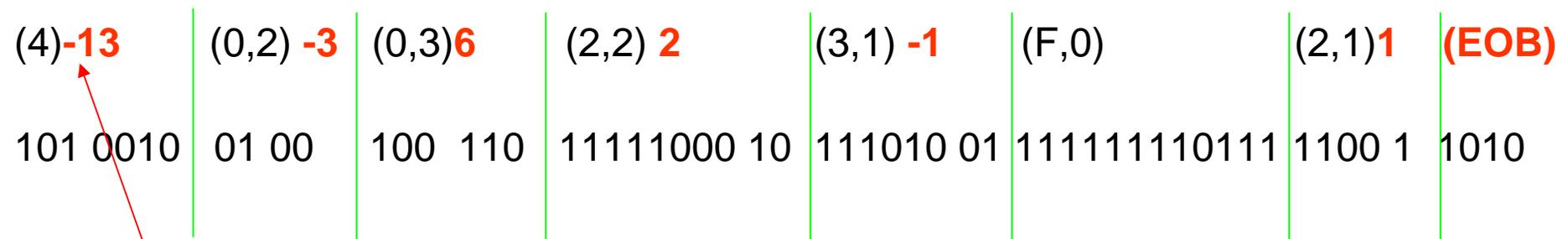
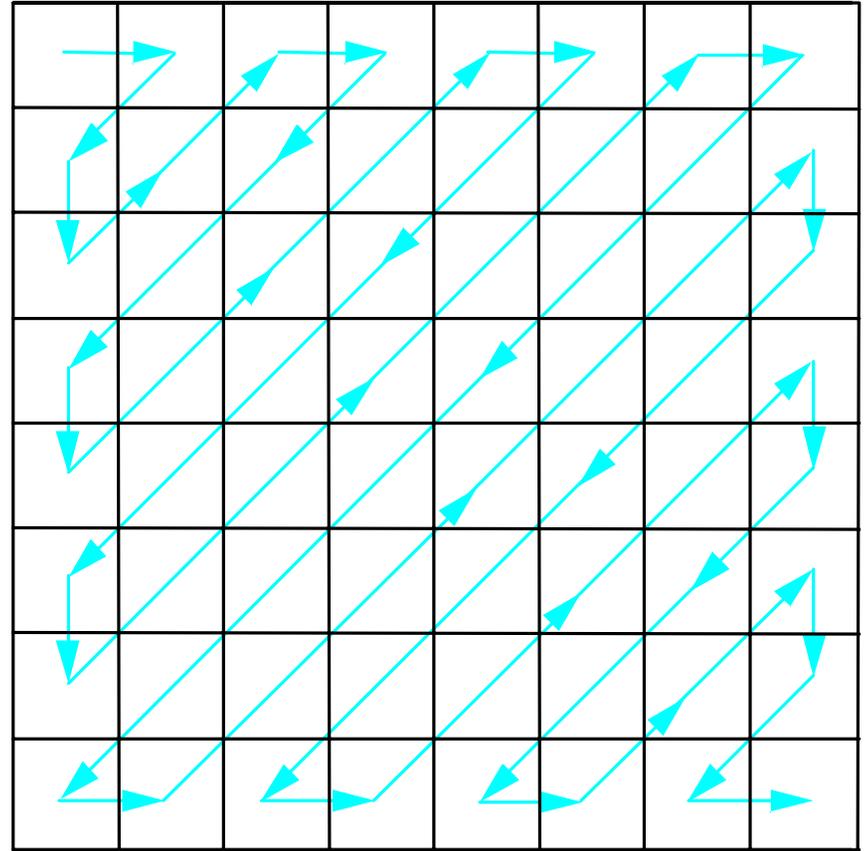
C/A	111111111100010	26
D/1	1111111010	12
D/2	111111111100011	18
D/3	111111111100100	19
D/4	111111111100101	20
D/5	111111111100110	21
D/6	111111111100111	22
D/7	111111111101000	23
D/8	111111111101001	24
D/9	111111111101010	25
D/A	111111111101011	26
E/1	11111110110	13
E/2	111111111101100	18
E/3	111111111101101	19
E/4	111111111101110	20
E/5	111111111101111	21
E/6	111111111110000	22
E/7	111111111110001	23

# 表6.3-5: JPEG 預設亮度交流值(N,S)編碼

6/8	1111111110101100	24
6/9	1111111110101101	25
6/A	1111111110101110	26
7/1	11111001	9
7/2	1111111001	13
7/3	111111110101111	19
7/4	111111110110000	20
7/5	111111110110001	21
7/6	111111110110010	22
7/7	111111110110011	23
7/8	111111110110100	24
7/9	111111110110101	25
7/A	111111110110110	26

E/7	1111111111110001	23
E/8	1111111111110010	24
E/9	1111111111110011	25
E/A	1111111111110100	26
F/0	111111110111	12
F/1	1111111111110101	17
F/2	1111111111110110	18
F/3	1111111111110111	19
F/4	111111111111000	20
F/5	111111111111001	21
F/6	111111111111010	22
F/7	111111111111011	23
F/8	111111111111100	24
F/9	111111111111101	25
F/A	111111111111110	26

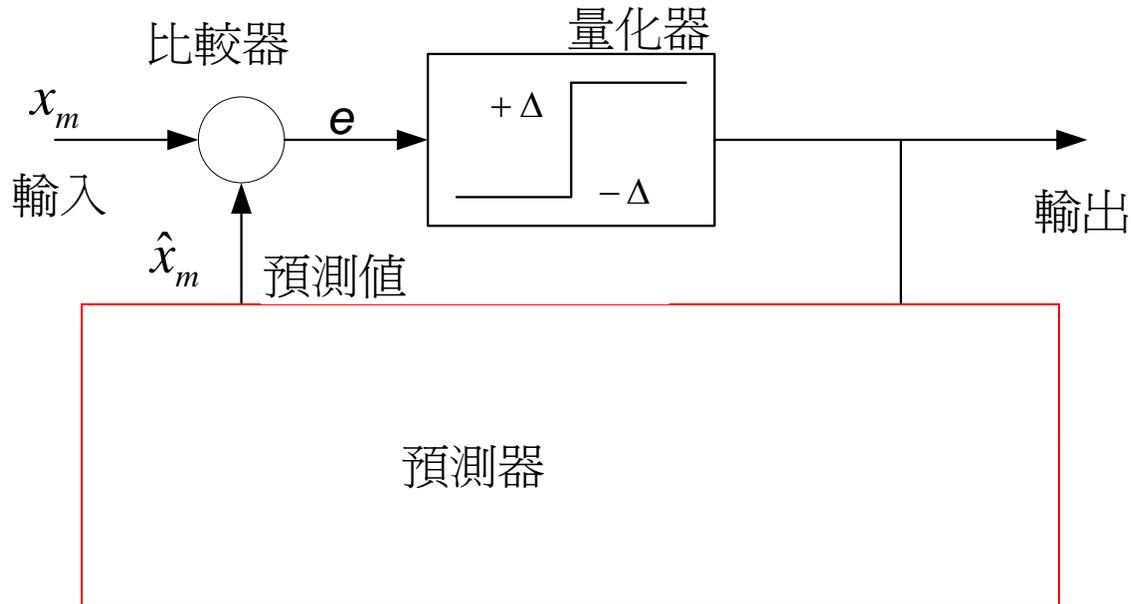
-13	-3	2	0	0	0	1	0
6	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
-1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0



假設此為第一張8x8切割圖像

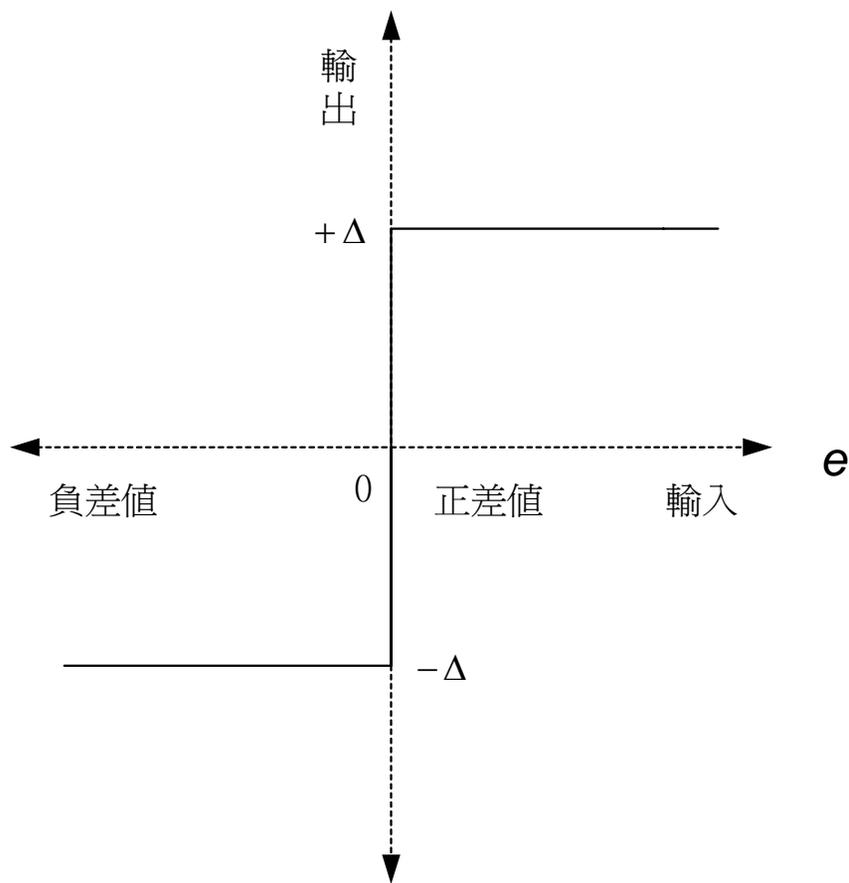
7. 如果是彩色影像，則將三張(紅藍綠)影像轉成  $YC_bC_r$ 。
8. 在解碼端，編碼位元串經過Huffman 解碼之後，重建二維的正常化矩陣。隨後，該矩陣中每一個係數藉著正常化矩陣中相對應位置的放大數值而回復為原來的值。當所有的係數值回復之後，便可做DCT 之逆轉換而到重建方塊。重建誤差決定於由正常化矩所控制的量化階數。

# 直流值的DPCM預測編碼 (DM與DPCM簡介)

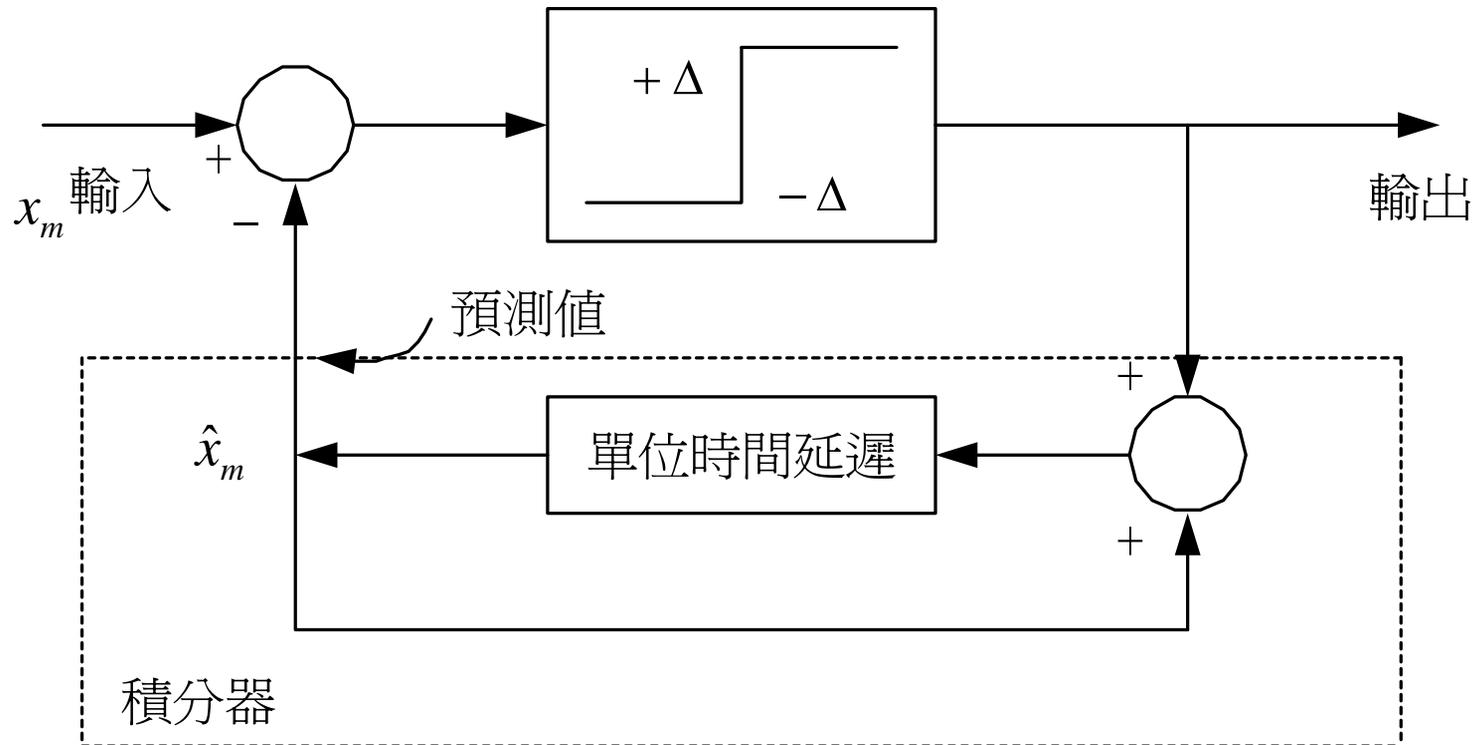


- 增量調變 (Delta modulation, DM)
  - 差值脈碼調變 (Difference pulse code modulation, DPCM)
- 輸入
- 預測值

# 量化器



# 增量調變(DM)

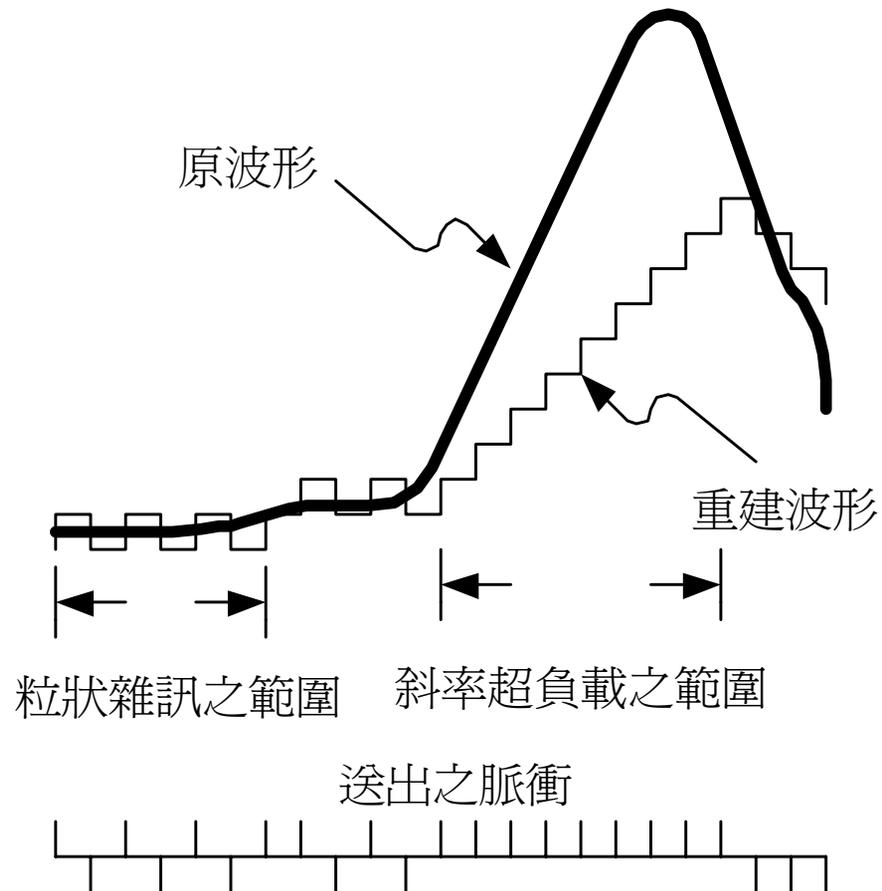


傳統的Delta編碼由三個部分構成：

比較器一個、二階量化器一個、積分器一個

(包括一個延遲單元的正回饋(positive feedback))

# 增量調變(DM)



# 增量調變(DM)

- **Delta 編碼**
- 編碼器的輸出由輸入的取樣值與積分器所產生的預測之取樣值所決定。兩者的差值決定了輸出是正脈衝或是負脈衝。
- 解碼端則需要累加編碼端所送出的正負脈衝即可得到解碼的訊號。

- 基本限制：

斜率的超負載(slope overload)

因為訊號的波形大於Delta編碼所能產生的最大斜率

$$\text{最大斜率} = (\Delta) \times (\text{取樣頻率})$$

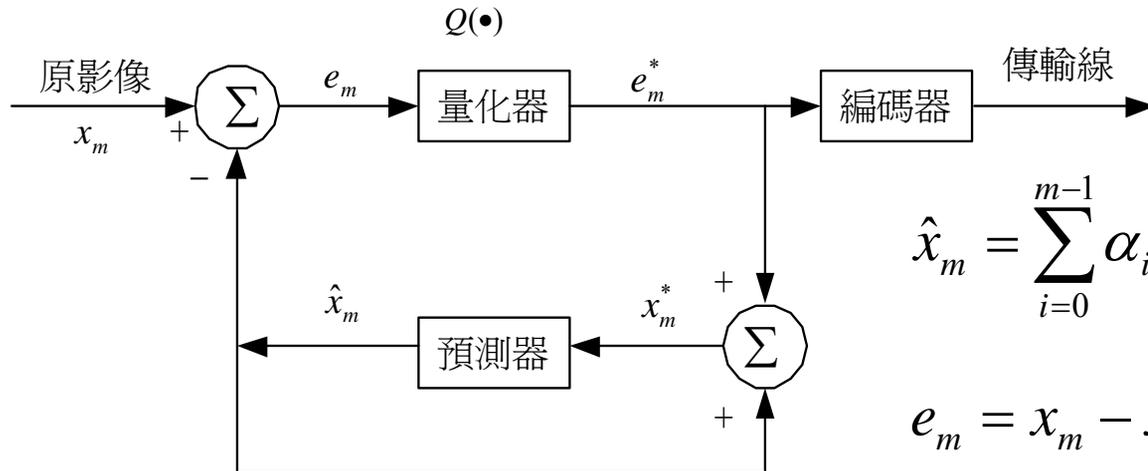
- 粒狀雜訊(granular noise)

發生在波形平坦之處

由於計算實際值與誤差值之差所得的結果很接近零，使編碼端輸出如亂數般的正與負值，於是這部分的重建波形就沿著原本的水平波形不規則的上下跳動。

# 差值脈碼調變 (DPCM)

傳送端

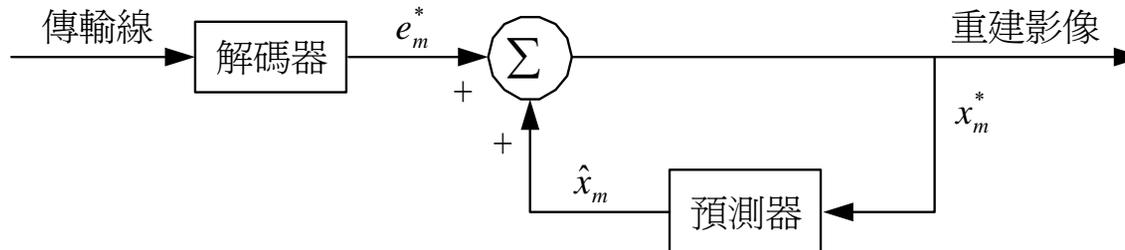


$$\hat{x}_m = \sum_{i=0}^{m-1} \alpha_i x_i \cong \sum_{i=0}^{m-1} \alpha_i x_i^*$$

$$e_m = x_m - \hat{x}_m = x_m - \sum_{i=0}^{m-1} \alpha_i x_i$$

$$e_m^* = Q(e_m)$$

接收端



# DPCM預測器

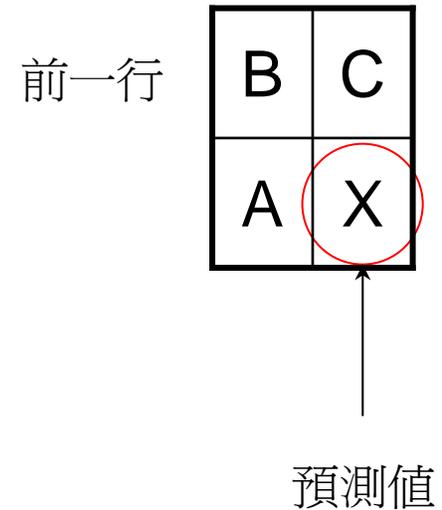
對影像而言，預測器的設定方式

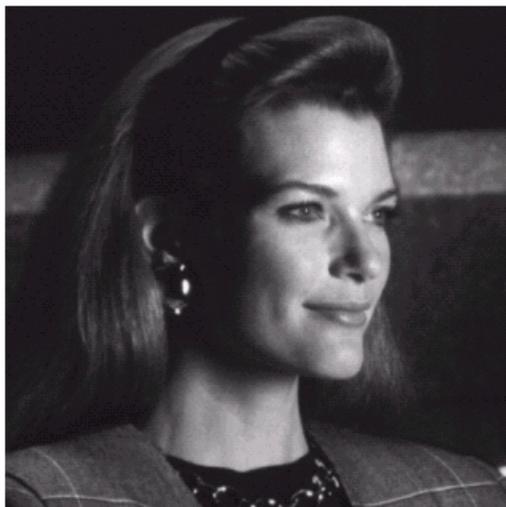
(1)  $\hat{x} = 0.97A$

(2)  $\hat{x} = 0.5A + 0.5C$

(3)  $\hat{x} = 0.75A - 0.5B + 0.75C$

(4)  $\hat{x} = \begin{cases} 0.97A & \text{if } |B - A| \geq |B - C| \\ 0.97C & \text{otherwise} \end{cases}$





(1)



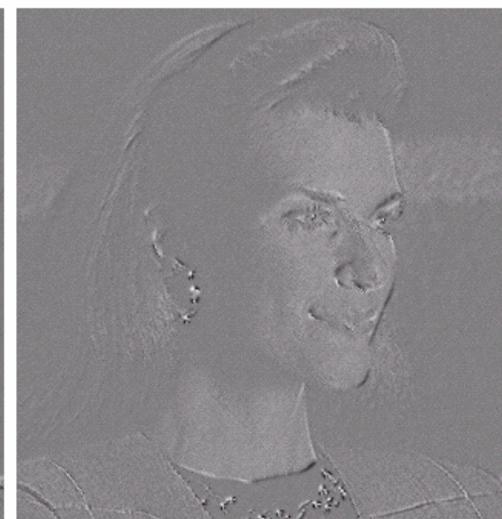
(2)



(3)

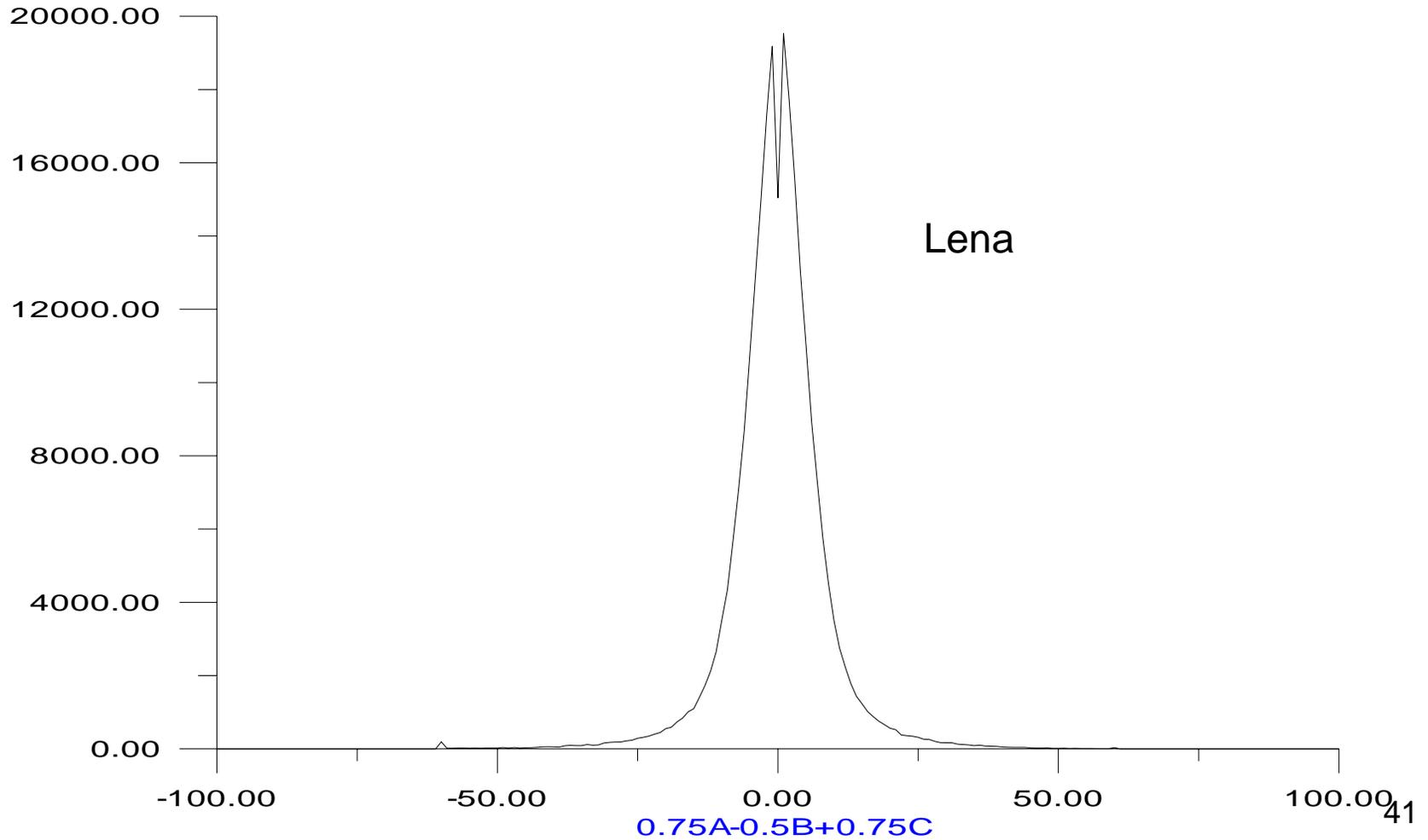


(4)



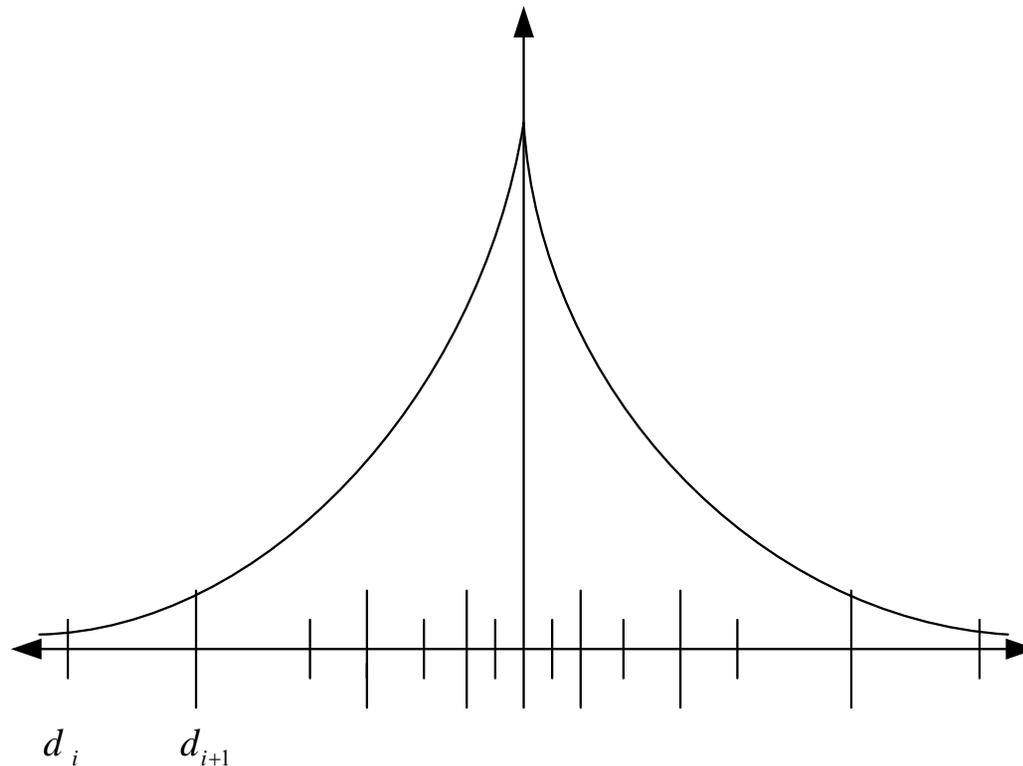
四種預測器的平均錯誤值為(1)4.9個灰階  
(2)3.7個灰階(3)3.3個灰階  
(4)4.1個灰階

# DPCM (Predictor)

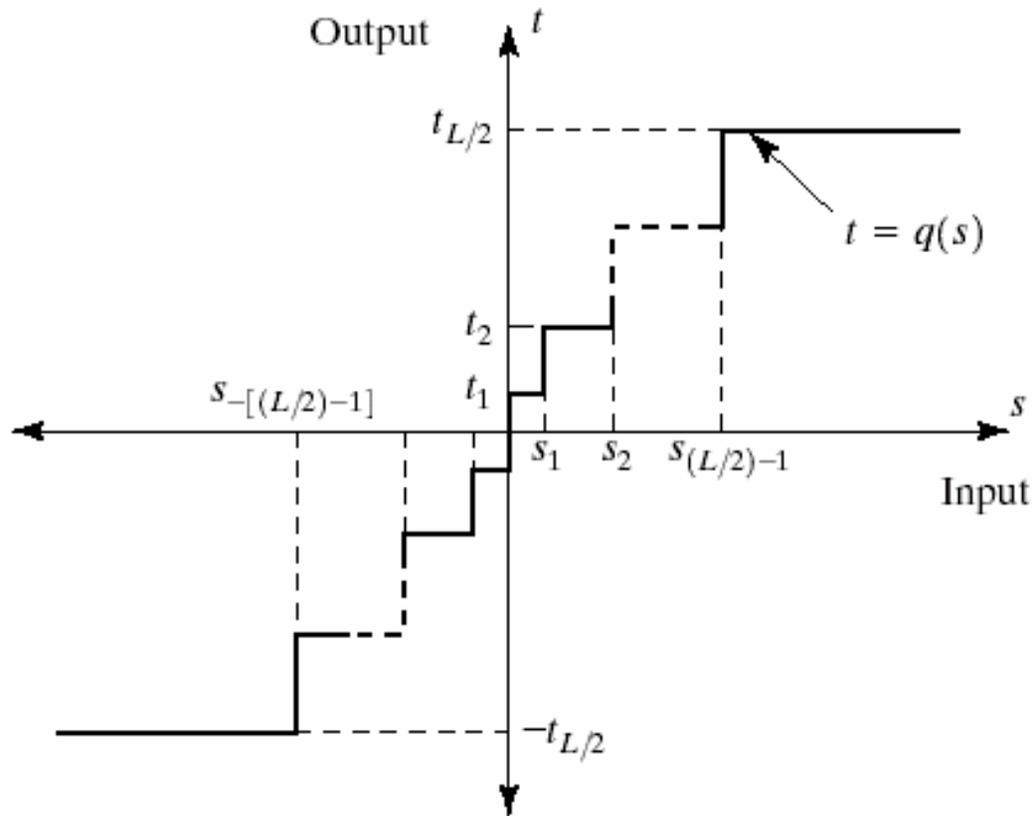


# DPCM

- 例如  $N=8$ , 誤差之機率分佈 :Laplacian distribution



# 最佳化量化器(Lloyd-Max Quantizer)



## 最佳量化(如前一頁)

是一個典型的量化函數，最佳化準則是使均方量化誤差最小化，其條件是

$$\int_{s_{i-1}}^{s_i} (s - t_i) p(s) ds = 0 \quad i = 1, 2, \dots, \frac{L}{2}$$

$$s_i = \begin{cases} 0 & i = 0 \\ \frac{t_i + t_{i+1}}{2} & i = 1, 2, \dots, \frac{L}{2} - 1 \\ \infty & i = \frac{L}{2} \end{cases}$$

$$s_{-i} = -s_i \quad t_{i-1} = -t_i$$

Levels <i>i</i>	2		4		8	
	$s_i$	$t_i$	$s_i$	$t_i$	$s_i$	$t_i$
1	$\infty$	0.707	1.102	0.395	0.504	0.222
2			$\infty$	1.810	1.181	0.785
3					2.285	1.576
4					$\infty$	2.994
$\theta$	1.414		1.087		0.731	

針對 Laplacian distribution of unit variance  
與不同階層, Lloyd-Max 量化器之設計

# DPCM

i	$(d_i, d_{i+1}] \rightarrow r_i$	Probability	Huffman code
0	$(-255, -16] \rightarrow -20$	0.025	111111
1	$(-16, -8] \rightarrow -11$	0.047	11110
2	$(-8, -4] \rightarrow -6$	0.145	110
3	$(-4, 0] \rightarrow -2$	0.278	00
4	$(0, 4] \rightarrow 2$	0.283	10
5	$(4, 8] \rightarrow 6$	0.151	01
6	$(8, 16] \rightarrow 11$	0.049	1110
7	$(16, 255] \rightarrow 20$	0.022	111110

8-level Lloyd-max quantizer for Lena

# Results

LENA			
Technique	Bit rate bits/pixel	RMSE (0-255)	SNR (dB)
1-D DPCM	1.00	18.67	22.71
1-D DPCM	2.00	9.44	28.63
1-D DPCM	3.00	5.11	33.96
2-D DPCM	1.00	14.58	27.74
2-D DPCM	2.00	6.93	31.32
2-D DPCM	3.00	3.71	36.74

***1D predictor***

$$\hat{x} = 0.97A$$

***2D predictor***

$$\hat{x} = 0.75A - 0.50B + 0.75C$$

# Results

- DPCM at 1.0 bpp (bits/pixel)



# Result

- DPCM at 2.0 bpp



# Result

- DPCM at 3.0 bpp



# 交流係數可變長度編碼 (Run Length Coding, RLC)

- 無失真影像壓縮的最簡單方法是縮減編碼冗餘性，首先統計各個run-length符號(如表6.1)出現的次數，再利用霍夫曼編碼(如下之例)。如果不做統計，則需使用表6.3-1 ~ 6.3-5是JPEG預設交流值編碼方式。

## 霍夫曼編碼 (Huffman coding)

Original source		Source reduction			
Symbol	Probability	1	2	3	4
$a_2$	0.4	0.4	0.4	0.4	0.6 0.4
$a_6$	0.3	0.3	0.3	0.3	
$a_1$	0.1	0.1	0.2	0.3	
$a_4$	0.1	0.1			
$a_3$	0.06	0.1			
$a_5$	0.04				

Original source			Source reduction			
Sym.	Prob.	Code	1	2	3	4
$a_2$	0.4	1	0.4 1	0.4 1	0.4 1	0.6 0
$a_6$	0.3	00	0.3 00	0.3 00	0.3 00	0.4 1
$a_1$	0.1	011	0.1 011	0.2 010	0.3 01	
$a_4$	0.1	0100	0.1 0100	0.1 011		
$a_3$	0.06	01010	0.1 0101			
$a_5$	0.04	01011				

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

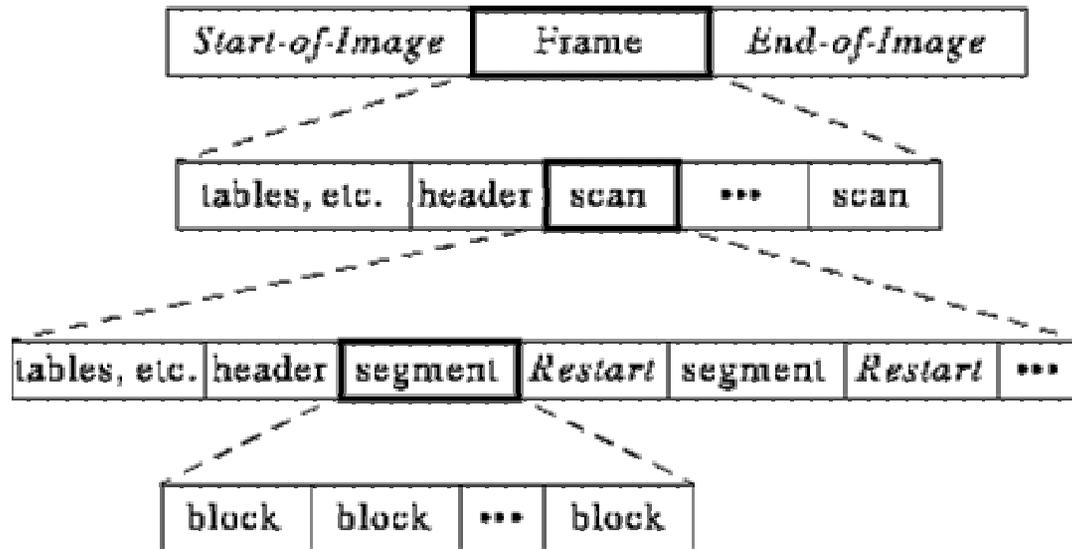
$$= 2.2 \quad \text{bits/symbol}$$

$$\text{entropy} = 2.14 \quad \text{bits/symbol}$$

$$\text{efficiency} = 0.973$$

# JPEG

- 綜觀 JPEG 位元串



- a “Frame” (圖像), a “scan” (彩色圖像中Y、或 $C_b$ 或 $C_r$ 的壓縮後的資料)，a “segment” (一串8x8方塊壓縮後的資料), a “block” (一個8x8方塊壓縮後的資料).

## 6.4 JPEG 流程範例

- 從Lena 圖取出一個8×8 方塊

$$f(x, y) = \begin{bmatrix} 139 & 144 & 149 & 153 & 155 & 155 & 155 & 155 \\ 144 & 151 & 153 & 156 & 159 & 156 & 156 & 156 \\ 150 & 155 & 160 & 163 & 158 & 156 & 156 & 156 \\ 159 & 161 & 162 & 160 & 160 & 159 & 159 & 159 \\ 159 & 160 & 161 & 162 & 162 & 155 & 155 & 155 \\ 161 & 161 & 161 & 161 & 160 & 157 & 157 & 157 \\ 162 & 162 & 161 & 163 & 162 & 157 & 157 & 157 \\ 162 & 162 & 161 & 161 & 163 & 158 & 158 & 158 \end{bmatrix}$$

- **JPEG** 所使用之**DCT** 其定義如下：

$$F(i, j) = \frac{1}{4} C(i)C(j) \sum_{x=0}^7 \sum_{y=0}^7 f(x, y) \cos\left[\frac{(2x+1)i\pi}{16}\right] \cos\left[\frac{(2y+1)j\pi}{16}\right]$$

- 這個方塊的**DCT**轉換結果為：

$$F(i, j) = \begin{bmatrix} 1260 & -1 & -12 & -5 & 2 & -2 & -3 & 1 \\ -23 & -17 & -6 & -3 & -3 & 0 & 0 & -1 \\ -11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\ -7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\ 2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\ -1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\ -3 & 2 & -4 & -2 & 2 & 1 & -1 & 0 \end{bmatrix}$$

- 量化矩陣 :DCT 轉換後之係數，由所選擇的量化矩陣予以量化(正常化)。
- JPEG 使用的典型量化矩陣  $Q(i, j)$  為

$$Q(i, j) = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

- 量化(正常化)後的係數  $F^*(i, j)$  ，公式為

$$F^*(i, j) = \mathit{round}\left(\frac{F(i, j)}{Q(i, j)}\right)$$

- 其中  $\mathit{round}(x)$  表示最接近  $x$  的整數。

$$F(i, j) = \begin{bmatrix} 79 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- 經量化(正常化)的過程後，有許多係數值是0。這些量化後的DCT係數接著以鋸齒形掃描（請看圖6.1），其結果為79, 0, -2, -1, -1, -1, 0, 0, -1, EOB。
- 直流係數(需再經DPCM的處理)與交流係數接著分別以各自的Huffman 表編碼，其中這兩個Huffman 表必須當成檔頭資訊 ( Header information)的一部份，送給接收端。

- 就直流係數值79而言，先求出這個方塊的直流係數值與前一個方塊的直流係數直之間的DPCM值，然後這個DPCM值再用Huffman表編碼。
- 交流係數值0, -2, -1, -1, -1, 0, 0, -1, EOB，使用表6.1的Huffman表進行編碼。考量0的位置，參考(N,S)表示，原交流係數值可表示為(1,2)01, (0,1)0, (0, 1)0, (0,1)0, (2, 1)0, EOB。若不統計它們的出現機率，則表6.3-1 ~ 6.3-5便可使用。