Course Code 005636 (Fall 2019)

Multimedia

Video Compression Principles

Prof. S. M. Riazul Islam, Dept. of Computer Engineering, Sejong University, Korea

E-mail: riaz@sejong.ac.kr

Contents

Principles of Video Compression

- Concepts of Motion Estimation and Compensation
- H.261
- MPEG

MPEG

- The first standard to be finalized for video compression was MPEG-1 for interactive video on CD and for digital audio broadcasting

Video Compression

- Video is a temporal combination of frames. Each frame can be considered as an image comprising spatial combination of pixels.
- Exploiting spatial redundancy as of JPEG

- Since the frame rate of the video is often relatively high (e.g: > 15 frames per second) and the camera parameters (focal length, position and viewing angle) usually do not change rapidly between-frames, the contents of consecutive frames are usually similar, unless certain objects in the scene extremely fast.
- Exploiting the temporal redundancy of a video.

Example: Temporal redundancy

- Consecutives frames are often almost identical

Video Compression Based on Motion Compensation

- Temporal redundancy is often significant and it is exploited, so that not every frame of the video needs to be coded independently as a new image. Instead, the difference between the current frame and other frame(s) in the sequence is coded.
- If redundancy between them is great enough, the difference images could consist mainly of small values and low entropy, which is good for compression.

- Although a simplistic way of deriving the difference image is to subtract one image from the other (pixel by pixel), such an approach is ineffective in yielding a high compression ratio.

- Since the main cause of the difference between frames is camera and/or object motion, these motion generators can be "compensated" by detecting the displacement of corresponding pixels or regions in these frames and measuring their differences.

Video Compression Based on Motion Compensation

The three main steps of these algorithms are:

1. Motion estimation (motion vector search)
2. Motion compensation-based prediction
3. Derivation of the prediction error—the difference.

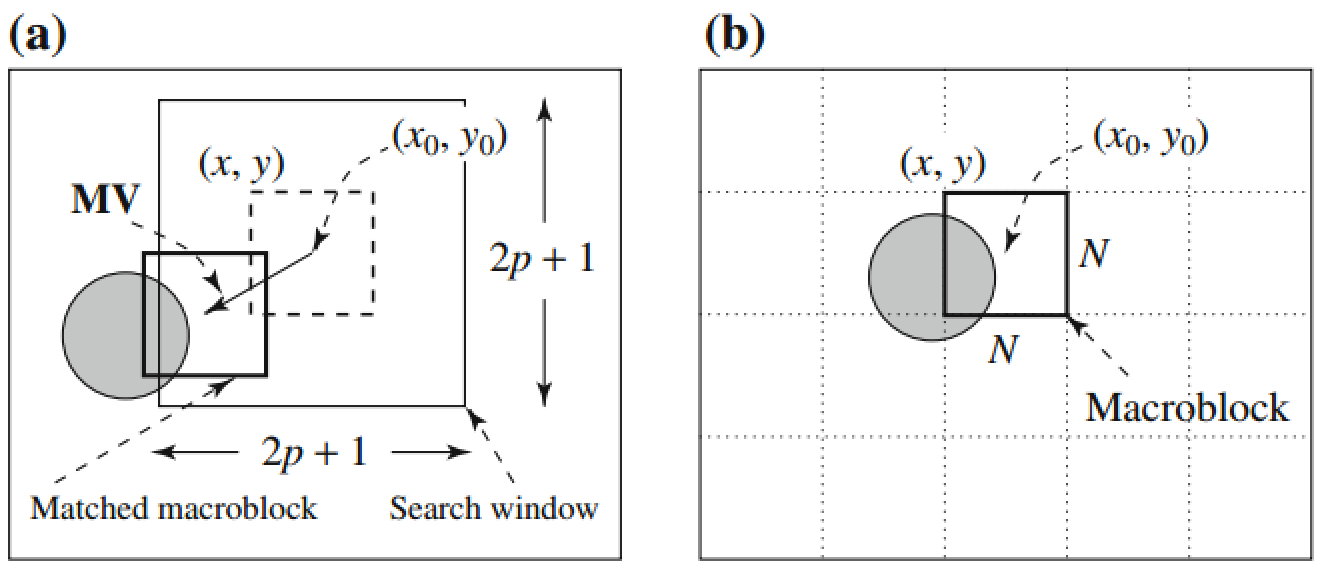
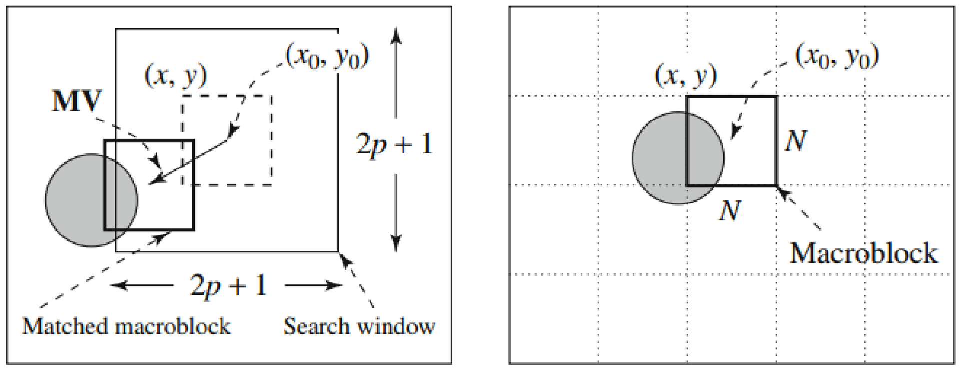


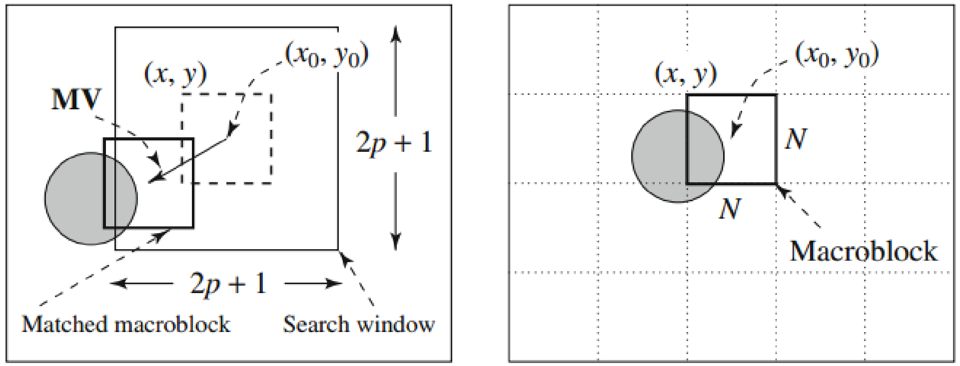
Fig. 10.1 Macroblocks and motion vector in video compression: **a** reference frame; **b** target frame

Video Compression Based on Motion Compensation



- The displacement of the reference macroblock to the target macroblock is called a motion vector **MV**.
- Figure 10.1 shows the case of forward prediction, in which the Reference frame is taken to be a previous frame.
- If the Reference frame is a future frame, it is referred to as backward prediction.
- The difference of the two corresponding macroblocks is the prediction error.

Motion Vector Search



Mean Absolute Distance (MAD)

$$MAD(i, j) = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} |C(x + k, y + l) - R(x + i + k, y + j + l)| , \quad (10.1)$$

where N is the size of the macroblock.

The goal of the search is to find a vector (i, j) as the motion vector $\mathbf{MV} = (u, v)$, such that $MAD(i, j)$ is minimum:

$$(u, v) = [(i, j) \mid MAD(i, j) \quad \text{is minimum}, \quad i \in (-p, p), j \in (-p, p)] \quad (10.2)$$

?? MSE

Motion Vector Search

Sequential Search

- The simplest method for finding motion vectors is to sequentially search the whole $(2p + 1)\times (2p + 1)$ window in the Reference frame.

- A macroblock centered at each of the positions within the window is compared to the macroblock in the Target frame, pixel by pixel, and their respective M AD is then derived using Eq. (10.1).

- Clearly, the sequential search method is very costly. From Eq. (10.1), each pixel comparison requires three operations (subtraction, absolute value, addition).

- Thus the cost for obtaining a motion vector for a single macroblock is $(2p + 1).(2p + 1) \cdot N^2 \cdot 3 \Rightarrow O(p^2N^2)$

Motion Vector Search

Sequential Search

As an example, let's assume the video has a resolution of 720×480 and a frame rate of 30 fps; also, assume $p = 15$ and $N = 16$. The number of operations needed for each motion vector search is thus

$$(2p + 1)^2 \cdot N^2 \cdot 3 = 31^2 \times 16^2 \times 3.$$

Considering that a single image frame has $\frac{720 \times 480}{N \cdot N}$ macroblocks, and 30 frames each second, the total operations needed per second is

$$\begin{aligned} \text{OPS\_per\_second} &= (2p + 1)^2 \cdot N^2 \cdot 3 \cdot \frac{720 \times 480}{N \cdot N} \cdot 30 \\ &= 31^2 \times 16^2 \times 3 \times \frac{720 \times 480}{16 \times 16} \times 30 \approx 29.89 \times 10^9. \end{aligned}$$

This would certainly make real-time encoding of this video difficult.

Solutions: 2D Logarithmic Search , Hierarchical Search

Motion Vector Search

Table 10.1 Comparison of computational cost of motion vector search methods according to the examples

Search method	OPS_per_second for 720 × 480 at 30 fps	
	$p = 15$	$p = 7$
Sequential search	29.89×10^9	7.00×10^9
2D logarithmic search	1.25×10^9	0.78×10^9
Three-level hierarchical search	0.51×10^9	0.40×10^9

H.261

- H.261 is an earlier digital video compression standard. Was developed as a standard for digital telephony for ISDN services
- Its principle of motion compensation–based compression is very much retained in all later video compression standards.

- Limits the image of just two sizes:
  - The common intermediate format (CIF)
  - Quarter CIF (QCIF)

CIF	**352**	**288**	**8 bits/pixel**	**15 fps**	**24.33 Mbps**	**H.261 112 Kbps**
QCIF	176	144	8 bits/pixel	10 fps	4 Mbps	< 64 Kbps

H.261

- I (Intra-) Frame: Self-contained JPEG encoded, appears periodically.
- P (Predictive) Frames: Block-by-block difference with the preceding I or P frames.

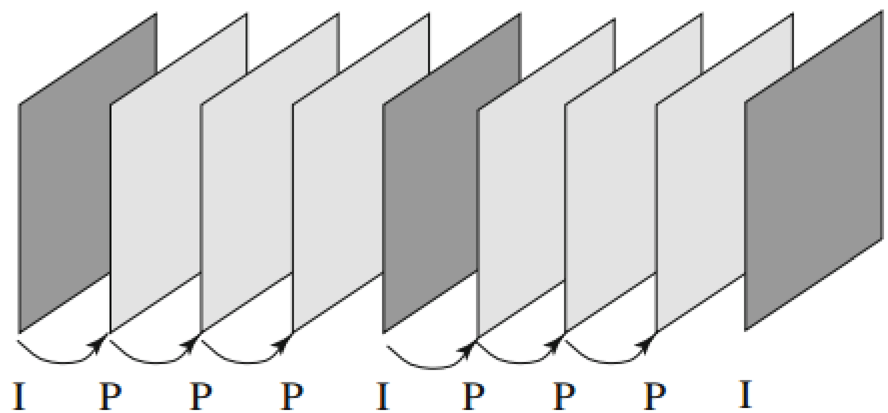


Fig. 10.4 H.261 Frame sequence

H.261

- I (Intra-) Frame: Self-contained JPEG encoded, appears periodically.

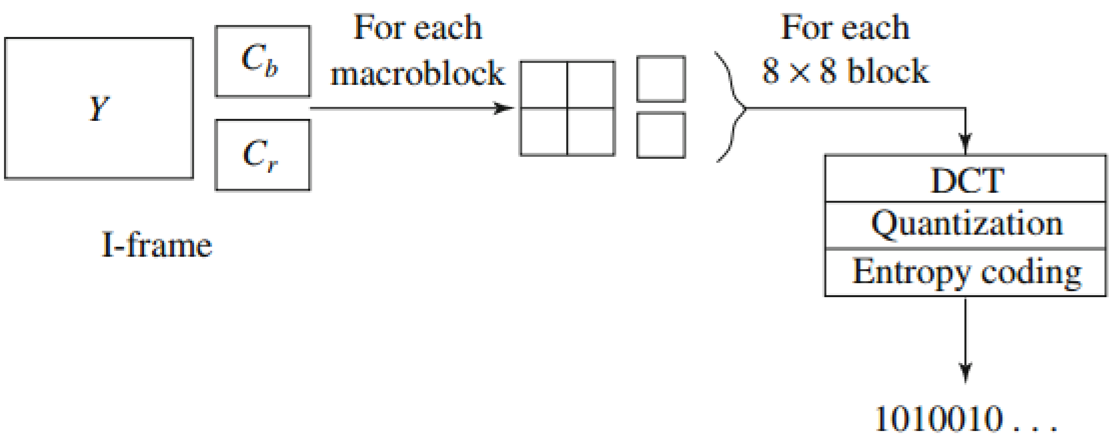


Fig. 10.5 I-frame coding

H.261

- P (Predictive) Frames: Block-by-block difference with the preceding I or P frames.

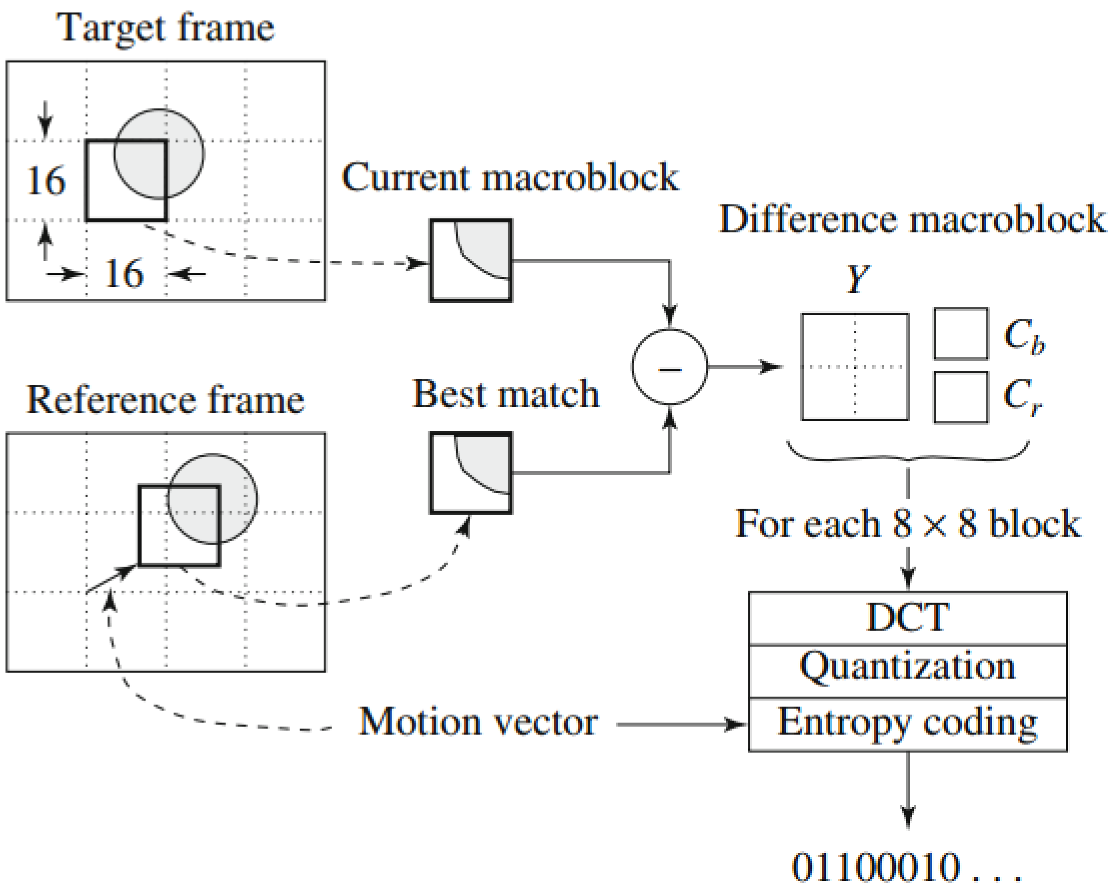


Fig. 10.6 H.261 P-frame coding based on motion compensation

H.261

Group of Block (GOB) layer

- H.261 pictures are divided into regions of 11 × 3 macroblocks (i.e., regions of 176× 48 pixels in luminance images), each of which is called a Group of Blocks (GOB).
- Example: The CIF image has 2 × 6 GOBs, corresponding to its image resolution of 352 × 288 pixels

GOB 0	GOB 1
GOB 2	GOB 3
GOB 4	GOB 5
GOB 6	GOB 7
GOB 8	GOB 9
GOB 10	GOB 11

CIF

MPEG-1

- In general, MPEG-1 adopts the CCIR601 digital TV format, also known as Source Input Format (SIF).
- MPEG-1 supports only noninterlaced video.
- Normally, its picture resolution is 352 × 240 for NTSC video at 30 fps, or 352 × 288 for PAL video at 25 fps. It uses 4:2:0 chroma subsampling

- I (Intracode) Frame: Self-contained JPEG encoded, appears periodically.
- P (Predictive) Frames: Block-by-block difference with the preceding I or P frames.
- B (Bidirectional) Frames: Differences with the preceding and the following I or P frame.
- D (DC-coded) Frame: Block averages used for fast forward.

MPEG-1

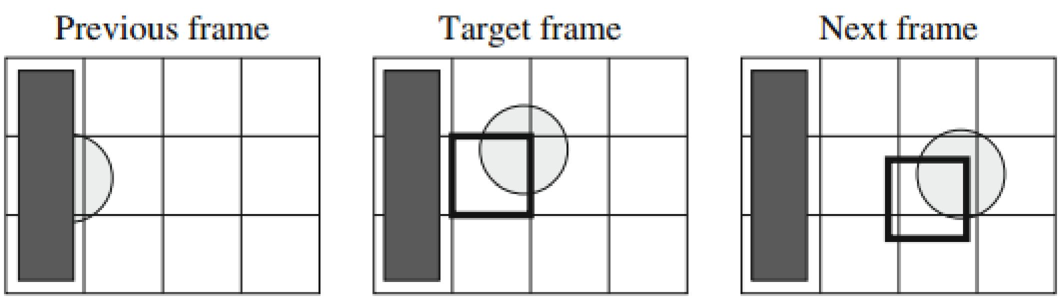


Fig. 11.1 The need for bidirectional search

MPEG-1

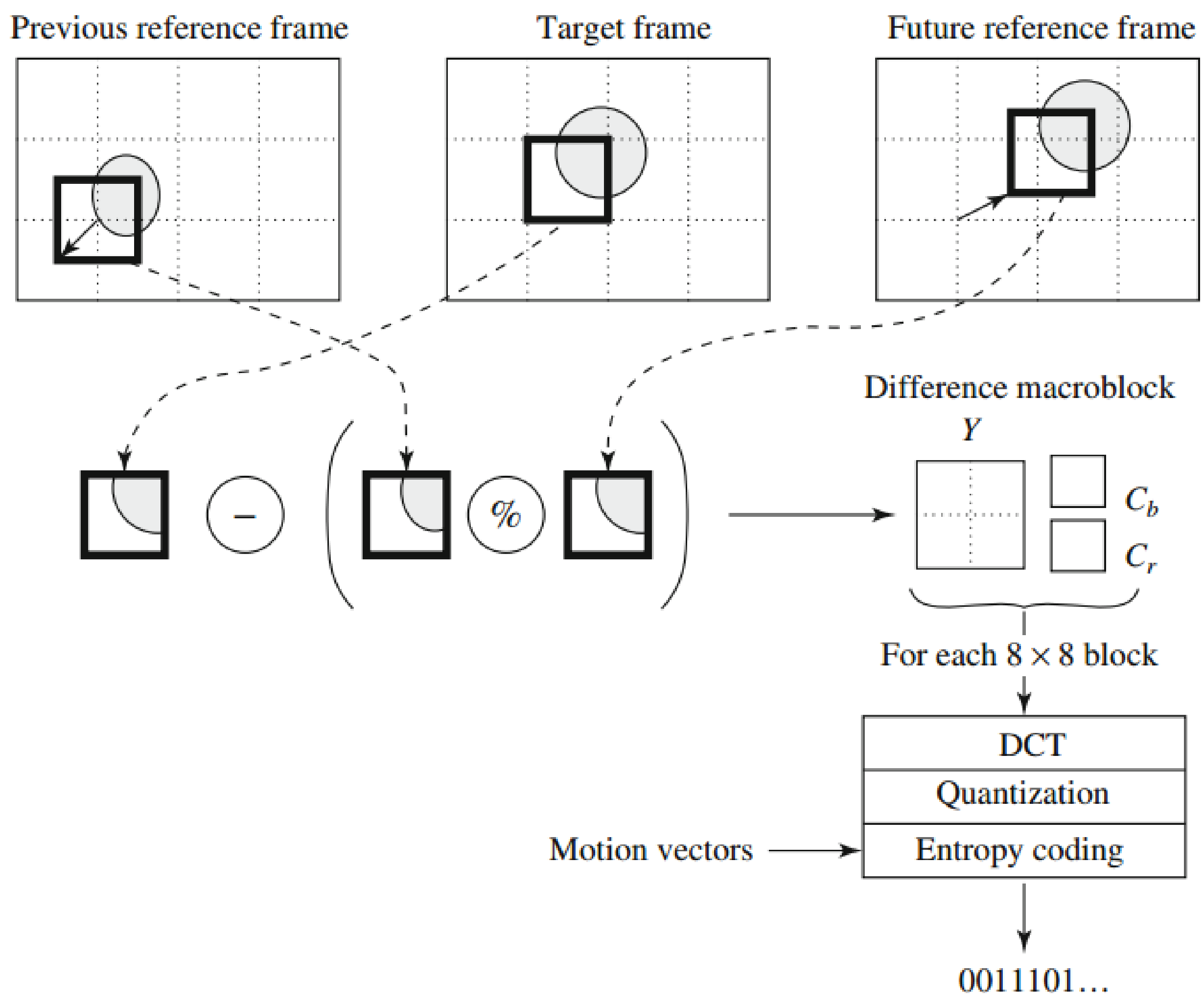


Fig. 11.2 B-frame coding based on bidirectional motion compensation

MPEG-1

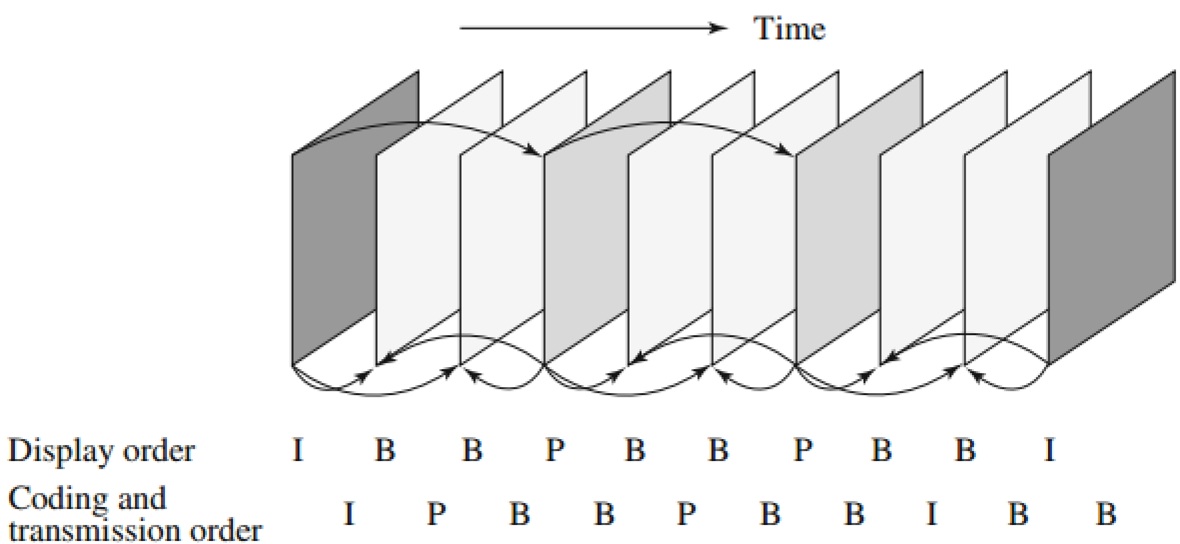
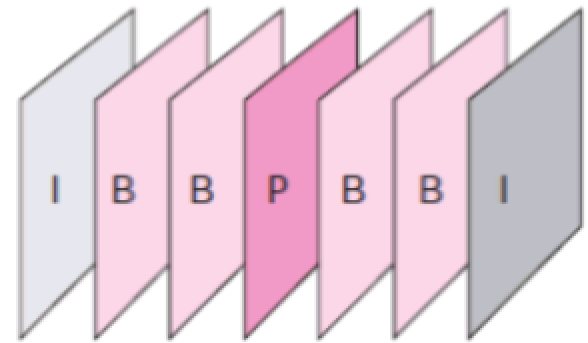


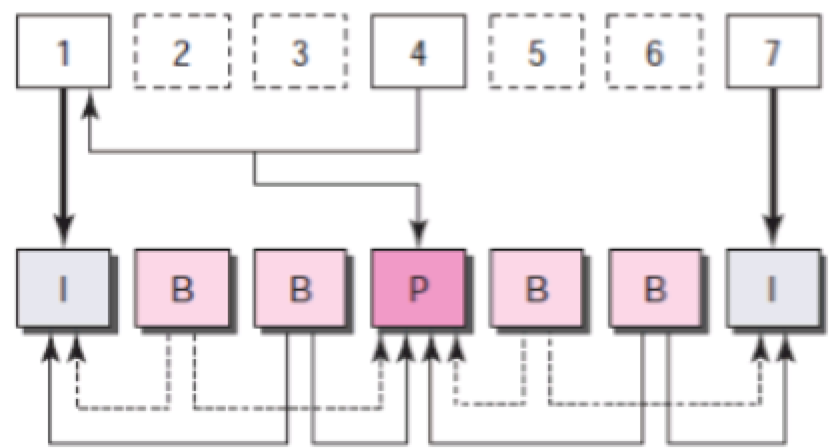
Fig. 11.3 MPEG frame sequence

MPEG Frames

- I-frames: Independent frame that is not related to any other frame.
- They are present at regular intervals



- P-frame: Related to the preceding I-frame or P-frame.
- P-frames can be constructed only from previous I- or P-frames.

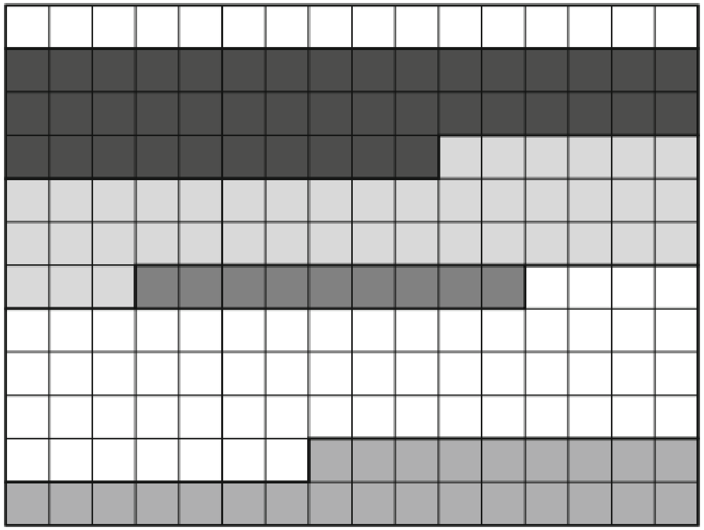


- B-frame: Each B-frame is relative to the past and the future.
- B-frame is never related to another B-frame.

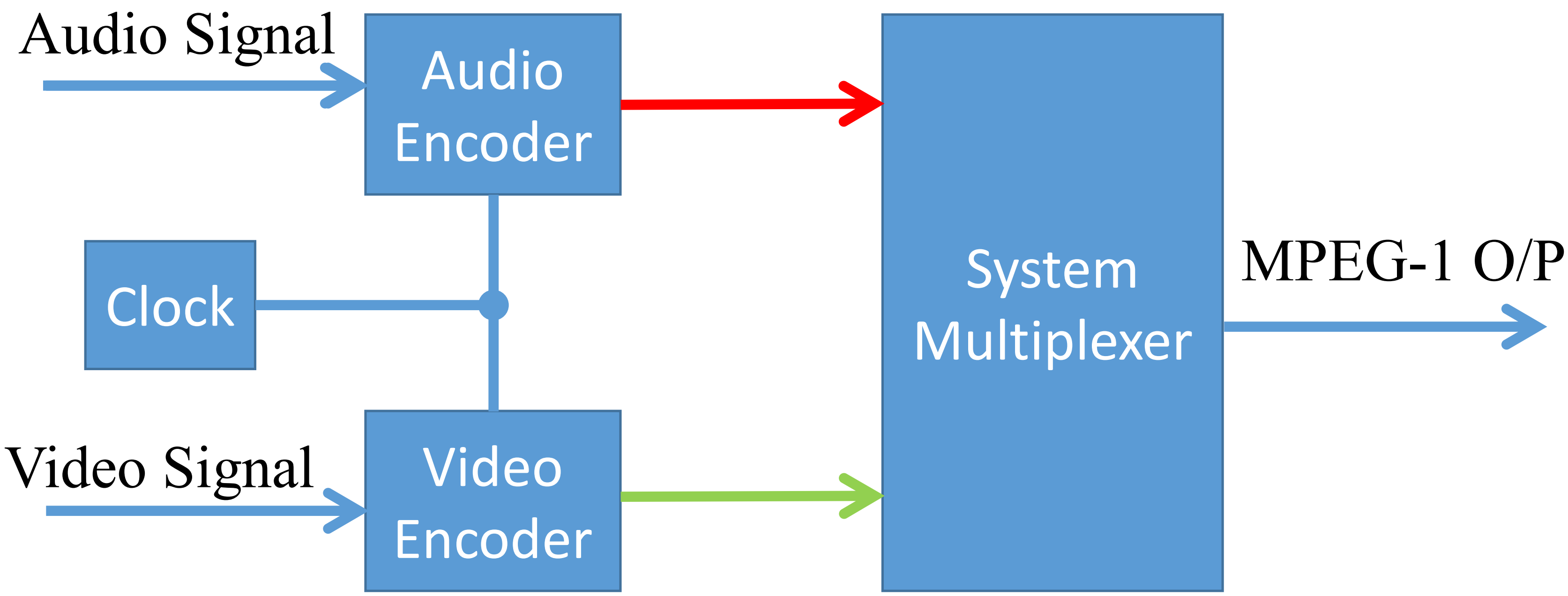
MPEG-1

Concept of Slices

- Instead of GOBs, as in H.261, an MPEG-1 picture can be divided into one or more slices, which are more flexible than GOBs.
- They may contain variable numbers of macroblocks in a single picture and may also start and end anywhere, as long as they fill the whole picture. Each slice is coded independently.
- For example, the slices can have different scale factors in the quantizer. This provides additional flexibility in bitrate control



MPEG-1



- MPEG-1 Components: Audio, Video and System
- 9o KHz clock: Provides the current time value (timestamps)
- Out of the specified total data rate of 1.5 mbps, 1.2 Mbps is intended for coded video, and 256 kbps(kilobits per second) can be used for stereo audio

Display Order and Coding Order

- Pictures are coded and decoded in a different order than they are displayed: Due to bidirectional prediction for B pictures.

- Source order and encoder input order:

I(1) B(2) B(3) P(4) B(5) B(6) P(7) B(8) B(9) P(10) B(11) B(12) I(13)

- Encoding order and order in the coded bit stream:

I(1) P(4) B(2) B(3) P(7) B(5) B(6) P(10) B(8) B(9) I(13) B(11) B(12)

- Decoder output order and display order (same as input):

I(1) B(2) B(3) P(4) B(5) B(6) P(7) B(8) B(9) P(10) B(11) B(12) I(13)

➢ Encoder can take minutes or hours to do the encoding, but decoder has to be fast!

Missing I-Frame in MPEG

Corrupted Video!

https://youtu.be/GGcWLY5lfTI

MPEG-2

- Similar to MPEG-1, but developed for digital TV.

Difference

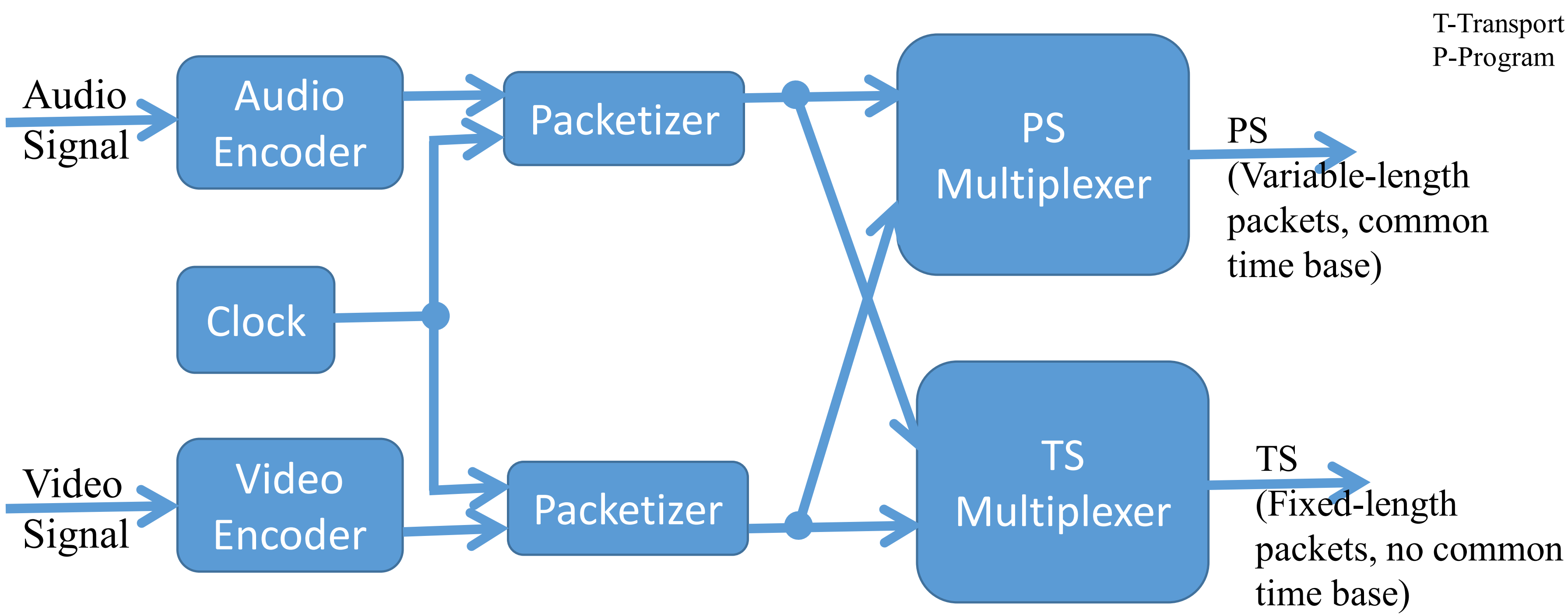
- Support interlaced video
- D-frames are not supported
- DCT is 10x10 for better quality
- Supports higher resolutions
- Support five different profiles

MPEG-1: VCD/VCR
MPEG-2: DVD

MPEG-2

HDTV	**1920**	**1080**	**24 bits/pixel**	**60 fps**	**2986 Mbps**	**25-34 Mbps**
TV	720	576	24 bits/pixel	25 fps	498 Mbps	3-6 Mbps

MPEG-2



- More general way of multiplexing
- Each streams are packetized with time stamps
- The output of each packetizer is a packetized elementary system (PES) having 30 header fields.

MPEG-4

- Object-based coding approach (unlike block-based or frame-based coding as in MPEG-1 and MPEG-2)

- Started as a standard for a very low bitrates for use in portable applications like videophones.

- It includes much more than just data compression

MPEG-4

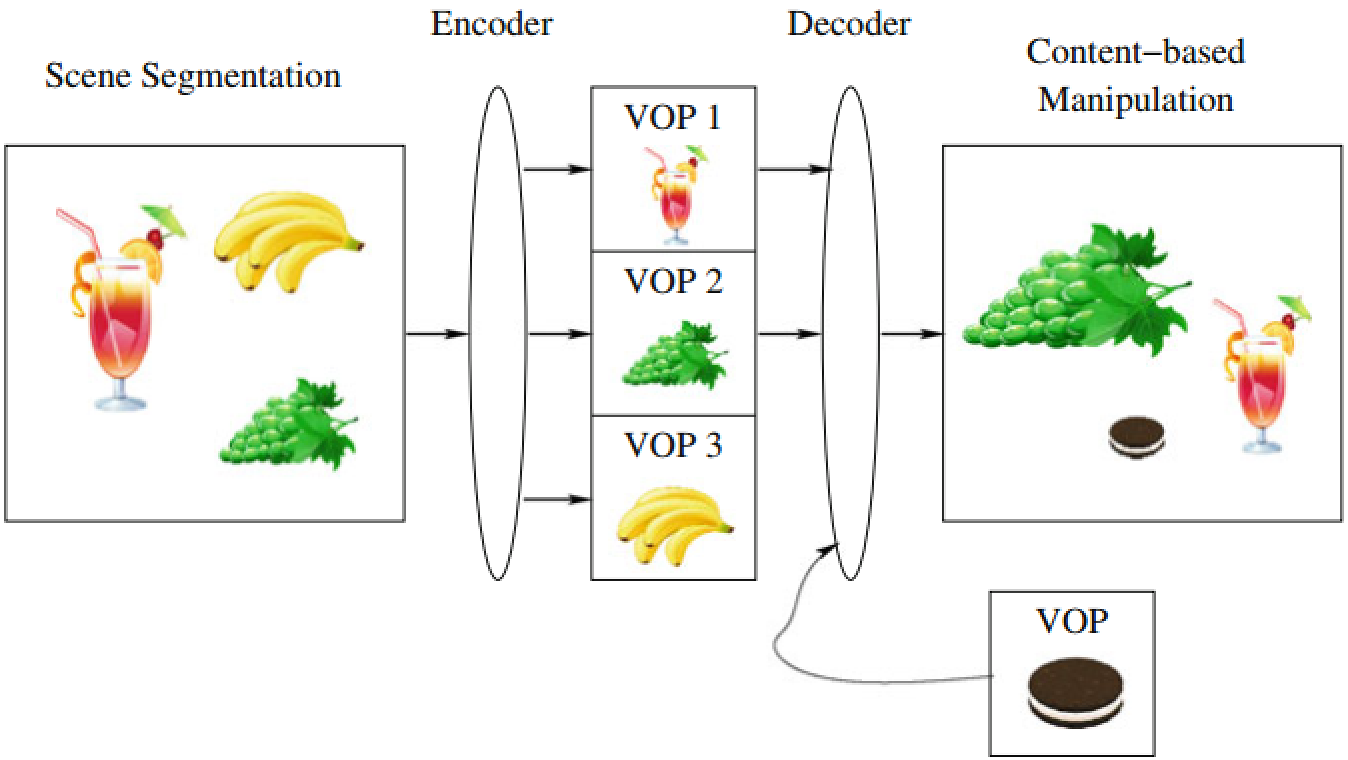


Fig. 11.11 Composition and manipulation of MPEG-4 videos (VOP = Video object plane)

MPEG-4

Important Functionalities

- Content-based MM access tools
- Content-based manipulations and bit-stream editing.
- Hybrid natural and synthetic data coding
- Improved temporal random access; and Improved coding efficiency
- Coding of multiple concurrent data streams
- Robustness in error-prone environment
- Content-based scalability.

MPEG Standards

- MPEG-1 is intended for intermediate data rates, on the order of 1.5 Mbit/sec.
- MPEG-2 is intended for high data rates of at least 25 Mbit/sec.
- MPEG-3 was intended for HDTV compression but was found to be redundant and was merged with MPEG-2.
- MPEG-4 is intended for very low data rates of less than 64 Kbit/sec.

Q&A