ANALYSIS OF HIERARCHICAL B PICTURES AND MCTF

Heiko Schwarz, Detlev Marpe, and Thomas Wiegand

Fraunhofer Institute for Telecommunications – Heinrich Hertz Institute, Image Processing Department
Einsteinufer 37, 10587 Berlin, Germany, [hschwarz|marpe|wiegand]@hhi.fraunhofer.de

ABSTRACT

In this paper, an investigation of H.264/MPEG4-AVC conforming coding with hierarchical B pictures is presented. We analyze the coding delay and memory requirements, describe details of an improved encoder control, and compare the coding efficiency for different coding delays. Additionally, the coding efficiency of hierarchical B picture coding is compared to that of MCTF-based coding by using identical coding structures and a similar degree of encoder optimization. Our simulation results turned out that in comparison to the widely used IBBP… structure coding gains of more than 1 dB can be achieved at the expense of an increased coding delay. Further experiments have shown that the coding efficiency gains obtained by using the additional update steps in MCTF coding are generally smaller than the losses resulting from the required open-loop encoder control.

1. INTRODUCTION

The increased flexibility of H.264/MPEG4-AVC [1] in comparison to prior video coding standards as MPEG-2 Visual, MPEG4 Visual or H.263 is one of the main reasons for its improved coding efficiency. However, this new flexibility on a picture/sequence level is still not a sufficiently well investigated topic. In contrast to previous video coding standards, the coding and display order of pictures in H.264/MPEG4-AVC is completely decoupled. Furthermore, any picture can be marked as reference picture and used for prediction of following pictures independent of the corresponding slice types. The set of pictures that is stored in the decoded picture buffer (DPB) and used for the prediction of following pictures can be adaptively controlled. These features allow the selection of arbitrary coding/prediction structures, which are not supported by previous standards.

In this paper, we analyze classes of hierarchical prediction structures regarding their encoding/decoding delay, memory requirements, and coding efficiency. Furthermore, commonalities and differences to the motion-compensated temporal filtering (MCTF) approach are described. A fair comparison of MCTF-based coding and H.264/MPEG4-AVC conforming coding in terms of coding efficiency is achieved by using identical temporal prediction structures.

2. HIERARCHICAL B PICTURES

A typical hierarchical prediction structure with 4 dyadic hierarchy stages is depicted in Fig. 1. The first picture of a video sequence is intra-coded as IDR (instantaneous decoder refresh) picture; so-called key pictures (black in Fig. 1) are coded in regular (or even irregular) intervals. A picture is called key picture, when all previously coded pictures precede the picture in display order. As illustrated in Fig. 1, a key picture and all pictures that are temporally located between the current key picture and the previous key picture are considered to build a group of pictures (GOP). The key pictures are either intra-coded (e.g. in order to enable random access) or inter-coded using previous (key) pictures as references for motion-compensated prediction (MCP). The remaining pictures of a GOP are hierarchically predicted as illustrated in Fig. 1 and coded using the bi-predictive (B) slice syntax of H.264/MPEG4-AVC.

Hierarchical prediction structures can also be used for supporting several levels of temporal scalability. For this purpose it has to be ensured that only pictures of a coarser or the same temporal level are employed as references for MCP (cp. Fig. 1). Then, the sequence of key pictures represents the coarsest supported temporal resolution, and this temporal resolution can be refined by adding the temporal refinement pictures of finer temporal levels.

It should be noted that the usage of hierarchical coding structures is not restricted to the dyadic case. Furthermore, the prediction structure can be adaptively adjusted over time. Also, the concept of multiple reference pictures can be combined with hierarchical coding structures, even though in the specific example of Fig. 1 only neighboring pictures of a coarser or the same temporal level are used for MCP.

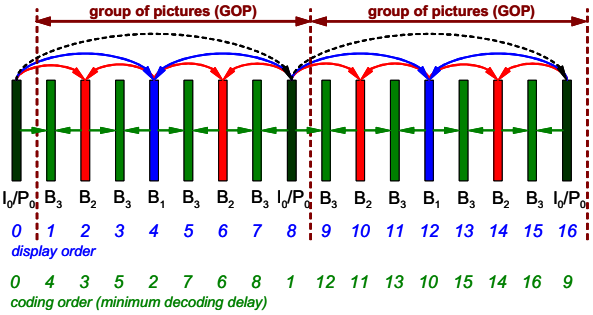


Fig. 1. Hierarchical coding structure with 4 temporal levels.

2.1. Coding Order and Delay

The coding order has to be chosen in a way that reference pictures are coded before they are employed for MCP. This can be ensured by different strategies, which mostly differ in the associated decoding delay and memory requirement.

In the following, we briefly describe a coding order inside a GOP that guarantees a minimal decoding delay. First, all pictures that are directly or indirectly used for MCP of the first picture of a GOP in display order and the first picture itself are coded. For the example in Fig. 1, this means that first the key picture (picture no. 8 in display order), and then the first pictures of all hierarchy stages (picture no. 4, 2, and 1) are coded. Next, all pictures that are required for coding the second picture of the GOP and the second picture itself are coded, etc. In Fig. 1, this coding order is illustrated for two groups of 8 pictures. The associated decoding delay in units of pictures is equal to the number of hierarchy levels minus 1 (equal to 3 pictures for the example of Fig. 1). Note that the encoding delay is independent of the coding order inside a GOP; it is always identical to the number of pictures in a GOP minus 1. However, both the encoding and decoding delay can be reduced down to a vanishing structural delay of 0 pictures by restricting MCP from using pictures as a reference that are located in the future.

2.2. Memory Requirement and Temporal Scalability

For the following analysis of the memory requirements we assume that the reference picture lists for all pictures are constructed using only directly neighboring pictures of a coarser or the same temporal level (cp. Fig. 1). Since the DPB will generally hold additional pictures that are marked as "used for reference", it is still possible to use multiple reference pictures for MCP. We further assume that the pictures are coded in the order, as described in the previous section. It can easily be verified that it is not possible to generate a different coding order that allows a smaller DPB size. In addition, as pointed out above, the selected coding order guarantees minimal decoding delay. Pictures of the highest temporal level (e.g. the pictures B_3 in Fig. 1) are always coded as non-reference pictures. These pictures don't need to be stored in the DPB and can be outputted just after decoding, since all of these non-reference pictures are coded in display order. Hence, the required DPB size (in units of pictures) is equal to the maximum number of reference pictures that need to be stored in the DPB, and consequently, it is also equal to the minimum required value of the syntax element num_ref_frames in H.264/MPEG4-AVC [1].

When using the coding order described in Sec. 2.1, it is always sufficient, when the 2 surrounding key pictures and 1 picture for each hierarchy level – with exception of the finest temporal level – are marked as "used for reference". Thus, when a key picture is decoded, it should replace the key picture before the previous key picture in the DPB. All other pictures that are coded as reference pictures should replace the previous picture of the same temporal level. This strategy can either be realized via memory management control operation (MMCO) commands of type 1 [1], or by storing all reference pictures as long-term picture using two long-term frame indices for key pictures and one additional long-term frame index for each further hierarchy level, for which the pictures are coded as reference pictures. Storing all reference pictures as long-term pictures has the advantage that temporal scalability is supported, which is not the case when MMCO 1 commands are applied, since then a removal of temporal refinement pictures generally results in an invalid bit-stream. Either method significantly reduces the memory requirement in comparison to the default sliding window marking process. The minimum required DPB size in pictures is equal to the number of hierarchy levels H. As an example, a DPB of only 6 frames is sufficient for coding groups of 32 pictures with a dyadic hierarchical structure. With the sliding window marking process it is not even possible to encode groups of 32 pictures, since the maximum allowed number of frame storages (16) would be exceeded.

2.3. Operational Encoder Control

For all experimental results that are presented in Sec. 4 below, a rate-distortion optimized H.264/MPEG4-AVC encoder following [2] has been used. Motion estimation and mode decision is performed as specified in the high-complexity mode of the Joint Model [3]. However, in order to further improve the coding efficiency for hierarchical prediction structures with B pictures, two details are slightly modified. While in the Joint Model [3] motion vectors for bi-predicted blocks are determined by independent motion searches for each reference list, it is a well-known fact that the coding efficiency can be improved when the combined prediction signal (weighted sum of list 0 and list 1 predictions) is considered during the motion estimation process. We employ a joint motion search for both reference picture lists using the iterative algorithm presented in [4].

The coding efficiency for hierarchical prediction structures is also highly dependent on how the quantization parameters are chosen for pictures of different hierarchy levels. Intuitively, the key pictures should be coded with highest fidelity, since they are directly or indirectly used as references for MCP of all other pictures, and thus the quality of the key pictures determines the quality of the prediction signal for all B pictures inside a GOP. For the next hierarchy level (B_1 in Fig. 1) a larger quantization parameter should be chosen, since the quality of these pictures influences less pictures. Following this rule, the quantization parameter should be increased for each subsequent hierarchy level.

An optimal selection of the quantization parameters can be achieved by a computationally expensive rate-distortion analysis similar to the strategy presented in [5]. In order to avoid such a complex encoder operation, we have chosen the following strategy, which proved to be sufficiently robust for a wide range of tested sequences. Based on a given quantization parameter for key pictures QP_0, the remaining

quantization parameters for pictures of a given temporal level $k > 0$ are determined by $QP_k = QP_{k-1} + (k = 1 \; ? \; 4 : 1)$.

Although this strategy for cascading the quantization parameters over hierarchy levels results in relatively large PSNR fluctuations inside a GOP of pictures, subjectively, the reconstructed video appears to be temporally smooth without any annoying temporal pumping artifacts.

3. MCTF VERSUS HIERARCHICAL B PICTURES

Motion-compensated temporal filtering (MCTF) [6] has been introduced as wavelet-based decomposition of image sequences along the temporal axis. It is generally realized using a motion-compensated lifting representation of the Haar or 5/3 spline filters. In [7], an MCTF extension of H.264/MPEG4-AVC has been presented. The employed temporal coding structure is very similar to the dyadic hierarchical structure described in Sec. 2. And actually, the only change in [7] compared to H.264/MPEG4-AVC is that motion-compensated update operations are introduced, in which a shifted and scaled version of the prediction error signal is added to the original signal of the corresponding reference pictures to be used in motion-compensated prediction. By applying these update steps, pictures of coarser temporal levels are effectively low-pass filtered along the motion trajectory before encoding. The strongest low-pass filtering is applied to the coarsest temporal level. The motion parameters for the update steps are derived from motion parameters of the prediction steps. At the decoder side, the inverse operations are applied in reverse order using the coded prediction error signal.

It is often believed that MCTF, and thus the additional motion-compensated update steps, results in superior coding efficiency when compared to classical hybrid video coding. This believe is then often based on the assumption that by using MCTF a certain noise reduction effect can be achieved for the reference signal. However, according to this line of reasoning, the impact of the corresponding temporal coding structure is mostly neglected. Typically, comparisons are made against hybrid video coding with the classical "IBBP…" structure. Furthermore, due to the temporal filtering structure, the motion-compensated prediction and update operations at the encoder side have to be carried out in reverse order of the decoder operations. Thus, it is not possible to apply a closed-loop encoder control for MCTF. This implies in particular that quantization errors can accumulate, since an MCTF encoder cannot compensate for quantization errors of the reference signal. Also, it has to be noted that the update steps result in significantly increased implementation costs due to the corresponding inverse MCP processes at the decoder side.

4. EXPERIMENTAL RESULTS

In the following, the coding efficiency for hierarchical prediction structure is evaluated and compare to that of a corresponding the MCTF extension of H.264/MPEG4-AVC.

4.1. Coding efficiency for hierarchical B pictures

The coding efficiency for dyadic hierarchical prediction structures with GOP sizes of up to 32 pictures is compared to "classical" prediction structures as IPPP… and IBBP… for a large set of test sequences. In Fig. 2 rate-distortion plots for two representative sequences are depicted. For sequences like "Mobile" that are characterized by high spatial detail and slow regular motion, we observed coding gains of more than 1 dB, when comparing hierarchical prediction structures with classical IBBP… coding. For this class of sequences, coding efficiency can be continuously improved by enlarging the GOP size up to about 1 second. Enlarging the GOP size, on the other hand, implies an increased depth of the temporal hierarchy and therefore, results in an increased coding delay. For sequences with faster and more complex motion like "Football", the maximum coding efficiency was reached with a GOP size of about 4 pictures. However, even for these sequences the subjective quality can be significantly improved as illustrated in Fig. 3, where reconstructed pictures coded with the IBBP… structure and a GOP size of 16 (about half a second) are compared. Fine-detailed regions of the background are noticeably better pre-

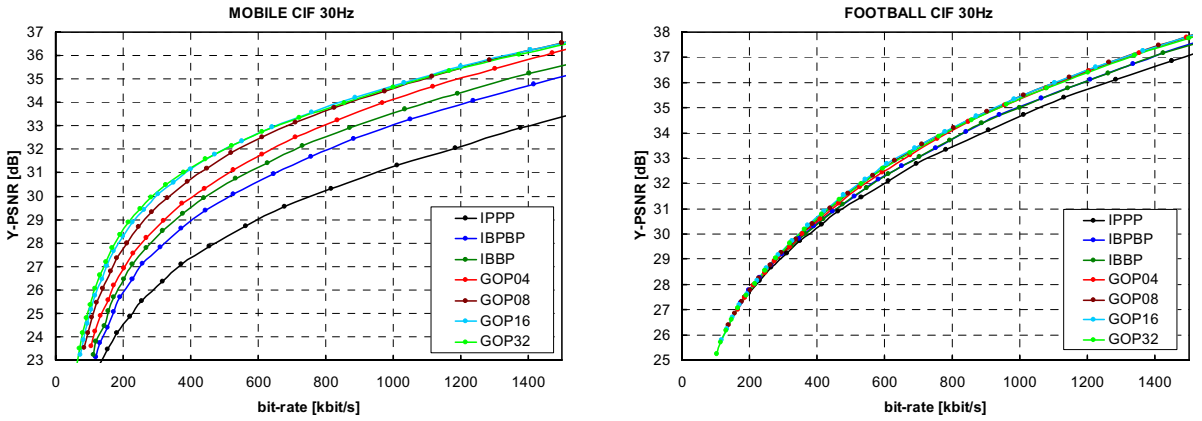
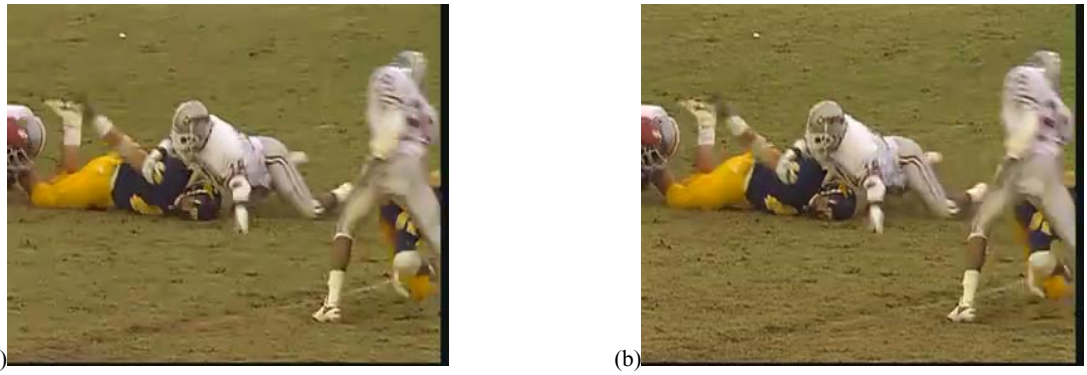


Fig. 2. Coding efficiency comparison of hierarchical prediction structures and IPPP, IBPBP, and IBBP coding structures.



(a) (b)

Fig. 3. Subjective comparison of frame 206 of the sequence "Football": (a) IBBP coding, (b) GOP size of 16

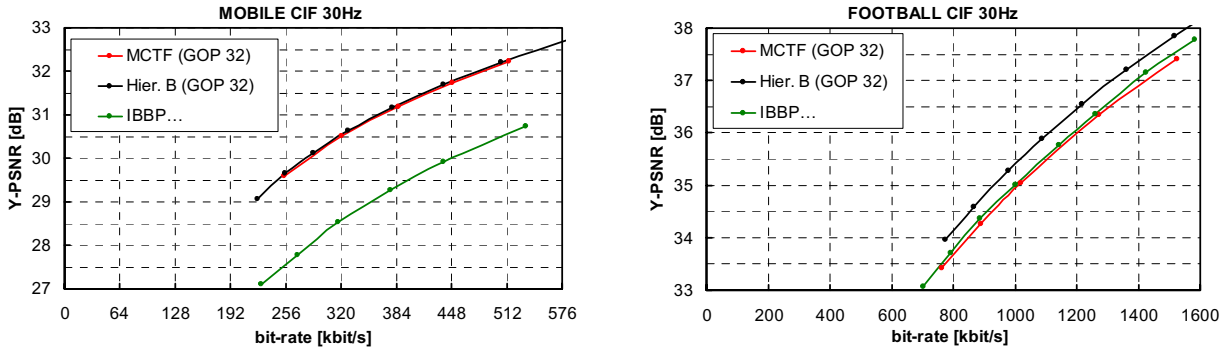


Fig. 4. Comparison of hierarchical B picture coding with MCTF coding for a GOP size of 32 pictures

served by using larger GOP sizes. Note that the selected pictures were deliberately chosen to represent low to medium quality in order to demonstrate the effects. The picture in Fig. 3(b) does not represent a high quality key picture.

4.2. Comparison of hierarchical B pictures and MCTF

In Fig. 4, the hierarchical B picture coding is compared with MCTF-based coding using the approach of [7]. Note that there are only two differences between the two codecs. First, additional motion-compensated update steps are performed in the MCTF coder while, secondly, the H.264/MPEG4-AVC conforming coder employs a closed-loop encoder control in contrast to the open-loop control of MCTF-based encoding, which is dictated by the corresponding temporal filtering structure. In general, we observed that the additional update steps lead to a smaller increase in coding efficiency than that obtained for the closed-loop encoder control. For sequences with fast and complex motion (e.g. "Football"), where MCP is generally less effective, the coding efficiency of the closed-loop coder is significantly higher than that of the MCTF coder.

5. CONCLUSION

The coding with hierarchical prediction structures was analyzed regarding the parameters of coding delay, memory requirements, and coding efficiency. Experiments showed that coding efficiency can generally be improved by increasing the coding delay. A comparison with MCTF-based coding using identical coding structures indicated that MCTF does not improve the coding efficiency, mainly because the open-loop coder control of an MCTF encoder cannot compensate for quantization errors of the reference pictures.

REFERENCES

[1] ITU-T Rec. & ISO/IEC 14496-10 AVC, "Advanced Video Coding for Generic Audiovisual Services," version 3, 2005.

[2] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. Sullivan, "Rate-Constrained Coder Control and Comparison of Video Coding Standards," *IEEE Trans. CSVT*, vol. 13, no. 7, pp. 668-703, July 2003.

[3] K.-P. Lim, "Text Description of Joint Model Reference Encoding Methods and Decoding Concealment Methods," *Joint Video Team*, JVT-L046, Redmond, USA, July 2004.

[4] M. Flierl, T. Wiegand, B. Girod, "A Locally Optimal Design Algorithm for Block-Based Multi-Hypothesis Motion-Compensated Prediction," *Data Comp. Conf.*, April 1998.

[5] K. Ramchandran, A. Ortega, M. Vetterli, "Bit Allocation for Dependent Quantization with Applications to Multiresolution and MPEG Video Coders," *IEEE Trans. Image Proc.*, vol. 3, no. 5, Sep. 1994.

[6] M. Flierl, "Video Coding with Lifted Wavelet Transforms and Frame-Adaptive Motion Compensation," *Proc. of VLBV*, Sep. 2003.

[7] H. Schwarz, D. Marpe, and T. Wiegand, "MCTF and Scalability Extension of H.264/AVC," *Proc. of PCS 2004*, Dec. 2004.