On the influence of motion vector precision limiting in scalable video coding

Marta Mrak, G.C.K. Abhayaratne and Ebroul Izquierdo

Department of Electronic Engineering, Queen Mary, University of London
Mile End Road, E1 4NS, London, UK
{marta.mrak, charith.abhayaratne, ebroul.izquierdo}@elec.qmul.ac.uk

Abstract

Recent studies on scalable video coding have not only substantiated the need for such technology but also made evident that many related problems remain open and need to be tackled if truly scalable video coding is to be achieved. One of these challenges relates to the coding of motion vectors. In conventional coders motion vectors are treated and coded in a non-progressive manner. Since scalable video coding targets decoding at several resolutions and a wide range of quality levels, the motion information needs to be encoded in an adaptive way. We propose a simple, yet efficient, strategy for scalable motion vector coding. The results show improvements of resolution scalability performance at lower-bit rates, while overcoming any negative influence at high resolutions and bitrates.

1. Introduction

Scalable video coding (SVC) provides a solution for the need of a video COding-DECoding scheme (codec) that is capable of serving a broad range of applications that require different quality levels, spatial and temporal resolutions of video. Generally, an SVC codec should provide at least the three main scalability functionalities: spatial resolution, temporal and quality (also known as SNR, i.e., Signal-to-Noise Ratio metric).

Recent research [1]-[6] has shown efficient combinations of techniques for achieving scalable video coding. Most of such frameworks consist of a two-step signal transformation: motion compensated temporal filtering (MCTF) and 2D spatial decomposition. The MCTF mechanism uses motion estimation and wavelet filter banks in lifting realisation in order to remove temporal redundancy. The multi resolution (MR) structure resulting from the wavelet transform in MCTF enables achieving temporal scalability. Similarly, the MR structure resulting from the 2D subband decompositions enables achieving spatial resolution scalability. These signal transforms followed by a 3D embedded coding scheme enable providing all three scalabilities, individually or as combinations.

The MCTF step produces a set of motion descriptors and residuals, also known as texture information, for each of the predicted (high pass) frames in a group of pictures (GOP). The effectiveness of the MCTF technique in a scalable coding system greatly depends on the way such motion information is encoded. For example, non scalable coding of the motion vector fields that have been generated for high spatial resolutions and high quality levels results in a fixed overhead when such coded bit streams are decoded at lower spatial resolutions and lower quality levels.

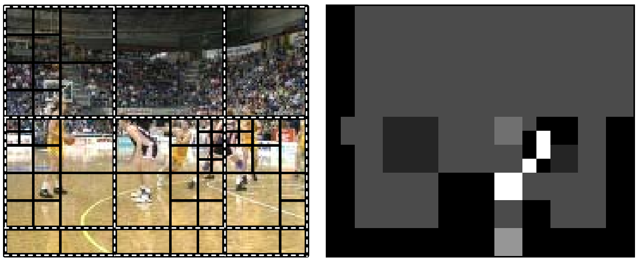
In this paper we present a precision limited coding (PLC) scheme for motion vectors in order to achieve a scalable motion information coding scheme that enhances the rate-distortion performance of SVC codecs. With this scale (in the spatial MR structure) dependent precision limiting scheme one can achieve a simple, yet efficient, method for layered coding of motion vectors. For the experiments shown in this paper we consider the spatial domain MCTF (SD-MCTF) framework, also known as t+2D, in which MCTF is performed prior to spatial filtering. We used the Motion Compensated Embedded Zero Block Context (MC-EZBC) codec [2], [7] in order to test the precision limited motion vector coding scheme proposed in this paper.

The rest of the paper is organised as follows: In section 2, we discuss the motivation for this work; Scalable motion vector coding based on the scale dependent precision limiting is presented in section 3; Experimental results are shown in section 4, followed by the conclusions in section 5.

2. Motivation

In video coding, motion estimation is usually achieved by block matching, i.e., partitioning the frame to be predicted into blocks and then finding the corresponding matched blocks from a reference frame by minimising a distant measure for each of the partitioned blocks. The positions of these matched blocks are usually denoted in terms of a displacement vector. Most of the current video coding standards, such as MPEG-1/-2, H.261 and H.263, use block matching with fixed block sizes. Recently, more efficient variable block size frame partitioning has been introduced in the H.264/AVC video coding standard [8]. A similar approach that uses hierarchical variable size block matching (HVSBM) [9] is widely used in MCTF coding schemes.

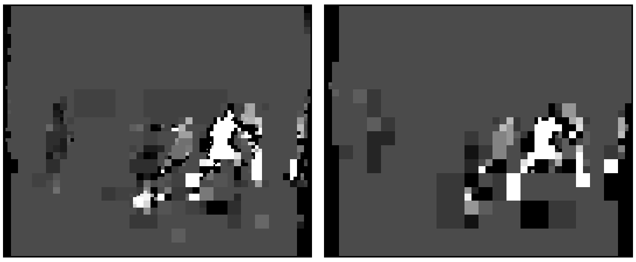
The test codec, MC-EZBC, also uses HVSBM for motion estimation. In HVSBM, a bottom-to-top quadtree pruning algorithm is used in order to find the precise motion estimation for a given block. Such schemes not only estimate motion accurately, but also increase the cost of motion information, both in terms of the number of sub



a) Optimal partitioning of a frame for HVSBM

b) Magnitudes of the corresponding motion vectors

Figure 1: Motion estimation using HVSBM.



a) 4CIF

b) CIF

Figure 2: The optimal motion fields.

blocks leading to the number of vectors and the precision of the vectors leading to the number of bits per vector. Thus, a balance between the gain introduced by high precision motion estimation and the cost of coding high precision motion must be maintained in such motion estimation schemes.

The motion information for a given frame in MC-EZBC codec consists of a motion map that is represented by a quadtree structure and motion vector values attached to each quadtree leaf. An example of the optimal quadtree partitioning for a frame is shown on Figure 1.a. The areas where high motion in various directions is present are highly partitioned into smaller sub blocks, whereas, the areas with uniform motion are not partitioned into sub blocks. Figure 1.b shows magnitudes of the corresponding motion vectors in the horizontal direction. The values are scaled so that the brighter areas correspond to motion vectors with large magnitudes.

A significant observation on HVSBM-based motion estimation is that the optimum quadtree structures obtained for two different spatial resolutions of the same frame are different. For example, we show the magnitudes of the horizontal motion vectors for the corresponding optimised quad tree structure for 4CIF and CIF resolutions in Figure 2. It is evident from Figure 2, that for higher resolutions, the motion map is more detailed.

Another observation is that if the motion information is coded in a non-scalable way, then the cost of motion vectors will remain as a fixed cost in the subsequent decoding points, irrespective of the lower spatial resolutions or lower quality decoding. An example is given in Table 1 using the test sequence "Basket". Table 1 shows motion information content as a percentage of the decoded bit rate, where the encoded and decoded bit rates are the highest bit rates possible for a given spatial resolution in MC-EZBC codec. The first three rows demonstrate the first observation discussed above, whereas, the last two rows demonstrate the second one.

Table 1: The percentages of motion information content at different encoded and decoded resolutions for the highest possible bit rates.

Encoded resolution	Decoded resolution	Motion information content as percentage of the decoded bit rate
4CIF	4CIF	1.22 %
CIF	CIF	1.52 %
QCIF	QCIF	1.77 %
4CIF	CIF	3.15 %
4CIF	QCIF	9.53 %

The last two rows also show the effect of non-scalable motion information coding on the spatial resolution scalability. Further it is evident that in such cases, the bit budget available for residual information is reduced by 1.63% (3.15%-1.52%) and 7.76% (9.53%-1.77%) when 4CIF to CIF and 4CIF to QCIF decoding is performed, respectively. Consequently, this contributes to poor visual quality at lower spatial resolutions. This loss of space for residual information is usually badly felt when the bit streams are decoded at low bit rates.

The above observations justify the necessity for a scalable coding scheme for the motion information in scalable video coding. In the MC-EZBC codec, the motion block map from the highest resolution is used as the motion block map for subsequent lower resolutions and the original motion vectors from the highest resolution are scaled down accordingly. This results in unnecessary motion vector precision at lower spatial resolutions. Thus we propose the scale dependent precision limited coding scheme for the motion vectors in order to achieve scalable motion information leading to better visual quality at lower spatial resolutions and lower bit rates.

3. Scalable Motion Vector Coding

In the MC-EZBC codec, the motion information is coded by first coding the quadtree map of the initial partition and then coding the motion vector values at the quadtree leaves by computing the running difference between the previous vector and the current vector. The running differences are coded using variable length codes (VLC). In the proposed precision limited coding (PLC), each vector is composed into the precision limited component and corresponding refining bits. Then we use the precision limited component of each vector to compute the difference, which will be coded using VLC. The number of possible states V for VLC is calculated using the motion search range, initial accuracy of the motion vectors ($1/M$ pel) and the number of precision reduction levels (N). A block diagram of the PLC based motion vector coding is shown in Figure 3.

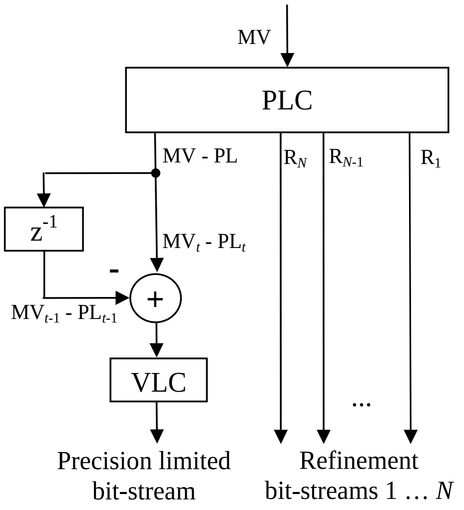


Figure 3: The block diagram for scalable motion vector (MV) coding.

3.1 Computing the precision limiting map

Let X_r be the magnitude of a motion vector component, N be the number of precision limiting levels and $1/M$ be the initial accuracy of motion vectors. Then the precision limiting component of a motion vector is computed as follows:

Step 0 Map X_r into an integer value (X_0);

$$X_0 = X_r \cdot M \tag{1}$$

Set $n = 1$ and $X'_0 = X_0$.

Step 1 Compute the precision limited component (X'_n) at level n;

$$X'_n = \lfloor X'_{n-1} / 2 \rfloor, \tag{2}$$

where $\lfloor \cdot \rfloor$ denotes downward rounding operation.

Step 2 Compute the refinement bit (R_n) at level n;

$$R_n = X'_{n-1} - X'_n \cdot 2 \tag{3}$$

Steps 1 and 2 are repeated until $n = N$. This partitions X_0 into components (X'_N, R_N, R_{N-1}, … , R_1).

3.2 Examples of precision limiting

Two examples of computing precision limiting components for motion vectors with 1/8 accuracy are shown in Table 2 and Table 3.

It can also be shown that the steps 1 and 2 can be computed easily at bit plane levels, by computing the two's complement of the mapped integer value X_0, i.e., the output of step 0. Then the least significant bits (LSB) represent the refinement bits, according to the required precision limiting level (N) and the rest of the bits form the precision limited component. An example is shown in Figure 4.

Table 2: Precision limiting and refinement bits computation for X_r = -1.625.

		X'_N	R_N
Vector value (X_r)	-1.625		
Mapped integer value (X_0)	-13		
$N = 1$		-7	1
$N = 2$		-4	1

Table 3: Precision limiting and refinement bits computation for X_r = 33.625.

		X'_N	R_N
Vector value (X_r)	33.625		
Mapped integer value (X_0)	269		
$N = 1$		134	1
$N = 2$		67	0

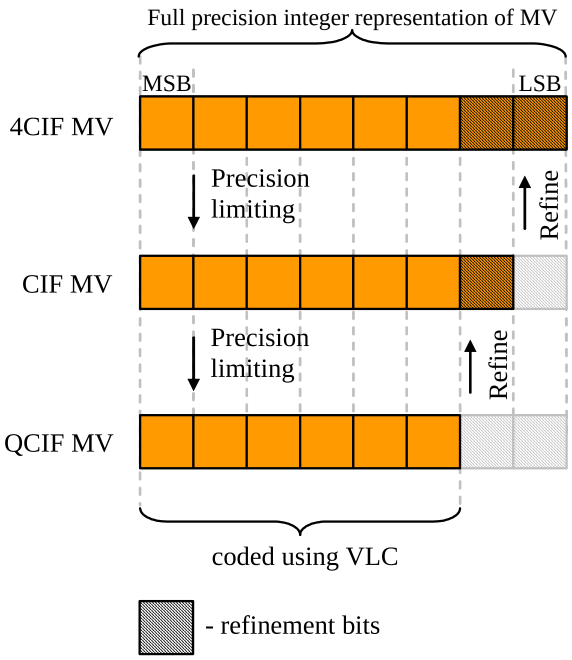
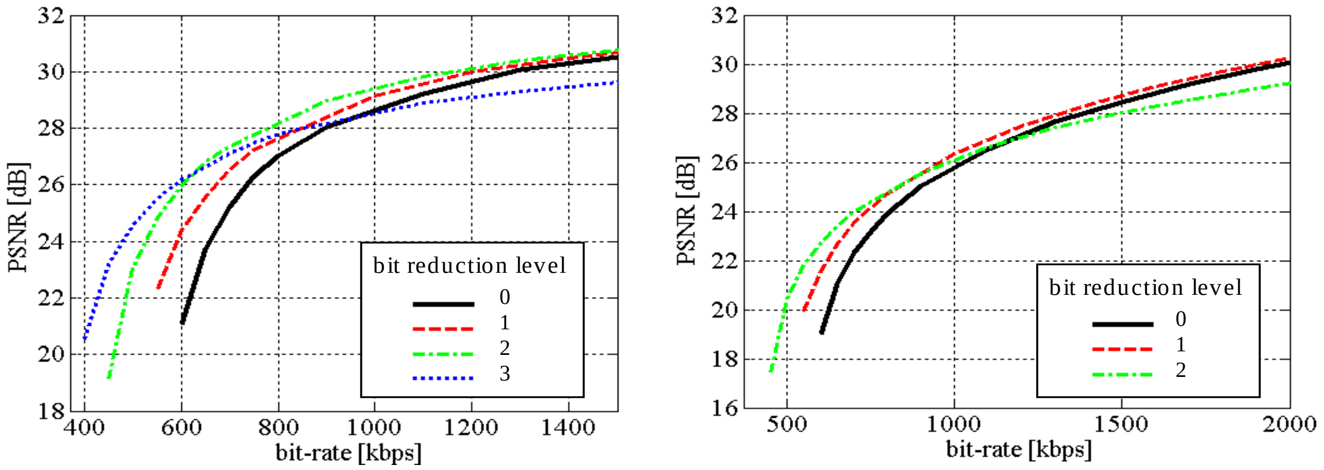


Figure 4: Bit plane domain motion vector (MV) precision limiting.

4. Experimental Results

We replace the non-scalable motion vector coding module in the MC-EZBC with the PLC scheme proposed in the previous section. The rate distortion (R-D) performance plots for decoding at CIF and QCIF spatial resolution levels using a bit stream originally coded at the highest bit rate 4CIF "Basket" test sequence are shown in Figure 5. The anchors for luminance *PSNR* computations at CIF and QCIF resolutions were obtained by down-sampling the 4CIF sequence using the 9/7 wavelet filter.

The plots show the R-D performances for several N values. $N = 0$ corresponds to the non-scalable motion vector coding. At low bit rates, the PLC scheme results in improved R-D performances. Finally an example of actual gain in visual quality is illustrated in Figure 6. It is evident from the figure that a gain of 2.8 dB at very low bit-rates introduces significant visual quality improvement due to bit savings caused by efficient motion vector coding.



a) QCIF

b) CIF

Figure 5: Resolution scalability rate-distortion performance using PLC for test sequence "Basket" (4CIF).

a) Non-scalable MV coding; $PSNR_Y$ = 22.02 dB

b) Scalable MV coding (PLC); $PSNR_Y$ = 24.80 dB

Figure 6: A portion of a frame from decoded QCIF sequence at 600 kbps extracted from 4CIF.

5. Conclusions

We proposed a scale dependent precision limited coding (PLC) scheme for scalable coding of motion vector values, in order to enhance the spatial resolution scalability performances at low bit rates for scalable video coding. The proposed PLC method is a simple, yet an efficient, scheme that limits the precision of the coded vector values according to the spatial resolution levels at the decoder side. The precision limited component that is used as the motion vector for the lowest spatial resolution and the subsequent refinement bits that refine the motion vectors for subsequent higher spatial resolutions form a bit plane domain layered partition of the motion vector field. This coding scheme results in significant gains in both the rate-distortion performance and the visual quality at low bit rates.

Acknowledgments

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

References

[1] J.-R. Ohm, "Three-dimensional Subband Coding with Motion Compensation," *IEEE Trans. on Image Processing*, Vol. 3, No. 5, pp. 559-571, Sept. 1994.

[2] S.-T. Hsiang, and J. W. Woods, "Embedded Video Coding Using Invertible Motion Compensated 3-D Subband/Wavelet Filter Bank," *Signal Processing: Image Communication*, Vol. 16, pp. 705-724, May 2001.

[3] S.-J. Choi, and J. W. Woods, "Motion-Compensated 3-D Subband Coding of Video", *IEEE Trans. on Image Processing*, Vol. 8, No. 2, pp. 155-167, Feb. 1999.

[4] D. Taubman, and A. Zakhor, "Multirate 3-D Subband Coding of Video", *IEEE Trans. on Image Processing*, Vol. 3, No. 5, pp. 572-588, Sep 1994.

[5] J. Xu, Z. Xiong, S. Li, and Y.-Q. Zhang, "Memory-Constrained 3D Wavelet Transform for Video Coding Without Boundary Effects", *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 12, No. 10, pp. 850-856, Oct 2002.

[6] Y. Andreopoulos, M. Van der Schaar, A. Munteanu, J. Barbarien, P.Schelkens, and J. Cornelis, "Complete-to-Overcomplete Discrete Wavelet Transforms for Scalable Video Coding with MCTF", *Proc. of SPIE Visual Communications and Image Processing (VCIP'03)*, Vol. 5150, pp. 719-731, July 2003.

[7] MC-EZBC software package available at NIST MPEG CVS Repository.

[8] ITU-T and ISO/IEC JTC1, "Advanced Video Coding for Generic Audiovisual Services," ITU-T Recommendation H.264 – ISO/IEC 14496-10 AVC, 2003.

[9] M. H. Chan, Y. B. Yu, and A.G. Constantinides, "Variable size block matching motion compensation with applications to video coding", *Communications, Speech and Vision, IEE Proceedings I* ,Vol. 137 , Iss. 4, pp. 205 - 212, Aug. 1990.