OPTIMISED COMPRESSION STRATEGY IN WAVELET-BASED VIDEO CODING USING IMPROVED CONTEXT MODELS

Toni Zgaljic, Marta Mrak and Ebroul Izquierdo

Multimedia and Vision Research Group, Queen Mary, University of London
Mile End Road, E1 4NS, London, UK
{toni.zgaljic, marta.mrak, ebroul.izquierdo}@elec.qmul.ac.uk

ABSTRACT

Accurate probability estimation is a key to efficient compression in entropy coding phase of state-of-the-art video coding systems. Probability estimation can be enhanced if contexts in which symbols occur are used during the probability estimation phase. However, these contexts have to be carefully designed in order to avoid negative effects. Methods that use tree structures to model contexts of various syntax elements have been proven efficient in image and video coding. In this paper we use such structure to build optimised contexts for application in scalable wavelet-based video coding. With the proposed approach context are designed separately for intra-coded frames and motion-compensated frames considering varying statistics across different spatio-temporal subbands. Moreover, contexts are separately designed for different bit-planes. Comparison with compression using fixed contexts from Embedded ZeroBlock Coding (EZBC) has been performed showing improvements when context modelling on tree structures is applied.

Index Terms— Context modelling, entropy coding, scalable wavelet-based video coding

1. INTRODUCTION

Entropy coding in video coding introduces additional compression to highly decorrelated coefficients obtained by spatial and motion-compensated temporal transforms. Entropy coding based on arithmetic coding supported by context models has been widely adopted as a tool for obtaining high compression in image and video coding standards - H.264 / AVC video coding standard [1] and JPEG 2000 still image compression standard [2]. It has also been applied in wavelet-based video coding, either in a form of entropy coding adapted from JPEG2000 or using popular Embedded ZeroBlock Codec (EZBC) [3]. The key to obtain high compression is in careful design of context models which are used to enhance probability estimation.

Context modelling is a process of designing contexts for symbols to be encoded based on the values of neighbouring symbols, i.e. context elements, with a target that the resulting length of the compressed data is minimised. Context modelling can either adaptively choose contexts according to the statistics of underlying source or it can be used as an offline tool for design of predefined contexts. In this paper we optimise context models for application in scalable wavelet-based video coding with a goal to obtain context models that enable higher compression than by using standard context models. It is used to obtain contexts for symbols generated by EZBC bit-plane encoder. The proposed approach builds contexts in a training phase and uses these contexts for coding of individual sequences. Adapted algorithms for growing and pruning of context trees are used to optimise contexts at different spatio-temporal subbands and different bit-planes. In the growing phase of the algorithm, each node of the context tree is assigned with context element which produces minimal length of the code in the current context tree node. In the pruning phase of the algorithm, context tree is pruned in the way so that its leaves correspond to the nodes of the full tree which give minimal overall code length. In this way the algorithm produces context models optimised for encoding of bit-plane coefficients of different spatio-temporal subbands created by a 3D wavelet transform.

The remainder of this paper is organised as follows. Section 2 provides overview of methods used in wavelet-based scalable video coding with an emphasis on EZBC algorithm. In section 3, a method for selecting optimised contexts for symbols generated by bit-plane encoder is presented. Selected experimental results are shown in section 4. Section 5 concludes this paper.

This research was partly supported by the European Commission under contract FP6-001765 aceMedia.

2. ENTROPY CODING WITH CONTEXT MODELS

In wavelet-based scalable coding motion compensated 3D wavelet transform is used to create multi-resolution representation of the input video. Results of such wavelet transform are motion vectors describing spatial displacements of objects in consecutive frames and wavelet coefficients organised in spatio-temporal subbands. Entropy coding of these wavelet coefficients generally consists of bit-plane coding and arithmetic coding of symbols generated by bit-plane encoder. Although these symbols are already highly uncorrelated, the remaining redundancy can be further exploited by careful selection of context models that drive probability estimation at the arithmetic coder. However, a context selection mechanism depends on the data to be encoded and the first step is to provide an efficient binary representation of wavelet coefficients, as described in the following.

Bit-plane encoding is performed by applying a successive approximation quantisation to wavelet coefficients applied in a bit-plane by bit-plane fashion from the highest bit-plane containing the most significant bits to the lowest bit-plane containing the least significant bits. In this way the resulting bit-stream is embedded which provides quality scalability. In order to improve rate-

distortion embedding of the final bit-stream, each bit-plane coding pass generally consists of at least two fractional bit-plane passes. These are the encoding of significance information and the encoding of refinement information. During the significant pass all coefficients that have not been found significant until the beginning of the current bit-plane pass are visited and the information if they have become significant in the current bit-plane is encoded. During the refinement pass, refinement bits of all coefficients found significant in previous bit-planes are encoded. In EZBC algorithm a quadtree representation of individual wavelet subbands is established in order to efficiently exploit information redundancy inherent to wavelet coefficients. Each quadtree node represents maximal value of all of its descendants. Encoding of significance information is performed by bit-plane encoding of quadtree nodes stating from the lowest level of the quadtree to the highest one. When a node on a quadtree level is found significant, its descendants are also checked for significance in a recursive way. This means that encoding of a node at some quadree level can actually be result of processing specific quadtree level (parent mode) or invoked by significance of some of its ancestors at higher quadtree level (descendant mode). It is important to differentiate between these two modes as encoding in descendant mode uses additional contextual information for significance encoding.

Bit-stream generated by EZBC bit-plane encoder is compressed by a binary arithmetic encoder. In order to exploit correlation between neighbouring symbols, arithmetic encoder is driven by conditional probabilities $p(x \mid CTX(x))$ where x is value of the symbol to be encoded and $CTX(x)$ is context in which x appears. Positions of symbols whose values are considered in creation of context define so-called context template. Thus, context template defines which neighbouring symbols (so-called context elements) are going to be considered for conditional probability estimation of the current symbol. Context template for encoding significance information of quadtree nodes in EZBC consists of eight neighbouring nodes as shown in Figure 1. EZBC contexts described in this paper are valid for intra-band context modelling. In inter-band modelling contextual information from parent wavelet subband is also used. In the following sections such context template is applied for both intra-coded frames as well as for motion-compensated frames. However, the final context models are separately designed for those two frame types. Moreover, the models are separately treated for different levels of temporal decomposition, i.e. different levels of motion compensation.

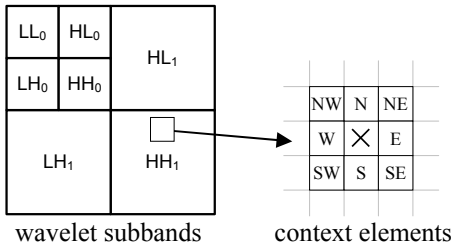

Figure 1 Context template used in EZBC; × represents a position of the symbol to encode.

Although application of contexts can significantly improve compression, using too many contexts can result with context dilution when probability estimation within contexts is inefficient since the number of symbols in each context is to low to obtain a good probability estimate. On the other hand, if too few contexts are used redundancies between symbols are not efficiently exploited resulting in less efficient compression. Thus, contexts have to be carefully selected. Considering these facts EZBC uses linear combination of significance bits at positions of predefined context elements and quantises them into contexts. While EZBC uses the same contexts across different spatial resolutions for the same types of wavelet subbands, it is expected that using different contexts for different spatial and temporal resolutions may further improve compression efficiency. Following this observation, in the work presented in this paper, contexts for symbols created by EZBC bit-plane encoder are optimised separately for different spatio-temporal subbands and different bit-planes using the method described in the following section. Additionally contexts are optimised for each level of quadtree since it is expected that the spatial redundancy of wavelet coefficients within one subband decreases with increasing quadtree level.

In addition to the context definition in EZBC using linear combination of values of context elements, an alternative approach for describing context is by arranging context elements into context trees. Using context tree representation, optimised contexts can be selected for a given context template knowing the statistics of values that are to be encoded. Also, in this approach, the optimisation takes into account the probability models in each context for arithmetic coding. It has been shown that application of Growing, by Reordering and Selection by Pruning (GRASP) algorithm [4] for context modelling using context trees in video coding, optimised context models can be found for various elements of compressed video syntax, improving the overall compression in H.264 / AVC for sequences of high spatial resolution. However, modified GRASP can be used in other compression schemes such as wavelet based-video coding. Starting from GRASP approach, in the following section we present a method modelling of context modelling for application with EZBC in wavelet-based scalable video coding.

3. OPTIMISATION OF CONTEXT MODELS FOR WAVELET-BASED VIDEO CODING

With a goal to optimise context models for different subbands in wavelet-based video coding the context tree structures are used. Modelling of tree structures is performed using statistics of symbols occurring in all possible contexts from the context template. Utilised context template corresponds to the EZBC template (Figure 1). Based on the knowledge of statistics in all possible contexts derived from available context elements, the optimised context models will be designed in the proceeding step.

In the initial step information is collected separately for each so-called syntax element. In contrast to the conventional EZBC, in our approach one syntax element is defined for each combination of different spatio-temporal subbands, colour components, quadtree levels / refinement pass and selected set of bit-planes. Additionally, cases where the symbols are created by encoding of quadtree nodes in the descendant or parent modes are differentiated.

In the actual optimisation step for each syntax element a context tree is optimised for data from training set. Here, a tree-building algorithm is described for one syntax element that occurs in G groups of pictures (GOPs) of a training set. As in GRASP algorithm, starting from an empty tree root at tree depth $d = 0$, an

index j of context element is assigned to the current node. y_j^d denotes assignment of value of the context element with index j, to a tree node at depth $d < D$, where D is the maximal tree depth, i.e. the number of context elements. For a current node at depth d, j is chosen from set $\{0, ..., D - 1\} \setminus \{j_0, ..., j_{d-1}\}$, so that the code length L_j is minimised. L_j is obtained by encoding using sequence of context assignments $Z_j = \left\{ y_{j_0}^0, y_{j_1}^1, ..., y_{j_{d-1}}^{d-1}, y_j^d = l \right\}$ to reach the node at the depth $d + 1$ from the root node. For each available context element it is computed as:

$$L_j = -\sum_{l=0}^{1} \sum_{g=1}^{G} \sum_{t=0}^{N(g,Zj)-1} \log_2 \frac{c_{b(t,g)}^{Zj}(t-1,g)+1}{c_0^{Zj}(t-1,g)+c_1^{Zj}(t-1,g)+2}, \quad (1)$$

where G denotes number of GOPs, $N(g, Z_j)$ is a number of symbols corresponding to observed syntax element that occur in the current node for GOP with index g in a sequence of context assignments Z_j. c_i are counters for binary symbols, $i \in \{0, 1\}$, and $b(t, g)$ is a symbol at the time instance t of GOP g in a corresponding context. Initial values of counters, $c_i(-1, g)$ are set to 0. With each occurrence of symbol $b(t, g)$, counter $c_{b(t, g)}$ is updated. The evaluation and selection of available context elements is performed recursively until the maximal tree depth is reached. In contrast to original GRASP algorithm, which considers only counts of binary symbols regardless of the time of their occurrence, this mechanism allows using additional features of probability modelling, such as scaling of counts. In this way it simulates the steps of the encoder producing the code exactly as the encoder would produce it.

Final context tree selection is performed in the opposite way: from context tree leaves towards the root using tree-pruning algorithm. The code length expected in each node, as given in (1) for selected context element, is compared to the code lengths of its child nodes. If the code length of the current node is smaller then the sum of the code lengths of its child node, the branch below current node is removed from the tree. Otherwise the current node is associated with the sum of the code lengths of its child nodes. This algorithm is also recursively performed and as a result an optimised context tree is found.

An example of context tree obtained with this proposed modelling algorithm is shown in Figure 2. Optimal tree nodes are labelled with symbols for context elements used in EZBC while the whole tree is built and evaluated by context modelling.

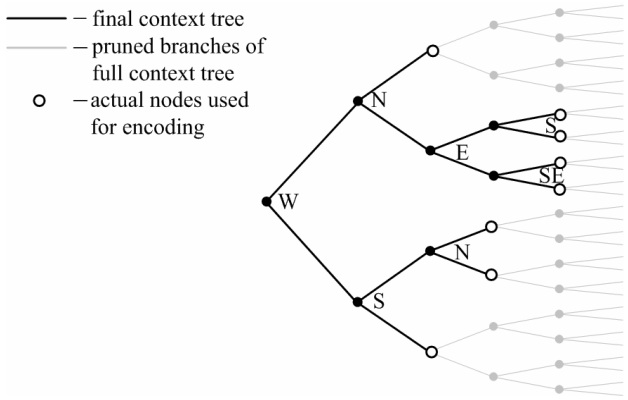

Figure 2 An example of context tree obtained by optimisation on context template from Figure 1.

Final context trees for each syntax element, as obtained by the pruning step, are then implemented in encoder and decoder and used for actual coding of video content.

4. EXPERIMENTAL RESULTS

Experiments were performed in scalable video coding environment [5], [6]. Training set used for context tree optimisation consisted of 4CIF (704 × 576) sequences of 30 fps. The sequences used in training phase were: "Basket", "City", "Crew", "Fair", "Harbour", "Ice", "Mobile" and "Soccer". Sequences were encoded using t + 2D decomposition scheme in which temporal decomposition is followed by spatial decomposition. Five levels of spatial wavelet transform using biorthogonal CDF 9/7 wavelet and four levels of temporal decomposition using bidirectional prediction (1/3 filter) were applied. Contexts were optimised jointly for all training sequences considering GOP structure of 32 frames. Context models for lower bit-planes $bp \in \{0, 1, 2, 3\}$ were optimised separately while higher bit-planes ($bp > 3$) were taken jointly.

Obtained results are shown in Table 1 for compressed data corresponding to encoding significance information of quadtree nodes in parent mode of 0-th, first and second quadtree level of luminance component. Results show average, minimum and maximum relative bit-rate savings averaged over all spatio-temporal subbands when contexts obtained by modified GRASP are used. Relative bit-rate saving is computed as

$$rbs = \frac{L_{wc} - L_c}{L_{wc}} \cdot 100\ [\%], \quad (2)$$

where L_{wc} denotes code length of all frames obtained without using contexts and L_c denotes code length of all frames when context are used. Average bit-rate savings are calculated for encoding of individual sequence in the following way: code length of all sequences was summed for cases without contexts and with optimised contexts obtained by GRASP and then savings were calculated according to (2). In Table 1, n represents the number of bit-planes removed from full quality bit-stream (≈ 45 dB, i.e. visually lossless). It is important to note here that all points for one sequence were extracted from a single scalable bit-stream. It can be seen that in all points application of GRASP contexts improves compression efficiency.

	Relative bit-rate saving by using contexts (*rbs*) [%]				
	$n = 0$	$n = 1$	$n = 2$	$n = 3$	$n = 3$
Avg.	2.05	2.39	2.86	3.54	4.50
Max.	2.97	3.31	3.63	4.13	4.87
Min.	1.50	1.75	2.16	2.82	3.89

Table 1 Performance results for encoding of individual sequences.

In Figure 3 the performance of the proposed context modelling algorithm based on GRASP is compared to the application of context models from EZBC. For the "Soccer" test sequence relative bit-rate saving are separately calculated for temporal low-pass subbands (t-L; intra-coded frames) and for

temporal high-pass subbands (t-H; motion-compensated frames). Application of optimised context-trees improves compression with respect to EZBC in both cases. Gains are higher for intra-coded frames because these frames have higher correlation which can be exploited with context optimisation. When lower bit-planes are removed from the bit-stream, the gain of GRASP over EZBC is lower because of fewer coefficients in higher bit-planes. In general, if fewer coefficients are expected the context trees should not be deep because of context dilution.

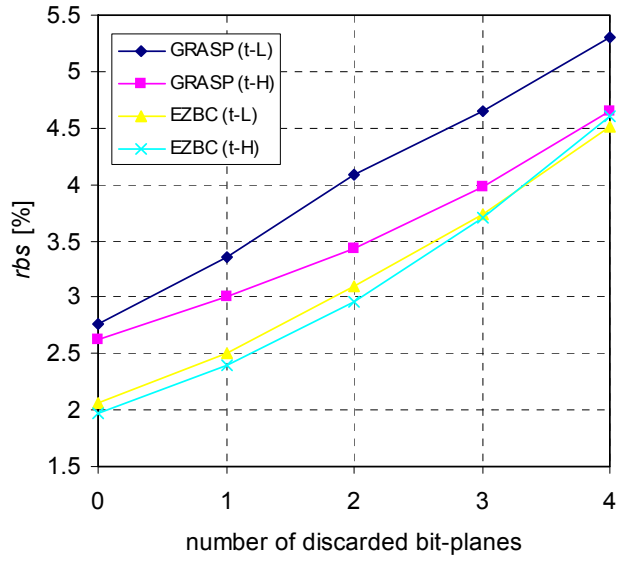

Figure 3 Relative bit-savings for low-pass and high-pass temporal frames (Soccer sequence).

Actual bit savings that are introduced by application of optimised context models are presented in Figure 4 for the "Harbour" test sequence. Results are displayed relatively to encoding from tree root, i.e. without context modelling, for all spatio-temporal subbands. Results obtained by using EZBC context are also included in comparison. Regardless of the number of bit-planes removed from the bit-stream, application of optimised context models gives higher savings than application of EZBC contexts.

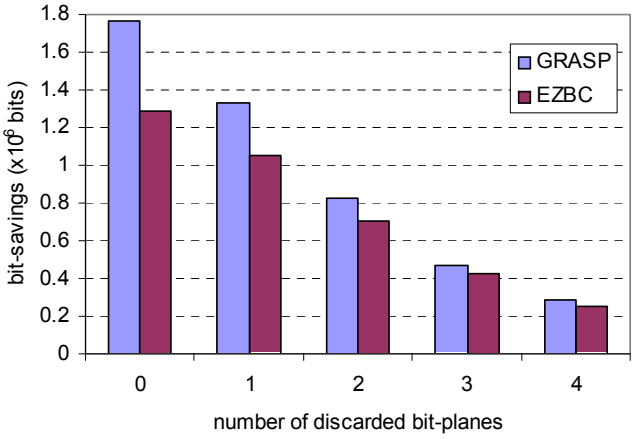

Figure 4 Bit-savings for the Harbour sequence.

5. CONCLUSION AND FUTURE WORK

A strategy for context optimisation in motion-compensated wavelet-based scalable video coding has been presented. Targeted coding elements are symbols generated by EZBC bit-plane encoder. Since it is expected that using of different contexts for different spatio-temporal subbands, quadtree levels, colour components and bit-planes improves compression, application of different context models for each of those combinations needs to be supported by the modelling technique. Chosen optimisation approach is based on GRASP algorithm for context modelling which is capable of selecting context models for various types of data in video coding. In the proposed approach the contexts are built according to data from a training set. These are then used as predefined contexts for encoding. In contrast to predefined contexts used in EZBC, contexts obtained by presented approach are arranged in context tree structures and are separately designed for each syntax element. Experiments measuring compression efficiency when optimised contexts are implemented have been performed for high resolution sequences and motion-compensated video coding. Presented results show that with application of new context models the bit-rate savings are achieved in compared to compression using EZBC contexts. Moreover, it has been observed that relative bit-rate savings depend on number of bit-planes discarded from the compressed bit-stream. For lower bit-planes of motion-compensated frames the application of context trees can use underlying source statistics in greater extend than EZBC contexts, while on higher bit-planes the gain is marginal. The highest gains of new approach are observed on intra-coded frames where remaining correlation of spatially transformed frames can be further exploited by careful context modelling on tree structures.

6. REFERENCES

[1] D. Marpe, H. Schwarz, Thomas Wiegand, "Context-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression Standard," *IEEE Trans. on Circuits and Systems for Video Techn.*, Vol. 13, No. 7, pp. 620-636, July 2003.

[2] D. Taubman, M. W. Marcellin, *JPEG2000 image compression: fundamentals, standards and practice*, Kluwer Academic Publishers, 2002.

[3] S.-T. Hsiang, "Embedded image coding using zeroblocks of subband/wavelet coefficients and context modeling," *Proc. Data Compression Conf.*, pp 83-92, March 2001.

[4] M. Mrak, D. Marpe, T. Wiegand, "A Context Modeling Algorithm and its Application in Video Compression," *Proc. Intl Conf. on Image Proc.*, ICIP 2003, September 2003.

[5] N. Sprljan, M. Mrak, T. Zgaljic, E. *Izquierdo, Software proposal for Wavelet Video Coding Exploration group*, ISO/IEC JTC1/SC29/WG11/MPEG2005, no. M12941, 75th MPEG Meeting, January 2006.

[6] T. Zgaljic, N. Sprljan, E. Izquierdo, "Bit-stream allocation methods for scalable video coding supporting wireless communications," *Signal Processing: Image Communication*, Vol. 22, No. 3, pp. 298-316, March 2007.