Improved Adaptive Arithmetic Coding in MPEG-4 AVC/H.264 Video Compression Standard

Damian Karwowski

Poznan University of Technology,
Chair of Multimedia Telecomunications and Microelectronics
dkarwow@multimedia.edu.pl

Summary. An Improved Adaptive Arithmetic Coding is presented in the paper to be applicable in contemporary hybrid video encoders. The proposed Adaptive Arithmetic Encoder makes the improvement of the state-of-the-art Context-based Adaptive Binary Arithmetic Coding (CABAC) method that is used in the new worldwide video compression standard MPEG-4 AVC/H.264. The improvements were obtained by the use of more accurate mechanism of data statistics estimation and bigger size context pattern relative to the original CABAC. Experimental results revealed bitrate savings of 1.5% - 3% when using the proposed Adaptive Arithmetic Encoder instead of CABAC in the framework of MPEG-4 AVC/H.264 video encoder.

1 Introduction

In order to improve compression performance of video encoder, entropy coding is used at the final stage of coding. Two types of entropy coding methods are used in the new generation video encoders: Context-Adaptive Variable Length Coding and more efficient Context-Adaptive Arithmetic Coding. The state-of-the-art entropy coding method used in video compression is Context-based Adaptive Binary Arithmetic Coding (CABAC) [4, 5] that found the application in the new worldwide Advanced Video Coding (AVC) standard (ISO MPEG-4 AVC and ITU-T Rec. H.264) [1, 2, 3]. Through the application of sophisticated mechanisms of data statistics modeling in CABAC, it is distinguished by the highest compression performance among standardized entropy encoders used in video compression. The goal of the paper is to present possibilities of further increasing the coding efficiency of CABAC-like algorithms by the use of even more complex data statistics modeling. Preliminary results in this field have been already published by author in [6, 7], where improved CABAC algorithm using Context-Tree Weighting (CTW) method [8, 9] was presented. Nevertheless, as it is in original CABAC simplified context pattern was used in older versions of improved CABAC. This paper presents new version of improved CABAC with context pattern of bigger size apart from accurate data statistics estimation mechanism based on CTW method.

2 CABAC entropy coding algorithm

CABAC algorithm has been already described in detail in many publications [4, 5]. In this paper, only main features of CABAC are presented that are important from the coding efficiency point of view. Three functional blocks can be distinguished in CABAC. These are binarizer, context modeler and core of binary arithmetic codec. In order to speed up whole entropy encoding process and reduce negative context dilution effect binary arithmetic codec core was used in CABAC. Therefore, each non-binary valued syntax element must be mapped to string of binary symbols at the first stage of CABAC. This is realized by the binarizer. The way in which binarizer works is very important because it directly affects number of binary symbols in input of arithmetic codec core. In order to reduce this number of symbols, several different binarization schemes were proposed in CABAC that exploit statistical features of syntax elements. Thus, working of binarizer is very similar to adaptive variable-length coding where probability of individual syntax elements is taken into consideration. Binary symbols in binarized strings still exhibit some statistical redundancy. In order to reduce this excess arithmetic coding is used that compresses symbols with respect to the probability of their occurrence. The manner in which probabilities are calculated to a large extend determines efficiency of arithmetic codec core. First of all, individual syntax elements show different statistical features. Secondly, individual binary symbols in the binarized string have different probabilities of occurrence. Therefore, total number of 460 statistical models were defined in CABAC. Statistical modeler of CABAC realizes multi-level adaptation to the current signal statistics to track probabilities of symbols accurately. In the coding process, individual statistical model is chosen based on: type of syntax element, image and image block type, binary symbol index in the binarized word and context of coded symbol (i.e. its values in neighboring blocks). Thus, highly sophisticated mechanism of data statistics estimation that is used in CABAC makes the power of entropy codec and allows to achieve extremely high compression performance. It certainly outperforms all other entropy encoders used in video compression.

3 Possibilities of improving the CABAC

As it was stated, methods of data statistics modeling used in CABAC belong to the most advanced that have ever found application in standardized video encoders. Nevertheless, in order to keep complexity of CABAC in the reasonable boundaries some simplifications were done in the context modeler block. First of all, each of 460 statistical models estimates probabilities of symbols with a dedicated finite-state machine (FSM) with 64 states only. In this way, only a limited set of 128 pre-defined values of probabilities can be used for coded symbols. Secondly, each of 460 FSMs uses the same pre-defined transition rule. It certainly negatively affects coding efficiency of CABAC in some situations. This paper presents improved version of CABAC algorithm that takes advantage of original improvements in data statistics estimation block.

4 Proposal of the improved CABAC

In order to increase compression performance of CABAC, original improvements were introduced to the context modeler block. Two main parts of the context modeler block were modified. These are: the way of calculating the conditional probabilities of symbols and the way of context modeling. The block diagram of the improved CABAC encoder is presented in Figure 1.

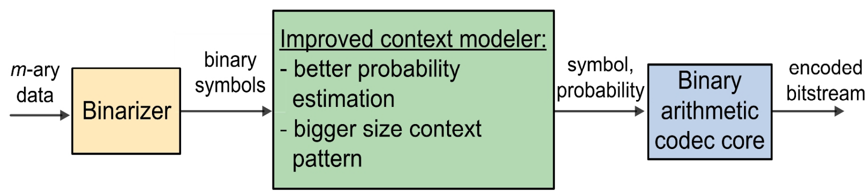

Fig. 1. Block diagram of improved CABAC encoder

Relatively simple method of the conditional probabilities estimation of symbols with FSMs in CABAC was replaced with more sophisticated mechanism of calculating the probabilities that exploits the well-known from literature Context Tree Weighting (CTW) method [8, 9]. This is done by substituting 460 FSMs of original CABAC with 460 binary context trees of depth D that are used in CTW method (see Figure 2).

A binary context tree is used in CTW. The context tree is a collection of nodes joined with branches to which binary symbols (0 or 1) are assigned. Thus, binary context tree is used to keep information on conditional statistics of 0 and 1 symbols that occurred in data stream in a given context. The context is determined by the previously coded symbols, these previously coded symbols determines the so-called context path on the context tree when moving from the tree root λ to individual node of the tree. When processing the symbols, conditional statistics are calculated and updated in each node that belong to the context path. At the given moment, it is unknown which conditional statistics will be the most appropriate to estimate the conditional probability of binary symbol. Therefore, data statistics modeling mechanism calculates probabilities in individual nodes of the context path and the CTW

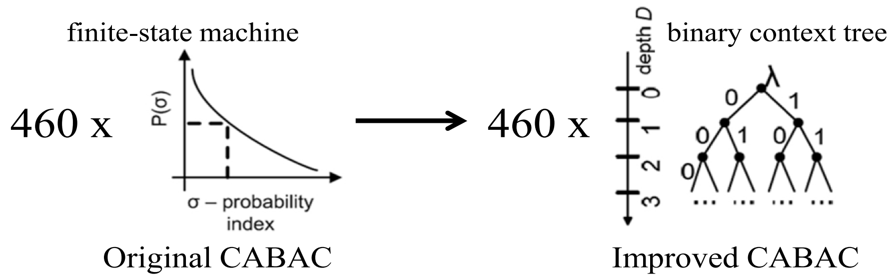

Fig. 2. Calculating probabilities in the improved CABAC

appropriately weights these probabilities to produce the final conditional probability in the root λ of the context tree. On the way of research works that were done by author previously [5, 6, 7] depth D=8 was chosen for each binary context tree in improved CABAC. Also experimental results for older version of improved CABAC were presented for some video sequences in [6, 7]. In the newest version of the improved CABAC the context modeling mechanism was also revised. Modifications were done for context modeling of transform coefficients data due to its significant percentage contribution in encoded bitstream. In original CABAC, a proper statistical model is chosen based on the number of transform coefficients with amplitude greater than or equal to 1 that were previously coded in the current transform block. When coding a transform coefficient, there is proposed more advanced method for context modeling that exploits knowledge on neighboring coefficients from the current transform block as well as neighboring transform blocks. Binarized strings of neighboring coefficients are used to form context information that is used in CTW-based data statistics estimation block (see Figure 3).

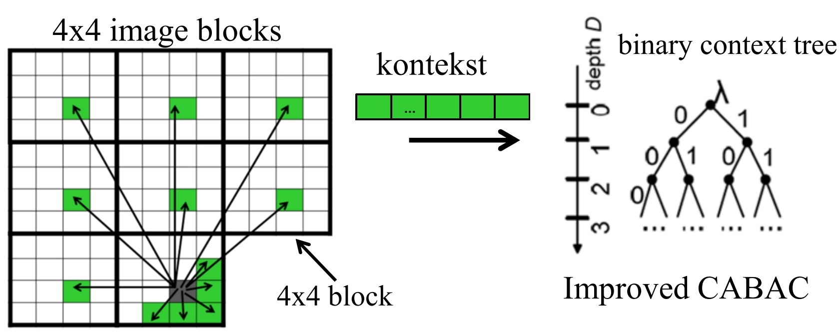

Fig. 3. Context modeling of transform data in improved CABAC

In this way, statistics of transform residual data can be track more accurately that it was done in the original CABAC.

5 Coding efficiency of improved CABAC - methodology of experiments

Presented improvements of data statistics modeling block were implemented and put to original CABAC. Resultant improved CABAC was subsequently activated in the framework of JM 10.2 reference software of MPEG-4 AVC/H.264 video encoder [10]. Coding efficiency of improved MPEG-4 AVC (with improved CABAC) was tested and confronted with performance of the original MPEG-4 AVC (with original CABAC). Experiments were done according to the following scenario:

- 1920x1080 resolution test video sequences were used. These are: *BasketballDrive, BQTerrace, Cactus, ParkScene.* The sequences were last recommended by ISO/IEC MPEG and ITU-T VCEG in works toward new video compression standard HEVC.
- Video sequences were encoded with JM 10.2 encoder forcing different values of quantization parameters for I-slices QP_I = 22, 27, 32, 37 (for P-slices QP_P = QP_I + 1, for B-slices QP_B = QP_I + 2). Quantization parameter is responsible for the quality of encoded sequence as well as the target bitrate (for used sequences QP_I = 22 gave the bitrate of 30Mbps and QP_I = 37 the bitrate of 2.5Mbps in average). IBBPBBP GOP structure was set.
- Both the improved and the original video encoders gave identical quality of reconstructed video sequences. Only bitrates in output of encoders were different.

6 Efficiency of improved CABAC - experimental results

Experimental results showed that application of more accurate mechanisms of data statistics estimation in CABAC allows reasonable increasing the coding efficiency of entropy encoder. In the framework of MPEG-4 AVC video encoder improved CABAC reduces bitrate by 1.5% - 3% relative to the original CABAC. It must be stated that the coding gain was obtained in lossless part of video encoder. Different gains were observed for different test video sequences. Different results were obtained for individual target bitrates (it corresponds to different quantization parameter values). Both the content of video sequence and the target bitrate significantly affect distribution of residual data that is a result of prediction and transform coding mechanisms in video encoder. In turn, efficiency of entropy encoder strongly depends on residual data nature.

In general, the more complex video sequence (rich texture, complicated motion model) the bigger difference in performance of tested entropy encoders. Previous author's results with older version of improved CABAC revealed that bitrate savings for such sequences can even reach 7% - 9% [7]. Additionally, improved CABAC better cope with data of low bitrates than original CABAC that uses pre-defined probability distribution. Detailed results for I-, P-, and B-slice types were presented in Figure 4. Results were averaged over 4 test video sequences.

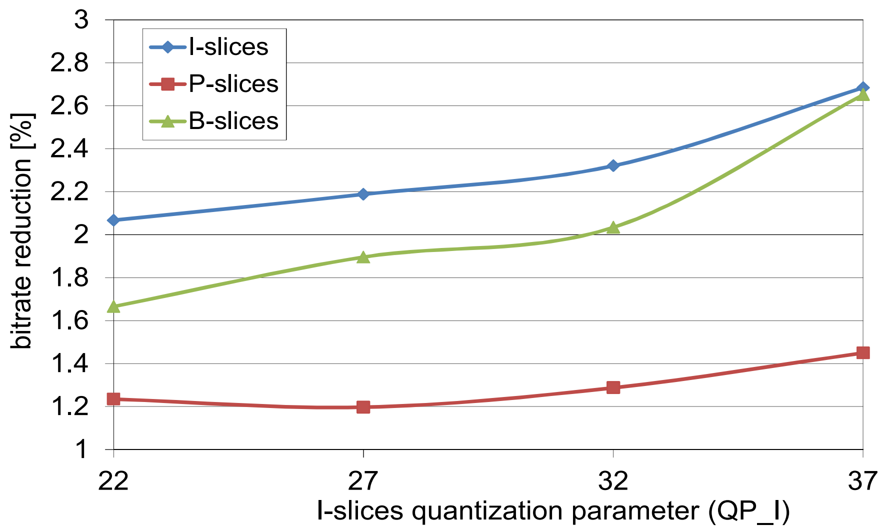

Fig. 4. Average bitrate reduction due to application of improved MPEG-4 AVC encoder instead of the original MPEG-4 AVC (for I-, P-, and B-frames)

As can be seen, different bitrate savings were noticed for individual slice types due to different statistics of residual data in I-, P-, and B-slices. Best results were observed for I-slices. Improved CABAC contains two revised mechanisms of data statistics modeling: more accurate probability estimation and bigger size context pattern. Preliminary experimental results showed that application of bigger size context pattern improves compression performance of entropy encoder slightly only (less than 0.3%). MPEG-4 AVC encoder realizes transform coding on low energy residual signal. Following quantization process additionally reduces energy of signal. Therefore, statistical modeling of such a signal is extremely difficult. Due to the bigger residual signal energy in I-slices, best results were observed for this slice type. Higher compression performance of improved CABAC was achieved at the cost of computational complexity increasing. Application of improved CABAC in MPEG-4 AVC video encoder only marginally increases complexity of video encoder (less than 5%). In the case of video decoder, computational effort increases up to 25% - 30% for useful range of bitrates below 8 - 10 Mbps.

7 Conclusions

Improved version of CABAC entropy encoder was presented in the paper. Revised entropy encoder contains two original improvements relative to the original CABAC. These are: more accurate CTW-based mechanism of data statistics estimation and bigger size context pattern for transform coefficients data. Both proposals of improvements for adaptive arithmetic encoder targets high performance video coding. In this scenario, improved CABAC yields 1.5% - 3% reduction of bitstream relative to original CABAC encoder in the framework of MPEG-4 AVC high performance video encoder.

Acknowledgment

The work was supported by the public founds as a research project in years 2010 - 2012

References

1. ISO/IEC 14496-10, Generic Coding of Audio-Visual Objects, Part 10: Advanced Video Coding, March 2006
2. Special issue on H.264/AVC video coding standard, IEEE Transactions on Circuit and Systems for Video Technology, Vol. 13, July 2003
3. I. E. G. Richardson, The H.264 Advanced Video Compression Standard. Second Edition, Wiley 2010
4. D. Marpe, H. Schwarz, and T. Wiegand, Context-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression Standard, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, No. 7, pp. 620-636, July 2003
5. D. Karwowski, Advanced Adaptation Algorithms of Arithmetic Coding in Hybrid Video Compression, Doctoral Dissertation, Pozna University of Technology, 2008 (http://www.multimedia.edu.pl/publications/doctoral/index.htm)
6. D. Karwowski, M. Domański, Improved Context-Adaptive Arithmetic Coding in H.264/AVC, European Signal Processing Conference EUSIPCO 2009, August 2009, Glasgow, Scotland
7. D. Karwowski, M. Domański, Improved Context - Based Adaptive Binary Arithmetic Coding in MPEG-4 AVC/H.264 Video Codec, Lecture Notes in Computer Science, no. 6375, ss. 25-32, Proceedings ICCVG 2010 Warsaw, Poland, September 20-22, 2010

8. F. M. J. Willems, Y. M. Shtarkov, and Tj. J. Tjalkens, The Context-Tree Weighting Method: Basic Properties, IEEE Transactions on Information Theory, Vol. 41, No. 3, pp. 653-664, May 1995
9. F. M. J. Willems, The Context-Tree Weighting Method: Extensions, IEEE Transactions on Information Theory, Vol. 44, No. 2, pp. 792-797, March 1998
10. H.264/AVC software coordination site - http://bs.hhi.de/∼suehring/tml