DECODER-SIDE MOTION VECTOR DERIVATION FOR HYBRID VIDEO INTER CODING

Steffen Kamp and Mathias Wien

Institut für Nachrichtentechnik, RWTH Aachen University, Germany
Email: {kamp,wien}@ient.rwth-aachen.de

ABSTRACT

The ongoing increase of computing performance facilitates a higher algorithmical complexity in video coding systems. The decoder may be able to estimate or derive prediction parameters based on the previously decoded signal. In this paper we present an extension to H.264/AVC where the explicit coding of motion parameters is adaptively replaced by a template matching algorithm that is performed identically at the encoder and decoder. The decision between explicit coding and derivation of motion parameters is done by the rate-distortion optimised mode decision and coded into the bitstream. Compared to previous work, the provided scheme has been extended to bidirectional prediction (B pictures). Simulation results show an improved coding efficiency over a wide range of test sequences, especially for higher spatial resolutions.

***Keywords*—** Video coding, Inter frame prediction, Template matching, Motion compensation, H.264/AVC

1. INTRODUCTION

Hybrid video coding schemes typically exploit spatial and temporal signal correlation in the source video material for bitrate reduction. Natural video sequences often expose low or homogeneous motion due to e. g. static backgrounds, consistent movement of objects or camera pans and zooms. This temporal consistency allows for high compression ratios due to temporal prediction. Motion vector fields are a common method for modelling similarities between temporally consecutive video frames. Many video coding systems such as [1, 2] take this approach by coding individual images on a regular grid of blocks, possibly subdivided into smaller, rectangular partitions. For each partition a motion vector may be present in the bitstream which associates the partition with a spatially displaced region in a previously coded frame (reference frame). The motion compensation process obtains the prediction signal by copying the associated region from the reference frame to the current partition. The motion vectors are determined by the video encoder and are written to the bitstream using some form of entropy coding.

Several methods have been proposed for reducing the amount of bits necessary to represent motion vectors in the coded bitstream. While schemes such as motion vector competition [3] aim at reducing the bits required to explicitly code motion vectors by selecting between different motion vector predictors, others completely avoid coding of vectors and instead derive the motion at the decoder side based on already decoded image samples [4, 5]. While previous works have shown coding performance gains in sequences using unidirectional prediction, modern video coding schemes use bidirectional prediction for improved coding efficiency. In this paper we are applying and adapting the idea of decoder side motion vector derivation to bidirectional prediction in the context of H.264/AVC B pictures. This allows the evaluation of this scheme using state of the art coding conditions.

The rest of this paper is organised as follows. The basic decoder-side motion vector derivation scheme and application to H.264/AVC B pictures are discussed in Section 2. Simulation results compared to JM 15.1 are provided in Section 3 followed by conclusions in Section 4.

2. DECODER-SIDE MOTION VECTOR DERIVATION

Instead of explicitly coding a motion vector into the bitstream, decoder-side motion vector derivation (DMVD) uses already reconstructed video samples for estimating suitable vectors. We define the region for which a prediction signal is required to be the *target* of the DMVD algorithm (see Figure 1). Potential vectors for a target are found using a template matching algorithm [6], defining the *template* to be an L-shaped region of width M in the partially reconstructed image adjacent to the target. The template is then matched with identically shaped and spatially displaced candidate regions in the available reference pictures using the sum of absolute differences (SAD) cost criterion. The derived vector is characterised by the spatial displacement of the candidate and the reference picture index. A prediction signal can be obtained by motion compensation using either the vector of the candidate with minimum cost or by averaging the prediction signals of the K candidates with lowest costs [5]. The multi-hypothesis prediction of K candidates does not require additional coding bits, given that K is fixed for a particular sequence. The DMVD target may be smaller than the actual macroblock partition, in this case DMVD is repeatedly performed until the prediction signal for the whole partition is obtained.

It is clear that the derived vectors will not always provide a suitable prediction signal. Therefore, a locally adaptive approach is utilised: During the encoding process the rate-distortion optimised mode decision selects between using DMVD and explicit motion vector coding for each partition. A flag representing the choice of the encoder is then coded into the bitstream for each partition.

2.1. Application to H.264/AVC B Pictures

In this paper we have applied the DMVD scheme to B pictures within the H.264/AVC framework. B pictures in H.264/AVC use two temporal prediction lists. Each list contains a set of reference pictures available for prediction and a reference picture may also be contained in both lists. Each macroblock partition may either be predicted using only one reference picture of list 0 or 1 (unidirectional prediction), or using two reference pictures, one from each list (bidirectional prediction). Considering 16×16, 16×8, and 8×16 partitions only (without skip and direct modes), a macroblock can be coded in 21 different configurations. For our experiments we allowed DMVD in all of these 21 macroblock types, with the prediction lists used for template matching being restricted by the prediction direction of the partition as signalled by the macroblock type (unidirectional list 0, unidirectional list 1 or bidirectional). For the 16×16 types one bit is added to the H.264/AVC macroblock layer syntax, specifying whether an explicit motion vector for the macroblock is present in the bitstream or no motion vector is coded and DMVD is used for prediction. For the 16×8 and 8×16 types, two bits are added to the macroblock layer syntax, specifying for each of the two partitions whether regular motion vector coding or DMVD is used.

For P pictures it was shown in [5] that multi-hypothesis prediction with DMVD provides gains over single-hypothesis prediction. We have examined two configurations with different settings for the number of hypotheses in B pictures as shown in Table 1.

	Configuration A		Configuration B	
	Hypotheses		Hypotheses	
Mode	List 0	List 1	List 0	List 1
Pred_L0	2	0	2	0
Pred_L1	0	2	0	2
BiPred	1	1	2	2

Table 1. Number of hypotheses used for DMVD prediction in the three possible uni-/bidirectional prediction modes.

2.2. Template Matching Search

The template matching search algorithm has to meet several criteria. In contrast to encoder motion estimation algorithms which can typically vary between implementations, the DMVD search algorithm must be fixed and identical in different decoder implementations in order to obtain identical decoded sequences. Also, the encoder must use the same algorithm for a reliable mode decision and synchronous operation with the decoder. Another important aspect is the computational complexity of the algorithm, as it needs to be performed during the decoding process. This is in contrast to conventional video coding systems, where the computational burden of motion estimation lies on the encoder only. However, given the ongoing increase in computing performance per unit cost, it becomes feasible to increase the decoding complexity in video coding systems. For our experiments we have adapted the predictive search algorithm from [7] to DMVD with bidirectional prediction.

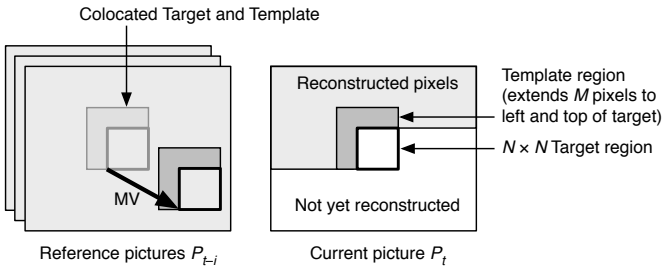


Fig. 1. Template matching.

The basic idea of the predictive search algorithm is to assemble a small set of vector candidates that have a high probability of referencing a suitable prediction signal. With the assumption that motion correlation in neighbouring macroblocks is high due to homogeneously moving backgrounds or objects whose area covers multiple macroblocks, motion vectors of partitions adjacent to the current DMVD target are used as candidate vectors. Obviously, candidate vectors can only be taken from the causal neighbourhood, i. e. partitions that are already decoded. In [7] the candidate set was composed of the vectors that are also used for calculating the motion vector predictor (MVP), namely the vectors from the adjacent left and upper-right partitions (if upper-right is not available, upper-left is used instead). Each candidate vector is initially associated with a specific reference picture. In order to adjust the candidate to other possible reference pictures, the vector $\mathbf{v}_{\mathrm{org}}$ is linearly scaled based on the temporal frame index of the current picture t_{cur} and the reference pictures (original t_{org} and destination t_{dst}):

$$\mathbf{v}_{\mathrm{dst}} = \frac{t_{\mathrm{dst}} - t_{\mathrm{cur}}}{t_{\mathrm{org}} - t_{\mathrm{cur}}}\mathbf{v}_{\mathrm{org}}. \qquad (1)$$

The candidate scaling is independent of the prediction list of $\mathbf{v}_{\mathrm{org}}$. If e. g. a neighbouring partition uses bidirectional prediction, both vectors (list 0 and list 1) of that partition are used as candidates for both prediction directions of the current partition. We have compared this approach to using vectors only as candidates for the same prediction list (i. e. list 0 vectors as candidates for list 0 prediction only, likewise for list 1 vectors) and found

Sequence	Res.	BD-Bitrate [%] Config A (P and B) Candidates for both lists
Flowervase	WVGA	-0.44
Mobisode2	WVGA	-1.47
Keiba	WVGA	-1.77
Kimono	1080p	-2.18
ParkScene	1080p	-0.85
Tennis	1080p	-2.80
average		-1.59

Table 2. Average bitrate difference (in %) of DMVD with candidates used for both prediction lists relative to DMVD with candidates used for the same list only.

Sequence	Res.	BD-Bitrate [%]			
		Configuration A			Config B
		P only	B only	P and B	P and B
Flowervase	WVGA	-1.73	-3.51	-5.14	-4.75
Mobisode	WVGA	-1.17	-4.40	-5.53	-4.62
Keiba	WVGA	0.04	-2.96	-3.15	-2.50
Kimono	1080p	-2.09	-8.26	-10.30	-9.08
ParkScene	1080p	-1.11	-4.68	-5.89	-5.21
Tennis	1080p	-0.73	-5.24	-6.19	-4.99
average		-1.13	-4.84	-6.03	-5.19

Table 4. Average bitrate difference (in %) of DMVD relative to JM 15.1.

that using vectors as candidates for both lists provides a higher coding efficiency (see Table 2 for results, coding conditions are stated in Table 3).

The best one or two candidate vectors per list (depending on the configuration, see Table 1) are additionally refined by performing template matching at the eight surrounding half-pel positions, followed by the eight quarter-pel positions around the best half-pel position.

3. SIMULATION RESULTS

We have implemented the described algorithm into the H.264/AVC reference software JM 15.1. Simulations have been performed using the coding conditions used for the anchor bitstreams of the MPEG Call for Evidence on High-Performance Video Coding (HVC) [8, 9] which are summarised in Table 3.

Simulation results for configurations A and B are summarised in Table 4, exemplary rate-distortion (RD) curves for the sequences *Kimono* and *Mobisode* are given in Figure 2. We have used the BD-Bitrate measurement [10] which calculates an average of the bitrate differences between two RD curves.

For configuration A individual results are provided for DMVD usage in P pictures only, B pictures only, and both P and B pictures. Evidently, with a coding structure using hierarchical B pictures, if DMVD is only applied in P pictures a limited bitrate reduction of 1.13 % on average is observed, while DMVD in B pictures saves 4.84 % bitrate. With DMVD in all inter coded pictures, the gains add up to 6.03 % on average with larger gains typically observed at higher spatial resolutions. The average bitrate savings observed for configuration B are slightly lower (5.19 %).

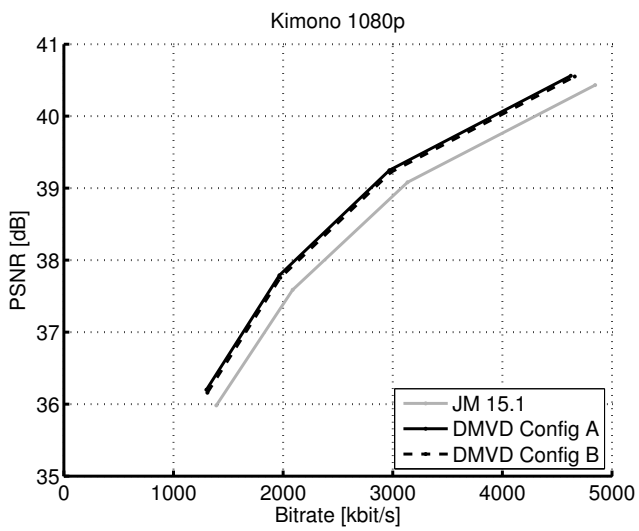
An analysis of the bitrate distribution within P and B pictures for the *Kimono* sequence is given in Figure 3. Compared to P pictures, the smaller temporal distance between B pictures and their reference pictures yields a more regular (or smoother) motion vector field and also a more accurate temporal prediction. This results in motion information accounting for a higher relative fraction (but smaller absolute amount) of the bits in B pictures compared to P pictures. Accordingly, relative bitrate savings due to DMVD are typically higher in B pictures. Another effect of DMVD is the slight increase of coded mode information which can be attributed to DMVD blocks partly replacing Skip mode blocks. While blocks coded in Skip mode require very few bits due to the motion vector not being coded but calculated as a component-wise median of neighbouring vectors, the distortion in Skip blocks may become relatively high as no residual signal is coded. In these cases, DMVD can be a middle course between Skip and regular motion vector coding, giving less distortion at low rates due to the signal adaptive derivation of motion vectors.

Sequences	Flowervase, Keiba, Mobisode (WVGA 30 Hz) Kimono, ParkScene, Tennis (1080p24)
Prediction structure	2-level hierarchical B: IbBbP One I picture each second
Profile	H.264/AVC High Profile (CABAC)
Quantisation	QPI: 25, 29, 33, 37 (WVGA) QPI: 25, 28, 31, 34 (1080p) QPP = QPI + 1, QPB = QPI + 2
Reference pictures	4 (P pictures), 2 per list (B pictures)
DMVD Template size	4 pixels
DMVD Target size	16×16 pixels (for 16×16 MB types) 8×8 pixels (for 16×8 and 8×16 MB types) 4×4 pixels (for 8×8 MB type, P pictures only)

Table 3. Simulation conditions.

4. CONCLUSION

In this paper we discussed the application of a decoder-side motion vector derivation scheme to video coding with bidirectional prediction. The algorithm has been implemented as an extension to H.264/AVC B an P pictures into the JM software. While the decoder-side motion vector derivation increases the computational decoding complexity, the reduction of motion information bits results in average bitrate savings of 6 % for a set of WVGA and 1080p test sequences.



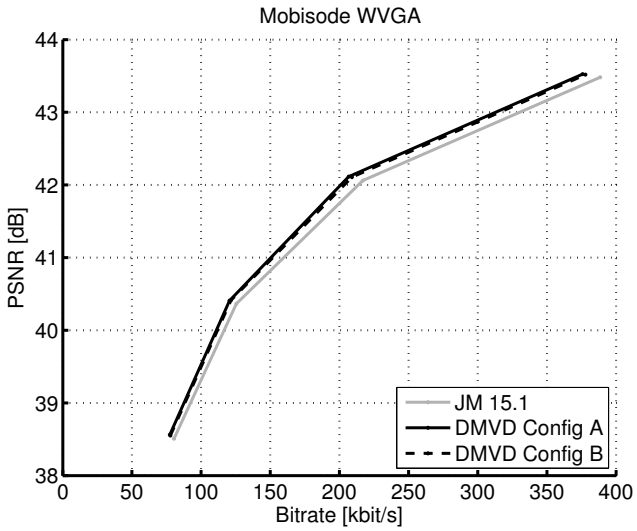


Fig. 2. Rate-distortion performance (luma component) for the sequences *Kimono* and *Mobisode*.

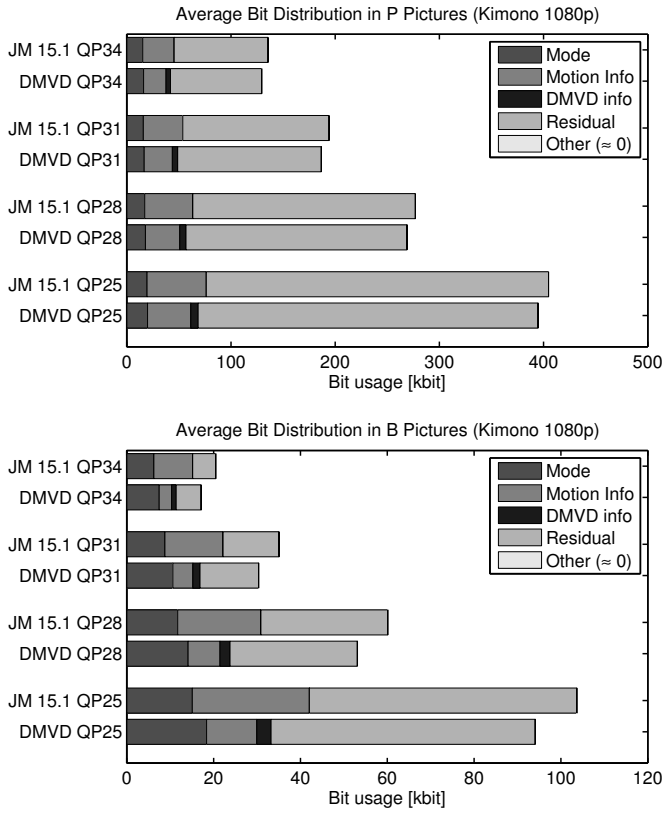


Fig. 3. Average bit distribution in P pictures (top) and B pictures (bottom) of *Kimono* for JM 15.1 and DMVD configuration A (configuration B is qualitatively similar). Note that when comparing DMVD and JM the same QP will result a slightly different PSNR so the bar lengths only approximate bitrate differences at the same quality.

5. REFERENCES

[1] ISO/IEC 13818-2, "Information technology – Generic coding of moving pictures and associated audio information: Video," 1996.

[2] *ITU-T Recommendation H.264: Advanced video coding for generic audiovisual services*, Mar. 2005.

[3] Guillaume Laroche, Joel Jung, and Beatrice Pesquet-Popescu, "RD optimized coding for motion vector predictor selection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 12, pp. 1681–1691, Dec. 2008.

[4] Kazuo Sugimoto, Mitsuru Kobayashi, Yoshinori Suzuki, Sadaatsu Kato, and Choong Seng Boon, "Inter frame coding with template matching spatio-temporal prediction," in *Proc. IEEE Int. Conference on Image Processing ICIP '04*, Singapore, Oct. 2004, pp. 465–468.

[5] Steffen Kamp, Johannes Ballé, and Mathias Wien, "Multihypothesis prediction using decoder side motion vector derivation in inter frame video coding," in *Proc. SPIE Visual Communications and Image Processing VCIP '09*, San José, CA, USA, Jan. 2009.

[6] Li-Yi Wei and Marc Levoy, "Fast texture synthesis using tree-structured vector quantization," in *Proc. 27th Annual Conf. on Computer Graphics and Interactive Techiques SIGGRAPH '00*, New York, NY, USA, July 2000, pp. 479–488.

[7] Steffen Kamp, Benjamin Bross, and Mathias Wien, "Fast decoder side motion vector derivation for inter frame video coding," in *Proc. International Picture Coding Symposium PCS '09*, Chicago, IL, USA, May 2009.

[8] "Call for evidence on high-performance video coding (HVC)," Maui, USA, Apr. 2009, MPEG2009/N10553.

[9] Steffen Kamp and Mathias Wien, "AVC anchor streams for evaluation of high-performance video coding (HVC)," Maui, USA, Apr. 2009, MPEG2009/M16463.

[10] Gisle Bjøntegaard, "Calculation of average PSNR differences between RD curves," Doc. VCEG-M33, ITU-T SG16/Q6 VCEG, 13th Meeting, Austin, TX, USA, Apr. 2001.