Very Low Bit Rate Video Coding Using Wavelet-Based Techniques*

Detlev Marpe and Hans L. Cycon**

Abstract

In this paper we propose a very low bit rate video coding scheme based on a discrete wavelet transform (DWT), block-matching motion estimation (BME) and overlapped block motion compensation (OBMC). Our approach reveals that the coding process works more efficiently if the quantized wavelet coefficients will be pre-processed by a mechanism exploiting the redundancies in the wavelet subband structure. Thus, we introduce a new framework of pre-coding techniques based on the concepts of partitioning, aggregation and conditional coding (PACC). Our experimental results show that our PACC coder outperforms the VM (Version 5.1) of MPEG4 both for coding of intra-frames (1–2 dB PSNR) and residual frames (up to 1.5 dB PSNR) of typical MPEG4 test sequences. The subjective quality of reconstructed video is in general superior to that obtained from the VM implementation. In addition, when restricted to intra-frame mode, the proposed coding algorithm produces results which are among the best reported for still image compression.

*Parts of this work were presented at Picture Coding Symposium '97 in Berlin, Germany. This work was supported by Deutsche Telekom AG, Technologiezentrum Darmstadt, and Deutsche Telekom Berkom GmbH, Berlin, Germany.

**The authors are with the Fachhochschule für Technik und Wirtschaft, Allee der Kosmonauten 20–22, 10315 Berlin, Germany.

I. Introduction

Very low bit rate video compression aimed for video communication systems operating on channels with low bandwidth has been a fast expanding research field in the last years. Besides the well established techniques based on a block-based motion compensation (BMC) and discrete cosine transform (DCT), which have been melted in the well elaborated H.263 standard [4], there emerged new ideas and techniques [6].

One of the most promising of these techniques is the wavelet transform scheme, where transform coding is combined with subband coding techniques. The wavelet transform is an invertible transformation which decomposes the signal into a dyadic structured tree of subbands by convolution-decimation operations. It decorrelates mutually dependent parts and performs an energy compaction of the samples representing the input signal. In addition, wavelets are well localized in phase space (i.e. space-frequency domain) thereby matching the characteristics of natural images and revealing their scale-invariant features. The transformation module of a wavelet-based coding scheme usually is followed by a quantizer which annihilates visually non-relevant information and in a final step, an entropy coder is employed to remove statistical redundancies in the data stream generated by the quantizer.

This generic wavelet-based coding scheme can be applied to still images and video provided that for the latter we add a mechanism to exploit the temporal redundancies. A plethora of work has been published aiming to provide a solution to the problem of extending the 2D wavelet-based scheme for video coding. The methods may be classified into 3 groups of different approaches. One group proposes extensions of the 2D wavelet transform or subband coding scheme to 3D-subband coding (3D-SBC) [16, 15, 21]. A second group attempts to capture temporal redundancies with help of a multiresolutional motion compensation (MRMC) in the wavelet domain [29]. Our approach follows the third idea presented in [18], where a scheme using a modified block matching algorithm, so-called overlapped block motion compensation (OBMC) [24, 27] was proposed. Like conventional block-based motion compensation, OBMC is a very effective technique for temporal predictive coding with the advantage of eliminating blocking artifacts in the prediction error signal. However, in contrast to 3D-SBC and MRMC, the hybrid OBMC/2D-DWT scheme is inherently incompatible with spatial scalability, which may be of some importance for special applications.

In this work we focus our optimization activities on the coding part, i.e. we introduce a framework of so-called *pre-coding* techniques which pre-processes the quantized wavelet coefficients prior to entropy coding. This framework is based on the concepts of *partitioning*, *aggregation* and *conditional coding* (PACC) introduced in a previous publication [13]. The idea follows the observation that the localization properties of the wavelet transform call for a mechanism which efficiently exploits the redundancies resulting from the characteristics of the wavelet

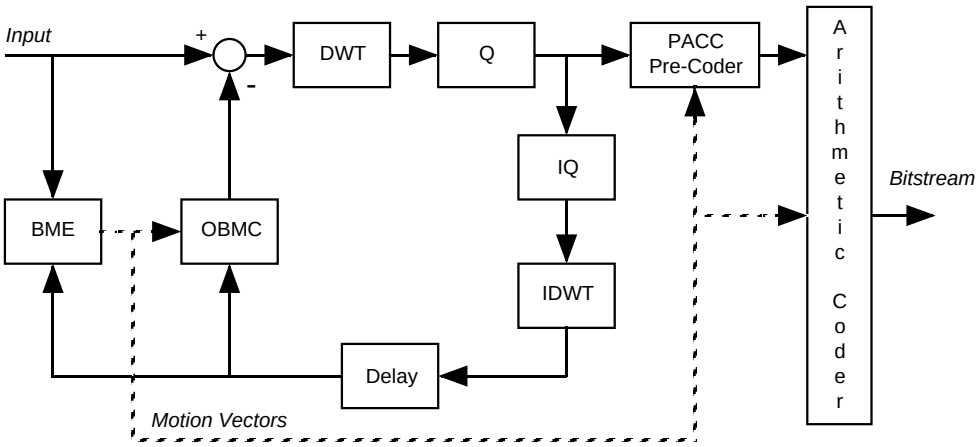


Fig. 1: Block diagram of the proposed PACC encoder

representation.

According to our PACC concept, we first split the data stream emerging from the quantizer into different subsources ("partitioning"). In a second stage, we capture correlations within and between different subsources by aggregating homogeneous elements into quadtree related data structures ("aggregation"). By using models based on conditional probabilities we are able to recover correlations between the structures constructed before as well as cross-correlations between different subsources, which will be utilized in a final arithmetic coding module ("conditional coding").

The organization of the paper is as follows. In Section II we give a brief overview of the proposed PACC video coder. Section III describes the algorithm, i.e. we review the properties of the used motion estimation and compensation, the DWT and introduce our quantization method. Following that, we formulate the essential principles of the PACC pre-coding framework and derive a collection of pre-coding methods from these principles. We conclude this section by a short discussion of our specific implementation of arithmetic coding. In Section IV we present and discuss our experimental results, which show the performance of our pre-coding methods embedded in a full codec for still images and for video sequences. The conclusion can be found in Section V.

II. Overview of the PACC-Codec

A simplified block diagram of our coding algorithm is given in Fig. 1. It essentially consists of three parts. First and main part is a temporal predictive feedback loop, where prediction is performed using a block-matching estimation (BME) and an overlapped block motion compensation (OBMC). The initial state of this loop processing, so-called *intra-frame* mode bypasses the predictive part

to enter directly the discrete wavelet transformation (DWT). After quantization (Q) the pre-coder performs a pre-processing of the quantized wavelet representation to allow an efficient exploitation of its inherent redundancies in the final arithmetic coding stage. A dequantization (IQ) and inverse DWT (IDWT) feeds the predictive loop with a reconstructed frame for processing of the next input frame in *inter-frame*, i.e. predictive mode.

III. Description of the Algorithm

A. Motion Estimation and Compensation

High compression performance in video coding relies on an efficient reduction of temporal redundancies in successive frames. Most of the successful algorithmic approaches to this problem are based on the assumption that groups or blocks of pixel intensities of a current frame can be estimated or predicted from related blocks of the previous frame by uniform translational motion.

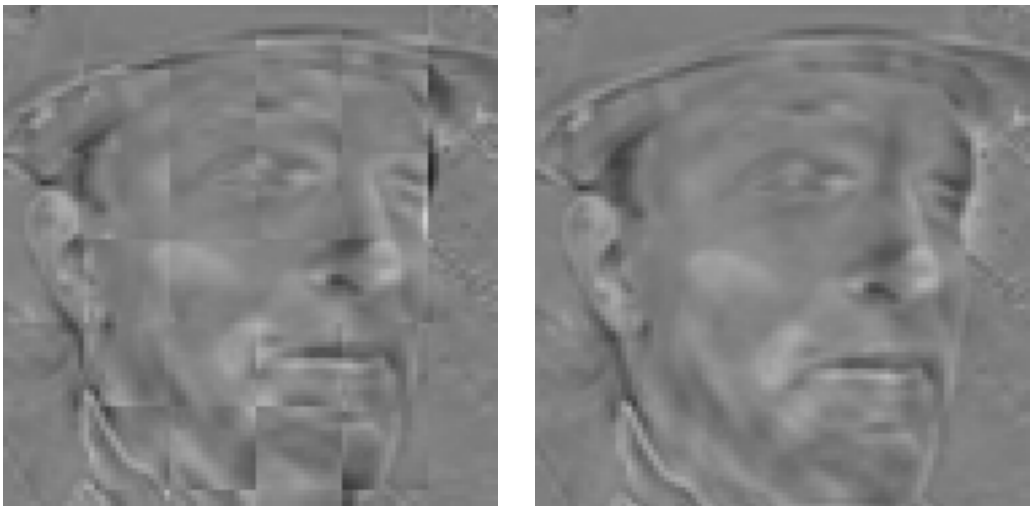
Given a partition into square blocks (typically of size 16×16 pixels) of a current frame, the block-based motion compensation algorithm consists of two steps: 1) Estimation of a motion vector field which relates each block of the partition to a displaced block in the previous frame, 2) Compensation of the estimated motion in order to substitute the current frame with its prediction residual (P-frame).

1) Block-Matching Estimation (BME): The motion estimation algorithm we use in our implementation is very similar to the schemes of MPEG4-VM* [20] or H.263 [4]. For each 16×16 block of pels in the current frame and a search range of ± 15 pixels in both directions relative to the center of the block, the best matching block of the previous decoded frame is found by minimizing some measure of the prediction error, typically so-called *sum of absolute difference* (SAD). To favor the zero motion vector, the SAD of the zero displacement is reduced by a fixed value of 100. Given the motion vector with minimal error norm, the 8 surrounding half-pel displacements are simulated using a bilinear interpolation of the image data and finally, the half-pel accurate motion vector corresponding to the block with minimal SAD is selected.

Note that the estimation process is performed on the luminance component of each frame. To obtain a prediction of the chrominance components, the (x, y)-components of each motion vector are divided by 2 due to the lower spatial resolution of the chrominance data. The resulting quarter-pel accurate motion vectors are modified towards the nearest half-pel positions.

In contrast to MPEG4-VM and H.263, we do not yet implement the optional *advanced prediction mode*, which in addition uses the four 8×8 sub-blocks of

*MPEG4 Video Standard Verification Model



(a) (b)

Fig. 2: Zoomed part of prediction error between frames 0 and 4 of FOREMAN: (a) Conventional Block Matching, (b) Overlapped Block Matching

each 16×16 block for a better modeling of the motion process in the BME.

2) Overlapped Block Motion Compensation (OBMC): After the BME stage, we can build a prediction error signal using the predicted blocks of the related previous frame, thus obtaining a signal with significantly lower energy than that of the original frame, in general. Due to the block-based nature of this conventional compensation technique, blocking artifacts are introduced in the prediction error frame. Fig 2 (a) shows an example of the prediction error image obtained after conventional BMC. For block-based DCT-coders like MPEG4-VM or H.263, the blocking artifacts are not critical as long as the block boundaries of transformation and block matching are well aligned. Wavelet-based coders, however, which transform an entire P-frame will have to sacrifice much of their coding efficiency by coding the artificial high-frequency information at the block boundaries of motion compensated P-frames.

To overcome this drawback, we consider a variation of the conventional block motion compensation in this work, so-called *overlapped block motion compensation* (OBMC), which has been proposed in [24, 27] and which is conceptually related to *lapped orthogonal transforms* [12]. The key element of this technique is the design of a smooth window function such that overlapping portions of windows from adjacent blocks add to unity. For our simulations, we have used the "raised cosine" window w given by

$$w(n,m) = w_n \cdot w_m, \quad w_n = \frac{1}{2}\left[1 - \cos\frac{\pi(n+\frac{1}{2})}{16}\right] \quad \text{for } n = 0,\ldots,31. \qquad (1)$$

It is defined on a 32×32 pixel support centered over a "core" block of size

16×16 pels. Using the OBMC method, the prediction error signal (P-frame) is formed by the weighted sum of the differences between all 32×32 overlapped block from the current frame and their related shifted overlapped blocks with displaced locations in the previous frame which have been estimated in the BME for the corresponding core blocks. In Fig. 2 (b) a zoomed part of the residual frame obtained by OBMC is shown. Comparing Fig. 2 (a) and 2 (b), it can be realized immediately that OBMC is more adequate than conventional BMC in the presence of a wavelet-based approach.

Note that there is a straightforward extension of the BME modifying the search algorithm as well to incorporate a weighting window function [27]. This feature together with a possible application of the OBMC method to the prediction of chrominance components will be a subject of further investigations. (In the current implementation of our coding algorithm OBMC is restricted to the processing of the luminance data, while for the chrominance components we use a conventional block motion compensation procedure.)

B. Wavelet Representation and Quantization

1) Multiresolution Analysis: The framework of multiresolution analysis allows an efficient implementation of the DWT using a perfect reconstruction two-channel filterbank. Given an input image I and a filterbank with lowpass-filter H and highpass-filter G, the first step of a wavelet decomposition is performed by repeated application of H and G on rows and columns, subsequently. Due to the separable nature of this filtering process, we get a representation with components in four subbands $\{W_{1,k} | \, k = 0, 1, 2, 3\}$ of different frequency localization. In our notation, the first subscript indicates the (first) *level* of decomposition and the second index denotes the *orientation* of the band according to the four possible combinations of applying H and G to rows and columns (HH, HG, GH, GG). Iterating the row- and column-wise operations of convolution-decimation on the resulting lowest frequency subband $W_{l,0}$ $(l \geq 1)$ yields an unbalanced logarithmic tree-structure of subbands which represents different resolution levels of the input image I.

For the purpose of image compression, the use of linear phase filterbanks associated with biorthogonal wavelet bases is preferable mainly for two reasons. First, the symmetry of the filters allows to solve the border extension problem in a non-expansive way (both in terms of information content and computational complexity). Second, iterated convolution-decimation of a non-symmetric filter may induce a shift of the filtered image by a different amount on different levels which is counterproductive to the utilization of interband correlations of coefficients belonging to the same spatial location. Moreover, from an implementation point of view it is important to note that symmetric filters require only half the number of multiplications compared to non-symmetric filters of the same length.

In our experiments, we have used a biorthogonal 9/7-tap filter [5] which proved

to be best suited to image compression applications both in terms of subjective and objective performance [22]. The maximum level of decomposition was chosen depending on the spatial resolution of the test material.

2) Uniform Scalar Quantization: An optimal quantization strategy would aim to eliminate information in the wavelet transformed image in such a way that under the constraint of a given target bit rate the resulting artifacts in the reconstructed image tend to be below the threshold of visibility. A solution of this problem would involve the design of a reliable mathematical model of human visual perception which, of course, is beyond the scope of our approach. However, starting with the simplest model of an uniform scalar quantizer with an overall stepsize q[†], we keep the option of a perceptual weighting of the "quality factor" q depending on the frequency content of a given subband $W_{l,k}$[25].

To absorb those wavelet coefficients, which are essentially related to noise, we have implemented a *deadzone*, i.e. a larger zero bin $[-\tau, \tau]$. The ratio $\eta = \frac{2\tau}{q}$ of zero bin size to stepsize has been chosen $\eta_{intra} = 1.5$ for intra-frame coding and $\eta_{inter} = 2.0$ for inter-frame coding. This was found empirically to be a good choice for all tested video sources at various bit rates. Moreover, following a mean-square error minimizing strategy, the choice of η_{inter} and η_{intra} has proven to be a good approximation of the optimum [11].

C. The PACC Pre-Coding Framework

In most transform coding schemes, a combination of predictive, run-length and variable length coding is applied to remove correlations of the quantized coefficients. These methods are known to offer a good trade-off between complexity and coding efficiency. To achieve further improvements in coding efficiency, more complex and sophisticated models have to be designed. Assuming some higher degree of statistical dependencies in our given data, a straightforward approach to a model of order N, for example, would require to estimate probability distributions (PD) for M^{N+1} different values of a random vector $\mathbf{s} = (s_0, \ldots, s_N)$ with a given alphabet size of M, which for sufficiently large N and M is impracticable, in general. Thus, we have to make some simplifications, while maintaining a maximal degree of statistical dependency.

The first step in this direction is to divide the source and to reduce the alphabet size M of the source with the instrument of partitioning. Fig. 3 illustrates that the initial partitioning process of our PACC pre-coding framework results in three subsources: the *significance map*, indicating the location of significant coefficients, the *magnitude map*, storing the absolute values of significant coefficients and the *sign map* with the phase information of the wavelet coefficients.

[†]Note that normalization of the DWT was chosen such that the filter coefficients of H sum up to $\sqrt{2}$.

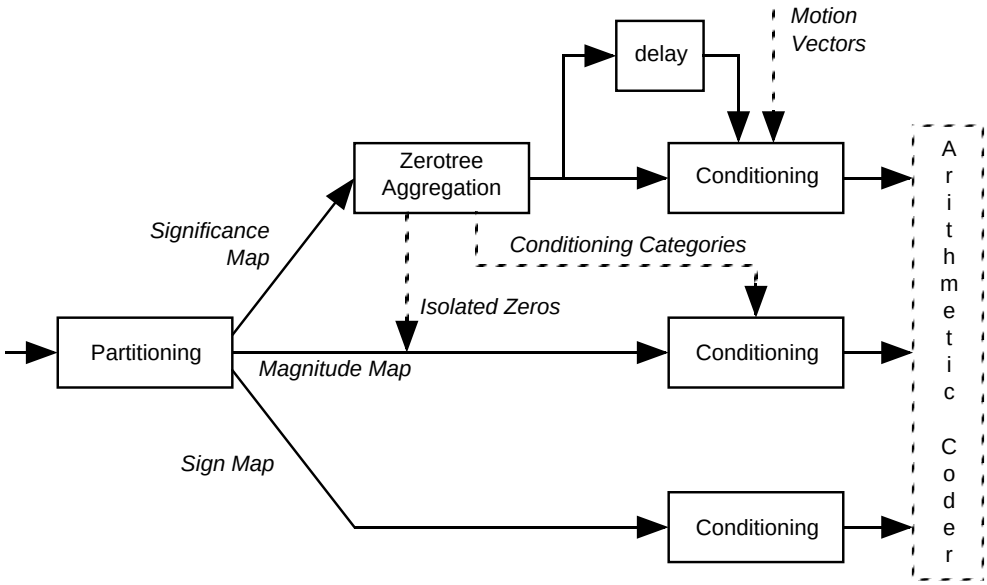


Fig. 3: Schematic representation of the PACC Pre-Coder

All three subsources inherit the subband structure from the wavelet decomposition, so that we obtain another partition of each subsource according to the dyadic band structure. Following the ideas of [10, 19], the second step of our pre-coding method consists of an aggregation of insignificant coefficients across different bands using the so-called *zerotree* data structure.

The main part of our pre-coder finally supplies the elements of each source with a "context", i.e. an appropriate model for the actual coding process in the arithmetic coder. Here we combine the two preceding methods with a context-based modeling which was initially introduced in [9] and later on successfully implemented in the JBIG[‡] standard [1]. It offers a very efficient, adaptive and flexible coding strategy for the removal of higher-order redundancies with a rather modest demand of computational resources.

1) Partitioning: The theoretical basis of partitioning is given by two theorems, which were initially introduced in a different context [8] and later discovered to be useful in wavelet-based image coding [23]. The first theorems states that the entropy rate of a source can be reduced by dividing the source into disjoint non-empty subsources [8].

In our coding scenario, this principle of source partition has one obvious application: we split the quantized and wavelet transformed image $\hat{c}$ according to its subband structure into subsources $\hat{c}_{l,k}$ with different PDs. Although this first step of partitioning may already decrease the overall (theoretical) entropy rate, we add two iterated steps of an adaptive range partition, characterized by the

[‡]Joint Bilevel Image Experts Group (ISO committee)

following theorem:

Theorem 1 (Adaptive Range Partition) *Let the dynamic range of a source $\mathcal{S}$ be given by $\mathcal{A} = \bigcup_i \mathcal{A}_i$, where $\mathcal{A}_i$ are disjoint, nonempty subsets of $\mathcal{A}$ and define the subsources $\tilde{\mathcal{S}}_i = \{s \in \mathcal{S} | \, s = \alpha, \alpha \in \mathcal{A}_i\}$ and the indicator set $\chi = \{x_k | \, x_k = i, \text{ if } s_k \in \mathcal{A}_i\}$, then the total entropy rate is given by*

$$\mathcal{R}(\mathcal{S}) = \sum_i \mathcal{R}(\tilde{\mathcal{S}}_i) + \mathcal{R}(\chi). \tag{2}$$

Adaptive range partitioning does not increase the entropy rate (which is the essential interpretation of Eq. (2)), but it allows to disentangle the information. Dividing the range in significant, i.e., non-zero quantized and insignificant, i.e., zero quantized values, will result in a subsource of significant coefficients $\hat{c}_{l,k}^{\text{sig}}$ and a source containing the side-information of the adaptive partition, the so-called *significance map* $\chi_{l,k}$. A further range partition according to the sign of significant coefficients finally yields using Eq. (2)

$$\mathcal{R}(\hat{c}_{l,k}) = \mathcal{R}(\chi_{l,k}) + \mathcal{R}(\hat{c}_{l,k}^{\text{mag}}) + \mathcal{R}(\varsigma_{l,k}), \tag{3}$$

where the *magnitude map* $\hat{c}_{l,k}^{\text{mag}}$ is the subsource containing the magnitudes of significant coefficients and the *sign map* $\varsigma_{l,k}$ holds the relevant sign information.

Eq. (3) tells us that our replacement of $\hat{c}_{l,k}$ with two binary-valued indicator maps $\chi_{l,k}$ and $\varsigma_{l,k}$ and a magnitude map $\hat{c}_{l,k}^{\text{mag}}$ does not alter the lower bound on the attainable coding rate, but, what is more important, it gives us insight into the way how we can approximate this ultimate bound. Since the partitioning process itself is a result of a higher level of abstraction, it allows a better utilization of the interdependence of different subsources either of different type or of different frequency content in the following two steps.

2) Zerotree Aggregation: For very low bit rate video coding two requirements are essential:

1. After quantization of intra-frames (I-frames) only a small fraction of coefficients should be left nonzero.

2. The output of the temporal prediction scheme should be a prediction error signal with low energy resulting in a quantized wavelet representation with few nonzero coefficients.

Although these requirements are rather stringent and not always fulfilled, especially in the presence of a scene with large motion, it is obvious, that there is a need for an efficient coding method of zero quantized coefficients.

Relating insignificant coefficients in the wavelet representation which share the same spatial location but reside in different levels, we can build balanced

quadtrees of insignificant coefficients, so-called zerotrees (ZT). In the pioneering work of [10, 19], the zerotree data structure has been recognized as an useful tool to exploit the complementary part of self-similar structures in the multiresolution representation. Our approach using the zerotree data structure differs from the previous ones mainly in two aspects.

First, the way we handle the zerotree root symbol is different from other proposals. Given the significance map $\chi_{l,k}$ ($l > 1$), we examine for each insignificant coefficient (which is not part of a ZT found on a previous level) whether it is a ZT root or not. If it is, we assign a "0"-symbol, else we put in a "1"-symbol (the symbol used for significant coefficients) and move the so-called *isolated zero* to the magnitude map $\hat{c}_{l,k}^{\mathrm{mag}}$.

As a result of this zerotree analysis we can replace $\chi_{l,k}$ and $\hat{c}_{l,k}^{\mathrm{mag}}$ by a binary-valued *zerotree map* $\chi_{l,k}^{\mathrm{ZT}}$ indicating the positions of ZT roots and its complementary part, i.e. coefficients which are not part of a ZT, together with a *modified magnitude map* $\hat{c}_{l,k}^{\mathrm{mm}}$ which includes isolated zeros. This leads to a more compact representation of insignificance at the expense of an enlarged magnitude map.

The second and main distinction from other zerotree-based methods concerns the use of the zerotree instrument. A careful experimental examination of the efficiency of zerotree-based coding in conjunction with conditional coding (see below) has shown that there is no further advantage of using the zerotree root symbol in bands $W_{l,k}$ below the maximum decomposition level $l < l_{max}$. Note that the number of zero coefficients aggregated in one zerotree root symbol at level $l > 1$ is given by $\sum_{k=0}^{l-1} 4^k$, so that given $l_{max} = 3$ and $l < 3$, a zerotree root contains less than 6 zero coefficients thereby producing an overhead of mispredicted isolated zero coefficients. Since the interband correlation mostly competes with a strong intraband correlation which can be efficiently absorbed by using conditional coding, it is intuitively clear that there is a diminishing benefit using the zerotree symbol.

Having confined the zerotree coding to the low-frequency subbands with $l = l_{max}$, we further improve the coding efficiency in inter-frame mode by connecting the root symbols in bands with indices $k = 1 \ldots 3$ to a zero coefficient in the lowest frequency band ($k = 0$) related to the same spatial location. This procedure allows to aggregate 64 zero quantized coefficients in one root symbol, so-called *integrated zerotree root* (IZT), if for example $l_{max} = 3$ is chosen. Note that an IZT has a close relation to a "block" of the original P-frame in the spatial domain with low energy.

3) Conditional Coding: For encoding of binary-valued indicator maps we can use all sorts of coding methods developed for the (lossless) compression of bilevel images. Run-length coding, as mentioned above, is one possibility which has a limited compression potential since it is not capable to capture the 2D correlations to a large extent. Alternatively, our approach is based on a model

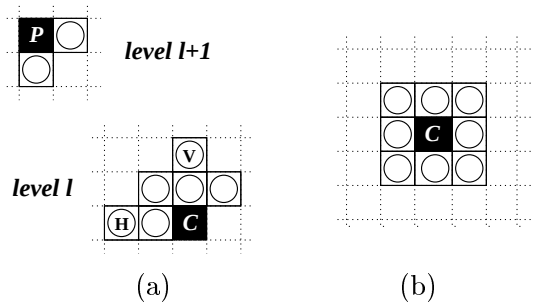


Fig. 4: (a) 7-element orientation-adaptive two-scale template (white circles) for context-based coding of the zerotree map. V and H indicate adaptive elements used for vertical and both horizontal and diagonal bands, respectively. (b) 8-neighborhood used for conditioning of C

using conditional probabilities where the conditioning "context" is created with the help of a so-called *template*. A template is usually made up of neighboring elements of the current element to encode.

Fig. 4 (a) shows the template, we have designed for the purpose of coding lower levels ($l < l_{max}$) of the zerotree map $\chi_{l,k}^{\mathrm{ZT}}$. It is similar to the differential-layer template used in the JBIG standard and covers elements of two levels: 5 surrounding elements of the current one (C) and two neighbors of the "parent" P of C, which allow a "prediction" of the non-causal neighborhood of C. In addition, we adapt the template to the orientation k of the band by choosing one of its elements (V, H) according to the direction of predominant correlations (cf. Fig. 4 (a)).

For zerotree maps on upper levels ($l = l_{max}$, $k > 0$) except for the lowest band, we choose a template consisting of the four nearest neighbors of the causal neighborhood. The processing of the lowest frequency band depends on the intra/inter-frame decision. In intra-frame mode, we suppose to have only nonzero coefficients, so that there is no need for coding a significance map. For inter-frames, however, we extend the four-element template by the related element of the zerotree-map of the previous P-frame together with a binary element indicating whether the motion vector of the related block was chosen to be the zero vector. In Fig. 3 this technique is illustrated by a small "branch" with a delay prior to conditioning of the elements of the significance map which allows an access to the significance map as well as to the motion vector field of the previous frame. One one hand this mechanism connects a potential IZT event representing a spatial domain "block" in the current P-frame with the "activity" of the related spatial location in the previous P-frame. On the other hand, the motion vector related to the same spatial location allows some kind of anticipation of an IZT event.

Further improvements on coding efficiency we have achieved by using conditional probabilities to encode the (modified) magnitude map. Since we established a definite order of processing the subsources band-wise by first coding

(and decoding) the zerotree map, we can use this (at the decoder) available information for the construction of conditioning categories. Fig. 4 (b) shows the 3×3-window of a current significant coefficient $c = \hat{c}^{\mathrm{mm}}_{l,k}[n,m]$ which is "mapped" on the corresponding part of the zerotree map in order to define the conditioning descriptor $\kappa = \kappa_{l,k}[n,m]$ of c

$$\begin{aligned} \kappa \;=\; & \chi[n-1,m] + \chi[n,m-1] \\ & + \chi[n+1,m] + \chi[n,m+1] \\ & + ((\chi[n-1,m-1] \vee \chi[n+1,m+1]) \\ & \wedge (\chi[n+1,m-1] \vee \chi[n-1,m+1]))\,, \end{aligned} \tag{4}$$

where $\chi = \chi^{\mathrm{ZT}}_{l,k}$ ($\vee, \wedge$: logical "or" resp. "and"). To reduce the overall learning cost in the adaptive arithmetic coder, we restrict the number of conditioning categories to six states characterized by the r.h.s of Eq. (4) which we found experimentally to yield best performance.

Our analysis of the sign map $\varsigma_{l,k}$ has shown that there are locally extended regions of constant sign and characteristic patterns of sign changes. These sign changes are usually due to edges having an orientation with a strong bias towards the orientation k of a given band. This observation motivates the use of a second-order model relating an actual sign to be encoded to a context built of its two preceding signs (relative to the given band orientation).

For encoding the lowest frequency subband of intra-frames, the models described above do no longer fit to the actual statistics which is similar to that of the original input frame. Thus, we use here a DPCM-like procedure with an adaptive Graham-predictor [7] and a backward driven classification of the prediction error contexts with a six state model.

D. Arithmetic Coding

The motion vectors and all symbols generated by the pre-coder are encoded using an adaptive arithmetic coder (AAC). Actually, we use a variation of the implementation given in [26] by restricting the AAC to binary alphabets. Multialphabet symbols like magnitudes of coefficients or motion vectors are first mapped to binary symbols with lengths proportional to their (expected) probability distribution thereby allowing a faster adaption of the models to the actual statistics. Moreover, it keeps the option of using a fast QM-coder [1] instead of our binary AAC for real-time applications.

For intra- and inter-frame mode we use separate models. Consecutive P-frames as well as consecutive motion vector fields are encoded using the updated related models of the previous P-frame and motion vector field, respectively. Note that for a 10-band decomposition of a YUV color representation there could be up to $3 \cdot 10 \cdot 2^7 = 3840$ different models for encoding of significance maps using the 7-element template, for example (cf. Fig. 4).

Bit Rate (bpp)	Lena			Goldhill		
	S&P	SFQ	PACC	S&P	SFQ	PACC
1.0	40.41	40.52	40.52	36.55	36.70	36.90
0.5	37.21	37.36	37.42	33.13	33.37	33.45
0.25	34.11	34.33	34.40	30.56	30.71	30.81
0.2	33.15	33.32	33.45	29.85	29.86	30.09

Table 1: Still Image Coding Results

IV. Experimental Results

This section presents results comparing our PACC-coder to current state-of-the-art coders. We first apply our algorithm to two well-known 512×512 standard test images in order to evaluate the still image compression performance of our coder. For video coding we report the results of two experiments using the four MPEG4 test sequences Akiyo, Hallmonitor, News, and Foreman in *QCIF*-format (176×144 pels) and 4:1:1 YUV color representation with 30 frames/sec. These test sequences are characteristic for a wide range of possible video coding applications such as video telephony, remote monitoring and video broadcasting.

First experiment is devoted to intra-frame coding and shows the performance of our coding algorithm in comparison with two methods of the emerging MPEG4 standard. Coding of entire sequences is the topic of our second experiment which finally demonstrates the performance of the wavelet-based PACC video codec compared to the MPEG4 DCT-based method at very low bit rates.

A. Intra-frame and Still Image Coding

We begin this section by briefly reporting results obtained by applying our algorithm to still images when restricted to intra-frame mode. In Table 1 our results for monochrome 512×512 standard Lena and Goldhill images are compared against those reported for the SFQ-coder by Xiong et al.[28] and the embedded coder of Said and Pearlman (S&P) [17], the best performing zerotree-based still image coders found in the literature.

The PACC-coder outperforms both the SFQ-coder and the S&P-coder at all bit rates at least in terms of peak signal-to-noise ratio (PSNR). The gain in objective performance is up to 0.35 dB PSNR. Note that, although all three algorithms are using the same filter (9/7-tap biorthogonal wavelet [5]), their computational requirements are very different. While the SFQ coding method is based on complex on-line computations to determine rate-distortion optimal quantization and coding parameters, the computational complexity of the PACC- and S&P-coder is rather modest.

(a) (b) (c)

Fig. 5: Part of GOLDHILL: (a) Original (8 bpp), Reconstructions of (b) S&P-Coder[17] (33.13 dB PSNR) and (c) PACC-Coder (33.45 dB PSNR), both at 0.5 bpp.

Fig. 5 gives an impression of the difference in subjective picture quality between the S&P-coded and the PACC-coded GOLDHILL image, both at 0.5 bpp. Notice that the structure of the roof is much better preserved in the reconstructed image obtained by the PACC-coder (Fig. 5 (c)).

In Table 2 we present results for coding of the first frame of three MPEG4 test sequences. We compare our results with those obtained with an improved DCT-based method [2] implemented in MPEG4-VM (Version 5.1) and with the zerotree-based texture coding method of MPEG4 (ZTE) [3, 14].

Intra-frame coding of PACC outperforms that of both VM and ZTE by 0.9–2.2 dB PSNR for the luminance component (Y) and 0.3–2.8 dB for the average PSNR of the two chrominance components (C). Note that the implementation of VM 5.1 already operates with an improved intra-frame coding method which compared to H.263 produces about 20% lower bit rates for the same quality due to better VLC-tables and DC/AC-prediction [2]. The subjective quality of the PACC reconstructions is much better compared to those of VM and ZTE. The DCT-based scheme produces annoying blocking artifacts especially at low rates while the wavelet typical ringing artifacts are far less pronounced in the PACC coded images than in those of ZTE at the same bit rate.

B. Video Coding

Coding results of our final experiment are shown in Tab. 3. Since we do not use a bit rate control, all results were obtained by adjusting the quantization parameters to meet a specific bit rate within a margin of 1–2%. We choose identical quantization step size for intra- and inter-frame coding (both for VM and PACC). The desired frame rates were obtained by temporally subsampling the original sequences. All results provided by Tab. 3 are averaged over 10 seconds of video, including the first frame. For the runs of VM, the advanced prediction mode, the bidirectional prediction mode and the use of the deblocking filter have

Sequence	kbit	Y/C	VM	ZTE	PACC
AKIYO	20.5	Y	37.81	37.54	39.75
		C	40.65	40.18	41.76
	31.5	Y	41.49	41.55	43.72
		C	43.58	42.58	45.34
HALL	20.9	Y	35.09	34.80	36.55
		C	39.78	39.77	40.23
	28.5	Y	37.82	37.87	39.53
		C	41.16	40.73	42.14
NEWS	20.8	Y	32.91	32.39	33.80
		C	37.32	36.39	37.71
	30.0	Y	36.13	35.50	37.41
		C	39.53	38.69	40.33
FOREMAN	19.3	Y	33.95	33.24	35.16
		C	40.56	39.95	40.86
	28.7	Y	37.11	36.32	38.57
		C	42.08	42.34	43.57

Table 2: Intra-frame Coding Results

been disabled. Both coding schemes are operating with a full and unrestricted motion vector search. The PACC algorithm was implemented with a maximum decomposition level of $l_{max} = 3$ both for luminance and chrominance components.

The coding results summarized in Tab. 3 shows that the PACC algorithm achieves in nearly all cases a significantly higher performance in terms of average PSNR compared to the MPEG4-VM. Improvements for AKIYO at 10 and 24 kbit/s are 0.57 dB and 0.98 dB for the luminance (Y), respectively. Fig. 10 (a) shows the detailed results for each coded frame of AKIYO at 24 kbit/s with a consistent superior PSNR performance of the PACC coder. This fact is also reflected in the visual quality of the reconstructed video. Fig. 7 shows a zoomed part of a single reconstructed frame both of the VM-coder and of the PACC-coder. The VM reconstruction is less detailed and suffers from blocking artifacts, while the PACC reconstruction appears more natural with less high-frequency noise patterns.

The (objective) performance of both methods for HALLMONITOR at the challenging rate of 10 kbit/s is approximately identical. Subjectively, reconstructed HALLMONITOR sequence of both VM and PACC suffers from severe visual degradation. The VM reconstruction shows blocking artifacts in the foreground and noise patterns around the moving persons, while the PACC reconstruction suffers from ringing artifacts at sharp object boundaries.

Sequence	kbit/s	fps	Y/C	VM	PACC	Gain
AKIYO	10.45	7.5	Y	34.28	34.85	0.57
			C	38.06	38.46	0.40
	24.15	10	Y	37.38	38.36	0.98
			C	40.54	41.48	0.96
HALL	10.20	7.5	Y	29.96	30.10	0.14
			C	37.68	38.00	0.32
	25.90	10	Y	33.75	34.51	0.76
			C	38.87	39.37	0.50
NEWS	49.10	7.5	Y	34.67	35.59	0.92
			C	38.54	38.98	0.44
	108.3	15	Y	36.91	38.15	1.24
			C	40.28	40.94	0.66
FOREMAN	50.30	7.5	Y	32.03	32.91	0.88
			C	37.96	38.12	0.16
	123.7	15	Y	34.29	35.16	0.87
			C	39.59	40.14	0.55

Table 3: Coding Results for Entire Sequences.

For the more complex NEWS sequence coded at 49 kbit/s and 7.5 fps, we have an average PSNR improvement of 0.82 dB PSNR and the plot in Fig. 10 (b) shows that the performance gap is nearly constant over the entire sequence. Although the DCT-based coder is operating in partial, i.e. block-wise intra-mode at the three scene cuts in the background, the relative gain in performance there is rather small. Comparing the visual quality of a particular reconstructed frame (no. 96) of the NEWS sequence coded by VM and PACC, it can be observed that the PACC reconstruction appears again more natural with a better preserving of image details (see Fig. 8). A large coding gain of 1.24 dB PSNR is achieved for NEWS at 108.3 kbit/s and 15 fps. Although these conditions are no longer in the range of "very low bit rate coding", it is interesting to note that the performance gain of PACC over MPEG4-VM increases when going to higher rates. This fact can also be observed in Fig. 6, which shows rate-distortion curves for AKIYO and HALLMONITOR at the fixed frame rate of 10 fps. At a bit rate of 40 kbit/s, the quality of the AKIYO reconstruction obtained by the PACC-coder is almost perfect with a coding gain of 1.5 dB PSNR compared to the VM at the same rate.

The experimental results for the very active FOREMAN sequence show that the good performance of our proposed method is not confined to a special class of video sources with (partial) static background. The gain in PSNR we achieved

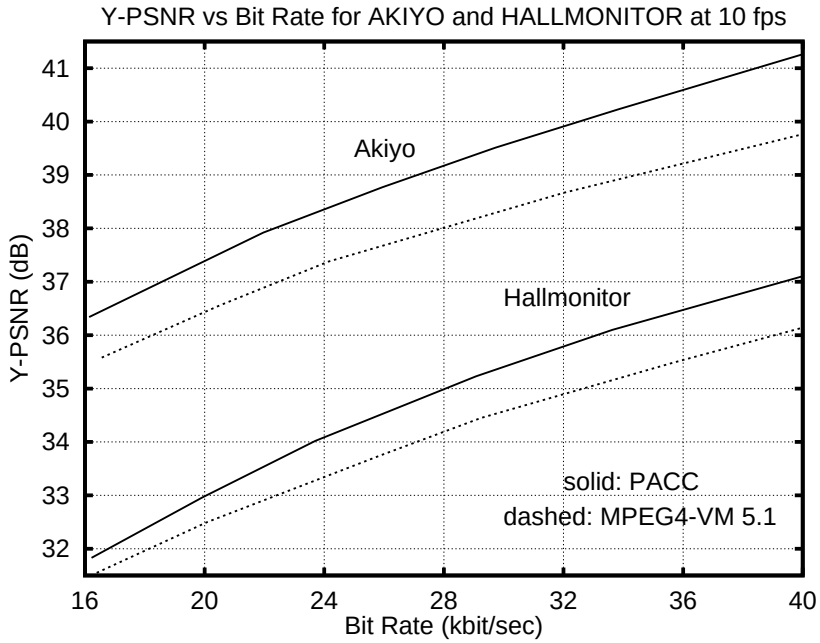


Fig. 6: Comparison of PACC and MPEG4-VM for AKIYO and HALLMONITOR.

for FOREMAN at 50 kbit/s and 7.5 fps is as much as 0.88 dB. The visual improvements are quite obvious, when comparing a sample reconstruction of both coding schemes at this bit rate (cf. Fig. 9).

V. CONCLUSION

In this paper we presented a video and still image coding algorithm with a new (pre-)coding strategy involving the concepts of partitioning, aggregation and conditional coding (PACC). The coder is optimized in the pre-coding part only, i.e. we used a standard DWT, simple uniform quantization and a conventional arithmetic coder as well as a slightly modified standard motion compensation technique. We could show that this wavelet-based coder is highly efficient in a quality vs. bit rate sense both for intra-frame/still image coding and for video coding. Our coding results for four MPEG4 test sequences demonstrate that the PACC coder achieves better performance than MPEG4-VM. Future research will be related to the incorporation of adaptive wavelet (packet) transforms and better motion models.

ACKNOWLEDGMENT

We would like to thank M. Palkow and A. Haderer for their contributions to the video coding simulation experiments.

(a) (b)

Fig. 7: Part of reconstructed frame 258 of AKIYO at 24 kbit/s and 10 fps: (a) MPEG4-VM, (b) PACC.

REFERENCES

[1] Document ISO/IEC JTC1/SC2, "Progressive bi-level image compression", ISO Standard CD11544, Sep. 1991.

[2] Document ISO/IEC JTC1/SC29/WG11 MPEG96/M1320, "Description and Results of Coding Efficiency Experiment T9", Chicago MPEG meeting, Sept. 1996.

[3] Document ISO/IEC JTC1/SC29/WG11 MPEG96/M1869, "Report of Core Experiment T1: Wavelet Coding of Intra Frames", Sevilla MPEG meeting, Feb. 1997.

[4] ITU-T Draft Recommendation "Video Coding for Low Bitrate Communication", Dec. 1995.

[5] A. Cohen, I. Daubechies and J.-C. Feauveau, "Biorthogonal Bases of Compactly Supported Wavelets", *Comm. on Pure and Appl. Math.*, Vol. 45, pp. 485–560, 1992.

[6] T. Ebrahimi, E. Reusens and W. Li, " New Trends in Very Low Bitrate Video Coding", *Proceedings of the IEEE*, Vol. 83, pp. 877–891, June 1995.

[7] R. E. Graham, "Predicitve Quantizing of Television Signals", *IRE Wescon Convention Record*, Vol. 2 (4), pp. 147–157, 1958.

[8] Y. Huang, H. M. Dreizen and N. P. Galatsanos, "Prioritized DCT for Compression and Progressive Transmission of Images", *IEEE Trans. on Image Proc. (IP)* , Vol. 2, No. 4, pp. 477–487, Oct. 1992.

[9] G. G. Langdon and J. J. Rissanen, "Compression of Black-White Images with Arithmetic Coding", *IEEE Trans. on Comm.*, Vol. 29, No.6, pp.858–867, 1981.

[10] A. Lewis and G. Knowles, "Image Compression Using the 2D Wavelet Transform", *IEEE Trans. on IP*, Vol. 1, No. 2, pp. 244–250, April 1992.

[11] S. Mallat and F. Falzon, "Understanding Image Transform Codes", *Proc. SPIE Aerospace Conf.*, Orlando, April 1997.

[12] H. S. Malvar and D. H. Staelin, "Reduction of Blocking Effects in Image Coding with a Lapped Orthogonal Transform", *IEEE Proc. ICASSP*, pp. 781–784, 1988.

(a) (b)

Fig. 8: Part of reconstructed frame 96 of NEWS at 49 kbit/s and 7.5 fps: (a) MPEG4-VM, (b) PACC.

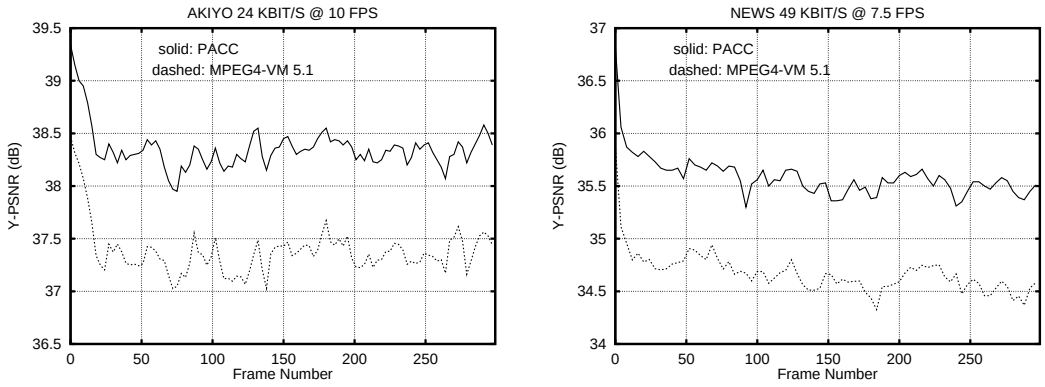
(a) (b)

Fig. 9: Part of reconstructed frame 4 of FOREMAN at 50 kbit/s and 7.5 fps: (a) MPEG4-VM, (b) PACC.

[13] D. Marpe and H. L. Cycon, "Efficient Pre-Coding Techniques for Wavelet-Based Image Compression", *Proceedings Picture Coding Symposium 1997*, pp. 45–50, 1997.

[14] S. A. Martucci, I. Sodagar, T. Chiang and Y. Zhang, "A Zerotree Wavelet Video Coder", *IEEE Trans. on Circuits and Systems for Video Technology (CSVT)*, Vol. 7, No. 1, pp. 109–118, Feb. 1997.

[15] J. R. Ohm, "Three-dimensional Subband Coding with Motion Compensation", *IEEE Trans. on Image Proc. (IP)*, Vol. 3, No. 5, pp. 559–571, Sep. 1994.



(a) (b)

Fig. 10: Frame vs. Y-PSNR: I- plus P-frame coding of (a) AKIYO at 24 kbit/sec with 10 frames/sec and (b) NEWS at 49 kbit/sec with 7.5 frames/sec

[16] C. I. Podilchuk, N. S, Jayant, N. Farvardin, "Three-dimensional Subband Coding of Video", *IEEE Trans. on IP*, Vol. 2, No. 2, pp. 125–139, Feb. 1995.

[17] A. Said and W. Pearlman, "A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees", *IEEE Trans. on CSVT*, Vol. 6, No. 3, pp. 243–250, June 1996.

[18] D. G. Sampson, E. A. B. da Silva, M. Ghanbari, "Low Bit-rate Video Coding using Wavelet Vector Quantisation", *IEE Proc.-Vis. Image Signal Processing*, Vol. 142, No. 3, pp. 141–148, June 1995.

[19] J. M. Shapiro, "Embedded Image Coding Using Zerotrees of Wavelet Coefficients", *IEEE Trans. on Signal Proc.*, Vol. 41, No. 12, pp. 3445–3462, Dec. 1993.

[20] T. Sikora, "The MPEG-4 Video Standard Verification Model", *IEEE Trans. on CSVT*, Vol. 7, No. 1, pp. 19–31, Feb. 1997.

[21] D. Taubman and A. Zakhor, "Multirate 3-D Subband Coding of Video", *IEEE Trans. on Image Proc. (IP)*, Vol. 3, pp. 572–588, Sep. 1988.

[22] J. D. Villasenor, B. Belzer, and J. Liao, "Wavelet Filter Evaluation for Image Compression", *IEEE Trans. on IP*, Vol. 4, No. 8, pp. 1053–1060, Aug. 1995.

[23] Y. Wang and E. Salari,"The Post Wavelet Transform Redundancy and its Reduction Techniques for Image Compression", *Proc. SPIE*, Vol. 2418, pp.164–173, 1995.

[24] H. Watanabe and S. Singhal, "Windowed motion compensation", *Proc. SPIE* Vol. 1605, pp.582–589, Nov. 1991.

[25] A. B. Watson, G. Y. Yang, J. A. Solomon and J. Villasenor, "Visibility of Wavelet Quantization Noise", to appear in *IEEE Trans. on IP*, 1997.

[26] I. Witten, R. Neal and J. Cleary, "Arithmetic Coding for Data Compression", *Comm. ACM*, Vol. 30, pp. 520–540, June 1987.

[27] R. W. Young and N. G. Kingsbury, "Frequency-Domain Motion Estimation Using a Complex Lapped Transform", *IEEE Trans. on IP*, Vol. 2, No. 1, pp. 2–17, Jan. 1993.

[28] Z. Xiong, K. Ramchandran and M. T. Orchard, "Space-frequency Quantization for Wavelet Image Coding", to appear in *IEEE Trans. on IP*, 1997.

[29] Y. Q. Zhang and S. Zafar, "Motion-Compensated Wavelet Transform Coding for Color Video Compression", *IEEE Trans. on CSVT*, Vol. 2, No. 3, pp. 285–296, Sept. 1992.