ISSN: 0975-8585

Research Journal of Pharmaceutical, Biological and Chemical Sciences

High Performance and Efficient Video Compression Using H.264 CABAC Algorithm.

Mary Pamila M*, Manonmani V[1], and Nirmala D[2].

Department of Electronics and Control, Sathyabama University, Chennai, Tamil Nadu, India.

ABSTRACT

H.264 / MPEG-4 Part 10, a recently developed international standard for video compression, offers significantly better video compression efficiency than previous universal standards. Video compression refers to decreasing the quantity of data used to represent video images and is a straight forward combination of image compression and motion compensation. Video compression systems used in many commercial products from consumer electronics devices such as digital camcorders, cellular phones to video teleconferencing systems. One of the tools is the CABAC algorithm used in the baseline profile of H.264 standard to compresses data more efficiently. In this paper, the encoding part is simulated by MATLAB which compress the input video sequence and ensures better performance and better video compression.

Keywords: H.264, MPEG-4, CABAC

Corresponding author

ISSN: 0975-8585

INTRODUCTION

One of the major challenges in enabling mobile multimedia data services will be the need to process and wirelessly transmit very large volumes of video data and video. This has motivated active research on multimedia video data compression techniques such as JPEG, JPEG2000 and MPEG/H.264. These approaches concentrate on achieving higher compression ratio without sacrificing the quality of the image. A problem however is that still image and digital video data rates are very large typically in the range of 150Mbits/sec. Data rates of this magnitude would consume a lot of the bandwidth. For this reason Video compression standards have been developed to eliminate picture redundancy, allowing video information to be transmitted and stored in a compact and efficient manner.

OVERVIEW OF VIDEO COMPRESSION

Video compression refers to reducing the quantity of data used to represent video images and is a straightforward combination of image compression and motion compensation. Compressed video can effectively reduce the bandwidth required to transmit digital video via terrestrial broadcast, via cable, or via satellite services .If the video is overcompressed in a lossy manner, visible (and sometimes distracting) artifacts can appear. Video compression typically operates on square-shaped groups of neighboring pixels, often called a macroblock. These pixel groups or blocks of pixels are compared from one frame to the next and the video compression codec (encode-decode scheme) sends only the differences within those blocks.In areas of video with more motion, more pixels change from one frame to the next. When more pixels change, the video compression scheme must send more data to keep up with the larger number of pixels that are changing. The size of the data in compressed form (C) relative to the original size (O) is known as the compression ratio (R=C/O). Image compression may be lossless where no information is lost during the compression process. The image produced by the decompression (also known as decoding) process is identical bit by bit with the original image. The widely used GIF formats a lossless image and video compression format. Image compression may be lossy where information is lost during the compression process. The widely used JPEG image compression standard is a lossy compression scheme. One of the most powerful techniques for compressing video is interframe compression which uses one or more earlier or later frames in a sequence to compress the current frame, while intraframe compression uses only the current frame, which is effectively image compression. Generally some elements within the data are more common than others and most compression algorithms exploit this property, known as redundancy. The greater the redundancy within the data, the more successful the compression of the data is likely to be. Fortunately, digital video contains a great deal of redundancy and thus is very suitable for compression.

H.264 ENCODER

The top-level block diagram of an H.264 Encoder is shown in Figure 1.

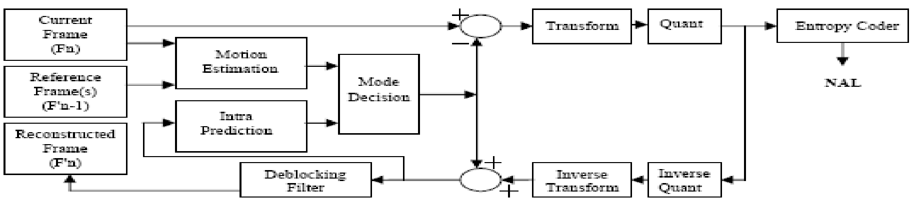

Fig: 1 H.264 encoder block diagram

As shown in Figure, an H.264 encoder has a forward path and a reconstruction path. The forward path is used to encode a video frame by using intra and inter predictions and to create the bit stream. The reconstruction path is used to decode the encoded frame and to reconstruct the decoded frame. Since a decoder never gets original images, but rather works on the decoded frames, reconstruction path in the encoder ensures that both encoder and decoder use identical reference frames for intra and inter prediction. This avoids possible encoder – decoder mismatches [1][3][4]. Forward path starts with partitioning the input frame into MBs. Each MB is encoded in intra or inter mode depending on the mode decision. In both intra and inter modes, the current MB is predicted from the reconstructed frame. Intra mode generates the predicted MB based on spatial redundancy, whereas inter mode, generates the predicted MB based on temporal redundancy. Mode decision compares the required amount of bits to encode a MB and the quality of the decoded MB for both of these modes and chooses the mode with better quality and bit-rate performance. In either case, intra or inter mode, the predicted MB is subtracted from the current MB to generate the residual MB. Residual MB is transformed using 4x4 and 2x2 integer transforms. Transformed residual data is quantized and quantized transform coefficients are re-ordered in a zig-zag scan order. The reordered quantized transform coefficients are entropy encoded. The entropy-encoded coefficients together with header information, such as MB prediction mode and quantization step size, form the compressed bit stream. The compressed bit stream is passed to network abstraction layer (NAL) for storage or transmission [1][3][4]. Reconstruction path begins with inverse quantization and inverse transform operations. The quantized transform coefficients are inverse quantized and inverse transformed to generate the reconstructed residual data. Since quantization is a lossy process, inverse quantized and inverse transformed coefficients are not identical to the original residual data. The reconstructed residual data are added to the predicted pixels in order to create the reconstructed frame. A deblocking filter is applied to reduce the effects of blocking artifacts in the reconstructed frame [1][3][4].Another application area for H.264 intra frame coder is in motion picture production, editing and archiving, where video frames are coded as I-frames only to allow for random access to each individual picture. For such applications, H.264 is shown to be superior to Motion-JPEG2000, especially at lower resolutions.

CABAC ALGORITHM

Context-based adaptive binary arithmetic coding (CABAC) is a form of entropy coding used in H.264/MPEG-4 AVC video encoding. It is notable for providing considerably better compression than most other encoding algorithms used in video encoding and is considered one of the primary advantages of the H.264/AVC encoding scheme. After a block has been predicted and DWT transformed, the resulting prediction modes and DWT coefficients must be stored in a stream. To minimize the number of bits needed to store these values, Advanced Image Coding uses the CABAC algorithms from the H.264 standard. It first converts all non-binary symbols to binary. Then, for each bit, the coder selects which probability model to use, then uses information from nearby elements to optimize the probability estimate. Arithmetic coding is then applied to compress the data.

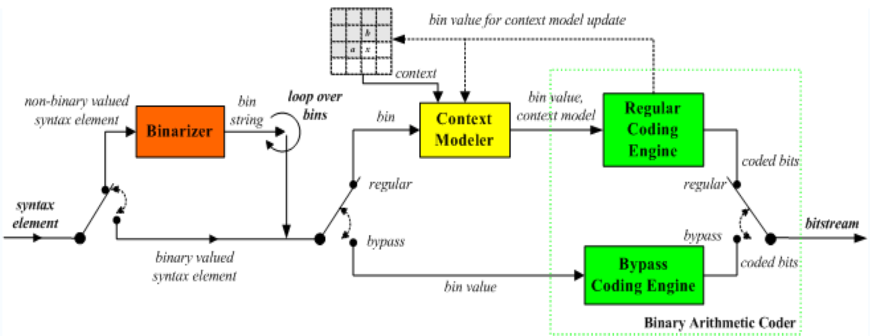

Fig :2 CABAC encoding block diagram

BINARIZATION

Binarizer is a form of pre-processing stage (before coding) that operation syntax elements and generates a unique intermediate binary codeword for a given syntax element. This intermediary codeword is called bin string and each binary value of it called a bin. This stage effectively reduces the alphabet size of the syntax element and allows more efficient operation of context modeling stage. There are four main binarization scheme employed in this stage: unary (U) binarizer, truncated unary (TU) binarizer, k-th order exp-Golomb (EGk) Binarizer and fixed-length (FL) binarizer.

CONTEXT MODELLING

Context modeler is the heart of context-adaptive capability of CABAC that differentiate it from other entropy coding techniques. It assigns a model probability distribution to given symbols which are used for generating the code at the subsequent coding stage. This model determines the code itself and controls the

ISSN: 0975-8585

efficiency of the coding. It is kept up-to-date at all times meaning its statistics is updated after coding of every new bit. It consists of a table of 399entries which each consists of a 6-bit probability value and a 1-bit MPS (Most Probable Symbol). The table is accessed and updated by binary arithmetic coding stage; it is initialized with some predefined values (3 variations of initial table exist that depending on encoding parameters one is selected for an encoding scenario) at the beginning of each slice.

BINARY ARITHMETIC CODER

Binary arithmetic coder is another differentiating feature of CABAC. It is based on recursive interval subdivision. The interval and its location are tracked at any time by two integer values. Based on the statistical property of the symbol being coded, the interval is divided to two regions proportional to probability of LPS and MPS. Update of this interval produces 0 or more output bits to be appended to the output stream. A context-model entry (associated with the symbol) provides the statistics of the symbol being coded. The 6-bit of context entry is the probability estimate of the Least Probable Symbol (LPS) while the 1-bit of the entry shows the polarity of MPS .Since the sub-interval size is reduced after each coding, a renormalization operation rescales the interval range and location to proper range. Actually this renormalization process generates the output bit as part of its rescaling process.

Arithmetic coding is a method for lossless data compression. Normally, a string of characters such as the words "hello there" is represented using a fixed number of bits per character, as in the ASCII code. Like Huffman coding, arithmetic coding is a form of variable-length entropy encoding that converts a string into another representation that represents frequently used characters using fewer bits and infrequently used characters using more bits, with the goal of using fewer bits in total. As opposed to other entropy encoding techniques that separate the input message into its component symbols and replace each symbol with a code word, arithmetic coding encodes the entire message into a single number, a fraction n where ($0.0 \le n < 1.0$).A disadvantage of binary arithmetic coding is that only two symbols, 0 and 1, can be encoded. Other symbols must first be binarised. In Block Prediction, the prediction mode itself is predicted from previous prediction modes. When the prediction is correct, the symbol 1 is encoded, otherwise the symbol 0. If the prediction is incorrect, the prediction mode is one of the other 8 available prediction modes, which can be represented using 3 bits. Each of these 3 bits is encoded as separate symbol.

5. SIMULATION RESULTS

The project has been successfully simulated and the following results were obtained

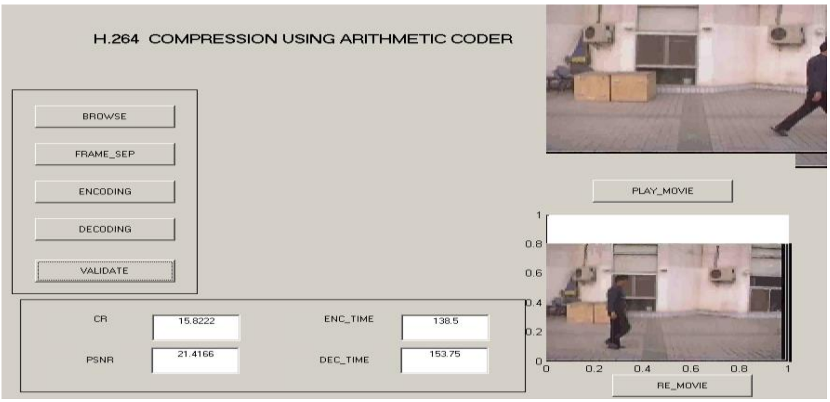

Fig: 3 Compression using arithmetic encoder

The compression ratio (CR) and noise ratio (PSNR) of the chosen file using arithmetic encoder is displayed in the above figure.

CONCLUSION

H.264/AVC was introduced with significant enhancement both in compression efficiency and error resilience. Compared with former video coding standards such as MPEG2 and MPEG4 part 2, it saves more than 40% in bit rate and provides important characteristics such as error resilience, stream switching, fast forward/backward etc. The peak signal to noise ratio has been obtained as 33.1 dB and the compression factor is about 75:1. The data rate is 0.4 bits per pixel. It is believed to be the most competitive video coding standard in this new era.

FUTURE ENHANCEMENT

The FPGA-based H.264 intra frame coder implementation can be modified as an ASIC implementation and prototypes can be fabricated. The power consumption of the H.264 intra frame coder hardware can be analyzed. Based on this analysis, low-power techniques such as clock gating and glitch reduction can be used to reduce its power consumption. A complete H.264 video encoder hardware can be implemented by integrating motion estimation, motion compensation, de-blocking filter, intra vs. inter mode decision and rate control modules to the H.264 intra frame coder hardware.

REFERENCES

[1] Schäfer R, Wiegand T and Schwarz H. "The Emerging H.264/AVC Standard",*EBU Technical Review*, January 2003.
[2] Personal Communication with Hasan Ateş.
[3] Wiegand T, Sullivan G, Bjøntegaard G and Luthra A. "Overview of the H.264/AVC Video Coding Standard", *IEEE Trans. on Circuits and Systems for Video Technology* 2003;13(7): 560–576.
[4] Richardson I. H.264 and MPEG-4 Video Compression*,* Wiley, 2003.
[5] Joint Video Team (JVT) of ITU-T VCEG and ISO/IEC MPEG, Draft ITU-T. Recommendation and Final Draft International Standard of Joint Video Specification, ITU-T Rec. H.264 and ISO/IEC 14496-10 AVC, May 2003.
[6] Tasdizen O, Hamzaoglu I. "A High Performance And Low Cost Hardware Architecture for H.264 Transform And Quantization Algorithms", 13th European Signal Processing Conference, September 2005.
[7] Digital Image processing GONAZLEZ.