IMPROVING THE CODING EFFICIENCY OF MPEG-4 FGS BY USING HYBRID CODING IN THE ENHANCEMENT LAYER

Matthias Narroschke

Institut fuer Theoretische Nachrichtentechnik und Informationsverarbeitung
University of Hannover, Appelstr. 9a, 30167 Hannover, Germany
narrosch@tnt.uni-hannover.de

ABSTRACT

MPEG-4 FGS encodes a video sequence into two bitstreams, the base and the enhancement layer bitstream. A video of a low SNR can be decoded from the base layer bitstream. High SNR can be achieved by decoding both bitstreams. The base layer is encoded using hybrid coding. In the enhancement layer, the resulting quantization error is transform coded. Since the reference image used for prediction is reconstructed just from the base layer signal, no temporal correlations are exploited in the enhancement layer. Therefore FGS requires about 80% additional bitrate for the same image quality in the enhancement layer compared to non scalable coding. In this paper, two techniques of different computational expense in the decoder are proposed that exploit temporal correlations also in the enhancement layer by using hybrid coding. Experimental results show that the first technique can reduce the required additional bitrate to 50%, the second one to 20%. Furthermore it is shown in an analysis that the higher coding efficiency is associated with a higher computational expense in the decoder. The first technique requires 51% and the second one 156% additional computational expense whereas FGS requires 18%.

1. INTRODUCTION

Due to a growing demand for streaming software, MPEG-4 Fine Granularity Scalability (FGS) [1], a video coding technique for SNR-scalable video coding, was developed in the standardization process of MPEG-4. It encodes a video sequence into two bitstreams, the base layer (BL) and the enhancement layer (EL) bitstream. A video of a low SNR can be decoded from the BL bitstream. High SNR can be achieved by decoding both layers. The BL is encoded using hybrid coding. This technique uses motion compensated prediction to reduce the redundancy of the video signal in the temporal direction. The spatial redundancy of the remaining prediction error signal is reduced by transform coding. The transform coefficients of the prediction error are quantized and transmitted. The resulting quantization error is requantized and transmitted in the EL. Figure 1 shows the block diagram of the FGS encoder.

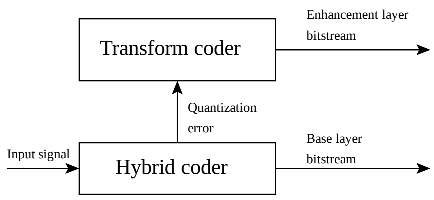

Fig. 1. Block diagram of the FGS encoder.

To avoid drift, the reference image used for prediction is always reconstructed just from the BL signal. Thus, no temporal correlations are exploited in the EL. Therefore FGS is associated with a decrease in coding efficiency in the EL compared to a non scalable MPEG-4 coder. For the same PSNR, FGS requires about 80% additional bitrate in the EL compared to a non scalable coder [2].
In this paper, two techniques based on the same idea are proposed that exploit temporal correlations also in the EL by using hybrid coding. The proposed techniques are explained in Section 2. Experimental results comparing the coding efficiency of the proposed techniques with the one of FGS and non scalable coding are given in Section 3. The computational expense in the decoder of both techniques is analyzed and compared to the one of FGS and non scalable coding in Section 4. Section 5 gives the conclusion.

2. SNR-SCALABLE VIDEO CODING USING HYBRID CODING IN THE ENHANCEMENT LAYER

Figure 2 shows the blockdiagram of the proposed encoder. In the BL, a standard hybrid coder (Hybrid coder 1) is used. The quantization of this coder is coarse to receive a low SNR resolution and a low bitrate. Thus, the PSNR of the reference image used for further predictions is also low.

In the EL, both, hybrid coding (Hybrid coder 2) and

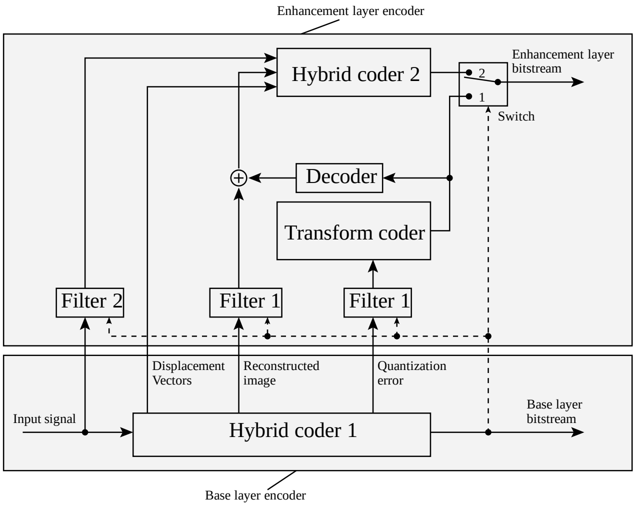

Fig. 2. Block diagram of the proposed encoder.

transform coding of the quantization error of the base layer signal is used adaptively. In opposition to FGS, the quantization error is transmitted in the EL only for parts of the image (Switch in position 1). These parts pass Filter 1 and not Filter 2. For all other parts of the image, which pass Filter 2 and not Filter 1, hybrid coding is used in the EL (Hybrid coder 2, Switch in position 2). This hybrid coder operates comparable to a non scalable hybrid coder with a reference image of a high PSNR reconstructed from both layers. Concerning motion compensation, all displacement vectors transmitted in the BL are used in the EL. If additional displacement vectors are required in the EL, they are estimated based on the reference image of the EL.
The proposed techniques differ in the type of adaptation which is controlled by the choice of Filter 1 and Filter 2. Two types of adaptation are tested which both need no extra side information to be transmitted. All control information for the adaptation is used from the BL bitstream that both, the encoder and the decoder have. Both techniques are explained in the next Subsections.

2.1. Block based adaptation

The first type allows an adaptation based on blocks of a size of 4 by 4 pel (4x4 blocks). This means that for whole 4x4 blocks, either the quantization error of the base layer signal or the prediction error of Hybrid coder 2 is sent in the EL. Because of the fact that the transmitted data in the BL is redundant for all blocks that are hybrid encoded in the EL, the use of hybrid coding just for blocks that are encoded with very little data in the BL allows a significant limitation of the redundancy. In MPEG-4 part 10/H.264, if all quantized transform coefficients of the prediction error of a 4x4 block are equal to zero, just one single bit is used to encode all of these coefficients. Furthermore, if all quantized transform coefficients of the prediction error and also all motion vectors of a whole macroblock which consists of sixteen 4x4 blocks are equal to zero the macroblock is called *skipped* and the whole macroblock is encoded with just one bit. Compared to these blocks, all other blocks that have transform coefficients unequal to zero require much more data for the transmission in the BL. Therefore to limit the redundant information, for all 4x4 blocks that have transform coefficients unequal to zero, the quantization error is transmitted in the EL. All other blocks, that are more than 40% of all blocks if using a coarse quantization (see Figure 3), are selected to be hybrid coded. For a finer quantization as used in the EL, the amount of blocks with transform coefficients that are all equal to zero is significantly lower, which can be seen in Figure 3. Therefore in the EL, most blocks can be efficiently hybrid coded as using a single layer hybrid coder.

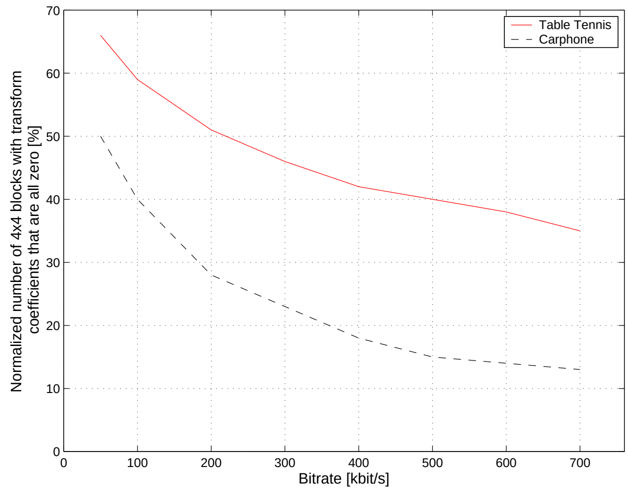

Fig. 3. Normalized number of 4x4 Blocks with transform coefficients that are all zero versus bitrate for different test sequences.

2.2. Transform basis function based adaptation

The second type allows an adaptation based on every transform basis function. Compared to the first type of adaptation, this one allows a finer adaptation and enables efficient hybrid coding to even more parts of each image.
Due to the fact that usually the energy of the transform coefficients of the prediction error signal is concentrated in the low frequencies, a so called zigzag scan an combined run and level coding is used. Small runs combined with small levels are coded with short code words, long runs combined with large levels with longer ones. If, at a certain point of

the scan, all following coefficients are equal to zero one specific *end of block* symbol (EOB) is sent which is encoded with only one bit. Due to a coarse quantization as used in the BL, for most of the 4x4 blocks only the very first coefficients of the scan, often just the DC coefficient, are unequal to zero and require long codewords. All other coefficients are encoded together using the EOB symbol. Thus, in the second technique, hybrid coding is used in the EL for all basis functions of the transform for which the quantized prediction error in the baselayer is equal to zero. The redundant information is mostly just the EOB symbol which is just marginal. For example, if just the DC coefficient of the quantized prediction error in the BL is unequal to zero, only for this single coefficient and therefore for this transform basis function the quantization error is transmitted. All other basis functions are hybrid encoded in Hybrid coder 2.

3. EXPERIMENTAL RESULTS

The new coding techniques were integrated in the reference MPEG-4 part 10/H.264 software (JM1.4). The displacement vector resolution was set to 1/4-pel and the frame rate to 10 Hz. Experimental results are shown in Figures 4 and 5 for the test sequences *Table Tennis* and *Carphone*. The operational rate distortion curves of the following five coding techniques are compared: MPEG-4 ASP (non scalable coding), MPEG-4 FGS, MPEG-4 part 10/H.264 (JM 1.4) and the new techniques using the two different types of adaptation, block based and transform basis function based adaptation.

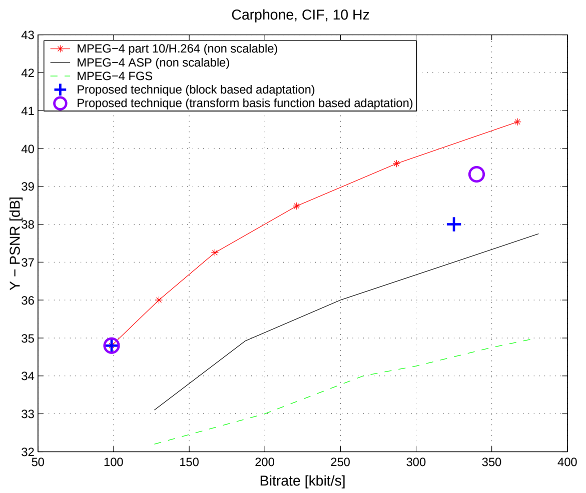

Fig. 4. Operational rate distortion curves for the test sequence *Carphone*.

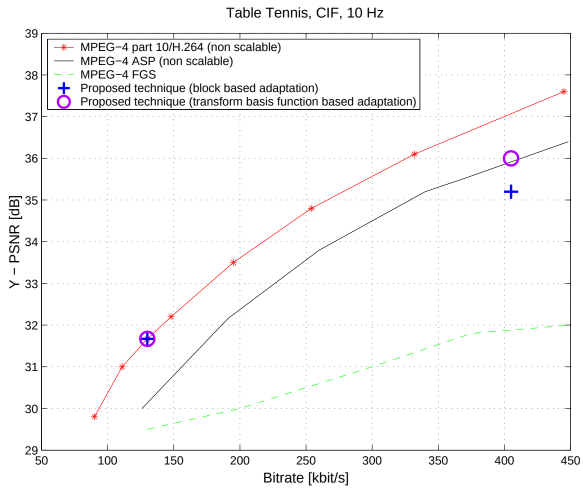

Fig. 5. Operational rate distortion curves for the test sequence *Table Tennis*.

It can be seen that the required overhead in form of additional bitrate to achieve the same PSNR in the EL as the according non scalable coder can be reduced from about 80% (FGS) to 50% using block based adaptation and to 20% using transform basis function based adaptation. At this moment, the proposed techniques support only two resolutions. More resolutions for a fine grained scalability can be supported in the future by integrating a bitplane coding for the transform coefficients in the EL [6].

4. ANALYSIS OF THE COMPUTATIONAL EXPENSE IN THE DECODER

In this section, the computational expense in the decoder $C_{Decoder}$ is analyzed. The computational expense is measured in cycle rates required for the performance of the decoder subtasks Motion Compensation (MC), Inverse Discrete Cosine Transform (IDCT) and Reconstruction of the image (REC).

$$\begin{aligned} C_{Decoder} = \; & C_{IDCT} \cdot N_{IDCT}(Decoder) + \\ & C_{MC} \cdot N_{MC}(Decoder) + \\ & C_{REC} \cdot N_{REC}(Decoder) \end{aligned} \quad (1)$$

The required cycle rates for the different subtasks can be found in [4]. They are:

$$\begin{aligned} C_{IDCT} &\approx 47.5 \text{ Cycles} \\ C_{MC} &\approx 60.8 \text{ Cycles} \\ C_{REC} &\approx 5.6 \text{ Cycles} \end{aligned}$$

The quantity of the processed subtasks N_i with $i \in \{IDCT,\ MC,\ REC\}$ is counted for each decoder for the first 22 Frames. In Table 1 and 2, the quantities N_i and the resulting computational expense are shown for the test-sequences *Carphone* and *Table Tennis*. It can be seen that both proposed techniques require significantly more cycles for the decoding than the corresponding non scalable decoder MPEG-4 part 10/H.264.

Coder	N_{IDCT}	N_{REC}	N_{MC}	$\frac{C_{Decoder}}{\text{Cycles}}$
MPEG-4 part 10/ H.264	67859	139392	132255	12045002
Proposed technique (block based adaptation)	86190	278784	209321	18381932
Proposed technique (transform basis function based adaptation)	315599	278784	231993	30657317

Table 1. Quantity of the processed subtasks and computational expense in the decoder for the test sequence *Carphone* (bitrate: 330 kbit/s).

Coder	N_{IDCT}	N_{REC}	N_{MC}	$\frac{C_{Decoder}}{\text{Cycles}}$
MPEG-4 part 10/ H.264	75939	139392	113143	11266792
Proposed technique (block based adaptation)	100986	278784	172880	16869129
Proposed technique (transform basis function based adaptation)	316890	278784	202702	28937747

Table 2. Quantity of the processed subtasks and computational expense in the decoder for the test sequence *Table Tennis* (bitrate: 400 kbit/s).

The normalized additional computational expense in each of the decoders $AC(Decoder)$ can be calculated as follows:

$$AC(Decoder) = \frac{C_{Decoder} - C_{MPEG-4\ part\ 10/H.264}}{C_{MPEG-4\ part\ 10/H.264}} \tag{2}$$

Table 3 shows $AC(Decoder)$ for different decoders and different test sequences. It can be seen that the proposed technique using block based adaptation requires 51% additional computational expense in the decoder, using transform basis function based adaptation 156% in average.

The average additional bitrate versus the average additional computational expense in the decoder, each compared to the one of the corresponding non scalable decoder, is illustrated in Figure 6. It can be observed that there is a dependancy between the coding efficiency and the required computational expense in the decoder. The higher the coding efficiency is the more computational expense is required in the decoder.

	Carphone	Table Tennis	Average
MPEG-4 part 10/ H.264	0%	0%	0%
Proposed technqiue (block based adaptation)	53%	49%	51%
Proposed technique (transform basis function based) adaptation)	155%	157%	156%

Table 3. Additional computational expense AC for different decoders and different testsequences.

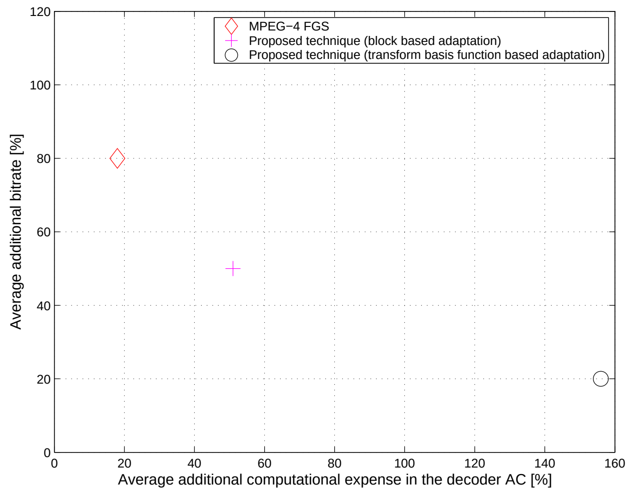

Fig. 6. Average additional bitrate versus average additional computational expense in the decoder (AC) for FGS and the proposed techniques.

5. CONCLUSION

In MPEG-4 FGS, the BL is encoded using hybrid coding. In the EL, the resulting quantization error is transform coded and transmitted. Since the reference image used for prediction is reconstructed just from the BL signal, no temporal correlations are exploited in the EL. Therefore FGS requires about 80% additional bitrate for the same image quality in the enhancement layer compared to non scalable coding. This paper presents two techniques for SNR scalable video coding using hybrid coding and transform coding of the quantization error of the base layer signal in the EL adaptively. Due to this hybrid coding, the temporal correlations are also exploited in the EL.
The first technique uses an adaptation based on full 4x4 blocks whereas the second one uses one based on transform basis functions. Experimental results show that the first technique can reduce the required additional bitrate from 80% to 50%, the second one even to 20%. An analysis shows that the higher coding efficiency can only be achieved

with a substantially higher computational expense in the decoder. The first technique requires 51% additional computational expense in the decoder, the second one 156%.

6. REFERENCES

[1] MPEG-4 FGS: ISO/IEC JTC1/SC29/WG11, ISO/IEC 14496-2 FDAM4: Information Technology - Coding of Audio-Visual Objects - Part 2: Visual, Amendment 4: Streaming Video Profile, ISO/IEC/JTC1/SC29/WG11 N3904, Pisa, January 2001.

[2] W. Li, Y. Chen, "Experiment Result on Fine Granular Scalability", Doc. ISO/IEC JTC1/SC29/WG11 MPEG99/M4473, March 1999.

[3] JVT: ISO/IEC and ITU-T, "Committee Draft of Joint Video Specification (ITU-T Rec. H.264 / ISO/IEC 14496-10 AVC)", Doc. JVT-C167, Fairfax, Virginia, USA, May 2002

[4] H.-J. Stolberg, M. Bereković, P. Pirsch, "A platform-Independent Methodology for Performance Estimation of Streaming Media App lications", in Proc. of IEEE International Conference on Multimedia and EXPO, 2002.

[5] M. Narroschke, "Functionalities and Costs of Scalable Video Coding for Streaming Services", in Proc. of 36th Asilomar Conference on Signals, Systems, and Computers , Pacific Grove, California, November 2002.

[6] W. Li et. al., "Bit-Plane Coding of DCT Coefficients", Doc. ISO/IEC JTC1/SC29/WG11 MPEG97/M2691, October 1997.