DECODER SIDE MOTION VECTOR DERIVATION FOR INTER FRAME VIDEO CODING

Steffen Kamp, Michael Evertz, and Mathias Wien

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

ABSTRACT

In this paper, a decoder side motion vector derivation scheme for inter frame video coding is proposed. Using a template matching algorithm, motion information is derived at the decoder instead of explicitly coding the information into the bitstream. Based on Lagrangian rate-distortion optimisation, the encoder locally signals whether motion derivation or forward motion coding is used. While our method exploits multiple reference pictures for improved prediction performance and bitrate reduction, only a small template matching search range is required. Derived motion information is reused to improve the performance of predictive motion vector coding in subsequent blocks. An efficient conditional signalling scheme for motion derivation in Skip blocks is employed. The motion vector derivation method has been implemented as an extension to H.264/AVC. Simulation results show that a bitrate reduction of up to 10.4 % over H.264/AVC is achieved by the proposed scheme.

***Index Terms*—**Video coding, Template matching, Motion compensation, H.264/AVC

1. INTRODUCTION

In video coding, the temporal and spatial correlation found in natural image sequences is exploited for bitrate reduction. Inter frame coding typically uses motion compensated regions from already decoded pictures as prediction signal for the currently coded picture. In order to perform the motion compensation at the decoder side, video coding standards such as MPEG-4 [1] and H.264/AVC [2] specify the coding of motion vectors which describe the translational displacement of rectangular blocks (forward motion coding). Motion estimation and rate-distortion optimisation [3] is used at the encoder side to determine the coding parameters for a specific block which are then coded into the bitstream. For inter coded blocks, a significant amount of the bitrate is spent for the coding of motion information such as motion vector data and reference picture indices.

Template maching (TM) algorithms have be shown as an effective method for texture synthesis [4]. This facilitates the use of texture synthesis in the area of intra prediction of still image and video coding [5, 6]. Inter prediction using TM has been studied in [7, 8, 9]. [7] performs a pixel-by-pixel prediction from the current picture and reference pictures similar to texture synthesis. In [8], H.264/AVC is extended by a macroblock type and sub-macroblock type which use TM to predict blocks of size 8×8. This method has been further improved in [9] by forming the prediction as a weighted average of multiple references.

In this paper we propose to signal the decoder-side motion vector derivation (DMVD) for the existing P macroblock modes in H.264/AVC. Motion information is determined using a template matching algorithm at the encoder and decoder. Additional flags are coded for the different P macroblock types to signal the usage of DMVD. For P-Skip macroblocks, motion refinement using DMVD with an efficient conditional signalling method is presented which is especially suited for lower bitrates as no residual data is coded for Skip macroblocks. Motion information is derived such that it directly replaces H.264/AVC syntax elements. This allows the use of derived motion information for calculating motion vector predictors (MVP) for subsequent blocks. In contrast to [7, 8, 9] our scheme exploits multiple reference pictures for improved prediction performance. The reference picture index to be used is derived in addition to motion vector data by performing TM on all reference pictures available to the decoder. It is further shown, that despite the use of multiple reference pictures most performance gains can be achieved with a small search range, keeping the complexity increase for the decoder relatively low. While our scheme achieves notable gains by using only a single predictor, further gains at the cost of additional complexity could probably be obtained by weighted averaging of multiple predictors as in [9].

The rest of the paper is organised as follows. The H.264/AVC inter prediction and the template matching schemes are introduced in Section 2. Section 3 describes the motion vector derivation parameters and bitstream signalling. Simulation results are provided in Section 4 followed by a conclusion in Section 5.

2. BACKGROUND

2.1. Inter Prediction in H.264/AVC P-Slices

Video picture coding in H.264/AVC is based on a regular grid of macroblocks (MB) consisting of 16×16 luma samples and associated chroma samples each. In P slices each MB may either be coded using intra prediction or one of the available P inter prediction types, allowing a subdivision of MBs into smaller partitions. The possible partition sizes of P MBs are depicted in Figure 1. If a MB is coded using the 8×8 partitioning, each partition is coded using one of four available sub-partitionings. A single motion vector is coded for each (sub-)partition. The reference picture to use for motion compensated prediction is coded using a reference list index for each partition. However, sub-partitions inside a single 8×8 partition use the same reference picture index. In addition to the motion parameters, the quantised prediction error signal (residual) is coded for each MB.

Motion vectors are coded predictively. A motion vector predictor (MVP) is calculated from neighbouring partitions of the currently coded block. The MVP calculation depends on the reference picture to be used for the currently coded block. Then, the motion vector difference (MVD) between the MVP and the actual MV is coded into the bitstream. Under the assumption that moving objects in the video sequence are commonly larger than the macroblock size, neighbouring macroblocks are likely to have similar or even identical motion vectors. This contributes to the efficiency of motion coding in H.264/AVC.

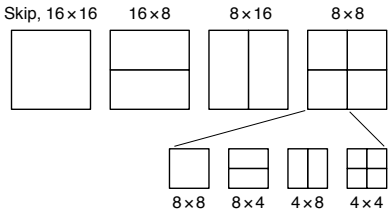

Fig. 1. P macroblock types (top) and sub-types (bottom).

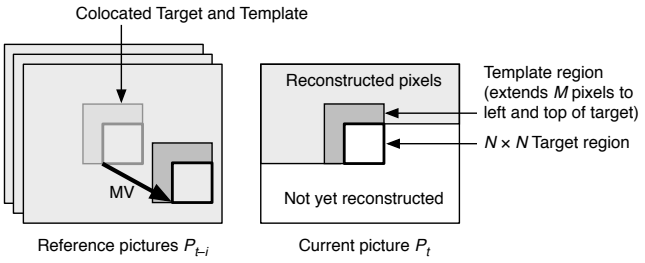

Fig. 2. Template matching.

A special predictive MB type is the Skip type, where no motion vector differences or reference picture indices are coded. Instead, the MVP and the first available reference picture are used for motion compensation of the 16×16 partition and no residual signal data is coded in this case. Therefore, signalling of Skip MBs is very inexpensive predestining this type for low bitrate video coding.

The (sub-)partitioning type to be used for a MB is decided by the encoder. Typically, the encoder performs motion estimation for several or all available predictive MB types. The final MB type and motion vectors are often selected by rate-distortion optimisation [3] using a Lagrangian cost criterion.

2.2. Template Matching

The proposed decoder side motion vector derivation scheme uses encoder and decoder side template matching (TM) to derive motion information. TM exploits correlation between the pixels from blocks adjacent to the prediction target block and those in already reconstructed reference pictures. The basic principle is shown in Figure 2: In order to derive motion information for a $N \times N$ target region in the current picture P_t, an inverse-L shaped template region is defined extending M pixels from the top and left of the target region. The template region only covers already reconstructed pixels of P_t. Then, the best displaced template region in the reference pictures P_{t-i} is determined by minimising the sum of absolute differences between the luma and chroma samples of the current and reference template regions. The spatial and temporal displacement of the best-matching template region are used as MV and reference picture index i for motion compensated prediction of the target area.

3. PROPOSED MOTION VECTOR DERIVATION

In our proposed scheme, the predictively coded Skip, 16×16, 16×8, 8×16 macroblock types and the 8×8 macroblock subtype are extended to optionally use TM for decoder side motion vector and reference picture derivation (DMVD). This section describes the signalling and the parameters used for each type. A block diagram of the decoder is shown in Figure 3.

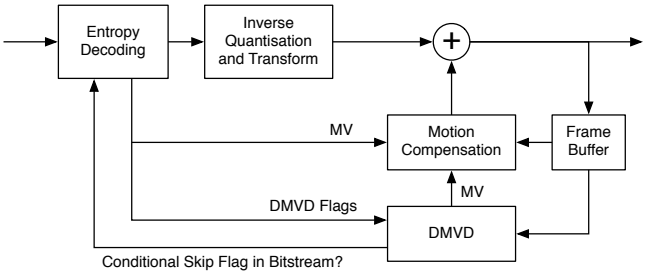

Fig. 3. Simplified block diagram of decoder with DMVD.

The template matching search is performed centred on the MVP for the target block using an exhaustive full-pel search with limited search range (± 1 pixels for Skip, ± 4 pixels for other types in this paper). Additionally, for the integration into H.264/AVC half-pel and quarter-pel refinement is performed around the best full-pel MV. Prediction is performed using sub-pixel accuracy using the H.264/AVC interpolation filter. The applicability of fast motion estimation schemes is subject to future research.

3.1. Blocks with Coded Residual

In H.264/AVC motion compensated prediction is performed for the 16×16, 16×8, 8×16 and 8×8 macroblock types. The resulting residual signal is then quantised and coded into the bitstream. We have extended these types to optionally use TM to derive motion information at the decoder side. As the MV derivation can be performed identically at the encoder and decoder side, the encoder only needs to signal that DMVD should be used to derive motion vectors and reference indices instead of explicitly coding the information into the bitstream.

For the 16×16 type one additional flag is present in the bitstream, signalling whether forward motion coding or DMVD is to be used for the macroblock. Only a single motion vector is derived in this case, i. e. the target size $N = 16$.

For 16×8 and 8×16, one additional flag is coded specifying whether DMVD is used for the macroblock. If so, a second flag is coded signalling which of the two partitions uses DMVD. $N = 8$ is used as target size.

In the 8×8 type, DMVD is only allowed for sub-blocks coded in the 8×8 sub-type. A single flag per 8×8 sub-type is coded for DMVD. The target size is $N = 4$ in this case.

Our simulation results have shown that the use of different target sizes for different macroblock types yields higher coding efficiency than using a fixed target size of $N = 4$ or $N = 8$ for all types. This can be attributed to the higher flexibility of the encoder in selecting target sizes suitable for the local signal characteristics. In regions with flat signal statistics and homogenous motion a target size of $N = 16$ with a corresponding larger template region leads to a more stable motion estimate compared to smaller partitions. For areas with irregular motion, smaller partitions provide a finer granularity for the motion compensation.

For each type the DMVD search is performed in all reference pictures available for the prediction of the current macroblock. Thus, the reference picture index can be derived in addition to the motion vectors, providing additional bitrate savings. Due to larger temporal distances, motion vector norms tend to increase when using higher reference picture indices. However, by centring the TM search win-

dow on the MVP for the respective reference picture, the search range can be limited.

3.2. Motion Refinement for Skip Blocks

The P-Skip type of H.264/AVC is especially suited for homogenous motion and low bitrates. For blocks coded using Skip no motion vector difference is coded into the bitstream. However, the limitation to the MVP makes the Skip type inappropriate for areas with slow and homogenous but non-translatory motion. We therefore propose a motion refinement for Skip types using DMVD with a very limited search range of ± 1 pixels. As the bitrate of Skip blocks is very low it is important to use an efficient signalling scheme for DMVD. We therefore used a conditional coding scheme: For each Skip macroblock, TM is performed and the candidate vector is compared to the MVP. If the vectors are identical, both unmodified Skip and DMVD would use the same vector, so no additional DMVD flag needs to be present in the bitstream. If the vectors are different a single flag specifies whether to use the derived vector or the MVP for motion compensation. No residual data is coded in both cases.

The conditional coding for the Skip type has some implications: As the presence of the DMVD flag depends on the TM result, TM must be performed during the bitstream parsing stage independent of whether DMVD is actually used, therefore leading to a increased decoder complexity. If reference pictures are lost or corrupted due to transmission errors, it is possible that the decoder side TM search yields a different vector than the encoder side. Under these circumstances the decoder might e. g. assume that a DMVD flag is present in the bitstream while it is not, making it almost impossible to decode the remaining MBs of the current slice. Therefore, the proposed Skip type signalling should only be used in applications with very low transmission error probabilities.

3.3. Reuse of Derived Motion for Motion Vector Prediction

Our proposed DMVD scheme has been designed such that the derived motion vectors and reference indices fit into the motion coding scheme of H.264/AVC and can transparently replace the corresponding syntax elements for MVD and reference picture index. I. e. motion vectors are derived at a minimal granularity of 4×4 pixels, reference frame indices at a minimal granularity of 8×8 blocks. This allows DMVD results to participate in the current motion vector prediction process without modifications, enhancing the MVP for subsequent partitions using either forward motion coding or DMVD.

4. SIMULATION RESULTS

To validate the performance, the proposed scheme has been implemented into the H.264/AVC reference software (JM12.3). The simulation conditions (unless noted otherwise) are summarised in Table 1. The DMVD flags as described in Sections 3.1 and 3.2 have been coded using context adaptive binary arithmetic coding (CABAC) [2] using separate CABAC contexts for each type of flag and initialised to a probability of 0.5. Coding efficiency comparisons are relative to the unmodified JM software using the Bjøntegaard average delta bitrate (BD-Rate) assessment method [10].

Figure 4 shows the BD-Rate reduction for different search ranges in the non-Skip types. It can be seen that the performance increases only minimally for higher ranges, justifying the use of a small search range in order to reduce the computational complexity of DMVD.

Sequences	Foreman, Mobile, Tempete (CIF, 30 Hz) Paris (CIF, 15 Hz) BigShips, ShuttleStart, City, Crew, Night (720p, 60 Hz, 150 frames)
Prediction structure	IPPP…
Entropy coding	CABAC
Quantisation parameter	QPI: 28, 32, 36, 40; QPP=QPI+1
Reference pictures	4
Template size M	4
Target size N	16 (Skip and 16×16 types) 8 (16×8 and 8×16 types) 4 (8×8 type)
Search range	± 1 pixel (Skip), ± 4 pixels (other types)

Table 1. Simulation conditions.

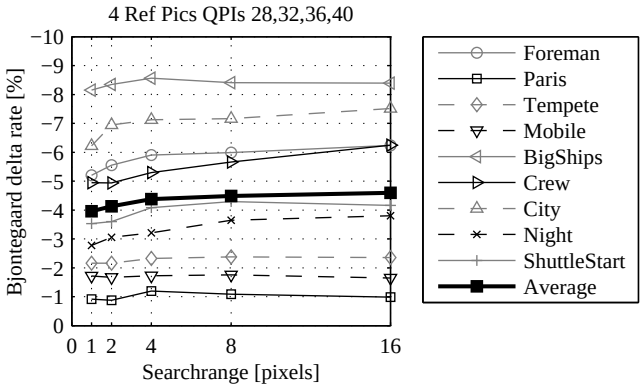

Fig. 4. BD-Rate results for different search ranges, search is centred around MVP.

The effectiveness of the DMVD motion vector reuse for MVP (Section 3.3) is shown in Figure 5 (left) averaged over all sequences. For 'Null' DMVD partitions were treated as intra coded blocks during MVP calculation, i. e. the DMVD partitions were treated as if no motion information was available. Because this breaks motion vector prediction chains and due to the DMVD signalling overhead, the coding performance actually decreases. By pretending the MVP has been used as MV in the DMVD partition ('MVP'), motion vector prediction chains are preserved and a bitrate reduction of 2.5 % can be observed. By using the actual DMVD-derived information ('DMV'), motion vector prediction can take advantage of the refined motion accuracy and the coding efficiency is improved to 4.5 % bitrate reduction on average.

Figure 5 (right) displays the impact of the number of reference pictures used for motion estimation for forward coding ('ME') and of the number of references used for 'DMVD'. For this figure, results are relative to an unmodified JM12.3 using only 1 reference picture. It can be observed that in this case DMVD with also only 1 reference picture already provides an average BD-Rate reduction of ~4.5 %. Allowing 4 reference pictures for forward motion coding without usage of DMVD significantly improves the coding efficiency. Enabling DMVD with a single reference picture saves further ~3 percentage points, using all 4 reference pictures saves ~4.5 percentage points in bitrate on average. Effectively, the bitrate savings achieved by DMVD are constantly ~4.5 % when the same number of reference pictures is used for both forward motion coding and DMVD.

The overall results of the proposed DMVD scheme are depicted in Figure 6. Results are shown for the DMVD with Skip type only, the non-Skip modes and all DMVD modes enabled. It can be ob-

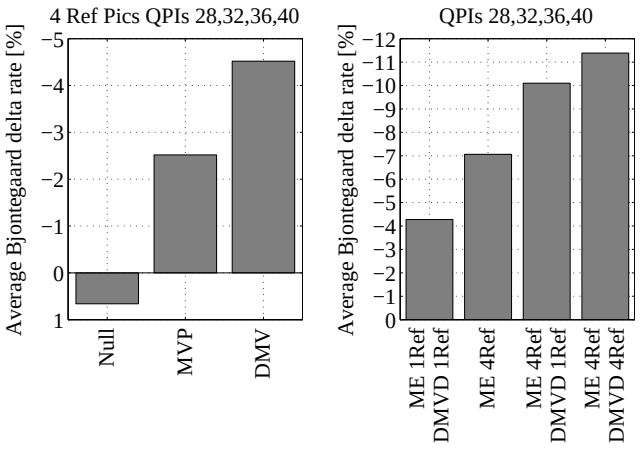

Fig. 5. Left: Derived MV reuse for MVP. Right: Effect of reference picture count (relative to JM using only 1 reference picture).

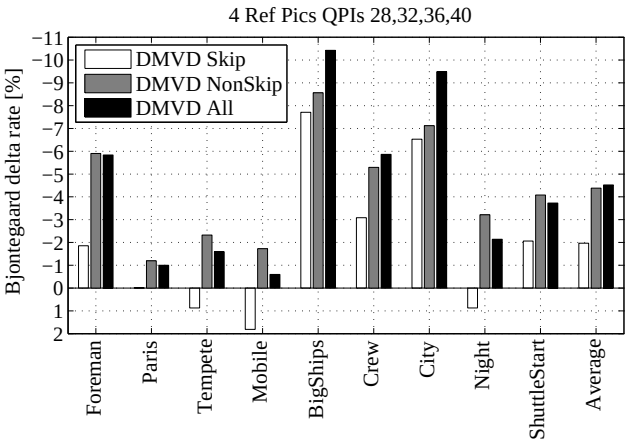

Fig. 6. Summarised BD-Rate results compared to JM12.3.

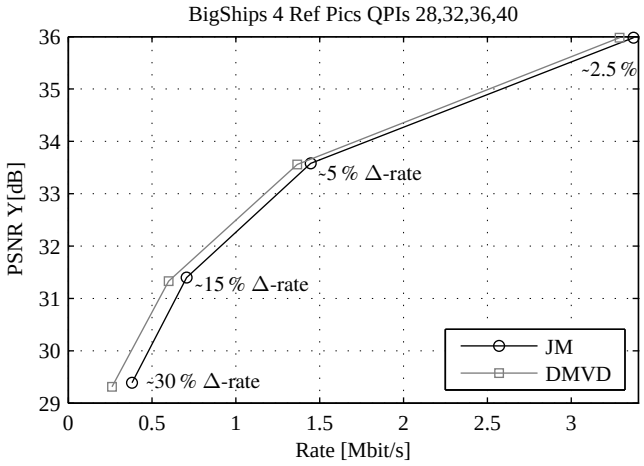

Fig. 7. Comparing JM12.3 with and without DMVD for sequence BigShips. Rate differences take PSNR differences into account.

served that the average bitrate can be reduced by up to 10.4 % for BigShips with a maximum bitrate saving of ~30 % for the lowest rate point (see Figure 7). For some sequences DMVD Skip leads to a slight bitrate increase when used without the other DMVD types and a penalty when using all DMVD types due to the signalling overhead. Future research is aimed at optimising Skip type signalling to reduce this effect. However, DMVD Skip accounts for a significant performance gain on its own for other sequences (BigShips, City). In general, better results can be obtained for higher resolutios. Averaged over all sequences and with all DMVD modes enabled, a BD-Rate reduction by 4.5 % is achieved by the proposed scheme.

5. CONCLUSION

This paper proposed a decoder side motion vector derivation scheme for inter frame video coding. A specific configuration suitable for the extension of H.264/AVC has been presented which uses multiple reference pictures, a small search range, improved motion vector prediction, and efficient conditional signalling for the Skip type. Simulation results show objective bitrate reductions of up to 10.4 %.

6. 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] Gary J. Sullivan and Thomas Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Processing Magazine*, pp. 74–90, Nov. 1998.

[4] 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.

[5] Thiow Keng Tan, Choong Seng Boon, and Yoshinori Suzuki, "Intra prediction by template matching," in *Proc. IEEE Int. Conf. on Image Processing ICIP '06*, Atlanta, GA, USA, Oct. 2006, pp. 1693–1696.

[6] Johannes Ballé and Mathias Wien, "Extended texture prediction for H.264/AVC intra coding," in *Proc. IEEE Int. Conf. on Image Processing ICIP '07*, San Antonio, TX, USA, Sept. 2007, pp. VI–93–VI–96.

[7] 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. Conf. on Image Processing ICIP '04*, Singapore, Oct. 2004, pp. 465–468.

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

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

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