FAST DECODER SIDE MOTION VECTOR DERIVATION FOR INTER FRAME VIDEO CODING

Steffen Kamp, Benjamin Bross, Mathias Wien

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

ABSTRACT

Decoder-side motion vector derivation (DMVD) using template matching has been shown to improve coding efficiency of H.264/AVC based video coding. Instead of explicitly coding motion vectors into the bitstream, the decoder performs motion estimation in order to derive the motion vector used for motion compensated prediction. In previous works, DMVD was performed using a full template matching search in a limited search range. In this paper, a candidate based fast search algorithm replaces the full search. While the complexity reduction especially for the decoder is quite significant, the coding efficiency remains comparable. While for the full search algorithm BD-Bitrate savings of 7.4% averaged over CIF and HD sequences according to the VCEG common conditions for IPPP high profile are observed, the proposed fast search achieves bitrate reductions of up to 7.5% on average. By further omitting sub-pel refinement, average savings observed for CIF and HD are still up to 7%.

***Index Terms*—** Video coding, H.264/AVC, inter frame prediction, motion compensation, template matching, fast algorithms

1. INTRODUCTION

Spatial and temporal correlation found in natural image sequences is commonly exploited for bitrate reduction in video coding. Inter frame prediction plays an especially important role by using motion compensated regions from previously coded pictures for signal prediction. Assuming that the content between temporally adjacent frames remains similar for many typical video sequences, temporal prediction is the main contributor to the coding efficiency in many video coding standards such as MPEG-4 [1] and H.264/AVC [2]. In most traditional video coding schemes, the encoder performs a motion estimation (ME) to determine translational displacements of regions and encodes the displacements as motion vectors into the bitstream (forward motion coding).

Recently, schemes using template matching (TM) [3] have been successfully applied to inter prediction, using pixel-by-pixel prediction [4], 8×8 block based prediction [5], multiple reference pictures [6] and multi-hypothesis prediction [7, 8]. These schemes allow for temporal prediction without the need to transmit actual motion vector data. Instead, motion vectors are derived using a template matching algorithm at the decoder side. While this increases the decoder complexity, significant bitrate reductions have been observed.

In this paper, we examine the applicability of fast motion estimation algorithms for complexity reduction of DMVD. Our proposed scheme employs a predictive search algorithm which uses only a very small number of candidate motion vectors taken from the spatial neighbourhood of the current DMVD partition. While this decreases the decoder complexity quite significantly, even compared to full search estimation with very limited search ranges, the compression efficiency of DMVD is maintained. With the candidate motion vectors being available in sub-pixel precision, further complexity reduction is possible by omitting sub-pixel refinement search at acceptable loss of coding efficiency.

2. DECODER-SIDE MOTION VECTOR DERIVATION

Decoder-side motion vector derivation (DMVD) has been shown to increase the compression efficiency in P slices of H.264/AVC. Instead of explicitly coding motion vector information into the bitstream, template matching (TM) is performed at the decoder to derive motion information from the already decoded signal. A target region of $N \times N$ pixels is predicted by performing a motion-estimation-like search. As the target region itself is not available at the decoder, an L-shaped region of size M located to the top-right of the target is used as reference signal for the search (see Figure 1). A motion vector is derived by minimising a cost function (e. g. sum of absolute differences, SAD) based on the difference of the reference signal values and the signal values of a displaced L-shaped region in the reference pictures. The derived motion vector is then used for motion compensated prediction of the

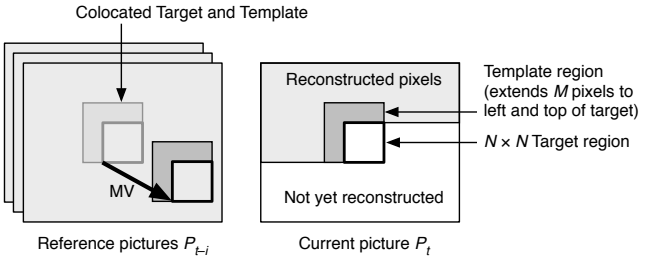

Fig. 1. For template matching already decoded pixel values of the current picture are used for a correspondence search in the available reference pictures. The spatio-temporal displacement of the best match is used as the derived motion vector.

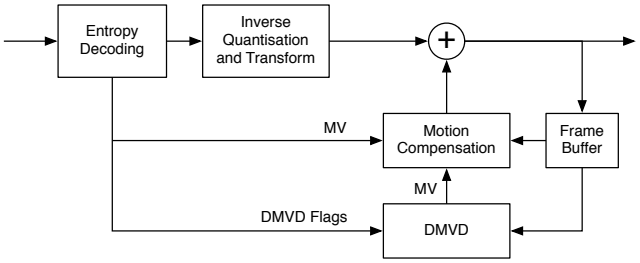

Fig. 2. Simplified block diagram of the DMVD decoder. Motion vectors are either coded into the bitstream directly or derived at the decoder depending on the DMVD flags in the macroblock header.

target area as in regular motion coding. Recently, DMVD has been extended to take advantage of multi-hypothesis prediction [8] for further improvements in coding performance. For multi-hypothesis prediction, the K best vectors found during DMVD are used for generating the prediction signal of the current block. The individual reference pixels are averaged using a pixel-wise arithmetic mean calculation of the K hypotheses. The parameter K is set on the sequence level, so all DMVD partitions use the same number of hypotheses.

As DMVD will not be able to find suitable motion vectors in all cases, the usage of DMVD for a specific macroblock (MB) is signalled using a flag in the MB header. The encoder can therefore perform a rate-distortion optimal decision between coding a MB using DMVD or explicitly coding the motion vector into the bitstream. As the DMVD algorithm is known to the encoder, encoder and decoder can run synchronously and no encoder-decoder drift is observed if the transmission is error free. A simplified block diagram of a decoder using DMVD is shown in Figure 2.

3. FULL SEARCH DMVD

Our previously presented DMVD schemes [6, 8] have been using full search template matching, where the template

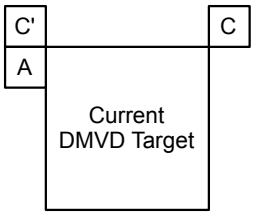

Fig. 3. Candidate motion vectors in the neighbourhood of the partition using DMVD. If motion data for partition C is not available (i. e. intra block or outside the coded image region), motion data from C' is used instead.

matching cost was calculated for all full-pel positions within a window of a fixed search range. The search window is centred on the motion vector predictor (MVP) for the respective reference picture index. Then, let K be the number of hypotheses to be used in DMVD prediction, the K best full-pel positions are subject to sub-pel refinement, searching the minimum cost of the eight surrounding half-pel positions and, subsequently, the eight surrounding quarter-pel positions.

Using the MVP as start for the search, relatively small search ranges are possible. While a search range of ± 4 has been used in [6], a search range of ± 1 already provides a significant portion of the observed gains, while reducing the number of search positions from 81 to 9 per reference picture. After full-pel search, the motion vector candidates are subject to a sub-pel refinement: First, the best of the 8 surrounding half-pel positions is determined followed by the best of the 8 quarter-pel positions. While all full pel candidates are initially unique, some candidates may be identical after sub-pel refinement. Therefore, sub-pel refinement is performed successively for the best full-pel candidates until K unique sub-pel exact vectors have been found.

4. CANDIDATE BASED PREDICTIVE SEARCH

In order to further reduce the number of search positions, a candidate based search is examined. Candidate based algorithms have been successfully applied to encoder side motion estimation or frame interpolation algorithms [9]. The principle of the candidate based search is for each reference picture index i to compose a set S_i of unique vectors for which the template matching costs are evaluated. Ideally, the number of vectors in S_i should be relatively small in order to significantly reduce the complexity of DMVD. With the assumption of a high spatial motion correlation in typical video sequences, we have chosen to derive candidates for the sets S_i from motion vectors of blocks adjacent to the currently estimated DMVD target block. Specifically, we only used the vectors from the neighbouring blocks A (left) and C (top-right if available, top-left otherwise) as used in motion vector predictor (MVP) calculation (see Figure 3) as original candi-

dates. We denote these candidates $c_A = (x_A, y_A, r_A)$ and c_C respectively, where x and y specify the spatial displacement and r being the associated reference picture index. We have tested two different methods for applying these candidates to the available reference pictures:

Independent of the Reference Picture
The two original candidates are used without modification for all tested reference indices. Given the original candidate vector c_A, the candidate $c_{A,i} \in S_i$ for reference picture i is obtained by setting $c_{A,i} := (x_A, y_A, i)$. Candidate $c_{C,i}$ is derived accordingly. In other words, for a given DMVD target, all S_i are composed of vectors with the same spatial displacement.

Reference Index Based Scaling (RIBS)
The two original candidates are scaled for other reference indices assuming a uniform motion. Given the original candidate vector c_A, we obtain the scaled candidate $c_{A,i} \in S_i$ for reference index i by setting $c_{A,i} := (\frac{i+1}{r_A+1}x_A, \frac{i+1}{r_A+1}y_A, i)$ and rounding the spatial components to quarter-pel accuracy. Candidate $c_{C,i}$ is derived accordingly. This scheme assumes that reference indices start at zero, a larger index corresponds to a larger temporal distance from the current picture and time differences between reference pictures are proportional to the difference between the respective reference indices. If these conditions can not be met, e. g. due to reference picture reordering, other means of assigning time to reference pictures have to be used such as an explicit coding of scaling factors.

Regarding sub-pel refinement, we have examined two strategies: (a) Performing a sub-pel refinement as in the full-search algorithm. (b) As the candidate vectors are taken from previous motion data, they are already available in quarter-pel accuracy, so a further sub-pel refinement may not be as important as for the full search, where the best vectors are initially found in full-pel precision. While this will further reduce the complexity, the coding efficiency is typically slightly reduced. Results for both schemes are shown in the next section.

It should be noted that the decoder needs to employ the same search algorithm as the encoder in order to ensure a correct rate-distortion optimal mode decision and drift-free operation.

5. SIMULATION RESULTS

The proposed DMVD algorithm has been implemented into the H.264/AVC reference software JM 13.2. Except for the search ranges and algorithms, DMVD settings are identical to [8] and shown in Table 1.

Simulations have been performed for CIF and HD sequences according to the VCEG common conditions for IPPP coding [10] using High Profile. BD-Rate results according to [11] and relative to JM 13.2 are shown in Figure 4. It can be observed that for full search the bitrate savings averaged over CIF and HD sequences are 7.4% for search range ± 4 and 6.7% for search range ± 1. Employing the candidate search without Reference Index Based Scaling (RIBS) reduces the bitrate savings to 6.3%, with a further decline to 5.3% when also omitting sub-pel refinement. Enabling RIBS for the candidate search raises bitrate savings to 7.5% with sub-pel refinement and 7% without sub-pel refinement.

Sequences	Foreman, Mobile, Tempete (CIF, 30 Hz), Paris (CIF, 15 Hz) BigShips, ShuttleStart, City, Crew, Night (720p, 60 Hz, 150 frames) RollingTomatoes (1080p, 60 Hz, 60 frames)
Prediction structure	IPPP…, High Profile
Entropy coding	CABAC
Quantisation parameter	QPI: 22, 27, 32, 37; QPP=QPI+1
Reference pictures	4
Hypotheses	2, fixed for all DMVD blocks
Template size M	4
Target size N	16 (16×16 type) 8 (16×8 and 8×16 types) 4 (8×8 type)
Search range	±1 or ±4 pixels for full search

Table 1. Simulation settings according to VCEG common conditions for coding efficiency experiments [10].

For the candidate based algorithm with only two candidates per reference picture and RIBS enabled, average bitrate savings are even slightly higher than using full search and search range ± 4. The reason for this behaviour is twofold. On the one hand, the candidate based method does not have a restricted search range and may capture larger displacements especially when using multiple reference pictures and RIBS, as can be observed for sequence City with relatively homogeneous global motion. On the other hand, as the lowest template cost does not guarantee to correspond to the optimal motion vector, the full search algorithm may be trapped in a local minimum. Using a preselection of MV candidates with a high probability of capturing the actual motion therefore stabilises the candidate based DMVD algorithm, yielding a more accurate prediction signal.

6. CONCLUSION

In this paper we have shown that it is feasible to reduce the complexity of DMVD by employing candidate based motion search techniques and omitting sub-pel motion refinement. By reducing the candidate set to motion vectors with a high probability of capturing the local motion and scaling the candidates based on the reference picture index, the gains previously reported for full search DMVD can be preserved while significantly reducing the number of required template cost calculations.

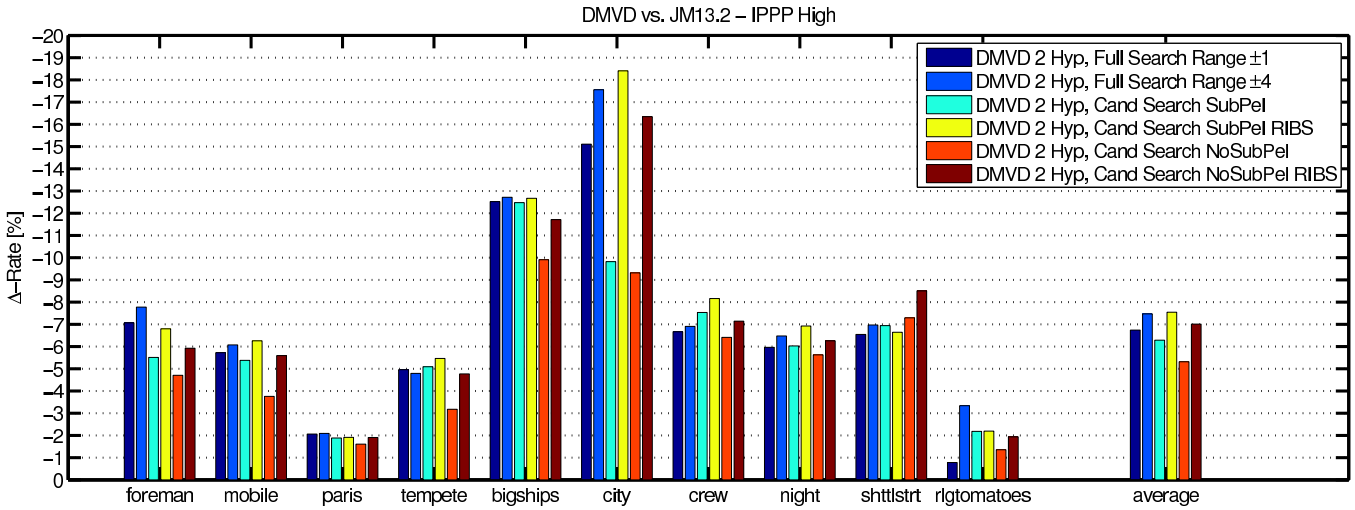

Fig. 4. Simulation results.

7. REFERENCES

[1] ISO/IEC 14496-2, "Information technology – Generic coding of audio-visual objects: Visual," 1998.

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

[3] 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.

[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] Yoshinori Suzuki, Choong Seng Boon, and Sadaatsu Kato, "Block-based reduced resolution inter frame coding with template matching prediction," in *Proc. IEEE Int. Conference on Image Processing ICIP '06*, Atlanta, GA, USA, Oct. 2006, pp. 1701–1704.

[6] Steffen Kamp, Michael Evertz, and Mathias Wien, "Decoder side motion vector derivation for inter frame video coding," in *Proc. IEEE Int. Conference on Image Processing ICIP '08*, San Diego, CA, USA, Oct. 2008, pp. 1120–1123.

[7] Yoshinori Suzuki, Choong Seng Boon, and Thiow Keng Tan, "Inter frame coding with template matching averaging," in *Proc. IEEE Int. Conference on Image Processing ICIP '07*, San Antonio, TX, USA, Sept. 2007, pp. III–409–III–412.

[8] 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.

[9] G. de Haan, P. W. A. C. Biezen, H. Huijgen, and O. Ojo, "True-motion estimation with 3-D recursive search block matching," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 3, no. 5, pp. 368–379, Oct. 1993.

[10] Thiow Keng Tan, Gary J. Sullivan, and Thomas Wedi, "Recommended simulation common conditions for coding efficiency experiments, revision 3," Doc. VCEG-AH010r3, ITU-T SG16/Q6 VCEG, 34th Meeting, Antalya, Turkey, Jan. 2008.

[11] 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.