Data Compression

Advanced Data Structures and Algorithms

Associate Professor Dr. Raed Ibraheem Hamed

University of Human Development, College of Science and Technology Computer Science Department

2015 – 2016

Objectives

After reading this topic, the reader should be able to:

- Realize the need for data compression.
- Differentiate between lossless and lossy compression.
- Understand three lossless compression encoding techniques: run-length, Huffman, and Lempel Ziv.
- Understand two lossy compression methods: JPEG and MPEG.

Data compression methods

Data compression means sending or storing a smaller number of bits.

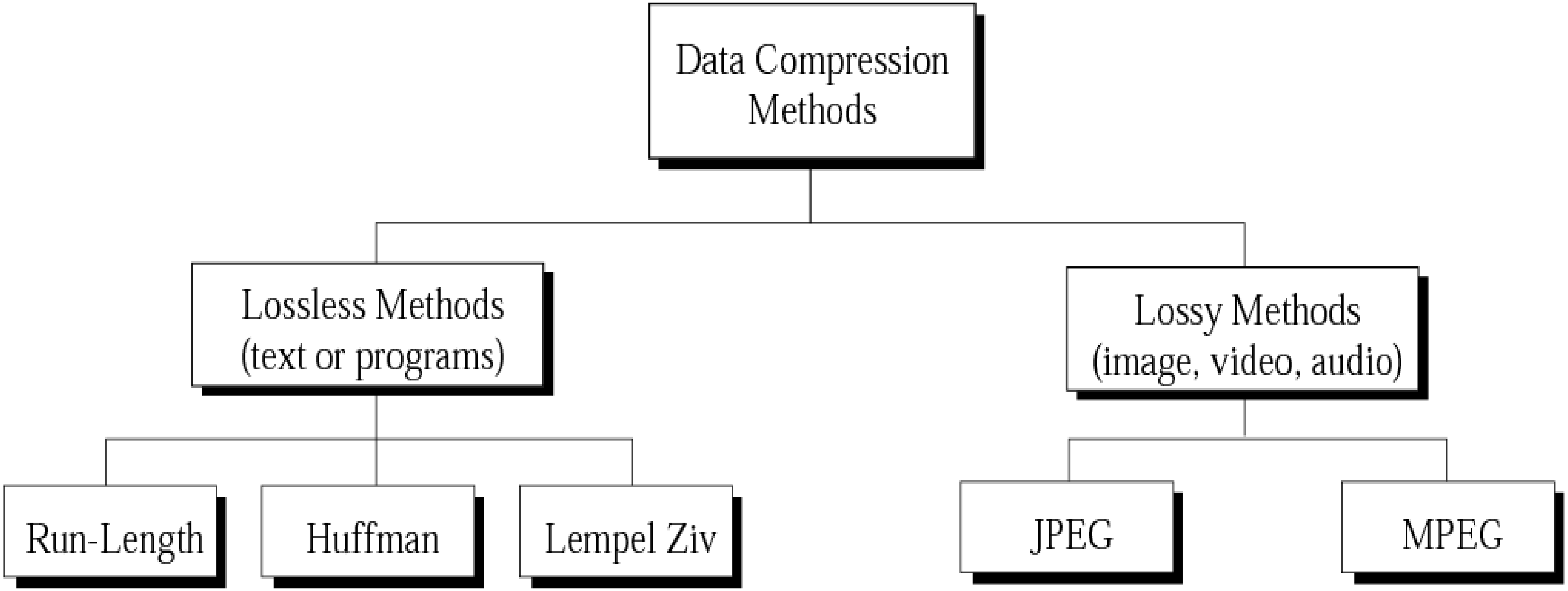


Figure 15-1

LOSSY COMPRESSION METHODS

Lossy compression methods

- Loss of information is acceptable in a picture of video.
- The **reason** is that our eyes and ears cannot distinguish subtle changes.
- Loss of information is **not** acceptable in a text file or a program file.
- For examples:
  - Joint photographic experts group (**JPEG**)
  - Motion picture experts group (**MPEG**)

JPEG and MPEG

JPEG is a commonly used method of lossy compression for **digital images**, particularly for those images produced by digital photography.

MPEG is a working group of authorities that was formed by **ISO** and **IEC** to set standards for audio and video compression and transmission.

International Organization for Standardization (ISO)
International Electrotechnical Commission (IEC)

Image compression: JPEG

JPEG gray scale example, 640 x 480 pixels

Gray scale:

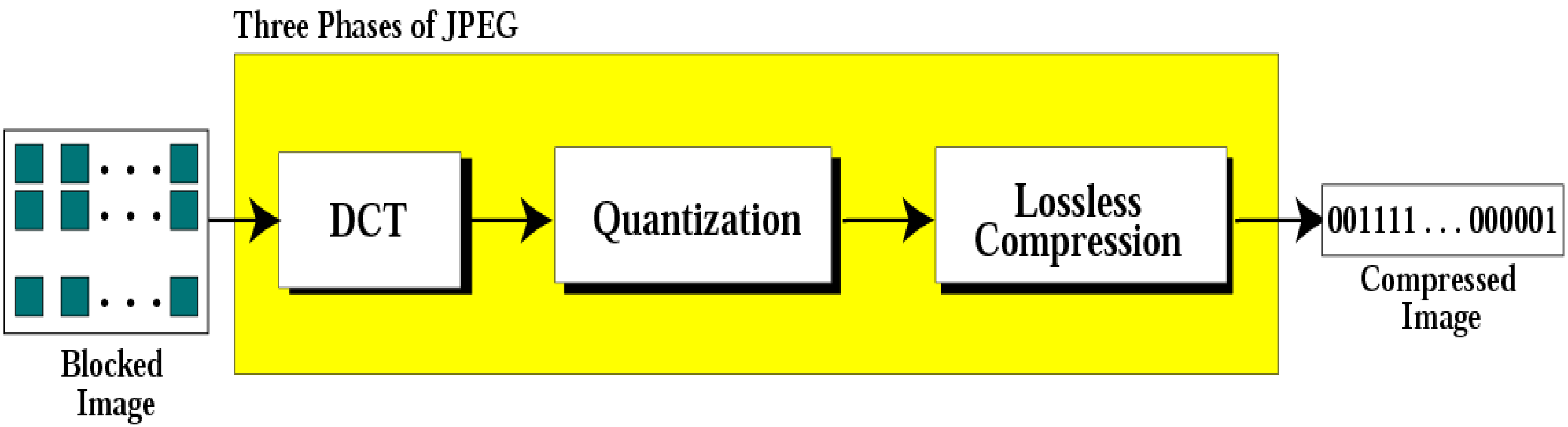
- Each pixel is usually stored as a byte (value between 0 to 255)
- A 640 x 480 grayscale image requires over **300 KB** of storage.

Monochrome:

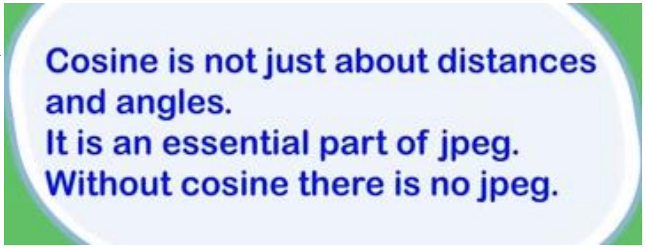
- Each pixel is stored as a single bit (0 or 1)
- A 640 x 480 monochrome image requires **37.5 KB** of storage.

Figure 15-11

JPEG process



1) **DTC: discrete cosine transform**
2) **Quantization**
3) **Compression**



DCT Process

The following is a general overview of the JPEG process.

1. The image is broken into 8×8 blocks of pixels.
2. The DCT is applied to each block.
3. Each block is compressed through quantization.

The DCT equation computes the i, j entry of the DCT of an image.

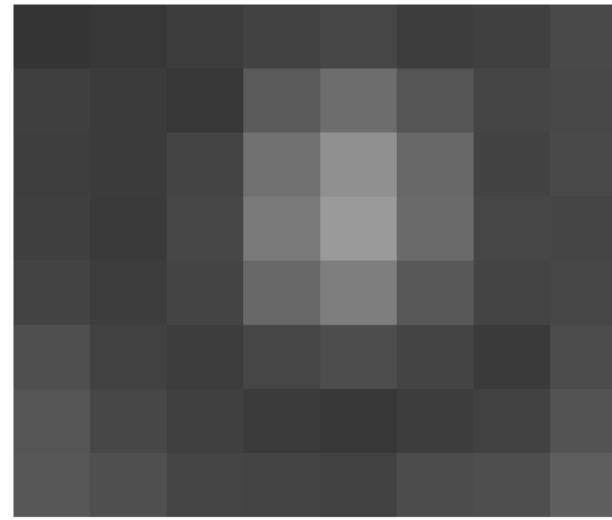
$$D(i,j) = \frac{1}{\sqrt{2N}} C(i)C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} p(x,y) \cos\left[\frac{(2x+1)i\pi}{2N} \right] \cos\left[\frac{(2y+1)j\pi}{2N} \right]$$

N is the size of the block that the DCT is done on

Equalization of Grayscale Images

The 8-bit grayscale image shown has the following values:

$$\begin{bmatrix} 52 & 55 & 61 & 66 & 70 & 61 & 64 & 73 \\ 63 & 59 & 55 & 90 & 109 & 85 & 69 & 72 \\ 62 & 59 & 68 & 113 & 144 & 104 & 66 & 73 \\ 63 & 58 & 71 & 122 & 154 & 106 & 70 & 69 \\ 67 & 61 & 68 & 104 & 126 & 88 & 68 & 70 \\ 79 & 65 & 60 & 70 & 77 & 68 & 58 & 75 \\ 85 & 71 & 64 & 59 & 55 & 61 & 65 & 83 \\ 87 & 79 & 69 & 68 & 65 & 76 & 78 & 94 \end{bmatrix}$$



grayscale values of the image.

Figure 15-12

Discrete cosine transform

Case 1: uniform gray scale

DCT is applied to blocks of 8 × 8 pixels.

Picture

20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20
20	20	20	20	20	20	20	20

$P(x,y)$

***T*(0, 0)**

160	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
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

$T(m,n)$

To get the matrix from DCT equation we will use the following equation:

$$T_{i,j} = \left\{ \begin{array}{ll} \frac{1}{\sqrt{N}} & \text{if } i = 0 \\ \sqrt{\frac{2}{N}} \cos\left[\frac{(2j+1)i\pi}{2N} \right] & \text{if } i > 0 \end{array} \right\}$$

Figure 15-13

Discrete cosine transform

Case 2: two sections

Picture

20	20	20	20	50	50	50	50
20	20	20	20	50	50	50	50
20	20	20	20	50	50	50	50
20	20	20	20	50	50	50	50
20	20	20	20	50	50	50	50
20	20	20	20	50	50	50	50
20	20	20	20	50	50	50	50
20	20	20	20	50	50	50	50

$P(x,y)$

280	-109	0	39	0	-25	0	22
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	0
0	0	0	0	0	0	0	0

$T(m,n)$

Figure 15-14

Discrete cosine transform

Case 3: gradient gray scale

Picture

20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90
20	30	40	50	60	70	80	90

$P(x,y)$

400	-146	0	-31	-1	3	-1	-8
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	0
0	0	0	0	0	0	0	0

$T(m,n)$

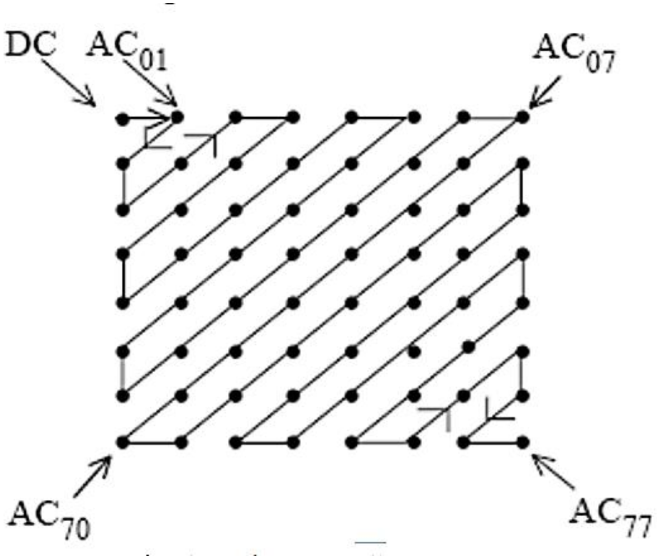
DCT discussion

The **DCT** transformation creates table *T* from table *P*.

The **DC** value gives the average **value** of the pixels.

The **AC** values gives the **changes**.

Lack of changes in neighboring pixels creates 0s.



Quantization

After the T table is created, the values are quantized to **reduce the number of bits** needed for encoding.

Quantization:

- Divide the number by a constant and then drop the fraction.
- The quantizing phase is **not** reversible.
- Some information will be lost.

Compression

After quantization, the values are read from the table, and **redundant 0s** are removed.

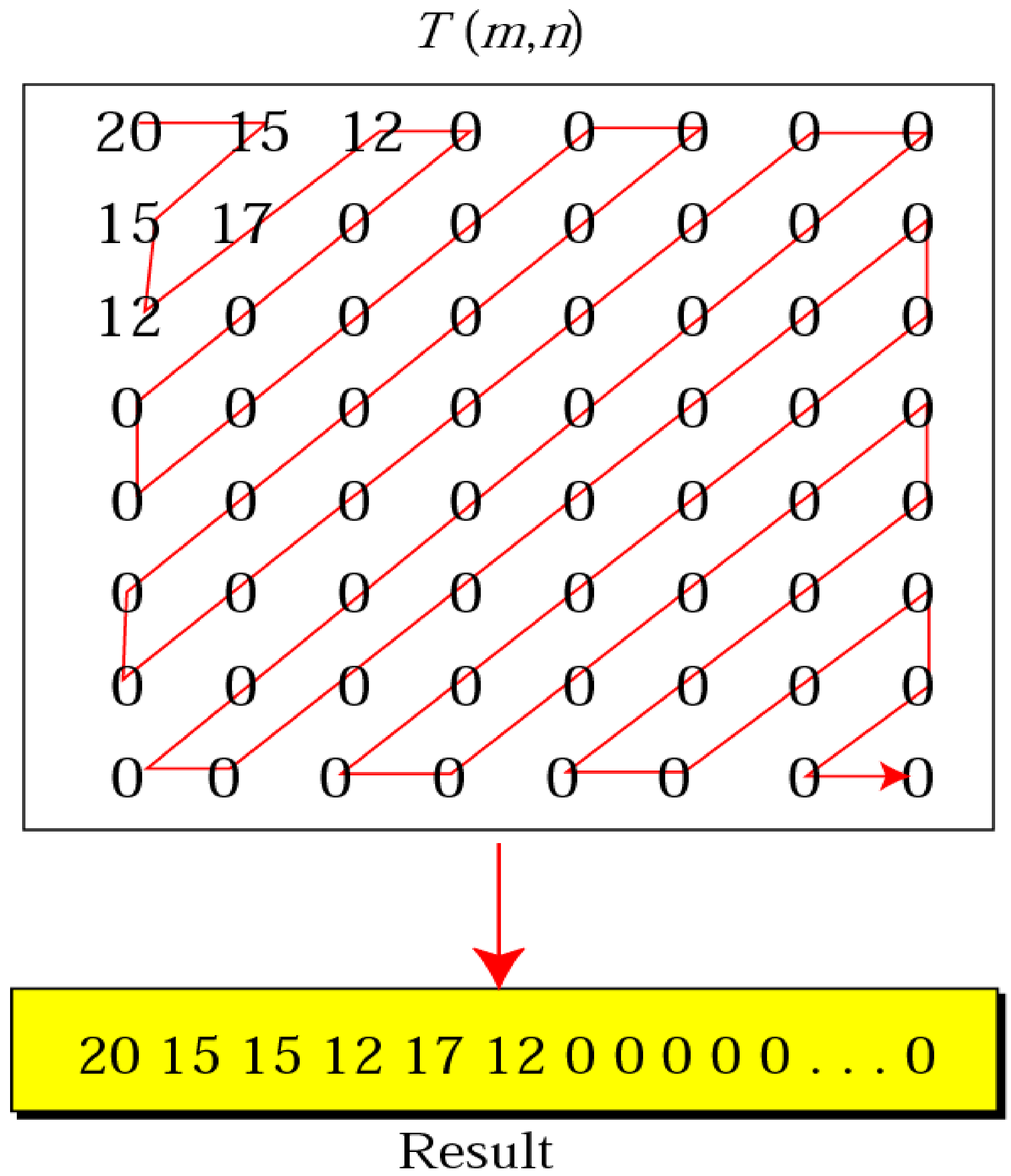
The reason is that if the picture does not have fine changes, the **bottom right corner** of the T table is all **0s**.

Fig. 15.15

Figure 15-15

Reading the table

Zig Zag Sequence



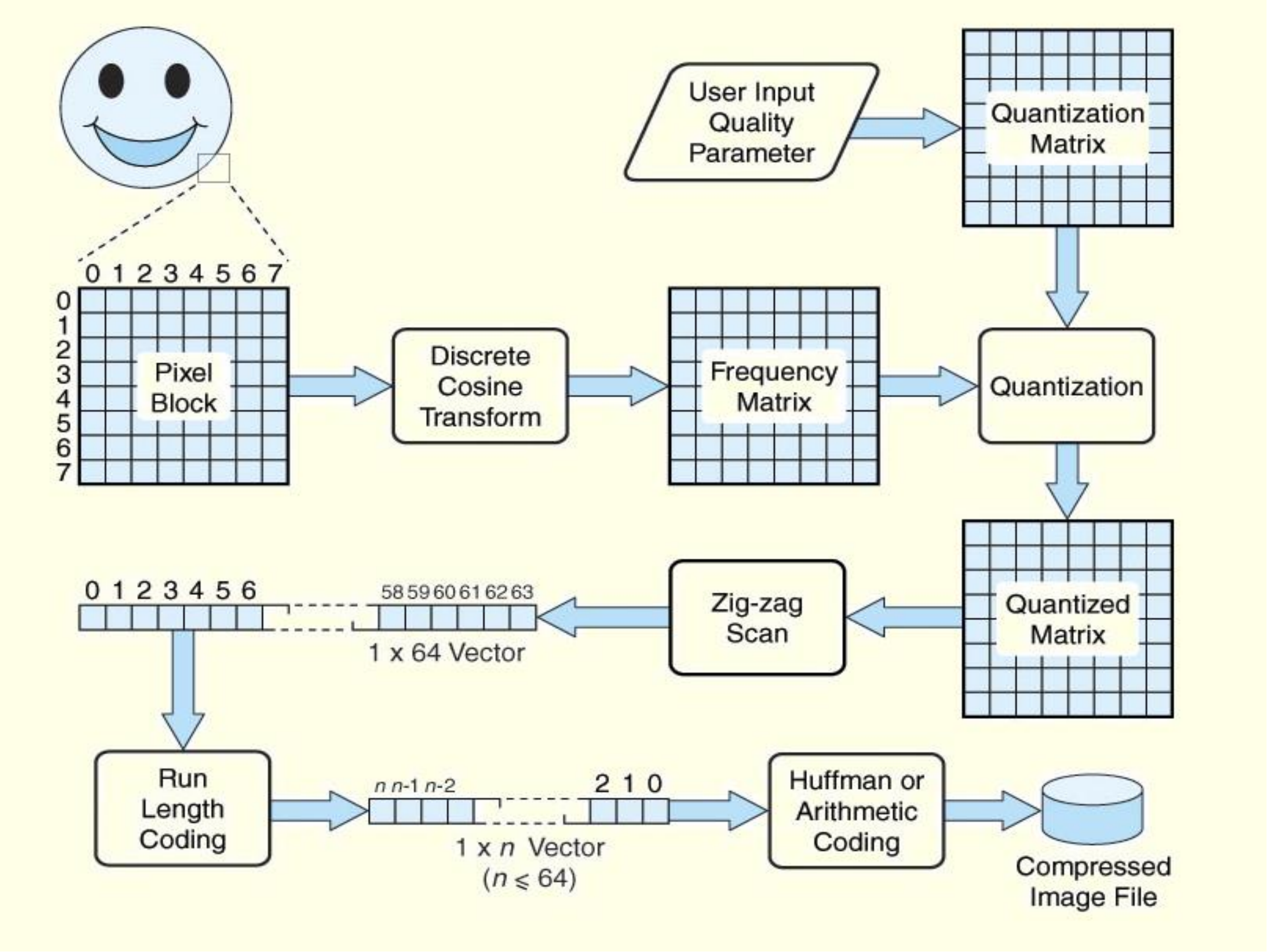

100%
BSM


BSM
50%


10%


5%

The JPEG algorithm is illustrated on the next slide.



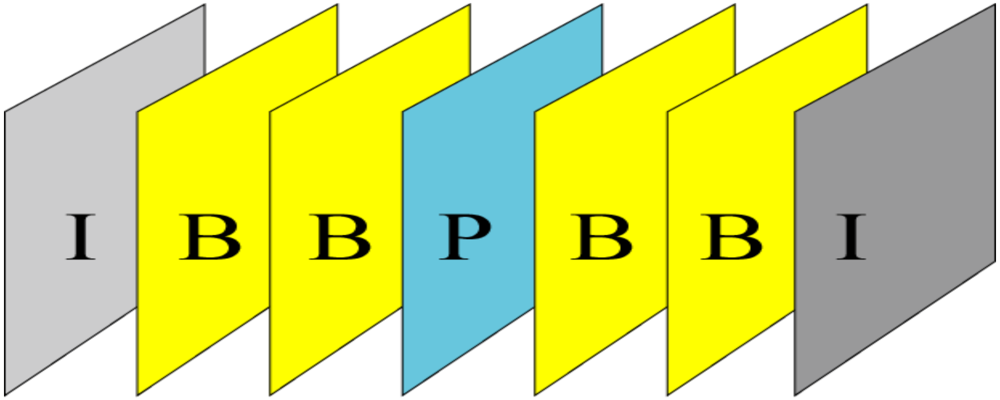
Video compression--MPEG

MPEG method

- **Temporal compression**
  - The temporal compression removes the redundant frames.
  - MPEG method first divides frames into three categories: I-frames, P-frames, B-frames.

Figure 15-16

MPEG frames

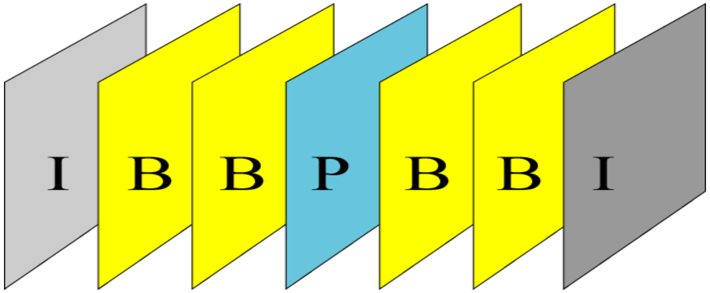


I-frames: (intra-coded frame)

- It is an independent frame that is **not** related to any other frame.
- They are present at **regular intervals**.
- I-frames are independent of other frames and **cannot** be constructed from other frames.

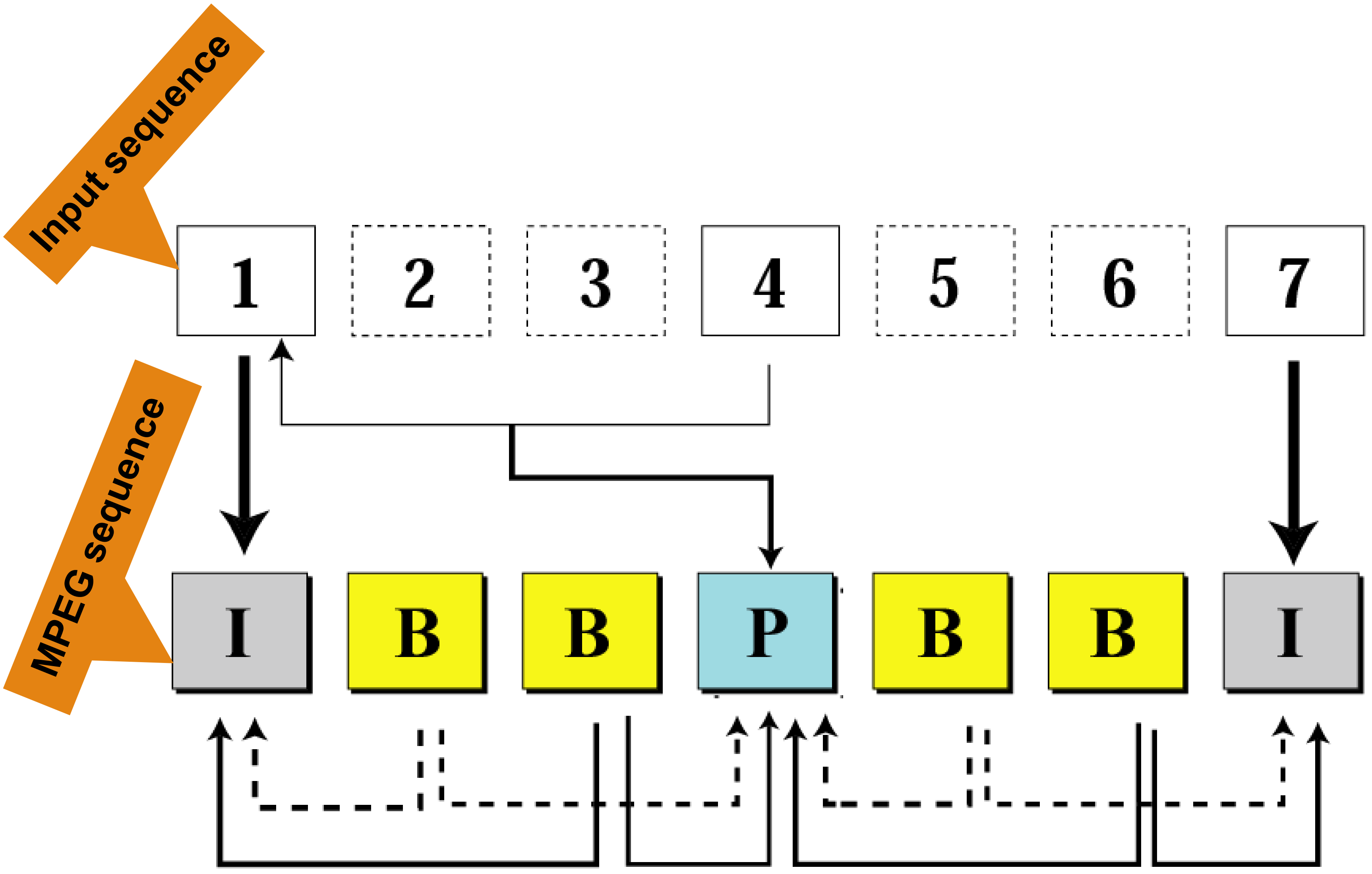
Figure 15-16

MPEG frames



- **P-frames**: (predicted frame)
  - It is related to the preceding I-frame or P-frame.
  - Each P-frame contains only the changes from the preceding frame.
  - P-frames can be constructed only from previous I- or P-frames.
- **B-frames**: (bidirectional frame)
  - It is relative to the preceding and following I-frame or P-frame.
  - Each B-frame is relative to the past and the future.
  - A B-frame is never related to another B-frame.

MPEG frame construction




Thank
you