

# A New Review on H.264 Video Coding Standard



## Engineering

**KEYWORDS :** Video Coding, H.261, H.263, H.264/AVC, MPEG-2, MPEG-4, Bitstream

\* **L.C.Manikandan**

Head & Asst. Professor of CSE, Mar Ephraem College of Engineering and Technology, Marthandam, Kanyakumari, Tamil Nadu, India. \* Corresponding Author

**Dr.R.K.Selvakumar**

Head & Professor of IT, Agni College of Technology, Chennai, Tamil Nadu, India.

### ABSTRACT

We give a review on H.264/AVC video coding standard. We use diagram to show the encoding and decoding process. The aim of this review is to provide the succeeding researchers with some constructive information about H.264 video coding, those who are interested in implementing video codec in H.264.

### INTRODUCTION

Video Coding is a necessary function of recording Video and TV signals onto a computer Hard Drive. Because raw video footage requires lots of space, without video coding, video files would quickly eat up gigabytes of hard drive space, which would result in only short amounts of video or TV recorded onto the computer's hard drive. With video coding, smaller video files can be stored on your computer's hard drive, resulting in much more space for video files.

Video coding for telecommunication applications has evolved through the development of the ITU-T H.261, H.262 (MPEG-2), and H.263 video coding standards [1] (and later enhancements of H.263 known as H.263+ and H.363++), and has extended from ISDN and T1/E1 service to embrace PSTN, mobile wireless networks, and LAN/Internet network delivery. Throughout this evolution, continued efforts have been made to maximize coding efficiency while dealing with the modification of network types and their characteristic formatting and loss/error robustness requirements. Recently the MPEG-4 Visual (MPEG-4 part 2) standard [1] has also begun to emerge in use in some application domains of the prior coding standards. It has provided video shape coding capability, and has similarly worked toward broadening the range of environments for digital video use.

In early 1998, the Video Coding Experts Group (VCEG) ITU-T SG16 Q.6 issued a call for proposals on a project called H.26L, with the target to double the coding efficiency (which means halving the bit rate necessary for a given level of fidelity) in comparison to any other existing video coding standards for a broad variety of applications. The first draft design for that new standard was adopted in October of 1999. In December of 2001, VCEG and the Moving Picture Experts Group (MPEG) ISO/IEC JTC 1/SC 29/WG 11 formed a Joint Video Team (JVT), with the charter to finalize the draft new video coding standard for formal approval submission as H.264/AVC [2,3] in March 2003. H.264 also has the flexibility to support a wide variety of applications with very different bitrate requirements. For example, in entertainment video applications include broadcast, satellite, cable and DVD.

The main goals of the H.264/AVC standardization effort have been enhanced compression performance and provision of a "network-friendly" video representation addressing "conversational" (video telephony) and "non-conversational" (storage, broadcast, or streaming) applications. H.264/AVC has achieved a significant improvement in rate-distortion efficiency relative to existing standards [5].

The rest of this paper is organized as following. In Section 2, we give an outline of the H.264/AVC coding system. In Section 3, we describe the H.264/AVC layer design. In section 4, we list the applications of H.264 and finally in Section 5 conclude the paper.

### THE H.264/AVC CODING SYSTEM

H.264 is an industry standard for video coding, the process of converting digital video into a format that takes up less capacity when it is stored or transmitted. Video coding is an essential technology for applications such as digital television, DVD-Video, mobile TV, video conferencing and internet video streaming. Standardizing video coding makes it possible for products from different manufacturers (e.g. encoders, decoders and storage media) to inter-operate. The encoding algorithm is applied to the source video to create the compressed file. To play the compressed file, the inverse decoding algorithm is applied to produce a video that shows the same content as the original source video [4]. A pair of algorithms that works together is called a video codec (encoder/decoder). Figure 1 shows the encoding and decoding processes and highlights the parts that are covered by the H.264 standard [4], [7].

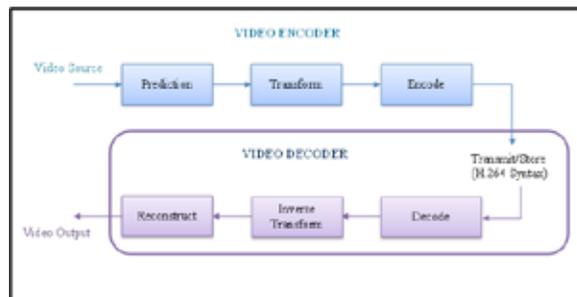


Figure 1: The H.264 video coding and decoding process

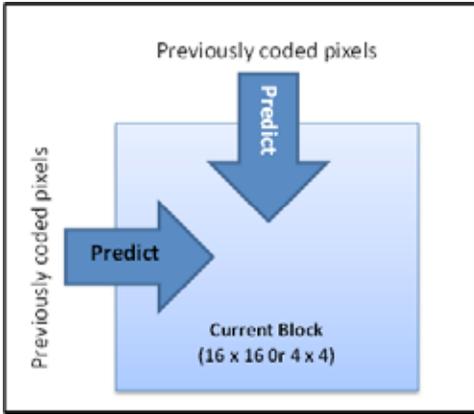
#### Encoder processes

Prediction, transform and encoding processes (see Figure 1.6) are carried out by the H.264 video encoder to produce a compressed bitstream.

#### Bit-stream encoding :

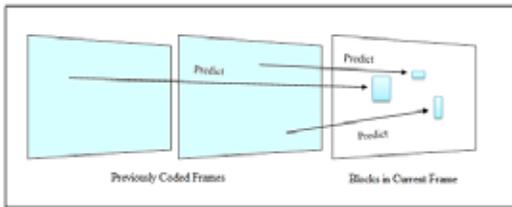
The encoder processes a frame of video in units of a Macroblock (16 x 16 displayed pixels). The predictions of the macroblock is formed based on previously coded data, either from the intra prediction or from inter prediction. To form a residual the encoder subtracts the prediction from the current macroblock [4]. The H.264 supports prediction methods and is more flexible than those in previous standards, empowering accurate predictions and hence competent video compression.

Intra prediction uses 16 x 16 and 4 x 4 block sizes to predict the macroblock from surrounding; previously coded pixels within the same frame (see Figure 2).



**Figure2: Intra Prediction**

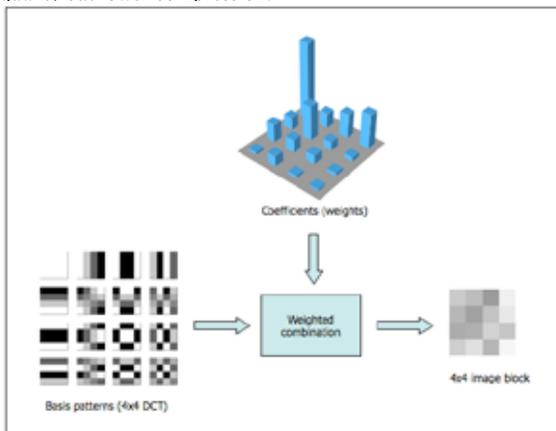
Inter prediction uses a range of block sizes (from 16 x 16 down to 4 x 4) to predict pixels in the current frame from similar regions in previously coded frames (see Figure 3).



**Figure 3: Inter Prediction**

Transform and quantization: A block of residual samples is transformed using a 4 x 4 or 8 x 8 integer transform, an approximate form of the Discrete Cosine Transform (DCT). It produces as a set of coefficients, each of which is a weighting value for a standard basis pattern. When combined, the weighted basis patterns re-create the block of residual samples. Figure 4 shows how the inverse DCT creates an image block by weighting each basis pattern according to a coefficient value and combining the weighted basis patterns [4].

A block of transform output coefficients is quantized, i.e. each coefficient is divided by an integer value. Quantization reduces the precision of the transform coefficients according to a quantization parameter (QP). Setting quantization parameter to a high value means that more coefficients are set to zero, resulting in high compression at the expense of poor decoded image quality. Setting QP to a low value means that more non-zero coefficients remain after quantization, resulting in better decoded image quality but lower compression.



**Figure 4: Inverse transform: combining weighted basis patterns to create a 4 x 4 image block**

The Karhunen-Loève Transforms (KLT) and the Discrete Cosine Transform (DCT) are the two well-known examples of transformation.

Bit-stream encoding: The video coding process produces a number of values includes Quantized transform coefficients, Information to enable the decoder to re-create the prediction, Information about the structure of the compressed data and the compression tools used during encoding, Information about the complete video sequence, that must be encoded to form the compressed bit stream [4]. Using variable length coding and/or arithmetic coding, these values and parameters (syntax elements) are converted into binary codes. Each of these encoding methods produces a competent, compacted binary representation of the information then the encoded bit stream can be stored and/or transmitted.

• **Decoder processes**

To produce a decoded video sequence the H.264 video decoder carries out the complementary processes of decoding, inverse transform and reconstruction.

Bitstream decoding: The compressed bitstream is received by the video decoder then it decodes each of the syntax elements and extracts the information described above (quantized transform coefficients, prediction information, etc.). Then used this information to reverse the coding process and recreate a sequence of video images.

Rescaling and inverse transform: The quantized transform coefficients are re-scaled. Each coefficient is multiplied by an integer value to restore its original scale. An inverse transform is applied to re-create each block of residual data. These blocks are combined together to form a residual macroblock.

Reconstruction: For each macroblock, the decoder forms an identical prediction to the one created by the encoder. The decoder adds the prediction to the decoded residual to reconstruct a decoded macroblock which can then be displayed as part of a video frame.

**H.264 / MPEG-4 / AVC LAYER DESIGN**

H.264 follows a two layer design: a network abstraction layer (NAL) and a video coding layer (VCL). VCL is designed to efficiently represent the video content, and a NAL formats the VCL representation of the video and provides header information to package that data for network transport [6].

• **H.264/AVC NAL**

The NAL is designed in order to provide “network friendliness” to enable simple and effective customization of the use of the VCL for a broad variety of systems. The full degree of customization of the video content to fit the needs of each particular application is outside the scope of the H.264/AVC standard itself, but the design of the NAL anticipates a variety of such mappings. Some key building blocks of the NAL design are NAL units, parameter sets, and access units [9]-[11]. A short description of these concepts is given below.

NAL Units: The coded video information is organized into NAL units, every of that is effectively a packet that contains number of bytes. NAL units are classified into VCL and non-VCL NAL units. The VCL NAL units contain the data that represents the values of the samples in the video pictures, and the non-VCL NAL units contain any associated additional information such as parameter sets and supplemental enhancement information.

Parameter Sets: A parameter set contains vital header data that may apply to an oversized variety of VCL NAL units. There are two types of parameter sets: sequence parameter sets, which ap-

ply to a series of consecutive coded video pictures called a coded video sequence. Picture parameter sets, which apply to the decoding of one or more individual pictures within a coded video sequence.

Access Units: The set of VCL and non-VCL NAL units that's related to one decoded image is spoken as an access unit.

• **H.264/AVC VCL**

As all told previous ITU-T and ISO/IEC JTC one video standards

since H.261 [12] the VCL design follows the block-based hybrid video coding approach (see in Figure 5). Each coded picture is represented in block-shaped units of associated luma and chroma samples called macroblocks. The basic source-coding algorithm is a hybrid of inter-picture prediction to exploit temporal statistical dependencies and transform coding of the prediction residual to exploit spatial statistical dependencies. There is no single coding element in the VCL that provides the majority of the significant improvement in compression efficiency in relation to prior video coding standards.

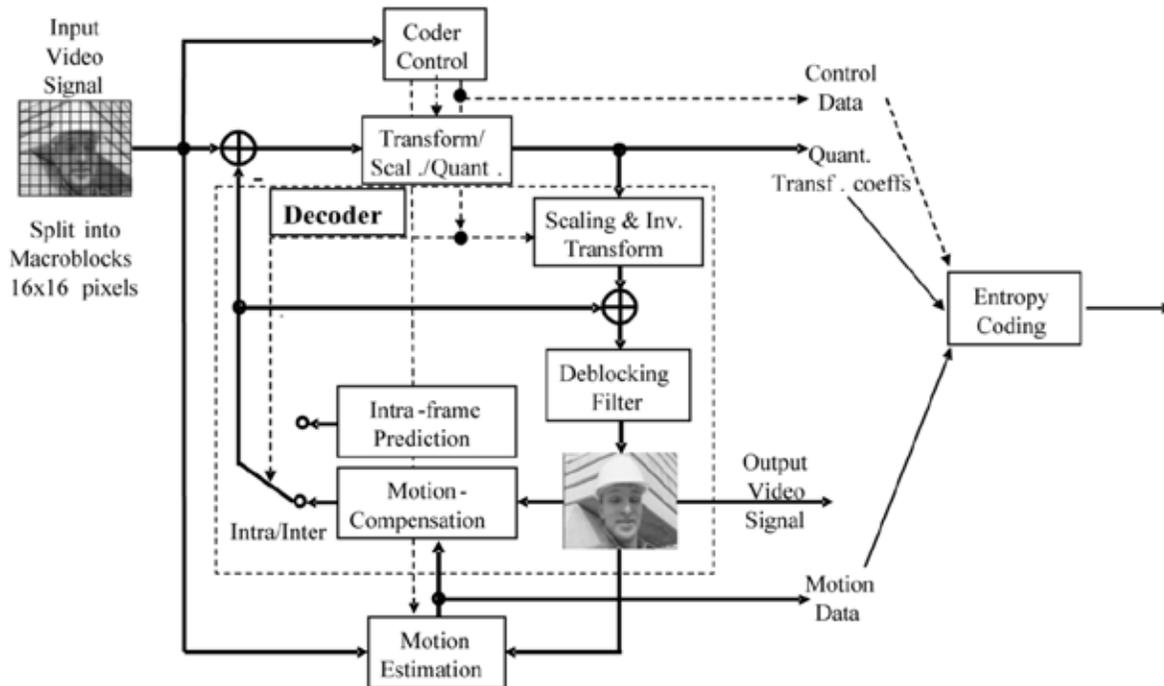


Figure 5: Basic coding structure for H.264/AVC for a macroblock

**APPLICATIONS**

The H.264/AVC video coding standard provides integrated support for transmission or storage. The compressed format and features of H.264 help to minimize the effect of transmission errors. H.264/AVC is being adopted for an increasing range of applications [8], [9] and [13] like, High Definition DVDs (HD-DVD and Blu-Ray formats), High Definition TV broadcasting, Apple products including iTunes video downloads, iPod video and MacOS, NATO and US DoD video applications, Mobile TV broadcasting, Internet video and Video conferencing.

**CONCLUSIONS**

In this paper, an overview of the H.264 video coding standard is presented. Now H.264 is a widely adopted standard, and represents the first time that the ITU, ISO and IEC have come together on a common, international standard for video coding. H.264 involves substantial enhancements in coding competence, latency, complexity and robustness. It provides new possibilities for creating better video encoders and decoders that provide higher quality video streams at maintained bit-rate (compared to previous standards), or conversely, the same quality video at a lower bit-rate.

**REFERENCE**

[1] Thomas Wiegand, Gary J. Sullivan, "Overview of the H.264/AVC Video Coding Standard", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, No. 7, July 2003. | [2] "Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264/ISO/IEC 14 496-10 AVC", in Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVTG050, 2003. | [3] Ostermann, J.; Bormans J.; List P.; Marpe D.; Narroschke M.; Pereira F.; Stockhammer T.; Wedi T.: Video coding with H.264/AVC. | [4] Iain Richardson, "An Overview of H.264 Advanced video coding" 2007 whitepaper. | [5] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard", IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 7, pp. 560-576, Jul. 2003. | [6] Gary J. Sullivan, and Thomas Wiegand, "Video Compression—From Concepts to the H.264/AVC Standard", Proceedings of the IEEE, Vol. 93, No. 1, January 2005. | [7] Iain E. G. Richardson, H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia, John Wiley & Sons, 2003. | [8] T. Wedi, "Motion Compensation in H.264/AVC", in IEEE Transactions on Circuits and Systems for Video Technology. | [9] T. Stockhammer, M. M. Hannuksela, and T. Wiegand, "H.264/AVC in wireless environments", IEEE Trans. Circuits Syst. Video Technol., vol.13, pp. 657-673, July 2003. | [10] S. Wenger, "H.264/AVC over IP", IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 7, pp. 645-656, Jul. 2003. | [11] S. Somnath Sundar and K. Sathiyaa, "Hybrid Video Encoding for Advanced Video Compression (AVC)", International Journal of Scientific & Engineering Research, Vol. 4, Issue 5, pp 1267-1272, May-2013. | [12] "Video codec for audio visual services at px64 kbit/s", Int. Telecommun. Union-Telecommun. (ITU-T), Recommendation H.261, version 1, 1990; version 2, 1993. | [13] S. Vetrivel et al., "An Overview of H.26x Series and its Applications", International Journal of Engineering Science and Technology Vol. 2(9), pp 4622-4631, 2010. |