Module 8

VIDEO CODING STANDARDS

Lesson 25

MPEG-4 Standard

Version 2 ECE IIT, Kharagpur

Lesson Objectives

At the end of this lesson, the students should be able to:

1. State the basic objectives of MPEG-4 standard.
2. Explain the concept of content based interactivity.
3. Explain the toolbox approach of MPEG-4.
4. Define video object planes, video objects and video object layers.
5. Define VOP image window and shape-adaptive macro block grid.
6. Explain shape coding, motion estimation and texture coding applicable to MPEG-4.
7. Explain the spatial and temporal scalability aspects of MPEG-4.
8. State the basic philosophy of sprite coding.
9. Explain the facial feature animation capabilities of MPEG-4.

25.0 Introduction

In lesson-23 and lesson-24, we presented the MPEG-1 and MPEG-2 standards, which even today are the most widely used video coding standards. In 1994, the MPEG committee introduced a new standardization phase, called MPEG-4, which finally became a standard in 2000. Unlike its predecessors, MPEG-4 coding did not remain confined to the domain of rectangular-sized pictures but adopted an object based coding concept in which arbitrarily shaped and dynamically changing individual audio-visual objects in a video sequence can be individually encoded, manipulated and transmitted through independent bit-stream. It was standardized to address a wide range of bit-rates- from very low bit rate coding (5-64 Kbits/sec) to 2 Mbits/sec for TV/film applications. In recent times, MPEG-4 has found widespread applications in internet streaming, wireless video, digital video cameras as well as in mobile phones and mobile palm computers.

In this lesson, we are going to present the key features of MPEG-4 standard. We shall introduce the concepts of content interactivity. The representation and coding of video objects will be explained. Two additional features of MPEG-4, namely sprite coding and the abilities to combine synthetic and natural video will also be covered.

25.1 Basic objectives of MPEG-4 standard

The MPEG-4 standard was conceptualized with an objective to standardize algorithms for audio-visual coding in multimedia applications, with flexibility for

interactions, universal accessibility and high compression. Following features can be listed as its objectives:

(a) Support for content-based manipulation and bit stream editing,

(b) Ability to combine synthetic scenes or objects with natural scenes and objects,

(c) Provisions for efficient random access of video frames or objects.

(d) Better visual quality at comparable bit rates, as compared to its earlier standards.

(e) Ability to encode multiple views, for example stereoscopic video

(f) Provisions for error robustness to allow access to a variety of wireless and wired networks, storage media

(g) Scalability with fine granularity in content, quality and complexity.

25.2 Content-based interactivity

In MPEG-4, audio and video data are *content based*, which allow independent access and manipulation of audio-visual objects in the compressed domain. Transformation of existing objects (re-positioning, scaling, and rotations), addition of new objects, removal of existing objects etc. are all within the scope of manipulation. The object manipulations are possible through simple operations performed on the bit stream. The audio-visual objects are layered, as will be explained later in section 25.4 and each layer is encoded into an elementary stream (ES) of bits.

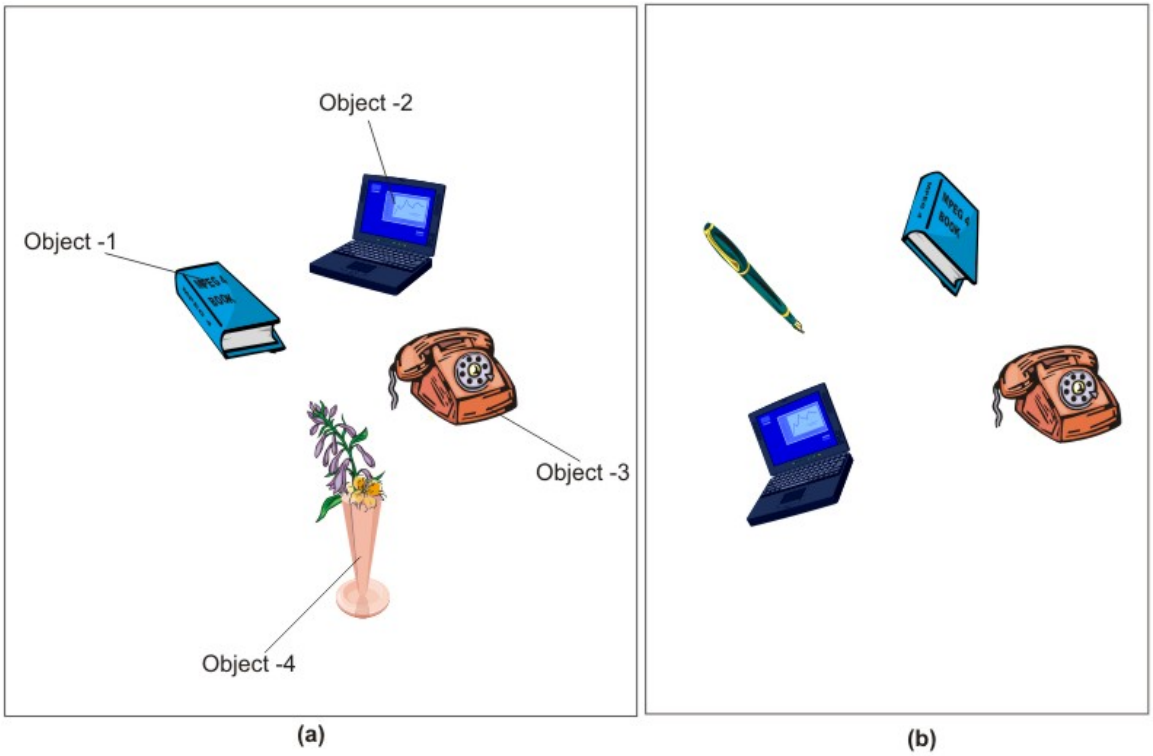

Figure 25.1 Content - manipulation of video frame.
The contents shown in (a) original and (b) reconstructed scene

Fig 25.1 illustrates the concepts of content based representation of data. Each of the contents (objects) shown in fig 25.1 (a) are encoded and decoded independently. While reconstructing the scene in fig 25.1 (b), some objects have been re-positioned, rotated or deleted and new objects added. As mentioned, the bit-stream is *object layered* as the shape and transparency of each object as well as the spatial coordinates and additional parameters that describe object scaling, rotation etc are described in the bit-stream of each object layer. The receiver can either reconstruct the object in its entirety or do some manipulation at the bit stream level to present the object in a different way. These capabilities are given to both natural and synthetic video objects and the reconstructed scene can have a combination of both.

25.3 Toolbox approach of MPEG-4

The MPEG-4 has followed a different approach towards the standardization of algorithms, as compared to MPEG-1 and MPEG-2. In the two earlier standards, complete algorithms for audio, video and system aspects were standardized. MPEG-4 in contrast, follows a *toolbox approach* in which tools are standardized. Video tools include a complete algorithm, or individual modules such as shape coding, motion compensation, texture coding etc. These independent coding tools can be bound together using the MPEG-4 Systems Description Language (MSDL). The MSDL is also transmitted with the bit stream and it specifies the structure and the rules for the decoder.

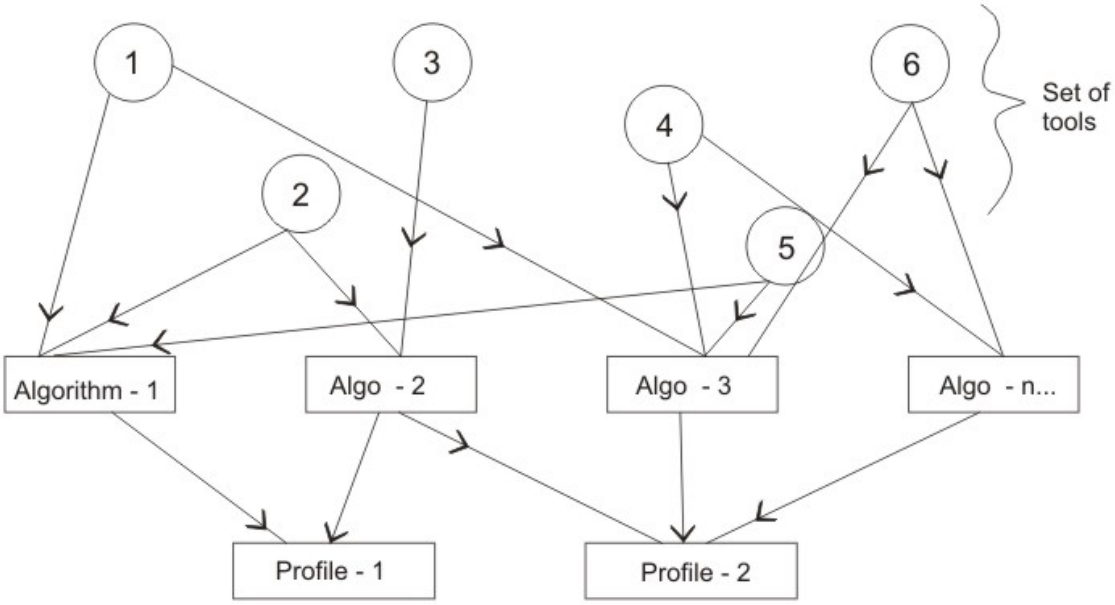

Figure 25.2 MPEG-4 "Toolbox" approch

The toolbox approach, illustrated in fig.25.2 offers flexibility to address variety of requirements. For example, the tools and algorithms for high compression applications and error-prone environments can not be the same. The tool box approach also supports future expandability and missing software tools can be downloaded at the receiver. The software implementation of the standard also facilitates implementation on general purpose DSP processors.

25.4 Video object representation and encoding layers

To achieve content-based interactivity, MPEG-4 has standardized on the video object representation. A sequence is composed of one or more audio visual objects (AVO). AVOs can be either an audio object resulting out of speech, music, sound effects, etc or a *video object* (VO) representing a specific content, such as a talking sequence of head-and-shoulder images of a person or a moving object, static/moving background etc. A video object may be present over a large collection of frames. A snapshot of a video object in one frame is defined as the *video object plane* (VOP) and is the most elementary form of content representation.

For content representation using VOPs, an input video sequence is segmented into a number of arbitrarily shaped regions (VOPs). Each of the regions may possibly cover particular image or video content of interest. The shape and the location of the region can vary from frame to frame.

The shape, motion and texture information of the VOPs belonging to the same VO is encoded and transmitted into a Video Object Layer (VOL). Since typically there are several video objects, the bit stream should also include information on how to combine the different VOLS to reconstruct the video.

Fig. 25.3 Snapshot of a Video Sequence

Fig 25.4 Binary alpha-plane

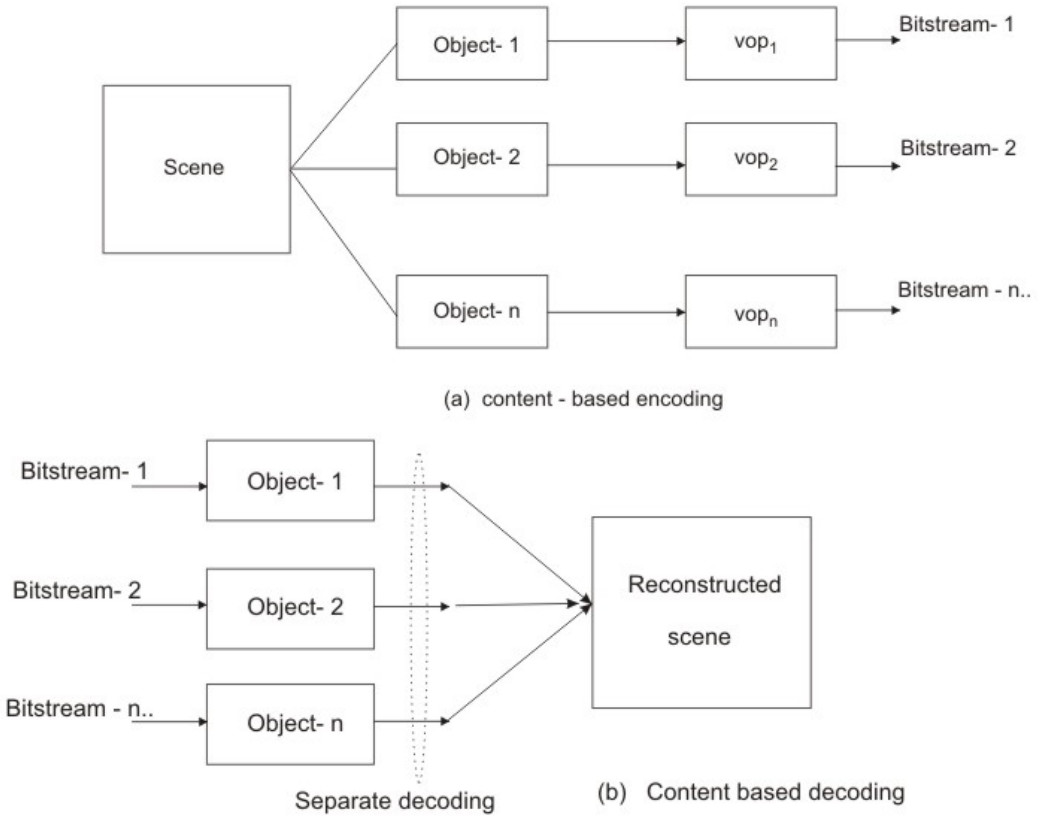

Figure 25.5 Content based encoding and decoding in MPEG - 4

Fig 25.3 shows a snapshot (frame) of a Video sequence, segmented into an arbitrarily shaped foreground VOP, and a background VOP2. Fig 25.4 shows a binary alpha-plane for the same frame, which is a binary segmentation mask specifying the location of the foreground content VOP. Fig 25.5 illustrates the scheme of content-based encoding and decoding. The scene is first segmented into a number of VOPs, each of which specifies particular image sequence content and is coded into a separate VOL. It is possible to reconstruct the original video if all the VOLs are considered. However, contents can be decoded by considering only a subset of all VOLs and this allows content based interactivity. Each VOL encoding has three components –

- Shape (contour) coding
- Motion estimation and compensation
- Texture coding

We may note that the frame-based functionalities of MPEG-1 and MPEG-2 form a subset of content based functionalities supported in MPEG-4. While MPEG-4 supports multiple VOPs, the former two standards support only one VOP containing the entire picture of fixed rectangular size.

25.4.1 VOP Window and shape-adaptive macroblock grid:

To encode shape, motion and texture information in arbitrarily shaped VOPs, the concept of VOP image window and a shape adaptive macro block grid has been introduced.

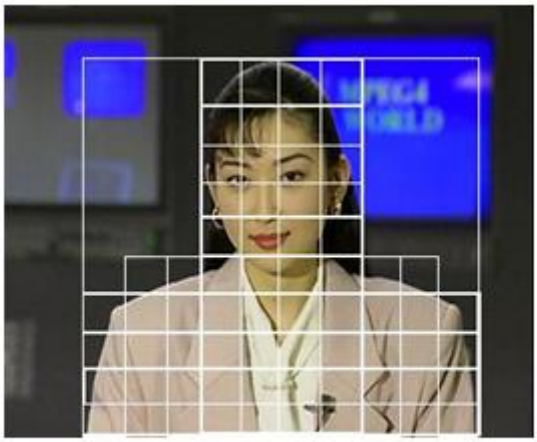

Fig 25.6 VOP Window and VOP macroblocks

The VOP image window, as illustrated in fig.25.6 is a rectangular window having size in multiples of 16 pixels in each image direction that surrounds the foreground VOP. This window is adjusted to collocate with the top-most and left-most border of the VOP. The position of the VOP image window is defined with respect to a reference window of constant size by specifying a shift parameter.

The VOP image window is composed of macro block of size 16 x 16 pixels, which are of three types:

- Macroblocks which do not belong to the VOP at all. These are inactive macroblock with respect to the VOP and are not encoded in the VOL.

- Macroblocks which partly belong to the VOP. These are the boundary macroblocks for the VOP and require some special consideration during its encoding.

- Macroblocks which fully belong to the VOP. The are the standard macroblock for the VOP.

The last two categories of macroblocks are the active macroblocks and the grids that define those are referred to as the shape-adaptive macroblock grid marked in fig 25.6. This plays a major role in VOP encoding, to be explained in the following section.

25.5 Encoding of VOPs

As already mentioned, the VOLs compose the bit-streams for the VOPs and their encoding have three major components, namely the shape, motion and texture. We are now going to discuss each of these in the context of content based functionalities of MPEG-4.

25.5.1 Shape coding in MPEG-4

Since video objects in MPEG-4 are of arbitrary shape, encoding of shapes form an essential part of encoding. Whether a pixel belongs to the VOP or not is specified by a binary map known as alpha plane which has an entry of “1” if the pixel belongs to VOP and is “0” otherwise.

Shape coding techniques may be broadly classified as (a) contour based and (b) bit-map based. The contour based techniques extract and encodes a description of the closed contour enclosing the shape. The bit-map based techniques are applied directly to the binary alpha-plane, within the conventional block-based framework.

The contour based technique adopted in MPEG-4 is the vertex-based coding that approximates the shape using a polygonal approximation. First, the longest axis of the shape is found and its two end points are used as the initial polygon. For each polygon line, it is checked if the approximation lies within the tolerance. If not, a new vertex is inserted at the point of largest prediction error. Each new polygon side is checked again for approximation and the process is iteratively repeated.

Bit map based shape coding techniques may be broadly categorized as:

- Modified Read (MR) approach, used in fax
- Context based arithmetic encoding (CAE), which has been adopted in JBIG standard.

The MPEG-4 standard has adopted CAE for encoding the bit-map of binary alpha planes. The pixels of alpha-planes are grouped into binary alpha block (BAB) of size 16 x 16 pixels. BABs may be intra-coded using CAE or inter-coded using motion compensation and CAE.

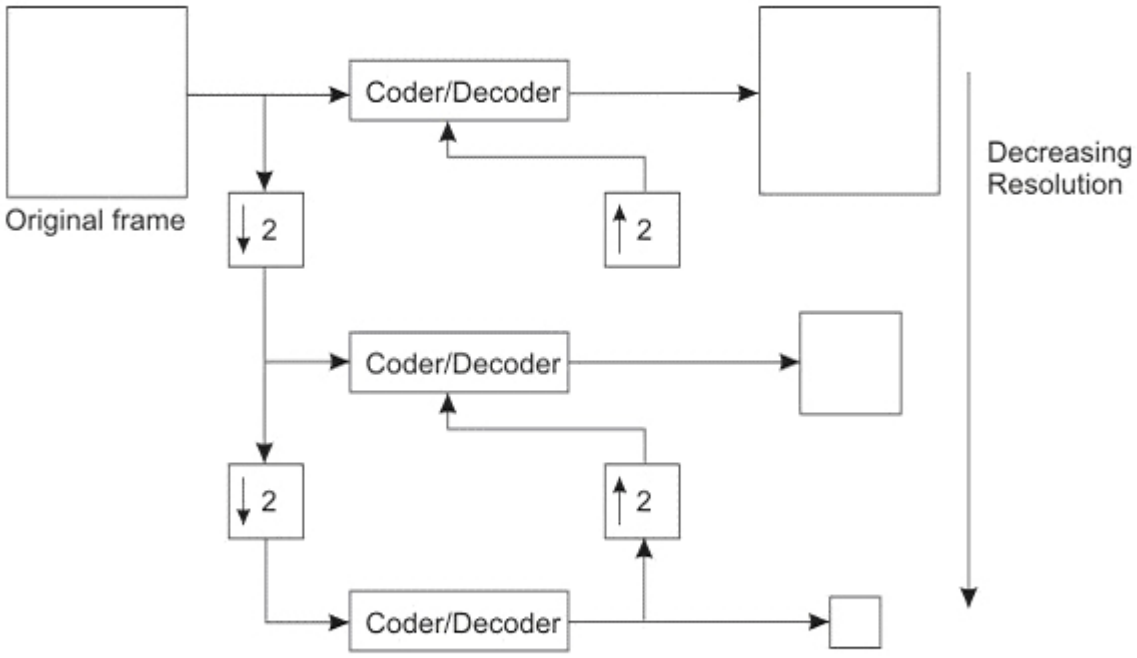

FIGURE 25.7 Spatial scalability of VOPs in MPEG - 4

25.5.2 Motion estimation in VOPs:

The shape-adaptive macroblock grid defined in section 25.4.1 is used for motion estimation in VOPs. The standard macroblock within the grid are motion compensated, following the approaches adopted in the earlier two MPEG standards. However, a different approach is adopted for the contour macroblocks. An image padding method is employed in the reference for these macroblocks, which can be seen as an extrapolation of pixel values outside the VOP based on the values inside the VOPs. After padding the reference VOP, a *polygon matching technique* is employed for motion estimation and compensation. "Polygon" refers to the part of the contour macroblock which belongs to the active area inside the VOP frame to be coded and excludes the pixels outside this area.

Based on the motion estimation and motion compensation philosophy, three types of VOPs can be defined :

- ***I-VOP*** **:** These are the intra-coded VOPs, similar to the intra coded frames (I-picture) where no motion estimation is employed and only texture coding is done.

- ***P-VOP :*** These VOPs use forward prediction for motion compensation, very similar to the P-picture.

- ***B-VOP:*** These VOPs are bi-directionally predicted, very similar to the B-picture.

It may be noted that the encoding of standard I-picture, P-pictures and B-pictures are still possible in MPEG-4 for the special case of single, rectangular VOPs.

25.5.3 Texture Coding :

Texture coding is to be performed on the I-VOP or the residual errors after the motion compensation in the P-VOPs and B-VOPs. For texture coding too, the shape adaptive macroblock grid is used. For each macroblock, a maximum of four 8 x 8 luminance blocks and two 8 x 8 chrominance block are employed.

Special adaptation is required for the contour blocks, where image padding technique is used to fill the macro block content outside a VOP before applying the DCT in intra-VOP. For motion compensated coding of P-VOPs, the contents of the pixels outside the active VOP area are set to128.

25.5.4 Multiplexing of shape, motion and texture information :

The video object layer (VOL) is formed by multiplexing the encoded VOP information in the following order:

- Shape encoding
- Motion vector encoding
- Texture coding

Two different encoding mechanisms are supported by MPEG-4. One is a joint motion vector, along with DCT coefficient encoding procedure to achieve high compression efficiency at very low bit rates. The second mechanism is to separately encode the motion vectors and the DCT coefficients.

25.6 Spatial and temporal scalability of MPEG-4

We have seen the scalability supports in the MPEG-2 standard, which can be used to make the lower resolution decoders (receivers) work from a scalable or layered bit stream. The same concept is extended to the encoding of arbitrarily shaped VOPs in MPEG-4. In this case, each VOP can be encoded to multiple number of VOLs of which only one forms the base layer and the remaining ones compose the enhancement layers. The layered bitstream has a major advantage in terms of prioritized transmission and error resiliency.

Two types of scalabilities are supported in VOP encoding process- one is the spatial scalability and the other is the temporal scalability.

25.6.1 Spatial scalability:

This is very similar to the spatial scalability support in MPEG-2. Here, multi resolution representations of the VOPs are formed by spatially downsampling the input video signal into a number of levels. The lowest resolution level supports the base-layer bit stream and the subsequent upper resolutions are predicted by upsampling from the lower resolution of the VOP. The spatial scalability concept is illustrated in fig 25.7.

25.6.2 Temporal Scalability

Very similar to the temporal scalability concepts for pictures in MPEG-2, the MPEG-4 standard supports temporal scalability for the VOPs. Like the spatial scalability temporal scalability too generates a layered bit stream in which the base-layer is formed by temporally subsampling the video objects and the enhancement layers are obtained by temporal prediction from the lower layers.

Using the MPEG-4 VOP temporal scalability approach, it is possible to have different frame rates for different video objects. For example, the foreground object may have a higher frame rate as compared to the relatively stationary background.

25.6.3 Sprite coding in MPEG-4

The sprite coding is an important feature of MPEG-4. The object based coding in MPEG-4 essentially requires video segmentation algorithm to extract the foreground from the background. This idea is extended to spite coding, in which the background is reconstructed and transmitted separately from the foreground, using a very sophisticated motion analysis and prediction strategies.

Sprite, also referred to *panorama* assumes a flat, static background. As the camera pans rotates or zooms over the scene, the spite coder learn more information about the background. By estimating the camera parameters from the successive frames, the background content of each frame can be added to or deleted from the panorama.

In the spite coding approach of MPEG-4, the large, static panorama picture is first transmitted to the receiver. For each frame, camera parameters are transmitted separately, which facilitates extraction of frame backgrounds from the panorama. The foreground is encoded separately and the receiver composes the scene from the separately transmitted foreground and the background.

Since sprite-coding requires only one time transmission of the background, substantial coding gain is usually achieved as compared to the usual block based encoding of the entire scene.

25.6.4 Facial feature animation capabilities of MPEG-4

The sprite coding concepts can be extended to the model based video coding for head-and shoulder video sequences. Such model based coding techniques use a 3-D wire mesh model of a human head and shoulders. A sprite image of a person is mapped on to the 3-D surface to represent the texture details of the person.

Both model and human –face sprites are required to be sent by the transmitter to the receiver in the beginning and subsequently, for each frame, only a few parameters, that represent the motion of the person are to be transmitted. Transmission of 2 to 6 motion parameters per frame is sufficient for excellent predictions of the face region.

The MPEG-4 uses a similar concept to animate synthetically generated faces. There are some specific control points in the wire mesh model to which motion parameters can be imparted to create the impression of a “talking head”.

25.7 Conclusions

We have seen that the MPEG-4 standard has significantly deviated from the conventional approach of frame based coding and introduced the object based coding concepts that permits flexibilities in reconstruction. The objects may be of arbitrary shape and encoding of video objects requires compression of shape, motion and texture contents of the object. The standard supports a wide range of bit-rate applications- from very low bit-rate, i.e. less than 64 Kbps to the HDTV applications at higher bit-rates. The sprite coding approach achieves significant compression efficiency.

In the next lesson we shall focus on the video coding standards proposed by the International Telecommunication Union (ITU).