MPEG-4 Part 2

Video Coding

White Paper

July 12, 2004

For Further information,
Please contact:

Ramin Soheili, Ph.D.

SEDA Solutions Corp.
One Daniel Burnham Court
Suite 240C
San Francisco, CA 94109
USA
Phone: (415) – 409 2544
Fax: (415) – 409 2546

ramin@sedasolutions.com

1 Table Of Contents

2 Table of Figures

3 Introduction

MPEG-4 is an ISO/IEC standard with designation ISO/IEC 14496. It was finalized and made into a formal international standard in 2000. Since then several additions were made to the standard notably the addition of the Simple Profile Level 0 with suitable complexity to be used in mobile environments such as the UMTS 3GPP networks. The MPEG-4 standard improves on the MPEG-2 standard in both compression efficiency and coding flexibility, and hence covers a wider range of applications. The MPEG-4 part 2 (visual coding) standard was developed to address applications in emerging fields in such areas, but not limited to, as digital television, interactive graphics with synthetic contents, interactive multimedia over the internet, and interactive video games.

3.1 Features in MPEG-4

Several features are added to the MPEG-4 visual standard, which distinguish it from prior video coders. Some are listed below:

- Coding of natural as well as synthetic video.
- Content based access to images or video.
- Arbitrary shape coding for video objects.
- 2D and 3D mesh objects.
- Animated human faces and bodies.
- Still texture and image coding.
- Spatial and temporal scalability.
- Pixel widths from 4 to 12 bits.
- Error resilience and robustness.
- Compatibility with ITU-T H.263 ‘short header’ video coding standard.

4 Applications

The MPEG-4 provides a bewildering number of tools. In order to allow effective implementation of the standard, subsets of these tools are grouped under ‘visual profiles’ that can be used for specific applications. The profiles together with ‘levels’ limit the set of tools that a decoder has to implement. The levels limit the computational complexity of both the encoder and decoder. Some MPEG-4 suited applications, but by no means limited to, are:

- Digital TV broadcasting, videoconferencing, and video storage.
- Content based coding, where individual objects in a picture are coded and sent separately. This can be used for high quality video editing and studio production environments.
- Computer graphics applications with synthetic 2D or 3D mesh geometry and animated human faces.
- Multimedia video over the Internet and mobile networks.

5 Profiles and Levels

The MPEG-4 standard uses a toolkit approach for coding visual information. A tool is a coding function such as interlaced video, data partitioning, or B-frames. There are no less than 24 tools defined in the standard. Subsets of these tools are called Profiles. Tools are grouped together in such a way to facilitate specific applications by controlling the complexity of the codec. There are at least 19 profiles defined in MPEG-4. The simplest being the ‘Simple’ profile for low-complexity coding of rectangular video frames to the most complex, the ‘Core Studio’ for object-based coding of high quality video. Each profile, in turn, can have 4 or more levels. Levels are used to place constraints on parameters of the bit stream such as picture size, bit rate, number of objects, and frame rates. These parameters ensure a maximum performance requirement for a compliant decoder to chosen profile and level. Up to 38.4Mbps video coding is defined for the Main profile Level 4 with a picture size of 1920x1088 pixels.

In order to address video broadcasting to mobile networks, a new level named Level 0 was added to the existing Simple profile. This profile limits the bit rate to 64kbps and the picture size to QCIF (176x144 pixels).

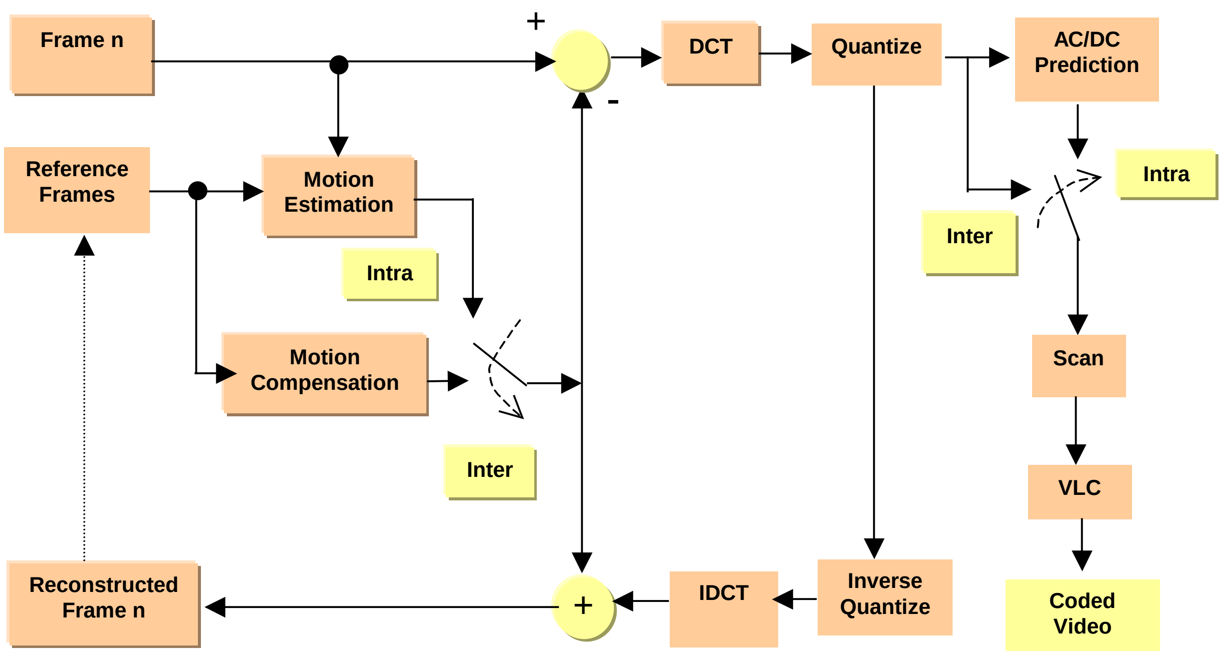

Figure 1 - Block diagram of MPEG 4 Part 2 Video Encoder

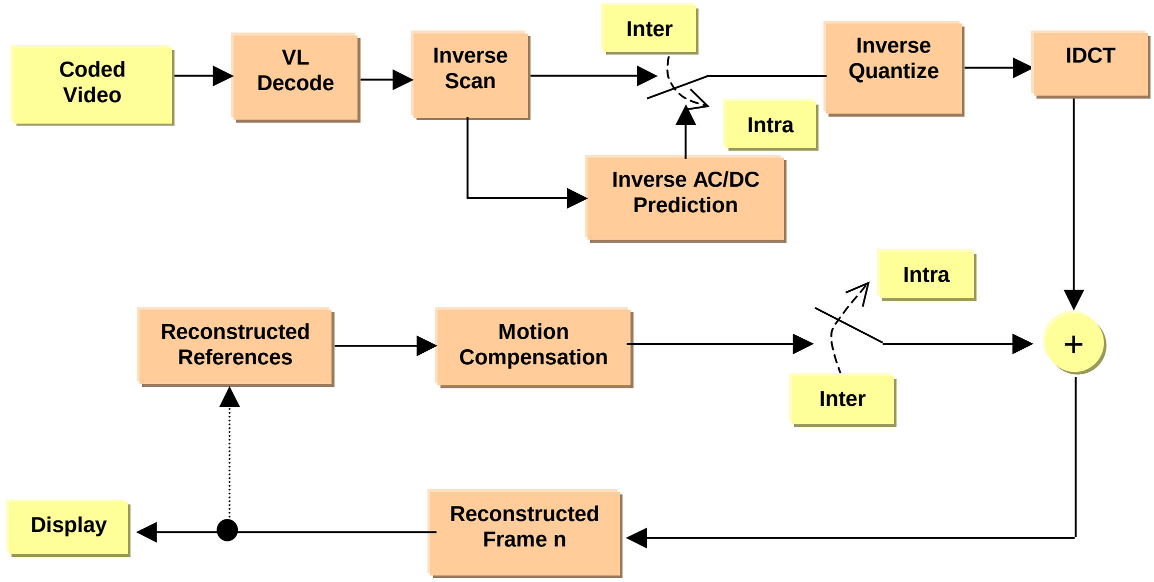

Figure 2 - Block diagram of MPEG 4 Part 2 Video Decoder

5.1 MPEG-4 Coding Features and Highlights

Traditionally video sequences are treated as a collection of rectangular frames. MPEG-4 digresses from this notion and defines 'video objects' (or VO) of arbitrarily shaped regions that can exist for an arbitrary length in time, which the user can access and manipulate. The traditional rectangular video shape is a subset of this definition. This powerful new notion provides the flexibility to divide a frame into small pieces and manipulate and code sections of a frame independent from each other. A video object plane (VOP) is an instance of the VO at a particular point in time. The simple profile in MPEG-4 is limited to rectangular VOPs and is the most popular application of the standard.

The MPEG-4 is based on the hybrid model for video coding where motion estimation/compensation is followed by a 2 dimensional transform. This model uses temporal (across time) and spatial (across space) prediction to achieve high levels of source compression. MPEG-4 uses many of the basic tools used in prior art for video compression with many additional coding and transport efficiency tools. The following are the main coding tools featured in the MPEG-4 standard:

- **Three frame types:** I, P, and B.

  I or Intra VOPs are coded without prediction from other VOPs. These are used at the start of a sequence and for periodic refreshing of any accumulated errors. They can also be used as entry points into the sequence during fast forward or reverse functions. P or Inter VOPs are coded using motion prediction from a previously coded I or P reference VOP. P frames achieve a high level of compression compared to I coded frames. B or Bi-directional frames use forward or backward prediction to achieve even higher compression than P frames. However, B frames do require more buffering and increase coding delay.

- **2 dimensional discrete cosine transform:** spatial domain to frequency domain conversion.

  MPEG-4 uses the traditional 8x8 pixel block DCT and IDCT transforms. Four luminance and two chrominance blocks are processed. The DCT/IDCT are floating point operations and do result in mismatches between decoders. A limit of +/-1 error in pixel value is indicated in the standard.

- **Quantization:** 31 parameter values are defined.

  Quantization is the main process in which large compression is achieved by dividing and truncating the transformed coefficients. Many coefficients are set to zero by this process. The larger the quantization parameter, the larger the number of zero coefficients, and the smaller the number of coded bits generated. Bit-rate vs. quality tradeoff is achieved by varying the quantization parameter. The simple profile uses a scalar quantization where all coefficients are scaled using the same quantization parameter. However, in the advanced profile, a user defined matrix quantizer is allowed where each coefficient can have a unique scalar. In the decoder, the inverse process of the mentioned is performed.

- **DC/AC prediction:** spatial compression in I frames.

  Most often the low frequency components of neighboring blocks, such as the DC component and the first row and column of the AC components, are correlated. Hence it saves bits to use the neighboring blocks as predictors of the current block. A residual is coded and transmitted instead of the full coefficients. The AC prediction is optional.

- **Motion estimation:** up to 4 motions vectors per macroblock.

  The default block size for motion estimation is 16x16 pixels. This translates to 1 motion vector per macroblock. However, motion estimation is more effective in smaller sizes especially for complex motions. The MPEG-4 standard, optionally, allows the encoder to subdivide the macroblock into 4-8x8 blocks and transmit 4 motion vectors per macroblock. Although this increases the overhead in transmitting 4 motion vectors, the energy in the transmitted motion compensated residual is minimized. The default resolution for the motion vector is ½ a pixel. However the advanced profile allows quarter pixel resolution motion vectors. In this case, the ½ pixel interpolated values are further interpolated into ¼ pixel values and used for motion estimation at a cost of increased complexity.

- **Unrestricted motion vectors:** improved motion compensation at VOP boundaries.

  The MPEG-4 standard allows the motion vectors to point outside the reference VOP if a better match is obtained during the motion estimation process by doing so. This is achieved by 'padding' or extrapolating the reference VOP beyond its boundary. This improves the efficiency of the motion compensation when objects are moving in and out of the frame.

- **Global motion compensation (GMC):** up to 4 global motion vectors per VOP

  Often a large number of macroblocks experience related motion vectors, such as camera panning, zooming, or rotation. In these cases, the encoder can opt to send up to 4 global motion vectors with the location of each to describe a global motion for the VOP. This eliminates the need to send individual motion vectors when the majority of the macroblocks undergoes similar related movement. The motion compensation calculates the individual motion vectors by interpolating the global motion vectors in relation to a pixel location to derive the motion at that location. Good compression efficiency is achieved in VOPs with linear or rotational motion.

- **Video packets:** prevents error propagation

  The MPEG-4 standard allows the option to packetize the coded bit stream. VOPs can consist of one or more packets. Packets contain structures such as headers, header extensions, and synchronization markers, all of which provide tools for error recovery, synchronization, and prevention of error propagation across packets.

- **Data partitioning:** resilience against transmission errors.

  Data partitioning goes one step further from packetizing by reordering the coded data within a packet to minimize error impact during transmission. Important information such as the DC coefficients in Intra coded VOPs, and motion vectors in Inter coded VOPs are separated by a marker and placed in two separate portions of a packet. A decoder can reasonably process a VOP using the first partition in case the second one gets lost during a transmission.

- **Run-length coding:** entropy coding is used for each symbol

  Re-ordering the quantized coefficients places most of the zero levels at the end of the array. The non-zero coefficients are coded using last-run-level coding. 'Last' indicates whether the coefficient is the last one in the array. 'Run' indicates the number of zeros preceding the coefficient. 'Level' indicates the value of the coefficient. The triplet is encoded by a variable length code (VLC) similar to Huffman coding where shorter codes are used for the more probable combinations.

- **Reversible VLC:** resilience against transmission errors

  This tool allows the decoder to process the variable length codes in the reverse direction. If a non-decodable error is encountered in the forward direction, the decoder stops and attempts to decode the data in reverse order up to the error location. Hence the data on either side of the error can be recovered.

The 'Core' and 'Main' profiles add several new tools that are worth mentioning but details are beyond the scope of this short treatment of the MPEG-4 standard.

- **Shape coding:** the VO is of arbitrary shape

  Three types of macroblocks are defined in the shape coding tool: 'transparent' if the macroblock is outside the VO, 'opaque' if the macroblock is inside the VO, and 'semi-transparent' if the macroblock is on the boundary of the VO. The boundary macroblock is coded as a binary alpha block (BAB). In turn, a set of tools is defined to deal with the BAB coding.

- **Scalability:** layered coding

  Scalability allows the encoder to send the coded video in layers; a 'base' layer, and 'enhancement' layers. The decoder always decodes the base layer and optionally may decode any number of enhancement layers based on complexity. Each enhancement layer adds an incremental complexity over the base layer. The scalability can be spatial where enhancements translate into increases in picture resolution, or it may be temporal with increases in the frame rate of the video. MPEG-4 also allows quality scalability, complexity scalability, and fine grain scalability.

- **Sprite coding:** S-VOP

  Sprite coding allows the encoder to code and transmit background images, that do not substantially change over time, only once.

Subsequently, the encoder transmits parameters, S-VOPs, to adjust or slightly modify the sprite instead of sending the complete background picture every frame.

6 Summary

The MPEG-4 part 2 standard provides efficient coding tools with great flexibility. The possible number of combinations and grouping of coding tools allow this standard to be used for a wide range of applications. By choosing the appropriate coding tools, low bit rate and low complexity applications, such as digital video over band limited mobile networks, all the way to high fidelity and high complexity digital studio distribution applications, can be easily supported.