SCALABLE HYBRID VIDEO CODERS
WITH DOUBLE MOTION COMPENSATION

Marek Domański, Łukasz Błaszak, Sławomir Maćkowiak, Adam Łuczak

Poznań University of Technology, Institute of Electronics and Telecommunications,
ul. Piotrowo 3a, 60-965 Poznań, Poland
contact: [domanski@et.put.poznan.pl]

ABSTRACT: Considered is the spatial scalability possibly mixed with the temporal and SNR scalability in hybrid video coders with block-based motion compensated interframe prediction. Scalability is implemented in a DCT-based coder (like MPEG-2 or H.263) or in new-generation Advanced Video Coder (AVC/H.264/ISO 14496-10). Here, the scalability is achieved in a coder structure consisting of two hybrid sub-coders with independent motion estimation and compensation. Considered are the rationale and consequences of such an assumption with respect to some reference standard coders.

KEYWORDS: scalable video coding, spatial scalability, hybrid video coder

1. INTRODUCTION

The history of the last twenty years as also the history of fascinating development of image and video compression tools that has open us the doors to a new world of digital media. Nevertheless, these techniques have been developed mostly for wired networks. Recently, the wireless networks has gained such a level of efficiency that they can be used for delivering multimedia content. It means that the video coding systems have to be adapted to unreliable wireless systems with their fades, transmission errors and bandwidth fluctuations [1,2]. Currently, various solutions are discussed. Among them, scalability [3] is quit often considered as the very important functionality needed for wireless multimedia networks.

Scalability means that a video data bitstream is partitioned into layers in such a way that the base layer is independently decodable into a video sequence with reduced spatial resolution, temporal resolution or signal-to-noise ratio (SNR). Enhancement layers provide additional data necessary for video reproduction with higher spatial resolution, temporal resolution or signal-to-noise ratio. This functionality is called spatial, temporal or SNR scalability, respectively, as already defined by the existing video coding standards: MPEG-2 [5] and MPEG-4 [6]. In the case of bandwidth decrease, the receiver decodes only the base part of the bitstream.

Unfortunately, the scalable coding schemes provided by MPEG-2 and MPEG-4 are not satisfactory in some aspects, like coding efficiency and bandwidth adaptation flexibility.

Although MPEG-4 [6] has adopted Fine-Granularity-Scalability (FGS) as a tool for precise tuning a bitstream to channel payload, its coding efficiency is not satisfactory because of lack of temporal prediction in the enhancement layer.

There were many attempts to improve spatially scalable coding of video. Great expectations are related to the inherently scalable wavelet-based techniques [7,8], which have been successfully exploited for flexibly scalable still image compression in the new international standard JPEG 2000 [9]. Recent developments of 3-D wavelet video coders [11,12,13] are extremely interesting for inherently scalable video compression. Another group of techniques exploits the hybrid coder structures based on motion-compensated prediction and transform block coding [15,16,21,22,23]. A similar approach has been proposed by the authors who introduced a concept of spatio-temporal scalability being a mixture of spatial and temporal scalability [17,18,19]. This approach was quit successful but mixing this technique with FGS provides even more flexible structure of the encoder [20].

The paper deals with an efficient coder structure that consists of two motion-compensated hybrid coders with independent motion estimation and compensation (Fig.1). The structure implements spatial scalability or mixed spatial and temporal scalability that can be combined with fine granular SNR scalability. The encoder exhibits extended capabilities of adaptation to network throughput.
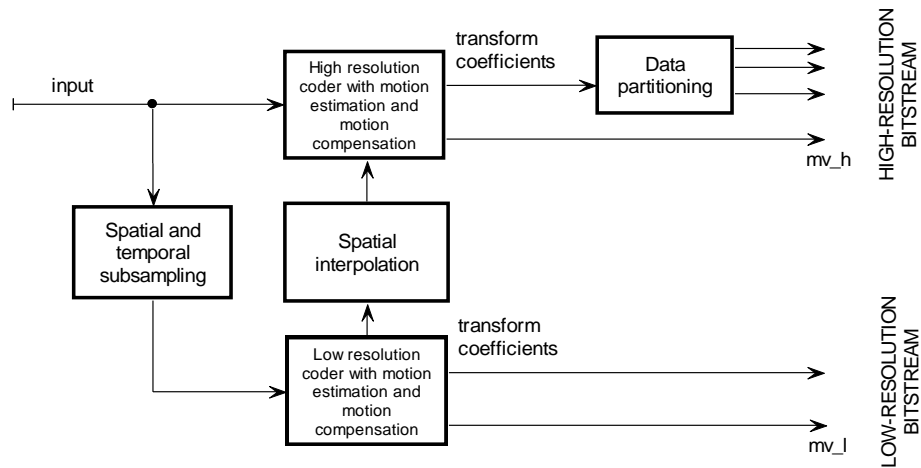
Fig. 1: A general structure of a two-loop scalable coder.

2. CODER STRUCTURE

The high resolution encoder (Fig. 2) is a modification of the H.264 [4] encoder. In this encoder, the interpolated image from the base layer is used as an additional reference frame [17]. A high resolution P-frame is predicted using the previous reference frame, as well as the interpolated current low resolution frame encoded in the base layer. For each macroblock, the best prediction is selected from three blocks, i.e. two motion-compensated blocks and their average.

The low resolution encoder produces a base layer bitstream with reduced spatial and temporal resolution. Temporal resolution reduction is achieved by partitioning the stream of P-frames: each second frame is not included into the base layer. The bitstream produced in the base layer is described by H.264 standard syntax.

The proposed encoder applies independent motion compensation loops in all layers. The motion vectors mv_l for the low resolution frames are estimated independently from the mv_h, which are estimated for the high resolution images. This solution was chosen after some experiments.

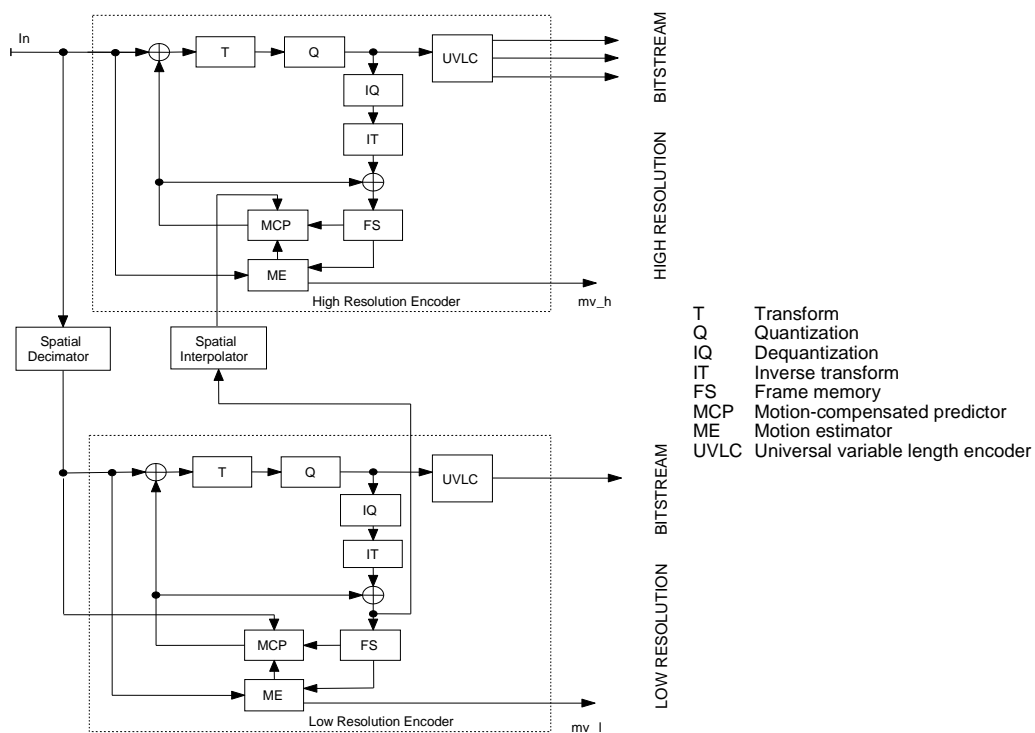|	
T	Transform
Q	Quantization
IQ	Dequantization
IT	Inverse transform
FS	Frame memory
MCP	Motion-compensated predictor
ME	Motion estimator
UVLC	Universal variable length encoder

Fig. 2: Detailed scheme of the encoder (temporal subsampling is not included in this figure).

The H.264 video coding standard is used as a reference but the results are also applicable to the MPEG-2/4 systems with minor modifications. The coder exhibits high level of compatibility with standard H.264 and MPEG 2/4 coders.

3. DOUBLE MOTION ESTIMATIONS AND COMPENSATIONS

Motion estimation is an important component of video coding systems because it enables us to exploit temporal redundancy in a video sequence. The scalable encoder proposed by the authors consists of two encoders. The low and high resolution layers produced by the low and high resolution encoders are characterized by different spatial and temporal resolution of the coded frames. Additionally, every second frame is skipped in the low resolution encoder. This means that motion estimation and compensation processes in the low and high resolution layers are performed for the frames from different time moments.

In the proposed encoder, it is possible to use independent motion estimation and compensation in the low and high resolution layers. Motion compensation processes in layers results in higher compression ratio. Therefore, the problem of efficient estimation of motion vectors in layers needs to be solved.

High efficiency of scalable encoding requires using some information from the base layer in the high resolution encoder. The high resolution layer encoder uses the interpolated decoded frame from the base layer in the prediction of the full resolution frame. It is assumed that the base layer represents a video signal with half the spatial resolution. Therefore one macroblock in the base layer corresponds to four macroblocks in the enhancement layer.

In the scalable encoder, motion estimation may be performed either once or twice. Estimation performed only once is characterized by low complexity, which is an advantage. Two estimations are more complex and require more calculation time. However, it may be expected that they will yield a more precisely predicted frame. The authors have conducted experiments to compare the coding efficiency in both cases.

Two independent motion estimation and compensation processes yield the best results because due to the estimation and compensation in the low resolution layer, there is coarse motion compensation for slowly moving objects in the high resolution layer. The second motion estimation and compensation give more precise prediction.

Since every second frame is skipped in the low resolution encoder, motion estimation and compensation processes in the low and high resolution layers are performed for the frames from different time moments.

This explains why one estimation and compensation process is not adequate in scalable encoding.

Let us consider scalable H.264 encoder and let us assume that the high resolution encoder produces a bitstream which represents a 352 x 288 30Hz progressive test sequence and that the base layer encoder produces a bitstream which represents a 176 x 144 15Hz progressive test sequence. Therefore there are 396 macroblocks in the high resolution layer and 99 macroblocks in the low resolution layer. For simplicity let us consider only the 16 x 16 prediction mode. In the enhancement layer the motion is estimated for high resolution images and full-frame motion compensation is performed. It means that there is a second, more precise motion compensation. Therefore the number of motion vectors mv_h estimated in the high resolution encoder is about four times bigger than the number of motion vectors mv_l estimated in the low resolution encoder. In fact, some enhancement macroblocks are interpolated from the low resolution layer and the respective motion vectors mv_h need not be sent in the high resolution layer. Therefore the total number of motion vectors is less than 25% bigger than the respective number of motion vectors of non-scalable bitstream.

Table 1. Comparison of motion estimation complexity of H.264 scalable and H.264 non-scalable encoders (This calculation is performed for full search estimation with half-pixel accuracy and 352 x 288, 30Hz input progressive test sequences, the search range D=±31).

		The number of operations for one macroblock	The number of operations per second
Non-scalable encoder (352 x 288 30Hz)		$(2D+1)^2 N_1 N_2$	11.668 x 10^9
Scalable encoder	Base layer (176 x 144 15Hz)	$(2D+1)^2 N_1 N_2$	2.917 x 10^9
	Enhancement layer (352 x 288 30Hz)	$(2D+1)^2 N_1 N_2$	11.668 x 10^9
	Overall		14.585 x 10^9

Complexity estimation constitutes another problem. Table 1 presents the comparison of motion estimation complexity of scalable and non-scalable encoders. This comparison was performed under the following conditions: the calculation was performed for half-pixel accuracy. In the low and high resolution layers the full search estimation was performed. The encoded progressive 30 Hz sequence in 1 second consisted of 1 I-frame and 29 P-frames in the enhancement layer and 1 I-frame and 14 P-frames in the base layer. Therefore, there were 14 motion estimations in the base layer and 29 motion estimations in the enhancement layer.

The scalable encoder with double independent estimation requires 2.917×10^9 calculations per second more than the non-scalable encoder.

4. MOTION VECTORS ENCODING IN H.264 SCALABLE ENCODER

The motion vector encoding method is derived from H.264 encoder. In proposed H.264 scalable encoder the motion vectors from the base and enhancement layers are encoded independently in the following way.

For every macroblock there may be from 1 up to 16 motion vectors transmitted. For every block a prediction is formed for the horizontal and vertical components of the motion vectors. The transmitted value signals the difference between the vector component to be used and this prediction. For all block shapes, except the 16x8 and 8x16, "median prediction" is used. For those two exceptions neighboring block is taken as a prediction. For block that are marked as "only interpolation from lower layer" vector is set to prediction value. It probably should be replaced by "different reference picture" marker, so this block shouldn't be taken into account in vector prediction.

5. FINE GRANULARITY

In the data produced by both encoders, there exist some granules that can be arbitrarily assigned to successive layers. The base layer bitrate may be reduced in the following way. For the whole low resolution bitstream, this bitrate is about 30 - 40% of the total bitrate. The base layer bitrate can be reduced if only part of non-zero DCT coefficients from the low resolution bitstream are allocated to the base layer, or more precisely, if the base layer comprises only a part of Huffman codes representing (run, level) pairs corresponding to the non-zero DCT coefficients from the low resolution bitstream. The process runs as follows. The encoder allocates headers and low resolution motion vectors mv_l to the base layer first. Then the codes of nonzero DCT coefficients are allocated, starting from the DC coefficients from all blocks, followed by the coefficients related to increasing frequencies in the zig-zag order from all blocks at once. The process ends when the bit budget is exhausted, thus leaving the remaining coefficients for the enhancement layers.

In this way, the base layer bitrate may be reduced below 15% of the total bitrate of the order of a few megabits per second. A similar strategy may be used in order to create a larger number of layers from both low and high resolution bitstreams. Except from headers and motion vectors, the bitstreams can be arbitrarily split into layers and multi-layer fine granularity can be achieved. All header data and the enhancement motion vectors mv_h may be treated as basic granules [20]. The next granules are constituted by DCT coefficients that are encoded as (run, level) pairs. The lower layer contains N_m first (run, level) pairs for individual blocks. The control parameter N_m influences bit allocation to layers. The bitrates in subsequent layers can be controlled individually. To some extent, nevertheless, each additional layer increases the bitrate overhead because at least slice headers should be transmitted in all layers in order to guarantee resynchronization after an uncorrected transmission error. The total bitstream increases by about 3% per each layer obtained using data partitioning.

The drawback of this strategy is accumulation of drift, since the base layer decoder has no access to full low resolution reconstructions. Drift is also generated by partitioning the high resolution bitstream. Moreover, when the enhancement layer bitstream is corrupted by errors during transmission, the enhancement layer DCT coefficients cannot be properly reconstructed due to the loss of DCT information. This causes drift between the local decoder and remote decoder.

In the authors solution, drift accumulation is also reduced because the total bitstream is divided into two drift-free parts, i.e. the low and the high resolution bitstream. Drift propagates within one part only when fine granularity is applied to high resolution bitstream. Furthermore, drift in this part may be reduced by more extensive use of the low resolution images as reference.

6. EXPERIMENTAL RESULTS

In order to evaluate the proposed scalable encoder the experimental results have been made using ITU-T Rec. H.264 (2002 E) / ISO/IEC 14496-10:2002 (E) codec Joint Model 2.1. The coder with both spatial and temporal scalability has been tested. Both temporal and spatial subsampling factor was set to 2. It means that the sequences were:

- Enhancement layer: progressive CIF 4:2:0 (352x288) 30 Hz,
- Base layer: progressive QCIF 4:2:0 (176x144) 15 Hz.

The experiments fulfilled the following conditions:

- I-P-P-P GOP structure,
- quantization parameter QP for first frame equal 15,
- quantization parameter QP for other frames equal 16,
- Hadamard transformation,
- one reference frame for prediction,
- all inter search modes,
- CABAC entropy coding method.

The comparison of coding performance between simulcast solution and scalable one is charted in Fig. 3. For all test sequences scalable encoder produces smaller bitstream than in case of simulcast, but larger then non scalable single layer encoder.
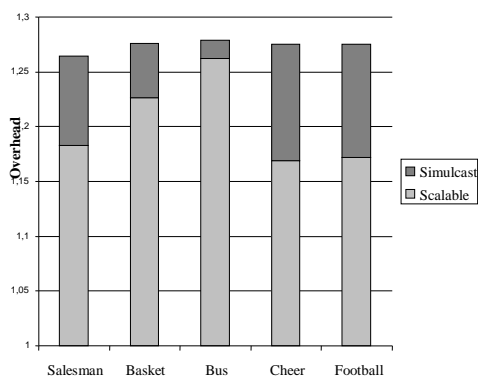

Fig. 3: The comparison of bitrate overhead between simulcast and scalable H.264 encoding

7. CONCLUSIONS

Described is a two-layer H.264 scalable encoder with the functionality of fine granularity. The major differences with respect to other proposals are: mixed spatio-temporal scalability, independent motion estimation for each motion-compensation loop and improved prediction of P-frames. These features are also the reasons for very good performance of the whole coder.

The bitrate of the base layer can be smoothly controlled starting from below 15% of the total bitrate. The bitrate overhead due to scalability exceeds 10% but is most below than simulcast overhead.

8. REFERENCES

[1] L. Hanzo, P. Cherriman, J. Streit, Wireless Video Communications, IEEE Press, New York, 2001.

[2] M. Al-Mualla, C. Canagarajah, D. Bull, Video Coding for Mobile Communications, Academic Press, Boston 2002.

[3] D. Wu, Y. Hou, Y. Zhang, "Scalable video coding and transport over broad-band wireless networks," Proc. of the IEEE, vol. 89, pp. 6-20, January 2001.

[4] ISO/IEC JTC1/SC29/ WG11MPEG02/ N4920, Text of Final Committee Draft of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC), July 2002.

[5] ISO/IEC IS 13818-2, Information Technology - Generic Coding of Moving Pictures and Associated Audio Information. Part 2: Video.

[6] ISO/IEC 14496-2/FPDAM4, "Coding of Audio-Visual Objects, Part-2 Visual, Amendment 4: Streaming Video Profile, July 2000.

[7] P. Topiwala (ed.), Wavelet image and video compression, Boston, Kluwer 1998.

[8] J.-R.Ohm, M. Beermann, „Status of scalable technology in video coding", Doc. ISO/IEC JTC1/SC29/WG11 MPEG01/M7483, Sydney, July 2001.

[9] ISO/IEC IS 15444-1 / ITU-T Rec. T.800, "JPEG 2000 image coding system."

[10] ITU-T, "Video coding for narrow telecommunication channels at < 64kbit/s", Recommendation H.263, 1996.

[11] J.-R. Ohm, T. Ebrahimi, „Report of Ad hoc Group on Exploration of Interframe Wavelet Technology in Video", ISO/IEC JTC1/SC29/WG11 MPEG02 /M8359, Fairfax, May 2002.

[12] J. W. Woods and P. Chen, "Improved MC-EZBC with Quarter-pixel Motion Vectors", ISO/IEC JTC1/SC29/WG11 MPEG2002 /M8366, Fairfax, VA, May 2002.

[13] S.T. Hsiang, J. W. Woods, "Embedded video coding using invertible motion compensated 3-D subband/wavelet filter bank", Signal Processing: Image Communication 16 (2001) 705–724, 2001.

[14] K. Rose, S. Regunathan, "Toward optimality in scalable predictive coding," IEEE Trans. Image Proc., vol. 10, pp.965-976, July 2001.

[15] Y. He, R. Yan, F. Wu, S. Li, "H.26L-based fine granularity scalable video coding," Doc. Doc. ISO/IEC JTC1/SC29/WG11 MPEG01/m7788, December 2001.

[16] U. Benzler, "Spatial scalable video coding using a combined subband-DCT approach", IEEE Trans. Circuits Syst. for Video Techn., vol. 10, pp. 1080-1087, Oct. 2000

[17] M. Domański, A. Łuczak, S. Maćkowiak, „Spatio-temporal scalability for MPEG video coding", IEEE Trans. Circ. Syst. Video Techn., vol. 10, pp. 1088-1093, October 2000.

[18] M. Domański, A. Łuczak, S. Maćkowiak, "On improving MPEG spatial scalability," Proc. Int. Conf. Image Proc. ICIP 2000, IEEE, pp. 2/ 848-851, Vancouver, Sept. 2000.

[19] A. Łuczak, S. Maćkowiak, M. Domański, "Spatio-temporal scalability using modified MPEG-2 predictive video coding", Signal Processing X: Theories and Applications, Proc. EUSIPCO-2000, pp. 961-964, Tampere, Sept.. 2000.

[20] M. Domański, S. Maćkowiak, „Modified MPEG-2 video coders with efficient multi-layer scalability," Int. Conf. Image Processing ICIP 2001, IEEE, vol. II, pp. 1033-36, Thessaloniki, October 2001.

[21] Y. He, R. Yan, F. Wu, S. Li, "H.26L-based fine granularity scalable video coding," Doc. Doc. ISO/IEC JTC1/SC29/WG11 MPEG01/m7788, December 2001.

[22] A. Reibman, L. Bottou, A. Basso, "DCT-based scalable video coding with drift," Int. Conf. Image Processing ICIP 2001, IEEE, vol. II, pp. 989-992, Thessaloniki, Oct. 2001.

[23] G. Cote, B. Erol, M. Gallant, F. Kossentini, "H.263+: video coding at low bit rates", IEEE Trans. Circ. and Syst. Video Technology, vol. 8, pp. 849-865, November 1998.