ROBUST MOTION VECTOR CODING AND ERROR CONCEALMENT IN MCTF-BASED VIDEO CODING

M. Stoufs, J. Barbarien, P. Schelkens, J. Cornelis, A. Munteanu

Vrije Universiteit Brussel (VUB) – Interdisciplinary Institute for Broadband Technology (IBBT)
Department of Electronics and Information Processing (ETRO),
Pleinlaan 2, B-1050 Brussels, Belgium

ABSTRACT

Error resilience is of paramount importance in video transmission over variable-bandwidth error-prone channels, such as wireless channels. In this paper, we investigate the influence of corrupted motion vectors in video coding based on motion compensated temporal filtering, and develop various error resilience and concealment mechanisms for this class of codecs. The experimental results show that our proposed motion vector coding technique significantly increases the robustness against transmission errors at the cost of less than 3% in terms of rate. It is also shown that our proposed spatial error-concealment mechanism leads to performance gains of up to 6 dB in comparison to a classical slicing-based approach employing no error concealment.

1. INTRODUCTION

An important property that modern communications have exhibited in recent years is concerned with network/user heterogeneity. Heterogeneity assumes various types of mediums for information transmission, from highly reliable, ultra-fast optical links to bandwidth-limited, error-prone wireless channels, coupled with an even larger variety of end-user terminals, ranging from low-power mobile devices to high-definition TVs or high-end desktop computers. From an application perspective, mobile video communications are expected to boost thanks to the introduction of new multimedia services. Streaming video, according to its inherent characteristics, will become a major consumer of network resources. Though in their beginning nowadays, lightweight mobile devices, such as mobile videophones will demand reliable video transmission over wireless channels. In general, video applications expanded over such variable-bandwidth, error-prone channels require a scalable video representation and robustness to transmission errors in order to allow for the adaptation to the inherently variable network conditions and terminal characteristics.

Recent findings – e.g. [1] show that videocodecs based on Motion Compensated Temporal Filtering (MCTF) can achieve compression performance competitive to that of the state-of-the-art non-scalable H.264, and at the same time, provide support for quality, frame-rate and resolution scalability. Until now, most of the research conducted in the field of such MCTF-based video coding systems has been focused on improving their flexibility and performance, without treating the error resilience aspects.

In this paper we propose error resilience mechanisms for motion vectors generated by a video codec based on unconstrained MCTF (UMCTF) [2], [3]. In order to increase the robustness against transmission errors we introduce and compare two motion-vector coding methods. Additionally, to further diminish the degrading effect of packet losses on the video quality we develop and compare various motion vector error-concealment mechanisms operating in the spatial, temporal and spatio-temporal domains respectively.

This paper is structured as follows. In section 2 UMCTF-based coding is briefly explained. The proposed robust motion vector coding techniques are described in section 3. In section 4, the proposed spatial, temporal and spatio-temporal error concealment techniques are presented. Comparative coding results are given in section 5. Finally, section 6 draws the conclusions of our work.

2. UMCTF-BASED CODING: PRINCIPLES

Motion-compensated temporal filtering (MCTF) [4], [5] is a combination of motion estimation and compensation and wavelet filtering in the temporal direction.

UMCTF [2], [3] is an extension of MCTF that supports the selective application of the update step in the temporal decomposition. UMCTF can be used in order to reduce the decoder delay, achieve non-dyadic temporal scalability and reduce the visual artifacts that are introduced by the failures of the motion model and inappropriate performance of the update step in MCTF [3].

To facilitate the description, in Figure 1 one illustrates the temporal decomposition achieved by using UMCTF when no update-step is performed. In this example, bi-directional motion estimation (ME) is performed to predict the odd-numbered frames in the group of pictures (GOP) from the even-numbered ones. The subsequently applied motion compensation (MC) and the production of the error-frames (H-frames) correspond to the predict step of the (2,2) wavelet transform, applied along the motion trajectory. The unchanged even-numbered A-frames of the first temporal level are subsequently used to produce the next temporal level by applying equivalent operations on them.

0-7803-8874-7/05/$20.00 ©2005 IEEE

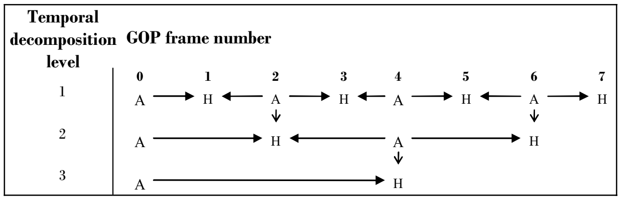

Figure 1: Temporal decomposition obtained by using UMCTF with no update step.

The temporal decomposition is followed by a spatial discrete wavelet transform; the resulting spatio-temporal subbands are subsequently encoded and entropy coded, as shown in the MCTF-based encoder architecture given in Figure 2.

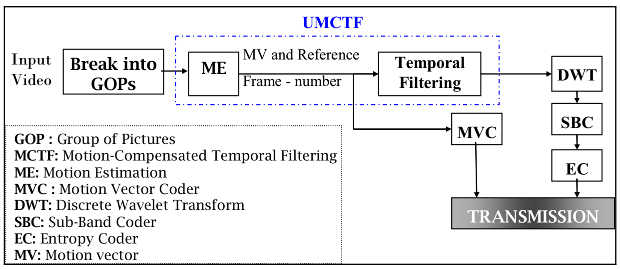

Figure 2: UMCTF-based video coder architecture.

3. MOTION VECTOR CODING

The video codec employed in this work uses the multi-hypothesis variable-size block-based motion estimation technique described in [6]. This motion estimator outputs the following motion information:

- Whether the macro-block is split or not. This will be referred to as the splitting information.
- For each separately predicted block:
  - The hypothesis, meaning the way the block is predicted (intra, by a single block in one reference frame, or by the average of multiple blocks lying in one of the reference frames).
  - Depending on the hypothesis, zero (intra coding), one, or more motion vectors and for each motion vector the associated reference frame index.

The motion vector coder (MVC) incorporated in our UMCTF-based architecture is the non-scalable version of the scalable MVC proposed in [7]. In this coder, the hypothesis, splitting and reference-frame information are coded using context-based adaptive arithmetic entropy coding. The motion vectors are encoded using median-based motion-vector prediction (cf. Figure 3) followed by lossless coding of the resulting prediction errors. The motion-vector prediction is performed by visiting each macro-block and each sub-block in raster-order and by using the motion vectors associated to the blocks (macro- or sub-blocks) that were already visited (causality). For more details about this MVC the reader is referred to [7].

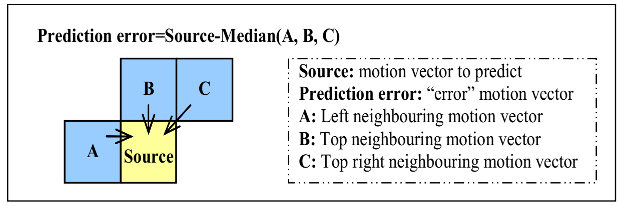

Figure 3: Motion vector prediction: principle

The drawback of this MVC method, and of prediction-based motion vector coding in general, is that when a motion vector is lost during transmission, all motion vectors that are predicted from that vector are automatically erroneous, leading to error propagation in the motion vector field and to a dramatic degradation of the decoded video quality.
To solve this problem, two robust motion-vector coding mechanisms are proposed and analyzed in the following.

3.1. Robust MVC-Mechanism 1

The first coding mechanism (which will be referred to as MVC_1), consists in reducing the prediction domain by dividing the frames into slices (cf. Figure 4), and imposing that the motion vector prediction can only use motion vectors from neighboring blocks (macro-blocks or sub-blocks) residing in the same slice as the block containing the predicted motion vector. This type of data partitioning technique is well-known in standards like H.263+, MPEG-1, -4 [8].

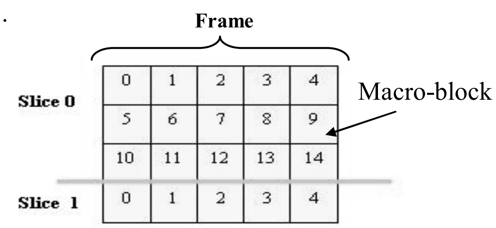

Figure 4: Robust MVC-Mechanism 1 (MVC_1).

3.2. Robust MVC-Mechanism 2

To further increase the robustness, we propose a new mechanism (referred to as MVC_2) that reduces the prediction domain even further. The frames are first divided into slices as in MVC_1 and next the rows and columns of the frames are also grouped by even and odd parity. This parity-based slicing mechanism results in a frame that is 'split' in four sliced sub-frames, as illustrated in Figure 5. The prediction of the motion vectors of each macro-block is then performed as in MVC_1, but on the sub-frames instead of on the whole frame.
The advantage of this second mechanism is that when a packet is lost, even fewer packets in the corresponding slice will be affected compared with MVC_1. The disadvantage is that motion vectors lying in neighboring macro-blocks are not used in the prediction, which can generate a loss in the motion vector coding performance.

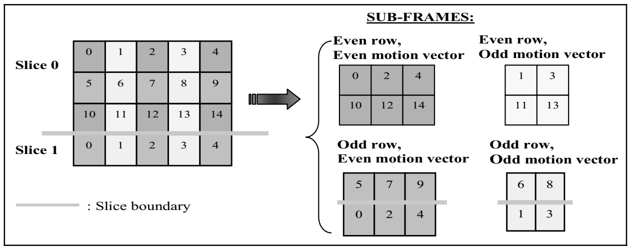

Figure 5: Robust MVC-Mechanism 2 (MVC_2).

	Original MVC - No Slicing (bytes per frame)	MVC_1 (bytes per frame)	Performance Penalty MVC_1 vs MVC (%)	MVC_2 (bytes per frame)	Performance Penalty MVC_2 vs MVC (%)
Container:	57.3	57.1	-0.35	58	1.2
Canoa:	1362	1368	0.44	1400	2.8
Football:	1140	1144	0.35	1174	3.0

Table 1: Performance penalty incurred when using MVC_1 and MVC_2, for frame-slicing into four consecutive rows. The average number of bytes per frame spent on motion vector information is reported.

4. ERROR CONCEALMENT

The proposed error concealment techniques are designed to recover the damaged motion vectors by using neighboring (correctly received) motion information. In the following we develop three types of error concealment techniques, operating in the spatial, temporal and spatio-temporal domains respectively.

4.1. Spatial concealment

The two developed spatial concealment techniques estimate a lost motion vector by using the median of the available surrounding motion vectors referring to the same frame as the vector to be concealed.
Our first spatial concealment method (SC_1) recovers the lost motion vectors from the median of the available motion vectors associated to the left, top and top-right macro-blocks/sub-blocks.
Our second spatial concealment method (SC_2) takes all eight surrounding macro-blocks/sub-blocks into account. This method is specifically designed for the coding mechanism MVC_2, as this is the only mechanism where all the motion vectors associated to the eight surrounding macro-blocks/sub-blocks can possibly be correctly received and decoded.

4.2. Temporal concealment

Temporal concealment (TC) of a lost motion vector involves the use of the motion vectors of the spatially corresponding macro-block in the previous frame. It was chosen to limit the temporal concealment only to the first temporal decomposition level. This is because the dissimilarity between the motion vector fields associated with two spatially corresponding macro-blocks increases with the temporal decomposition level.

4.3. Spatio-temporal concealment

Spatio-temporal concealment (STC) combines spatial and temporal information to estimate a lost motion vector. Following the same reasoning as before, the temporal information is only used when concealing frames of temporal level 1.

5. EXPERIMENTAL RESULTS

In our experiments the UMCTF uses uni-directional / bi-directional ME and does not employ any update step. Four temporal and four spatial levels are generated. Motion estimation is performed with 1/4-pel accuracy and the search range is chosen independently per temporal decomposition level (level 1: 16, level 2 1: 24, level 3: 32, level 4: 48). The macro-block size is set to 16 by 16 pixels and the macro-block splitting is restricted to one level.
First, the impact on the compression performance of using the proposed MVCs is investigated for three test CIF video sequences: 'Canoa' (220 frames), 'Container' (300 frames), and 'Football' (260 frames). One notices from the results of Table 1 that slicing the frames into four consecutive rows implies a negligible increase in motion vector information rate compared to the original MVC. Even in the extreme case where each slice consists of one row, the performance penalty is only 0.7%, 0.1%, 3% for MVC_1 and 1.6%, 3.7%, 3.6% for MVC_2 for the 'Container', 'Canoa' and 'Football' sequences respectively. However, in comparison to the original MVC, the proposed error-resilient MVCs increase the robustness against transmission errors significantly. Indeed, since no slicing is performed in the original MVC [7], even a single error generates the de-synchronization of the arithmetic entropy decoder, leading to both an erroneous decoding of the prediction-error vectors and to a recursive erroneous prediction of the motion vectors. In order to confine these error propagation phenomena within a slice, synchronization words are inserted at the beginning of each slice.
The error robustness of the proposed MVCs and the performance of the different concealment techniques are assessed in the following experiments. Video decoding at two different bit-rates is considered: 1024 and 2048 kbps. The frames are divided in groups of four consecutive rows. A packet size of one macro-block per packet is chosen. One simulates the transmission errors by randomly corrupting the motion vectors for different packet loss rates (PLRs): 0%, 0.1%, 1%, 2% and 5%. The quality of the decoded sequences is expressed as the average PSNR calculated over the whole sequence. Additionally, to provide a reliable comparison of the various techniques, one ensures that the corrupted motion vectors are always located at the same spatial positions for each tested technique. Each simulation is performed ten times, for every given bit-rate and PLR. Finally, the average PSNR over these rounds is computed. One notices from the resulting PSNR figures in Table 2 that our most robust coding method (MVC_2) performs significantly better (up to 3 dB) than MVC_1. Therefore, the error concealment

techniques are only tested when the motion vectors are encoded by MVC_2.

Out of all the error concealment approaches it is found that SC_2 provides the best results. Temporally concealing the lost motion vectors provides only for the 'Canoa'-sequence a very small improvement of the video quality compared with no concealment at all (MVC_2). Finally, the results show that spatio-temporal concealment (STC) does not result in an improvement of the video quality when compared with the technique employing spatial concealment alone (SC_2). We note that these techniques add a negligible overhead to the decoding process. Figure 6 provides an example of the visual improvement obtained when performing spatial concealment (SC_2) compared with no concealment at all (MVC_2).

1024 kbps						
Sequence Canoa	MVC_1	MVC_2	SC_1	SC_2	TC	STC
PLR 0%	28.76	28.73	28.73	28.73	28.73	28.73
PLR 0.1%	27.76	28.34	28.49	28.53	28.36	28.53
PLR 1%	23.37	26.01	26.82	27.02	26.10	27.03
PLR 2%	21.41	24.17	25.37	25.70	24.28	25.71
PLR 5%	19.77	21.87	23.05	23.44	21.91	23.45
Container						
PLR 0%	39.69	39.69	39.69	39.69	39.69	39.69
PLR 0.1%	39.53	39.61	39.66	39.67	39.61	39.67
PLR 1%	37.93	38.64	39.32	39.47	38.63	39.47
PLR 2%	36.48	37.69	38.96	39.21	37.72	39.21
PLR 5%	33.63	35.41	37.82	38.48	35.39	38.49
Football						
PLR 0%	31.32	31.29	31.29	31.29	31.29	31.29
PLR 0.1%	30.09	30.78	30.92	30.97	30.77	30.97
PLR 1%	25.15	27.94	28.76	29.02	27.93	29.02
PLR 2%	22.87	25.84	26.87	27.22	25.82	27.22
PLR 5%	20.85	23.23	24.27	24.72	23.20	24.71
2048 kbps						
Sequence Canoa	MVC_1	MVC_2	SC_1	SC_2	TC	STC
PLR 0%	32.10	32.07	32.07	32.07	32.07	32.07
PLR 0.1%	30.43	31.33	31.60	31.67	31.36	31.67
PLR 1%	24.19	27.54	28.69	28.98	27.67	28.99
PLR 2%	21.94	25.14	26.64	27.08	25.28	27.09
PLR 5%	20.15	22.45	23.79	24.23	22.57	24.25
Container						
PLR 0%	42.49	42.49	42.49	42.49	42.49	42.49
PLR 0.1%	42.25	42.37	42.44	42.45	42.37	42.45
PLR 1%	39.95	40.87	41.88	42.11	40.85	42.11
PLR 2%	37.97	39.48	41.28	41.68	39.52	41.68
PLR 5%	34.44	36.51	39.57	40.52	36.47	40.53
Football						
PLR 0%	34.76	34.74	34.74	34.74	34.74	34.74
PLR 0.1%	32.81	33.81	34.05	34.13	33.81	34.13
PLR 1%	26.01	29.39	30.49	30.86	29.38	30.85
PLR 2%	23.41	26.74	27.98	28.42	26.73	28.41
PLR 5%	21.24	23.79	24.93	25.43	23.77	25.42

Table 2: Performance comparison of the different robust motion vector coding and error concealment techniques for PLRs up to 5%. The average PSNR (dB) values are reported.

6. CONCLUSIONS

Our proposed parity-based slicing mechanism provides an improved robustness against transmission errors compared with the slicing technique that is typically used in current video-standards (H.263+, MPEG-4). Moreover, it is shown that the proposed robust motion vector codec combined with spatial concealment that considers all surrounding motion vectors and takes the reference frame information into account yields significantly better video quality when motion vector losses occur. It is also observed that adding temporal information to perform the concealment does not provide any gain. Finally, it is shown that our proposed spatial error-concealment mechanism leads to performance gains of up to 6 dB in comparison to a classical slicing-based approach employing no error concealment.

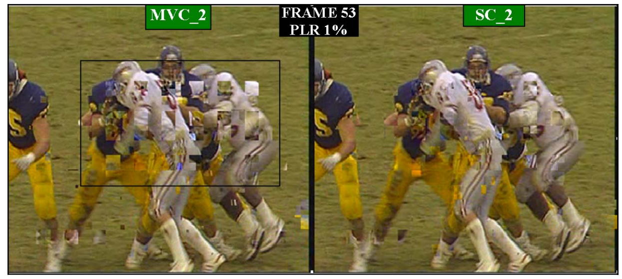

Figure 6: Visual improvement obtained by performing spatial concealment. Left: no concealment, wherein lost motion vectors are set to zero (MVC_2). Right: lost motion vectors are concealed spatially (SC_2).

7. ACKNOWLEDGMENTS

This work was supported by the DWTC (IAP Phase V– "Mobile") and by the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT-Vlaanderen, PhD bursaries M. Stoufs and J. Barbarien). P. Schelkens and A. Munteanu have post-doctoral fellowships with the Fund for Scientific Research – Flanders (FWO), Egmontstraat 5, B-1000 Brussels, Belgium.

8. REFERENCES

[1] J.-R. Ohm, "Registered Responses to the CfP on Scalable Video Coding," MPEG, Munich, Germany ISO/IEC JTC1/SC29/WG11 (MPEG), M10569, March 15-19, 2004.

[2] D. Turaga and M. van der Schaar, "Wavelet coding for video streaming using new unconstrained motion compensated temporal filtering," Proceedings of Proc. Int. Workshop on Digital Communications: Advanced Methods for Multimedia Signal Processing, Capri, IT, pp. 41-48, September 2002.

[3] D. S. Turaga, M. van der Schaar, I. Andreopoulos, A. Munteanu, and P. Schelkens, "Unconstrained Motion-compensated Temporal Filtering (UMCTF) for Efficient and Flexible Interframe Wavelet Video Coding," to appear.

[4] J.-R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 559-571, September 1994.

[5] S.-J. Choi and J. W. Woods, "Motion-compensated 3-D subband coding of video," *IEEE Transactions on Image Processing*, vol. 8, no. 2, pp. 155-167, February 1999.

[6] I. Andreopoulos, A. Munteanu, J. Barbarien, M. van der Schaar, J. Cornelis, and P. Schelkens, "In-band motion compensated temporal filtering," *Signal Processing: Image Communication*, vol. 19, no. 7, pp. 653-673, 2004, August.

[7] J. Barbarien, A. Munteanu, F. Verdicchio, Y. Andreopoulos, J. Cornelis, and P. Schelkens, " Scalable Motion Vector Coding," *IEE Electronics Letters*, vol. 40, no. 15, pp. 932-934, July 2004.

[8] S. Wenger and G. Knorr, "Error Resilience Support in H.263+," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 7, pp. 867-877, November.