

Flexible Motion-Adaptive Video Coding With Redundant Expansions

Adel Rahmoune, Pierre Vanderghyest, *Member, IEEE*, and Pascal Frossard, *Senior Member, IEEE*

Abstract—This paper presents a highly flexible video coding scheme, based on the use of a redundant dictionary of spatio-temporal three-dimensional (3-D) functions. Directionality and anisotropic scaling are key ingredients to the spatial components, which form a rich collection of two-dimensional (2-D) visual primitives. The temporal component is tuned to capture most of the energy in the temporal signal evolution, along motion trajectories in the video sequence. The video coding scheme (MP3D) first computes motion trajectories that are eventually entropy coded and sent as side information to the decoder. It then applies a spatio-temporal decomposition along motion trajectories, using an adaptive approximation algorithm based on matching pursuit (MP). Quantized coefficients and basis function parameters are entropy-coded in a embedded stream that is constructed to respect multiple rate constraints. The geometric properties of the 2-D primitive dictionary allow for flexible spatial resolution adaptation, so that the flexible MP3D stream enables decoding at different spatio-temporal resolutions, and multiple rates. The MP3D scheme is shown to provide rate-distortion performance that are comparable with state-of-the-art schemes, such as H.264, MPEG-4, at low and medium bit rate. However, the use of a redundant dictionary is penalizing at high coding rate, which makes the MP3D algorithm mostly interesting for low rate applications, or as a flexible base layer in hierarchical coding schemes.

Index Terms—Matching pursuit, redundant expansion, scalability, spatio-temporal atoms, video coding.

I. INTRODUCTION

HIGH scalability video coding is becoming a stringent requirement for video streaming and delivery over the Internet and heterogeneous networks. These networks are characterized by a fluctuating channel capacity and a wide range of clients with different computational and display capabilities. Hence, the goal of highly scalable video coding is to generate a single embedded bit stream that can be decoded efficiently at different bit rates and at various resolutions. This framework imposes strong constraints on the encoder: it has to operate without prior knowledge about the specific bit rate and the format at which the compressed video will be decoded. For this reason, video coding algorithms based on the predictive feedback approach, which combines motion compensation and a discrete cosine transform (DCT) transform, fail to achieve high scalability. An alternative method is to use a feed-forward or an

open-loop approach, where a spatio-temporal transform is followed by embedded quantization and coding.

Most scalable three-dimensional (3-D)-based approaches employ a separable two-dimensional (2-D) discrete wavelet transform (DWT) for the spatial information, and DWT with either a transversal or lifting implementation along motion trajectories. Recently however, it was pointed out that the separable 2-D wavelet transform is not ideally suited for representing images as it fails to capture regular geometric features (e.g., edges) [1].

In this paper, we elaborate a new 3-D coding method that tries to overcome these limitations. Stepping upon previous work on low bit rate image and video coding using redundant dictionaries [2]–[4], we introduce a spatio-temporal representation that consists in applying a sparse decomposition algorithm along motion trajectories. On the algorithmic point of view, using redundant dictionaries still represents quite a challenge. Nevertheless, the recent advances related to sparse approximations motivate the use of the pure greedy algorithm (also known as matching pursuit (MP) [5]) as a decomposition strategy for the video sequence. The MP3D scheme consists of two modules: 1) motion-adaptive 3-D transform and 2) embedded quantization and coding. The motion-adaptive 3-D transform algorithm comprises two steps, which are the motion trajectory prediction and MP decomposition. The embedded bitstream is generated through a rate-constrained quantization scheme, which is applied to the transform coefficients. Then, these quantized coefficients are compressed with the adaptive arithmetic coding, along with atom indexes. Finally, motion parameters are entropy-coded and sent as side information to the decoder. It has been found that at low and medium bit rate, (i.e., less than 500 kbps), the obtained compression performance of MP3D are very comparable to the state-of-the-art nonscalable coders H.264 and MPEG-4, and to the scalable schemes MC-EZBC and H.264 with scalability extensions (denoted in this paper as H.264/Ext). However at higher bit rates, our redundant dictionary approach becomes less efficient in terms of compression efficiency when compared to rate-distortion behavior.

An overview of the state-of-the-art 3-D-based video coding approaches is given in Section II. In Section III, we review some of the recent results, which are related in signal approximation with redundant dictionaries, and present an overview of the MP3D coding scheme is given in Section III-C. The motion-adaptive 3-D transform is presented in details in Section IV. Then, the generation of an embedded scalable bit stream is presented in Section V. Experiments carried out on standard test sequences illustrate the compression efficiency of MP3D in Section VI. Section VII highlights the scalability features of the MP3D coding scheme, which are the resolution and SNR scalability properties. Finally, conclusions and perspectives are provided in Section VIII.

Manuscript received August 8, 2004; revised May 2, 2005. This work was supported in part by the Swiss National Science Foundation under Grants PP002-68737 and 2001-066912.01. This paper was recommended by C.-W. Chen.

The authors are with the Signal Processing Institute ITS, Swiss Federal Institute of Technology, EPFL CH-1015 Lausanne, Switzerland (e-mail: adel.rahmoune@epfl.ch; pierre.vanderghyest@epfl.ch; pascal.frossard@epfl.ch).

Digital Object Identifier 10.1109/TCSVT.2005.859932

II. RELATED WORK

Most successful scalable video coding schemes are based on 3-D separable DWT. Karlsson and Vetterli initiated the use of DWT for subband video compression [6]. Then, in the method proposed by Ohm [7], a translational model is assumed for motion, where the video frames are divided into blocks that undergo a translation or a rigid motion. A separable 3-D wavelet transform is then applied on the displaced blocks. However, the effects of contraction and expansion in the motion field are observed by the appearance of disconnected pixels between the blocks. These disconnected pixels are handled differently, in order to make the transform invertible, which affects the overall coding performance. In the scheme proposed by Taubman and Zakhor [8], a warping operator is used in order to align the frames in the direction of motion prior to applying the 3-D transform. However, only global camera pan is treated in their warping operator in order to make it invertible. Kim and Pearlman [9] uses a 3-D separable transform, by extending the successful 2-D wavelet-based SPIHT algorithm [10] into the temporal dimension. However, without motion compensation, it produces some annoying ghosting artifacts at low bit rate, because of the temporal filtering. This motivates the exploitation of motion within the spatio-temporal transform. Subsequently, Choi and Woods [11] propose a motion-compensated 3-D scheme where a hierarchical block matching motion estimation algorithm with pruning is used with half-pel precision for displacements. In the same manner as in [7], this scheme applies a special processing for disconnected pixels, which enhances coding efficiency. In the same framework, we can also cite the 3-D embedded subband coding with optimized truncation (3-D ESCOT) [12], which has been proposed by Jizheng *et al.* In this scheme, a wavelet decomposition is first applied along motion trajectories, followed by a DWT decomposition in the spatial dimensions. It is worth noting that both warping-based and block displacement-based methods often use Haar wavelets. When longer filters are selected, they do no bring any significant improvement in terms of coding efficiency.

Recently, 3-D wavelet based methods have been redesigned by employing the lifting scheme in the temporal dimension in order to have perfect reconstruction regardless of the motion model. The LIMAT scheme [13] employs a lifting implementation of the DWT, in which each lifting step is compensated for the estimated scene motion. Mesh-based motion estimation algorithms are shown to perform better than hierarchical block-based ones. Another lifting-based scheme that satisfies the invertibility property has been presented in [14].

Finally, *in-band* motion-compensated temporal filtering (IBMCTF) [15], has been proposed in order to overcome the problem of the shift-variance property of the critically sampled wavelets, encountered when applying the MCTF operation in the spatial subbands. In fact, the IBMCTF performs the spatial decomposition of the image sequence first through the DWT, then it applies the MCTF operation in each subband. A shift-invariant wavelet representation is thus constructed by using a *complete-to-overcomplete* discrete wavelet transform in order to build a wavelet representation that is overcomplete.

This was shown to improve the scalability and the efficiency of video compression.

In summary, most of the existing scalable video compression schemes employ a separable 2-D wavelet transform for the spatial information and a temporal DWT along the motion trajectories with either a transversal or a lifting implementation. Our approach differs from the afore-mentioned schemes in that it uses an *overcomplete* dictionary for *adaptive* signal decomposition in both the spatial and the temporal dimensions. In other words, this adaptive signal decomposition provides a sparse representation, consisting of *only* the atoms that match well the signal components. Meanwhile, the remaining atoms in the dictionary are simply ignored in this signal decomposition. This method allows to achieve competitive performance at low and medium bit rate while ensuring scalability and flexibility of the compressed bit streams, which become a series of geometric features. These nice properties however usually come at a price, which is a higher computational complexity.

III. MP EXPANSION OF IMAGE SEQUENCES

A. Benefits of Nonlinear Sparse Approximations

Most acclaimed technical solutions to both image and video compression, namely the JPEG2000 and MPEGx/H.26x families of standards, rely on the transform coding. Moving to the transform domain is classically performed in order to obtain decorrelated sets of coefficients on which scalar quantization and entropy coding is performed. The choice of the transform is thus driven by its de-correlating performances as well as good properties under quantization and ease of entropy coding. Most techniques use two well controlled orthonormal basis (ONB): DCT and wavelets. Performing the transform by means of an ONB allows for the use of well studied data compression results, and in both cases fast algorithms help keeping a low complexity algorithm. Unfortunately, restricting a representation to an ONB fixes a very rigid structure on the components of the signals that are represented, and sometimes dramatically damages the coherence and quality of important visual primitives. This results in annoying artifacts at low bit rates on textures and edges. To cope with these problems, an interesting line of research consists in representing the image with a transform whose building blocks match important signal structures. Unfortunately, the price to pay for such a freedom is that no genuine ONB can be used and a new coding paradigm has to be adopted. In the following, we will basically derive a coding scheme that tries to preserve predefined structures in a sequence of frames. More specifically we view such a sequence as a 3-D space-time signal $I(x, y, t)$ and we will try to efficiently encode coherent spatio-temporal structures.

The approach we have chosen relies on expanding the signal as a linear superposition of N generalized waveforms g_γ , tuned to match the signal structures \mathcal{D}

$$I = \sum_{i=0}^{N-1} c_i g_{\gamma_i}. \quad (1)$$

These waveforms are selected among a vast library $\mathcal{D} = \{g_\gamma, \gamma \in \Gamma\}$ with the only constraint that \mathcal{D} is dense in the space of finite energy signals. In the following, we will

refer to g_γ as an atom and to \mathcal{D} as a dictionary. The parameter set Γ usually carries important information about the atoms, for example space and frequency localization. Of course, we would like also that the necessary parameters in this expansion, namely the set of coefficients c_i and indexes γ_i , yield good compression performances; this mainly leads to a generic requirement about (1), namely that this expansion is sparse enough.

Without more constraints on \mathcal{D} , and in particular if it is not an ONB, there is generally no unique solution to (1). One possible solution can be to look for the sparsest exact expansion, that is minimizing the number of coefficients in (1). This unfortunately leads to an NP hard, combinatorial optimization problem [16]. A close solution may be provided by relaxing this problem and trying to minimize the ℓ^1 norm of the coefficients; it leads to the Basis Pursuit algorithm, deeply studied by Donoho and collaborators [17]. Interestingly, this algorithm sometimes provides the optimal sparsest solution of (1) with particular dictionaries [18]–[20]. Alternatively, the MP algorithm [5] solves (1) by iteratively decomposing the signal using a greedy strategy. Starting with $R_0 = I$, the n th iteration reads

$$R_n = \langle R_n, g_{\gamma_n} \rangle g_{\gamma_n} + R_{n+1} \quad (2)$$

where the atom g_{γ_n} is the one having maximum correlation with R_n . It is given by

$$g_{\gamma_n} = \arg \alpha \max_{\mathcal{D}} |\langle R_n, g_\gamma \rangle| \quad (3)$$

where α is a positive constant that depends on the search strategy and is equal to 1 in the exhaustive search case. After N steps, MP yields a sparse approximation

$$I = \sum_{i=0}^{N-1} \langle R_i, g_{\gamma_i} \rangle g_{\gamma_i} + R_N \quad (4)$$

where R_N is the residual error. MP converges [21], that is $\|R_N\| \rightarrow 0$ when N tends to infinity and converges even exponentially in finite dimension [5]. It can be bounded as

$$\|R_N\|_2^2 \lesssim (1 - \alpha^2 \beta^2)^N \quad (5)$$

where β is a constant that solely depends on \mathcal{D} and is getting close to 1 when the redundancy increases. Recently, more constructive results have been obtained concerning the approximation properties of greedy algorithms [20] but their description is beyond the scope of this paper. As already shown in [3], MP is particularly well suited for low rate and adaptive coding of visual information because it easily yields scalable streams by a simple truncation. A good approximation is obtained with few well chosen components, mostly because MP will first pick the most prominent signal structures in the dictionary. This property makes it particularly useful at low and medium bit rates.

B. The 3-D Spatio-Temporal Overcomplete Basis

The video sequence is approximated in terms of a series of 3-D spatio-temporal atoms. These atoms, which consist of separable spatial and temporal components, should have a structure that is able to efficiently represent the spatial image content, as well as temporal evolution along motion trajectories. Many approaches have been proposed to improve image representation (e.g., curvelets [1], bandelets [22], contourlets [23])

and all underline that an efficient image representation should have the following properties: 1) multiresolution; 2) localization (the basis functions should be localized in space and frequency); 3) directionality (the basis functions should have different orientations); and 4) anisotropy (the basis functions should have a variety of elongated shapes with different scale ratios).

As for the spatial part of our dictionary, we use the same construction as proposed in [3], which we sketch here for the sake of completeness. Two spatial mother atoms have been proposed, satisfying *the localization* property, a 2-D Gaussian g_1 , an edge-like function g_2 built on a Gaussian, and its second partial derivative in the orthogonal direction

$$g_1(x, y) = \frac{1}{\sqrt{\pi}} e^{-(x^2+y^2)} \quad (6)$$

$$g_2(x, y) = \frac{2}{\sqrt{3\pi}} (4x^2 - 2) e^{-(x^2+y^2)}. \quad (7)$$

The 2-D Gaussian is used in order to extract the low frequency components and to generate a coarse approximation of image content. Whereas the motivation behind using the second partial derivative of Gaussian, besides the localization property, is the need to have a function that efficiently captures image singularities like edges and contours.

The overcomplete spatial dictionary is spanned by shifting, orienting, and scaling the spatial mother atoms using the following unitary operators:

- *Shift*:

$$\mathcal{U}_{(x_0, y_0)} g = g((x - x_0), (y - y_0)). \quad (8)$$

- *Orientation*:

$$\mathcal{U}_\theta g_2 = g_2(r_{-\theta}(x, y)). \quad (9)$$

- *Scaling*:

$$\mathcal{U}_a g_1 = \frac{1}{a} g_1\left(\frac{x}{a}, \frac{y}{a}\right) \quad (10)$$

$$\mathcal{U}_{(a_1, a_2)} g_2 = \frac{1}{\sqrt{a_1 a_2}} g_2\left(\frac{x}{a_1}, \frac{y}{a_2}\right). \quad (11)$$

For implementation issues, spatial position (x_0, y_0) sweeps the whole image and orientation may take 32 values $\theta = i\pi/32$, where $i = 0, \dots, 31$. The scaling factor a_j , $j = 1, 2$, is logarithmically distributed as $a_j = 2^{i/2}$, with $i = 0, \dots, 2[\log(\text{Image_size}/6)]$.

The temporal functions, on the other hand, should satisfy the following objectives. They should capture most of the signal energy in the low-pass temporal frequencies with few elements, as this will reduce the ghosting artifacts at low bit rate. They should satisfy the multiresolution and the localization properties in order to approximate well the temporal evolution of the video signal. These properties are achieved by selecting a β -spline $\beta^n(t)$ function [24]. Trading-off between temporal and frequency decay, the order of $\beta^n(t)$ should be $n \geq 2$. Empirically, the 3rd order β -spline, $\beta^3(t)$, was shown to provide good approximation performances for a group of pictures (GOP) size of 16. The temporal part of the dictionary is thus generated by shifting and scaling the β -spline

$$\mathcal{T}_{t_0, s} \beta^3 = \beta^3\left(\frac{(t - t_0)}{s}\right). \quad (12)$$

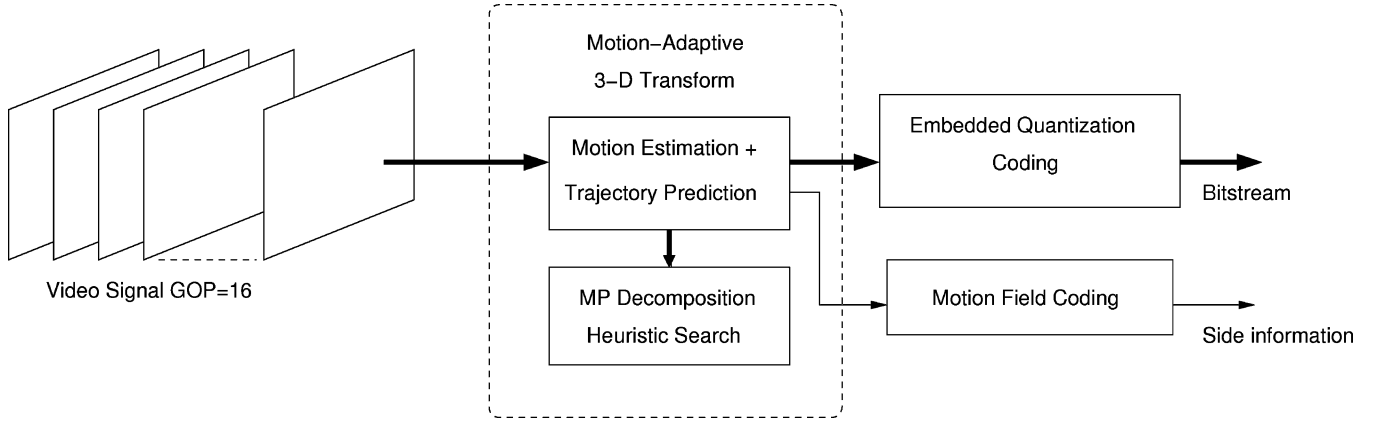


Fig. 1. Block diagram of the motion-adaptive MP encoder (MP3D).

The atom center t_0 sweeps the entire GOP size and the scale $s = 2^i$ varies according to $i = 0, \dots, \lfloor \log_2(\text{GOP}_{\text{size}}) \rfloor$. It is noteworthy that, in the temporal scale $s = 2^i$, i refers to the resolution or the number of frames that are processed in the signal. When $i = 0$, only 1 frame is processed, which can be interpreted as the existence of an abrupt motion, a scene change or an isolated feature. Whereas for $i = 1$, the support size is of 3 frames. It means that there is a smooth temporal evolution in the direction of motion localized in 3 frames. More generally, the larger the support size, the longer the motion trajectory.

To summarize, the redundant spatio-temporal dictionary is built by applying the coupled operators \mathcal{U} and \mathcal{T} on the 3-D mother atoms, along motion trajectories, in order to take advantage of the nature of the video signal.

C. Overview of the MP3D Coding Scheme

The building blocks of the novel MP video encoder proposed in this paper, are represented in Fig. 1. The MP3D coder consists of two main modules, namely 1) the motion-adaptive 3-D spatio-temporal transform and 2) the embedded quantization and coding.

The video sequence is first segmented into GOP of size N , with $N = 16$ in the remainder of this paper. The motion-adaptive 3-D transform performs a motion estimation in the GOP, in order to define the motion fields in each frame. These eventually generate motion trajectories along the successive frames of the GOP. MP is then implemented with a heuristic search algorithm. It provides a sparse representation of the video information in a series of spatio-temporal atoms, which are displaced along the motion trajectories. In a sense, this operation has the same objective as the motion-compensated temporal filtering (MCTF) [11], where the signal is filtered in the temporal dimension, along a given trajectory. Finally, the atom parameters are quantized and progressively encoded to generate a scalable video stream. Lossless coding (DPCM and arithmetic coding) is applied to the motion field parameters, which are sent as a constant rate side information layer to the decoder. The motion adaptive transform and embedded coding stages are described in details in the next sections.

IV. MOTION-ADAPTIVE 3-D TRANSFORM

A. Motion Estimation and Trajectory Prediction

A key element to efficient video coding consists in efficiently exploiting the temporal redundancy between pictures. This is generally done by motion estimation, which finds the best mapping between successive frames. The motion-adaptive 3-D spatio-temporal transform proposed in this work relies on the definition of motion trajectories that correspond to the movement of spatial atoms within the GOP.

The motion trajectories, which could be due to either local or global motion, are obtained through the estimation of the motion fields in video frames. Motion vectors (MVs), which form motion fields in a frame, are computed here using a block-matching (BM) based technique, in the backward direction. Dividing the frame domain into nonoverlapping blocks induces the smoothness of the estimated motion fields. Block matching is illustrated in Fig. 2. A block \mathcal{B}_m in the current frame i , is mapped to the best matching block \mathcal{B}'_m in the previous frame $i - 1$. Equivalently, the error between \mathcal{B}_m , and its parent in the previous frame, is minimized. The spatial displacement vector d_m between these two blocks is the MV assigned to each pixel in block \mathcal{B}_m . These MVs are then losslessly encoded and transmitted as side information to the decoder.

The 3-D atom is constructed by propagating its spatial component along the motion trajectory passing by its center in the *reference* frame, in the forward as well as the backward directions. The reference frame is chosen dynamically as the frame with the largest energy in the GOP. A motion trajectory is defined as the displacement of a pixel, or group of pixels in a frame, toward previous, as well as successive frames. The motion fields, which are produced through block matching, and associated to pixels in a given picture uniquely reference parent pixels in the preceding picture. Thus, the backward segment of the motion trajectory is directly given by the motion fields, and follows the MVs of the successive blocks toward previous frames in the GOP. However, the motion mapping defined by block matching is unfortunately not bijective, since parent pixels may have several children in the following pictures (see Fig. 3). There is no guarantee that two different blocks in frame i do not reference overlapping blocks in the frame $i - 1$. The motion

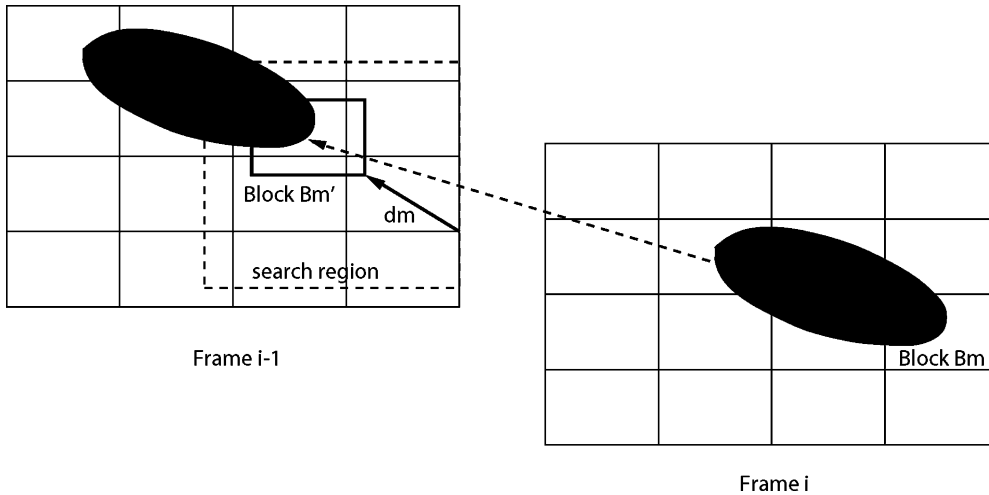
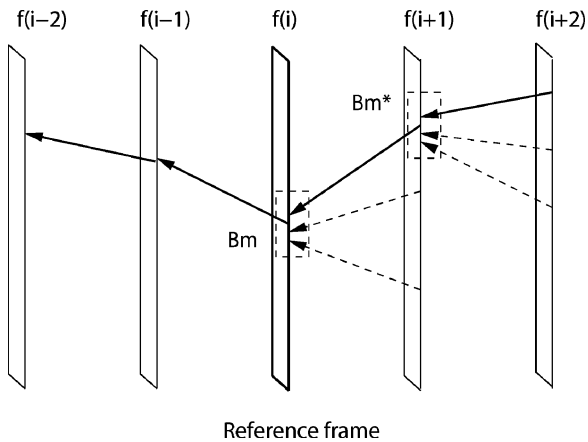


Fig. 2. Block-matching algorithm.

Fig. 3. Example of trajectory prediction, for a block in frame i .

trajectory segments, in the forward direction with respect to the reference frame, could be estimated with a forward motion prediction algorithm (i.e., where blocks in frame i are mapped to blocks in frame $i + 1$), but this solution is definitely too expensive in terms of complexity, and coding overhead. Instead, the forward segment of the motion trajectories is inferred from the backward predicted motion fields, as follows.

A selection strategy is established *a priori*, in order to compute the most likely forward trajectory of a group of pixels in frame i . A given block \mathcal{B}_m in frame f_i , is mapped to the best matching block \mathcal{B}_m^* in frame f_{i+1} using only the MVs derived from backward block matching estimation. The two following criteria allow to select the best trajectory 1) the minimum distance to the center of the block, and 2) the scanning order. In the case where more than one MV from the frame $i + 1$, point to a block that overlaps with block \mathcal{B}_m in frame f_i , the selection is based on the nearest neighbor criteria. In the low probability case where this criteria is not sufficient to choose the best candidate vector, the scanning order is determinant. Note that some blocks in frame f_i may not have any children in the frame $i + 1$, which simply means that the motion trajectory ends in frame i . The selection of the motion trajectories is represented in the re-

maining of the paper by the generic motion mapping operator \mathcal{W} as

$$\mathcal{W}_{i \rightarrow i+1}(I(x, y, i)) \approx I(x, y, i + 1) \quad (13)$$

where $I(x, y, i)$ denotes the samples of frame i in the video sequence I . Fig. 3 illustrates the steps involved during the trajectory prediction. The dotted lines correspond to possible paths that are discarded during trajectory prediction. Thanks to the motion mapping definition, the 3-D MP encoder only transmits backward MVs to the decoder, and all trajectories are computed similarly at encoder and decoder, according to the definition of \mathcal{W} . We noticed empirically that varying the size of the blocks \mathcal{B}_m does not influence significantly neither the approximation performance, nor the coding efficiency of this scheme, for different test sequences. For instance, a block size of 16×16 was employed for video sequences in CIF format. It is very important to clarify that we employed the MVs, in the backward direction, in order to construct the trajectories of the spatio-temporal atoms. Meanwhile, the temporal support of these atoms is adaptive, i.e., it can extend over a certain number of frames in the GOP depending on the video sequence content. In fact, the temporal support is determined by the scale s of the filter in (12). Moreover, this temporal adaptivity can be considered as the bi-directional prediction, which is encountered in the other video coding schemes.

B. MP Expansion

The second part of the 3-D transform consists in computing a decomposition of the GOP in a series of spatio-temporal atoms, by applying the MP algorithm along motion trajectories. A full 3-D search is obviously too computationally complex as it requires to compute all the scalar products between each atom and the residual video sequence. Hence, another effective search policy has been designed to trade off the complexity against the approximation performance, and is based on a heuristic search algorithm, described by Algorithm 1. The iterative search algorithm first selects the frame with the highest energy, in the GOP $I^n(x, y, t)$, where $I^0(x, y, t)$ represents the

original pictures. It then performs an exhaustive MP search in the selected frame, in order to find the M candidates among the spatial atoms in the dictionary, which best fit the picture characteristics. A fast MP implementation is used, based on a FFT algorithm: it allows for computing scalar products with all translated versions of a single atom using one FFT. Each one of the M candidates¹ is then used to build the spatio-temporal atoms, which are aligned on the motion trajectories according to \mathcal{W} , and have the temporal functions as defined in (12). Then, the best spatio-temporal atom ($g(x, y, t)$) is selected, i.e., the one having the largest inner product amplitude with the residual signal $I^n(x, y, t)$ among all the constructed spatio-temporal candidate atoms. The residual signal is updated accordingly to become $I^{n+1}(x, y, t)$. The process is repeated until the signal expansion has enough terms, or until a residual error energy threshold has been reached. Note that even if the MP does not perform a full search, the algorithm still converges quite rapidly, i.e., the constant α from (5) stays close to 1.

The property of choosing the reference frame dynamically in the GOP as the one having the maximum energy, at each iteration, makes the video signal decomposition very adaptive. This chosen reference frame acts like an *intra*-block, where M candidate spatial atoms are short-listed. Then, the prediction is performed in both directions by constructing the spatio-temporal atoms and selecting the one that achieves the best energy extraction or the best prediction. Actually, this property limits the effects of the error propagation problem, which is encountered in the feedback-based systems.

Algorithm 1 The Heuristic Search Algorithm.

- 1: Let $I^0(x, y, t), t = 1 \dots \text{GOP}_{\text{size}}$ be a group of frames
- 2: Select a reference frame r with the largest energy
- 3: Use the 2-D exhaustive search FFT-based algorithm to find the best M uncorrelated candidates among spatial atoms in frame r
- 4: Search for the best mapped 3-D atom starting from the M candidates
- 5: Update the residual $I^n(x, y, t)$ accordingly and iteratively get back to step 2.

C. The Computational Complexity

The Algorithm 1 is characterized by a computational complexity of order $O(pN_s \log_2 N_s + kN_s \text{GOP}_{\text{size}})$ per iteration, where N_s is the spatial size of the video sequence, p is a parameter related to the cardinality of the spatial functions of the dictionary and k is a constant related to M . On the other hand, if the exhaustive search algorithm were employed, then it would require an order of $O(q \text{GOP}_{\text{size}} \cdot N_s \log_2(N_s \cdot \text{GOP}_{\text{size}}))$, where the constant q depends on the cardinality of the dictionary \mathcal{D} and it is larger than p . Thus, the proposed algorithm can achieve a speed-up gain of order $(q/p) \cdot \text{GOP}_{\text{size}}$ with respect to the full search one. For instance, the GOP_{size} is set to 16 and q/p is at least 5, in our experiments.

¹ M is chosen to be proportional to the number of blocks in a picture, in the current implementation.

V. EMBEDDED CODING AND QUANTIZATION

A. Scalable Coding

After the motion-adaptive 3-D transform, coefficients and atoms have to be scalably encoded in order to provide a rate and geometry-adaptive video signal representation. This stage is key to fully benefit from the intrinsic scalability properties of MP expansions over a dictionary built on geometric laws. Each spatio-temporal atom index needs to be coded and transmitted along with its coefficient. The atom index is represented by the tuple $(p_x, p_y, p_t, a_x, a_y, a_t, \theta)$, which costs an average of ρ bits, where (p_x, p_y, p_t) , (a_x, a_y, a_t) , and θ , respectively represent the position of the atom, the scale and the spatial rotation parameters.

Since MP by nature produces a progressive series of atoms, where most energetic components appear first, it makes sense to code atoms in order of their appearance. The stream then offers a great flexibility, since even a simple truncation ensures that the most important features are preserved. Coefficients can be coded by taking into account the exponential decay of their magnitude, as proposed in [25]. An approach based on successive refinement of information [26] can also be implemented, since the statistics of the exponential distribution of coefficient magnitude are known. Both approaches allow to reduce the coding rate of the coefficients, and provide a fully progressive stream. However, no significant gain can be obtained on the atom indexes, since their order is mostly random. Hence, they both work well when atom index size ρ is small. However, the index size may be as large as a few tens of bits for 3-D atoms.

An alternative coding method consists in trying to reduce the index coding rate by changing the initial order of the atoms, possibly at the price of a higher coefficient coding rate. Interestingly, the spatial position parameters p_x and p_y are responsible for almost half of the index coding rate. A natural approach consists in sorting atoms along their spatial position, row- and column-wise, similar to the method proposed in [27]. Run-length coding of spatial position parameters, and lossless arithmetic coding of the remaining parameters could then save up to 7 bits on the average index size (in the case of CIF video format). Nevertheless, in such a scheme, the stream is no longer progressive, since atoms are not sorted along their magnitude anymore, but rather according to their spatial positions. Trying to get the best out of both coding alternatives, the embedded coding in the MP3D encoder initially divides the series of atoms in S disjoint subsets s_i , where each subset contains l_i elements as shown in Fig. 4. These subsets can be seen as energy subbands. Their number is dictated by scalability requirements (i.e., the number of target decoding rates), and represents a tradeoff between stream flexibility, and coding efficiency, which respectively increases and decreases with S . In each subset, atoms are sorted according to their spatial positions, that are further run-length encoded. Other index parameters and quantized coefficients are encoded with a context adaptive arithmetic encoder [28]. The resulting bitstream is now piecewise progressive, and optimal truncation points can be set at subset limits. The rate control problem belongs to a general class of bit allocation problems under multiple rate constraints, and is discussed in detail below.

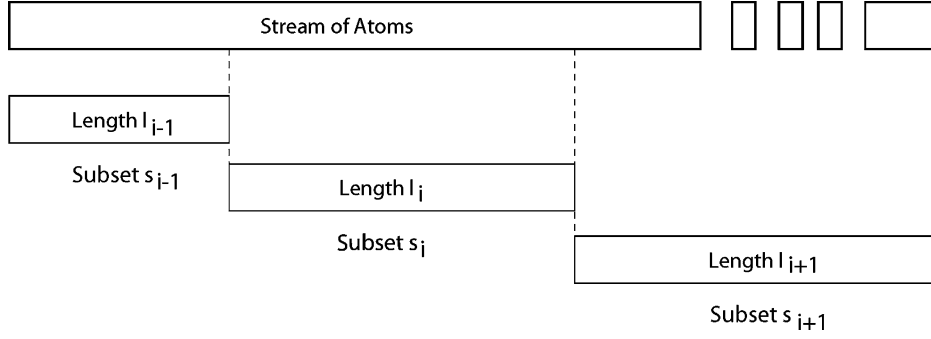


Fig. 4. Series of atoms is divided into energy subsets, as a compromise between coding efficiency and flexibility.

B. Coefficient Quantization

As already observed in [27], the distribution of MP coefficients magnitudes obeys an exponential probability density function (pdf):

$$f(x) = \mu^{-1} e^{-\frac{x}{\mu}}, \quad x > 0, \quad \mu > 0. \quad (14)$$

Without constraint on the number of quantization levels, the mean-square error optimal entropy constrained scalar quantizer (ECSQ) for the exponential pdf is the uniform scalar quantizer designed in [29]. We choose to use a uniform quantizer in each energy subband, allowing however for a different step size in each coefficient subset. This allows us to conveniently model rate and distortion, and develop an efficient rate allocation scheme. As introduced before, the subset s_i contains l_i coefficients, whose magnitudes belong to the interval $[t_{i+1}, t_i[$ (see Fig. 5). With these notations, s_0 gathers the highest energy atoms, and we assumed $t_0 \rightarrow \infty$. Under such assumptions, the number of coefficients in subset s_i is given by $l_i = L(\exp(-t_{i+1}/\mu) - \exp(-t_i/\mu))$, where L is the total number of atoms. Uniform quantization in the subset s_i then generates reconstruction centroids y_k^i that are given by

$$y_k^i(\alpha_i, t_{i+1}, \mu) = k\alpha_i + t_{i+1} + \delta_i \quad (15)$$

where k represents the quantization bin, and α_i is the quantization step size. Following the notation used in [29], the reconstruction offset in each set is given by

$$\delta_i = \mu - \frac{\alpha_i e^{-\frac{\alpha_i}{\mu}}}{1 - e^{-\frac{\alpha_i}{\mu}}}. \quad (16)$$

Using conditional probabilities, the entropy H_i in each subset can be written as

$$H_i = \frac{P_0}{\log 2} \left[\frac{\frac{\alpha_i}{\mu} e^{-\frac{\alpha_i}{\mu}} \left(1 - e^{-\frac{t_i - t_{i+1} - \alpha_i}{\mu}}\right)}{\left(1 - e^{-\frac{\alpha_i}{\mu}}\right)^2} - \frac{\log P_0 + \left(\frac{t_i - t_{i+1} - \alpha_i}{\mu} - \log P_0\right) e^{-\frac{(t_i - t_{i+1})}{\mu}}}{1 - e^{-\frac{\alpha_i}{\mu}}} \right] \quad (17)$$

with P_0 equal to

$$P_0 = \frac{1 - e^{-\frac{\alpha_i}{\mu}}}{1 - e^{-\frac{t_i - t_{i+1}}{\mu}}}. \quad (18)$$

Under the assumption that modern coding methods such as arithmetic coding [28] can achieve rates close to the entropy, the

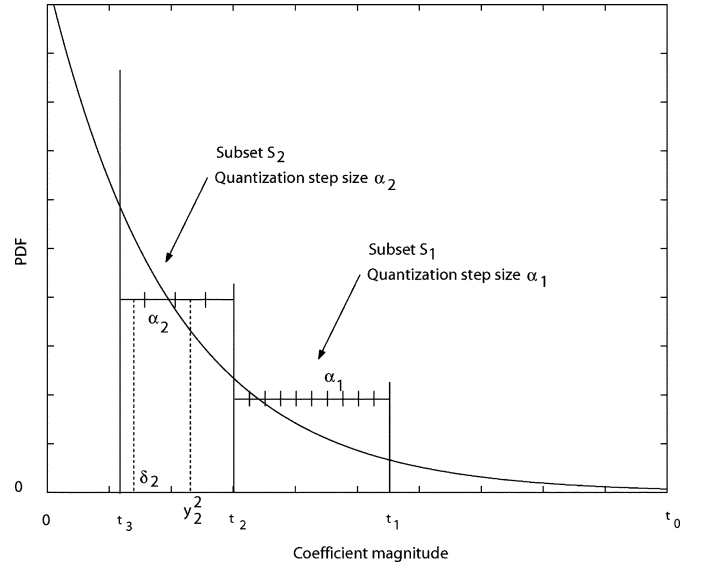


Fig. 5. Example of the construction of energy subbands s_i .

rate R_i required to code the atoms of the subset s_i is given by $R_i = l_i(H_i + \rho)$, where ρ is the average number of bits to code the atom index, which does not depend on s_i . The mean-square distortion D_i in each subset s_i is finally composed of two terms: the quantization error in s_i , $\Delta_q^i(\alpha_i)$, and the distortion $\Delta(t_{i+1})$ due to discarding the coefficients with magnitude smaller than t_{i+1} (see Fig. 5). Equivalently

$$D_i = \Delta_q^i(\alpha_i) + \Delta(t_{i+1}) \quad (19)$$

where

$$\Delta_q^i(\alpha_i) = \left(e^{-\frac{t_{i+1}}{\mu}} - e^{-\frac{t_i}{\mu}} \right) \left(\mu^2 - \frac{\alpha_i^2 e^{-\frac{\alpha_i}{\mu}}}{\left(1 - e^{-\frac{\alpha_i}{\mu}}\right)^2} \right) \quad (20)$$

and

$$\Delta(t_{i+1}) = 2\mu^2 \left(1 - e^{-\frac{t_{i+1}}{\mu}}\right) - t_{i+1} e^{-\frac{t_{i+1}}{\mu}} (t_{i+1} + 2). \quad (21)$$

Details about the computation of H_i and D_i are given in Appendix I.

The distortion of the reconstructed signal given that all the subsets s_j for $j \leq i$ are decoded, is denoted as d_i . In the case of $i \geq 1$, it is expressed as

$$d_i = \Delta(t_{i+1}) + \sum_{j=0}^i \Delta_q^j(\alpha_j) = D_i + \sum_{j=0}^{i-1} \Delta_q^j(\alpha_j). \quad (22)$$

Meanwhile for $i = 0$, d_0 is equal to D_0 .

C. Rate Allocation

For scalability issues, the bitstream is encoded efficiently at several target rates, which can be directly defined by the application. The rate allocation becomes an optimization problem with multiple constraints. The bit stream has to be efficiently decoded at target rates $\{r_0, r_1, \dots, r_{N-1}\}$, and the corresponding distortions $\{d_0, d_1, \dots, d_{N-1}\}$ should be as close as possible to the best distortion achievable at these target rates. Moreover, these distortions $(d_0, d_1, \dots, d_{N-1})$ must form a *successive refinement chain*, i.e., the subset s_i is built such that the distortion d_i is minimum for the rate r_i given that all the subsets s_j , for $j < i$, are decoded. In such a case, the distortion d_i refines d_{i-1} , which also refines d_{i-2} , and so forth, up to d_0 . The successive refinement chain condition is denoted as $d_0 \rightarrow d_1 \rightarrow d_2 \rightarrow \dots \rightarrow d_{N-1}$. This optimization problem can be formalized with Lagrange multipliers $\vec{\lambda}$, and becomes equivalent to minimizing the cost function $J(\vec{\lambda}) = \mathcal{D} + \vec{\lambda}^T \vec{\mathcal{R}}$ subject to 1) $\mathcal{R}_i \leq r_i$ for $i = 0, \dots, N-1$ and 2) the successive refinement chain condition, where $\mathcal{D} = \sum_{i=0}^{N-1} d_i$, $\vec{\mathcal{R}} = [R_0, R_1, \dots, R_{N-1}] \cdot U_1$, and U_1 is an upper triangular matrix of ones. The constrained optimization problem now consists in finding the thresholds, t_i , and the quantizer step sizes, α_i , that minimize $J(\vec{\lambda})$.

Instead of solving the complex global optimization problem, the proposed rate allocation strategy adopts a greedy approach, that optimizes the quantization in each independent subset successively. Since atoms are allocated to subsets according to their magnitude, subsets are naturally assigned different priority levels, with s_0 becoming the most important one. The rate allocation algorithms starts by minimizing $J_0(\lambda_0) = D_0 + \lambda_0 R_0$ under the rate constraint $R_0 \leq r_0$ to find either the threshold t_1 (or equivalently l_0) and the step size α_0 . It then optimizes iteratively $J_i(\lambda_i) = D_i + \lambda_i R_i$ under the rate constraint $R_i \leq (r_i - \sum_{k=0}^{i-1} R_k)$. The rate allocation is summarized in Algorithm 2.

Algorithm 2 The Rate Allocation Algorithm

```

Let  $\{r_0, r_1, \dots, r_{N-1}\}$  be the multiple rate constraints.
Set the rightmost threshold  $t_0 \rightarrow \infty$ 
Find  $(\alpha_0, t_1) = \arg \min(J_0(\lambda_0) = D_0 + \lambda_0 R_0)$  subject to
 $R_0 \leq r_0$ 
for  $i = 1 \dots N-1$  do
    Find  $(\alpha_i, t_{i+1}) = \arg \min(J_i(\lambda_i) = D_i + \lambda_i R_i)$  subject
    to  $R_i \leq (r_i - \sum_{k=0}^{i-1} R_k)$ 
end for
    
```

Various approaches can be used to minimize each individual cost function $J_i(\lambda_i)$. Heuristic approaches based on the pruning strategy operating on trees have been proven to be efficient [30], [31]. A tree of Q subtrees can be built for each subset s_i , where each subtree is a unitree of $\text{Card}(\mathcal{A})$ nodes, with \mathcal{A} the set of possible quantization step sizes α . A set \mathcal{A} of fifteen values has been verified to work well over a large range of bit rates, and therefore we set $\mathcal{A} = \{120, 100, 80, 70, 60, 50, 40, 35, 30, \dots, 10, 5\}$. Each node of a unitree corresponds to a different rate-distortion (R-D) tuple (R_i, D_i) , and the leaf is the lowest distortion node (respectively the node with the highest rate). Each unitree then

corresponds to a different number of atoms l_i in the subset (i.e., a different value of t_{i+1}), and the number of unitrees Q is an arbitrary parameter that is related to the search space dimension. The search can be limited to values of l_i that are close to $L_i = (r_i - \sum_{k=0}^{i-1} R_k) / \rho$, the maximum number of atoms permitted under the rate constraints, since the index size is generally larger than the coefficient entropy. In the current implementation, Q is set to 10, and l_i is uniformly distributed between $0.75L_i$ and L_i . Finally, the minimization of $J_i(\lambda_i)$ consists in pruning the unitrees for the nodes that violate the rate constraints $R_i \leq (r_i - \sum_{k=0}^{i-1} R_k)$, and in choosing among the Q unitrees, the leaf with the minimal distortion.

VI. EXPERIMENTAL RESULTS

A. Rate-Distortion Comparison Against H.264 and MPEG

In this section, we evaluate the R-D performances of our codec by comparing it with two reference schemes: MPEG4² and H.264.³ The standard Foreman, Football and Bus sequences in CIF format at 30 fps have been used to generate the results. In all experiments, we used a GOP size of 16. Moreover, in both schemes, we used only the I- and P-frame type, i.e., each GOP in the image sequence consists of the sequence having a structure as: IPP.PIPP. The I-frame is coded using the *Intra-frame* prediction whereas the P-frame is coded using the *Inter-frame* prediction. Notice that no frame of type B is used. It can be seen on Fig. 6(a) and (b) that the PSNR of MP3D is higher than the one of MPEG-4 by about 1–1.5 dB for Foreman and Football sequences and over a wide range of bit rates. Meanwhile, it is only slightly inferior to the performance of H.264, staying within a 1-dB gap (see Fig. 6). We noted that the results for the Foreman sequence are always penalizing our scheme at high rate. On the other hand, our scheme performs better on the Football sequence for example, where it stays close to H.264 over the whole range of bit rates under consideration. Our codec does not perform very well on textures (see Fig. 7): once most geometrical information has been encoded, PSNR tends to saturate at higher rates. Finally, one should remember that both H.264 and MPEG-4 are non-scalable video coding schemes optimized for compression performance, contrarily to MP3D. In the next section, we compare MP3D to another highly scalable scheme for completeness.

B. R-D Comparison Against MC-EZBC and H.264/Ext

An objective evaluation of MP3D against MC-EZBC⁴ and H.264 with the scalability extensions, which is referred to as H.264/Ext,⁵ in terms of PSNR performances, is given in Fig. 6. The GOP size was set to 16 in both schemes. For the MC-EZBC scheme, we set the parameter of the temporal levels to four. We used also a single layer coding for the MVs (MV_s) in order to have a fair comparison with MP3D since it has only one layer for MV_s in the current version of MP3D. In the case of H.264/Ext,

²MPEG-4 Reference Software [Online] Available: <http://megaera.ee.nctu.edu.tw/mpeg/>

³H.264/AVC Reference Software [Online] Available: <http://bs.hhi.de/suehring/tml/>

⁴MC-EZBC Software [Online] Available: <http://mpeg.nist.gov/cvsweb/>

⁵H.264 With Scalability Extension [Online] Available: <http://mpeg.nist.gov/cvsweb/>

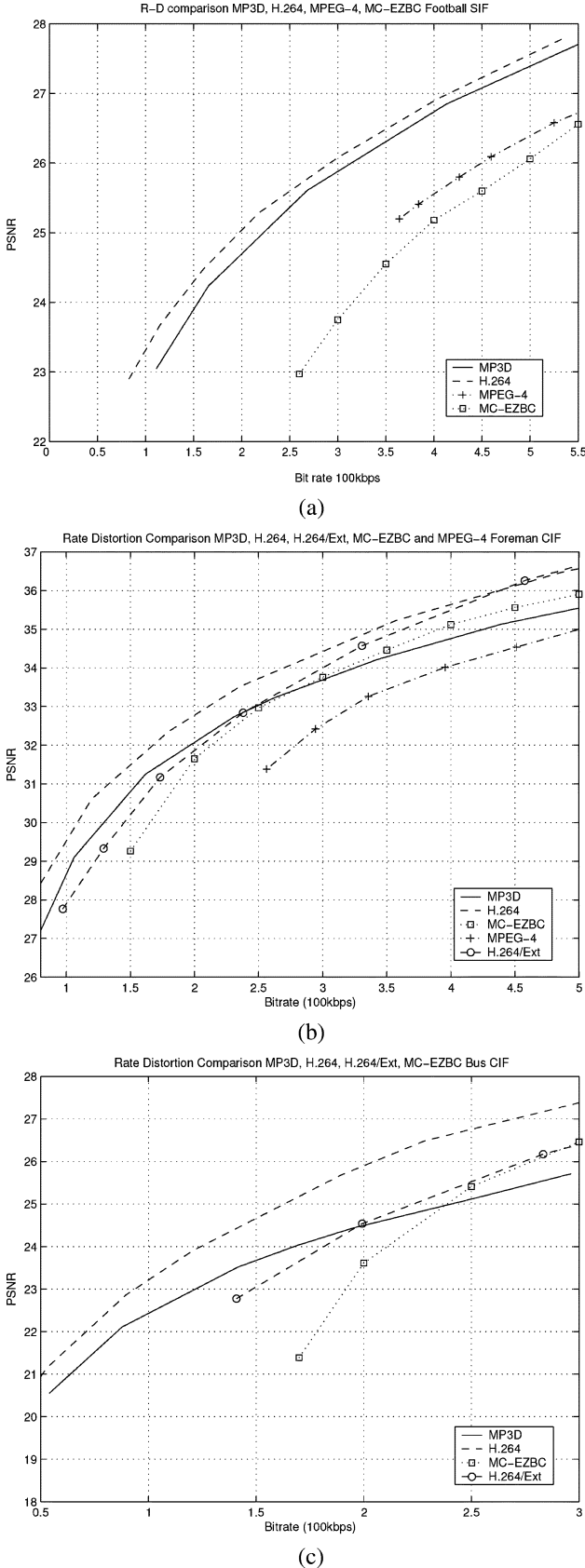


Fig. 6. Rate-distortion performances of MP3D. (a) Football sequence. (b) Foreman sequence. (c) Bus sequence.

we used only the I-and P-frames, i.e., the image sequence has the following structure: IPP.PIPP. The number of layers in the

H.264/Ext was set to five. For the Football sequence, MP3D has a higher PSNR than MC-EZBC over the whole range under consideration. For the Foreman and Bus sequences however, there exists a cross-over point where MP3D loses its advantage to H.264/Ext and MC-EZBC schemes. At some higher rate, coding the atoms results in a less significant PSNR improvement. Nevertheless, we observed that MP3D is always more efficient at low and medium rate (less than 500 kbps).

C. Visual Quality Comparison

Fig. 7 shows visual comparisons of the first frame from the Football sequence decoded at 550 kbps, using the schemes mentioned before. One can see that H.264 produces more uniform regions. The regions in MP3D are also very smooth, but most prominent edges are well captured due to the nature of the dictionary we used. On the other hand, MP3D lost most textures. The frame coded by MC-EZBC shows a trade-off in representing smooth areas and texture components, while MPEG-4 produces an overall slightly inferior visual quality. Of course these tests are not conclusive, but they allow to show the behavior of MP3D in capturing first the geometrical features in image sequences.

VII. SCALABILITY PROPERTIES

The scalability features are intrinsic in MP3D due to the multiresolution structure of the dictionary, the nature of MP and the embedded coding. All these characteristics make the bitstream highly scalable, offering 3-D geometric (i.e., spatio-temporal) and SNR scalabilities. The geometric properties of the dictionary enable a very easy sequence adaptation prior to decoding. As a result, a single bitstream can be decoded at any spatial resolution (as long as the re-scaling is isotropic) and at various frame rates, without resorting to costly re-encoding or post-processing operations. These properties significantly differ from simple transcoding schemes, and we chose to refer to them as geometric scalability.

For example, a coded video signal I of spatial size $W \times H$ with a frame rate F can be spatially decoded into a video signal \tilde{I} of spatial resolution $\alpha W \times \alpha H$ at the same frame rate as follows. First the full atom trajectory is reconstructed at the initial size using the motion field operator \mathcal{W} . Then each individual atom is analytically re-scaled by simply transcoding its index values (scales and positions) as described in [2]. The new signal reads

$$\tilde{I} = \sum_{i=0}^{N-1} \alpha c_i \mathcal{W}(\tilde{g}_{\gamma_i}) \quad (23)$$

where c_i are the atom coefficients and $\mathcal{W}(\tilde{g}_{\gamma_i})$ corresponds to the motion-mapped atom $\mathcal{W}(g_{\gamma_i})$ after transcoding. We noted that, when transcoding by $\alpha < 1$, thus to a lower resolution, possible aliasing from very small atoms saturate quickly PSNR quality as rate increases. The smallest atoms are simply discarding by stream truncation. Fig. 8 shows frame 1 of the Foreman sequence decoded in QCIF format from the bitstream corresponding to CIF format. One sees that spatial resolution adaptation nicely preserves edges after transcoding. These

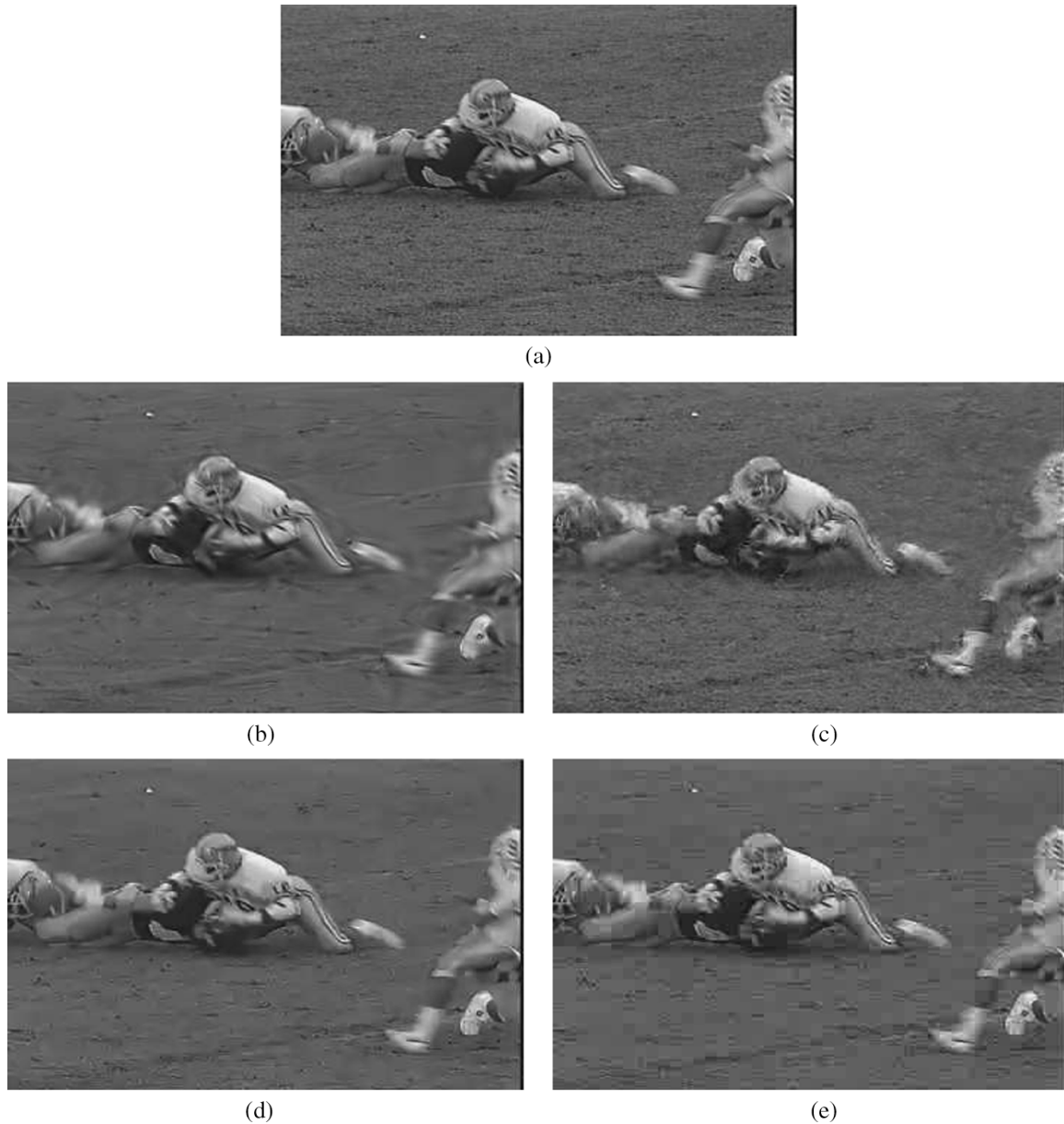


Fig. 7. Visual comparison for frame 1 of football decoded at 550 kbps. (a) Original. (b) MP3D. (c) MC-EZBC. (d) H.26L. (e) MPEG-4.



Fig. 8. Frame 1 of Foreman decoded in QCIF from the CIF bitstream.

structures are indeed well captured by our dictionary and the corresponding atoms are simply re-scaled, when decoded at a different resolution. This clearly brings a great advantage in visual quality.

Besides geometric scalability, MP3D provides natural signal-to-noise ratio (SNR) scalability because of the exponential decay of MP coefficients and the embedded quantization. It was noticed in Section V that the amplitude of atom coefficients has a small variation within most of the subsets (except for the first one). Therefore, simple truncation of the embedded bitstream in a given subset still ensures that the decoder receives most of the signal energy for the available bandwidth, as shown on Fig. 9.

VIII. CONCLUSION

In this paper, we introduced a video coding scheme based on motion-adaptive decompositions in a redundant dictionary of waveforms. The overcomplete dictionary is designed to

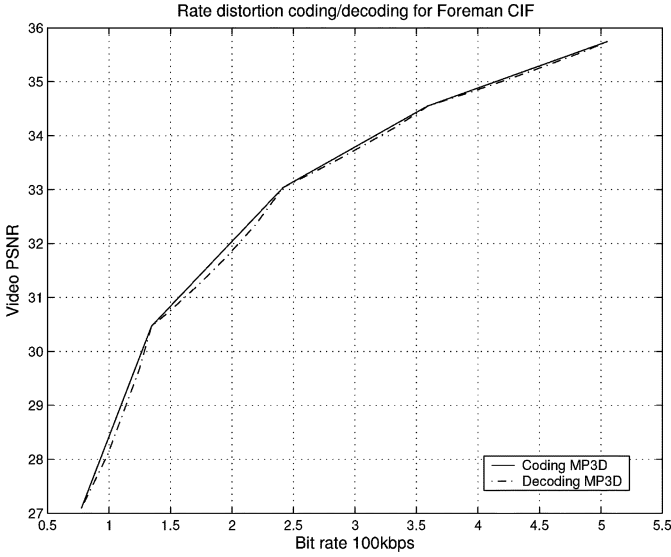


Fig. 9. PSNR of the Foreman sequence encoded according to the multirate constraint {75 135 245 360 500} kbps and decoded at various intermediate rates.

model image primitives, mostly edges, which are likely to display coherent trajectories over time. The MP algorithm is first used to compute a compact signal representation. In parallel, the sequence motion field is estimated by classical block matching techniques and used to recover the trajectories of most prominent image primitives. The data is then filtered along these trajectories using a redundant temporal dictionary. An embedded multirate allocation method was designed to offer a progressively refinable bit stream. Target rate points are defined at the encoder, and the decoder can recover those R-D points by truncating the stream. Suboptimal decoding is still possible in-between predefined target rates and simulations show only a slight degradation in R-D performance. The compressed video sequence can further be decoded at any spatial resolution due to the parametric structure of the redundant libraries used to represent the information. These geometric stream manipulations are lightweight and can be performed at the decoder or by some simple network intelligence. Comparisons with state-of-the-art scalable and nonscalable codecs illustrate the good performance of the proposed technique at low bit rates and motivate its possible use as a flexible base layer in a more general scalable framework. The computational complexity of our encoding scheme is clearly one of its main drawbacks, and faster implementations are currently under study.

APPENDIX I

ENTROPY AND DISTORTION IN COEFFICIENT SUBSETS

Let s_i be the subset of all possible atoms whose coefficient modulus belongs to $[t_{i+1}, t_i]$ according to the exponential distribution in (14). By assuming a uniform quantization step size of α , the total number of possible quantization levels (or the number of bins) n should be

$$n = \frac{t_i - t_{i+1}}{\alpha}. \quad (24)$$

The conditional probability of bin k given s_i is $p(k|s_i)$ (which is denoted by P_k) is

$$p(k|s_i) = \frac{p(k)}{p(s_i)} \quad (25)$$

$$= \frac{e^{-\frac{t_{i+1}+\alpha k}{\mu}} - e^{-\frac{t_{i+1}+\alpha(k+1)}{\mu}}}{e^{-\frac{t_{i+1}}{\mu}} - e^{-\frac{t_i}{\mu}}} \quad (26)$$

$$= \left(\frac{1 - e^{-\frac{\alpha}{\mu}}}{1 - e^{-\frac{t_i - t_{i+1}}{\mu}}} \right) e^{-\frac{\alpha k}{\mu}} \quad (27)$$

$$= P_0 e^{-\frac{\alpha k}{\mu}}. \quad (28)$$

with P_0 is defined in (18).

Now, the resulting entropy necessary to code all the quantized values in s_i is

$$\begin{aligned} H_i &= - \sum_{k=0}^{n-1} P_k \log_2 P_k \\ &= - \sum_{k=0}^{n-1} P_0 e^{-\frac{\alpha k}{\mu}} \log_2 \left(P_0 e^{-\frac{\alpha k}{\mu}} \right) \\ &= \frac{P_0}{\log 2} \sum_{k=0}^{n-1} e^{-\frac{\alpha k}{\mu}} \left(\frac{\alpha k}{\mu} - \log P_0 \right). \end{aligned}$$

By using the closed form of a finite sum of an arithmetico-geometric progression, we obtain

$$H_i = \frac{P_0}{\log 2} \left[\frac{\frac{\alpha}{\mu} e^{-\frac{\alpha}{\mu}} \left(1 - e^{-\frac{\alpha(n-1)}{\mu}} \right)}{\left(1 - e^{-\frac{\alpha}{\mu}} \right)^2} - \frac{\log P_0 + \left(\frac{\alpha(n-1)}{\mu} - \log P_0 \right) e^{-\frac{\alpha n}{\mu}}}{1 - e^{-\frac{\alpha}{\mu}}} \right] \quad (29)$$

Now substituting n of (24) in (29), we get the final expression of H_i as in (17).

When the mean-square error (MSE) measure is used, the reconstruction centroids are defined as

$$\begin{aligned} y_k^i(\alpha, t_{i+1}, \mu) &= \frac{\int_{t_{i+1}+\alpha k}^{t_{i+1}+\alpha(k+1)} x f_e(x) dx}{\int_{t_{i+1}+\alpha k}^{t_{i+1}+\alpha(k+1)} f_e(x) dx} \\ &= \alpha k + t_{i+1} + \frac{\int_0^\alpha x f_e(x) dx}{\int_0^\alpha f_e(x) dx} \\ &= \alpha k + t_{i+1} + \mu - \frac{\alpha e^{-\frac{\alpha}{\mu}}}{1 - e^{-\frac{\alpha}{\mu}}} \\ &= \alpha k + t_{i+1} + \delta_i, \end{aligned}$$

where δ_i is defined as in (16).

Finally, the distortion D_i is defined as the total distortion for coefficients whose magnitude is smaller than t_i which is composed of two terms, one Δ_q^i due to quantization error in s_i and the other $\Delta(t_{i+1})$ due to discarding the coefficients whose magnitude is smaller than t_{i+1} , i.e., $D_i = \Delta_q^i + \Delta(t_{i+1})$.

Δ_q^i is defined as the average cumulative distortion by using the reconstruction codewords y_k^i over the entire subset s_i , i.e.,

$$\Delta_q^i = \sum_{k=0}^{n-1} \int_{t_{i+1}+\alpha k}^{t_{i+1}+\alpha(k+1)} (x - y_k^i)^2 f_e(x) dx \quad (30)$$

$$= e^{-\frac{t_{i+1}}{\mu}} \sum_{k=0}^{n-1} e^{-\frac{\alpha k}{\mu}} \underbrace{\int_0^\alpha (x - \delta_i)^2 f_e(x) dx}_{\gamma} \quad (31)$$

Clearly γ does not depend on k and it is given as

$$\gamma = \mu^2 \left(1 - e^{-\frac{\alpha}{\mu}} \right) - \frac{\alpha^2 e^{-\frac{\alpha}{\mu}}}{1 - e^{-\frac{\alpha}{\mu}}}.$$

Now, by substituting γ and n in (30) we obtain Δ_q^i as

$$\Delta_q^i = \left(e^{-\frac{t_{i+1}}{\mu}} - e^{-\frac{t_i}{\mu}} \right) \left(\mu^2 - \frac{\alpha^2 e^{-\frac{\alpha}{\mu}}}{\left(1 - e^{-\frac{\alpha}{\mu}} \right)^2} \right).$$

The second term $\Delta(t_{i+1})$ is defined as follows:

$$\begin{aligned} \Delta(t_{i+1}) &= \int_0^{t_{i+1}} x^2 f_e(x) dx \\ &= e^{-\frac{x}{\mu}} (x^2 + 2x\mu + 2\mu^2) \Big|_{x=0}^{x=t_{i+1}} \\ &= 2\mu^2 \left(1 - e^{-\frac{t_{i+1}}{\mu}} \right) - t_{i+1} e^{-\frac{t_{i+1}}{\mu}} (t_{i+1} + 2) \end{aligned}$$

which gives the final expression of D_i as in (19).

REFERENCES

- [1] E. J. Candes and D. L. Donoho, "Curvelets—a surprisingly effective nonadaptive representation for objects with edges," in *Curve and Surface Fitting*, A. C. C. Rabut and L. L. Schumaker, Eds. Saint-Malo: Vanderbilt Univ. Press, 1999.
- [2] A. Rahmoune, P. Vanderghyest, and P. Frossard, "MP3D: a highly scalable video coding scheme based on matching pursuit," in *Proc. IEEE ICASSP*, vol. 3, May 2004, pp. 133–136.
- [3] R. Figueras i Ventura, P. Vanderghyest, and P. Frossard, "Low rate and flexible image coding with redundant representation," *IEEE Trans. Image Process.*, Feb. 2006, to be published.
- [4] P. Frossard, "Robust and multiresolution video delivery: From H.26x to matching pursuit based technologies," Ph.D. dissertation, EPFL, Lausanne, Switzerland, 2000.
- [5] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.
- [6] G. Karlsson and M. Vetterli, "Three-dimensional subband coding of video," in *Proc. IEEE ICASSP*, vol. 2, Apr. 1988, pp. 1100–1103.
- [7] J.-R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Trans. Image Process.*, vol. 3, pp. 559–571, Sep. 1994.
- [8] D. Taubman and A. Zakhor, "Multirate 3-D subband coding of video," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 572–588, Sep. 1994.
- [9] B.-J. Kim and W. A. Pearlman, "An embedded wavelet video coder using three-dimensional set partitioning in hierarchical trees (3d-spiht)," in *Proc. IEEE Data Compression Conf.*, Mar. 1997, pp. 251–259.

- [10] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 234–250, Jun. 1996.
- [11] S.-J. Choi and J. W. Woods, "Motion-compensated 3-d subband coding of video," *IEEE Trans. Image Process.*, vol. 8, no. 2, pp. 155–167, Feb. 1999.
- [12] J. Xu, Z. Xiong, S. Li, and Y.-Q. Zhang, "Three-dimensional embedded subband coding with optimized truncation (3-D ESCOT)," *Appl. Comput. Harmonic Anal.*, vol. 10, pp. 290–315, 2001.
- [13] A. Secker and D. Taubman, "Lifting-based invertible motion adaptive transform (LIMAT) framework for highly scalable video compression," *IEEE Trans. Image Process.*, vol. 12, no. 12, pp. 1530–1542, Dec. 2003.
- [14] B. Pesquet-Popescu and V. Botreau, "Three-dimensional lifting schemes for motion compensated video compression," in *IEEE Proc. Int. Conf. Acoust. Speech, Signal Process.*, May 2001, pp. 1793–1796.
- [15] Y. Andreopoulos, A. Munteanu, J. Barbarien, M. Van der Schaar, J. Cornelis, and P. Schelkens, "In-band motion compensated temporal filtering," *Signal Process. Image Commun.*, vol. 19, no. 7, pp. 653–673, 2004.
- [16] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations," in *Constructive Approximations*. New York: Springer-Verlag, 1997.
- [17] S. Chen and D. Donoho, "Atomic decomposition by basis pursuit," presented at the *SPIE Int. Conf. Wavelets*, San Diego, CA, Jul. 1995.
- [18] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decompositions," *IEEE Trans. Inf. Theory*, vol. 47, no. 7, pp. 2845–2862, Nov. 2001.
- [19] R. Gribonval and M. Nielsen, "Sparse representations in unions of bases," IRISA, Rennes, France, Tech. Rep. 1499, 2003.
- [20] A. C. Gilbert, S. Muthukrishnan, and M. J. Strauss, "Approximation of functions over redundant dictionaries using coherence," in *Proc. 14th Annu. ACM-SIAM Symp. Discrete Algorithms*, 2003, pp. 243–252.
- [21] L. Jones, "On a conjecture of huber concerning the convergence of projection pursuit regression," *Annal. Statist.*, vol. 15, pp. 880–882, 1987.
- [22] E. Le Pennec and S. Mallat, "Sparse geometric image representation with bandelets," *IEEE Trans. Image Process.*, vol. 14, no. 4, pp. 423–438, Apr. 2005.
- [23] M. N. Do and M. Vetterli, "The contourlet transform: an efficient directional multiresolution image representation," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 2091–2106, Dec. 2005.
- [24] I. Schoenberg, "Spline functions and the problem of graduation," *Proc. Nat. Acad. Sci.*, vol. 52, pp. 947–950, 1964.
- [25] P. Frossard, P. Vanderghyest, R. Figueras i Ventura, and M. Kunt, "A posteriori quantization of progressive matching pursuit streams," *IEEE Trans. Signal Process.*, vol. 52, no. 2, pp. 525–535, Feb. 2004.
- [26] W. H. R. Equitz and T. M. Cover, "Successive refinement of information," *IEEE Trans. Inf. Theory*, vol. 37, no. 2, pp. 269–275, Mar. 1991.
- [27] R. Neff and A. Zakhor, "Very low bit rate video coding based on matching pursuits," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 158–171, Feb. 1997.
- [28] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Commun. ACM*, vol. 30, pp. 520–540, Jun. 1987.
- [29] G. J. Sullivan, "Efficient scalar quantization of exponential and Laplacian random variables," *IEEE Trans. Inf. Theory*, vol. 42, no. 5, pp. 1365–1374, Sep. 1996.
- [30] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Optimal pruning with applications to tree-structured source coding and modeling," *IEEE Trans. Inf. Theory*, vol. 35, no. 3, pp. 299–315, Mar. 1989.
- [31] E. A. Riskin, "Optimal bit allocation via the generalized BFOS algorithm," *IEEE Trans. Inf. Theory*, vol. 37, no. 3, pp. 400–402, Mar. 1991.

Adel Rahmoune received the M.Sc. degree in electrical engineering in 1999, then the Ph.D. degree from the Swiss Federal Institute of Technology, Lausanne, Switzerland, in 2005.

He is a Researcher at the Signal Processing Institute of EPFL, Lausanne, Switzerland. His research interests are mainly focused on signal and image representation, processing, and compression, image analysis, and computer vision.



Pierre Vandergheynst (M'01) received the M.S. degree in physics and the Ph.D. degree in mathematical physics from the Université Catholique de Louvain, Louvain, Belgium, in 1995 and 1998, respectively.

From 1998 to 2001, he was a Postdoctoral Researcher with the Signal Processing Laboratory, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland. He is now an Assistant Professor of visual information representation theory at EPFL, in charge of the LTS2 Laboratory, where his research focuses on active in several research

topics ranging from time-frequency/time-scale analysis to mathematical image analysis, computer vision, data compression, and higher dimensional data processing.

Dr. Vandergheynst is co-Editor-in-Chief of *Signal Processing*.



Pascal Frossard (S'96–M'01–SM'04) received the M.S. and Ph.D. degrees, both in electrical engineering, from the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, in 1997 and 2000, respectively. In 1998, he graduated from the Graduate School of the Communications System Division of the EPFL.

From 1998 to 2000, he worked with the Signal Processing Laboratory, EPFL, as a research and teaching assistant under a grant from Hewlett-Packard. Between 2001 and 2003, he was a member of the

research staff at the IBM T. J. Watson Research Center, Yorktown Heights, NY, where he worked on media compression and streaming technologies. Since April 2003, he has been an assistant professor at EPFL. His research interests include image representation and coding, nonlinear representations, visual information analysis, joint source and channel coding, multimedia communications, and multimedia content distribution. He works on nonlinear signal transforms applied to visual information, and adaptive media streaming algorithms. In 2002, he was co-editor of a Special Issue on new visual coding methods for *Signal Processing*.

Dr. Frossard has been the General Chair of IEEE ICME 2002 (Lausanne, Switzerland), and member of the organizing or technical program committees of numerous conferences. He has served as Guest Editor of special issues on Streaming Media (IEEE TRANSACTIONS ON MULTIMEDIA), on Media and Communication Applications on General Purpose Processors: Hardware and Software Issues (*Journal of VLSI SPSS*), and on Image and Video Coding Beyond Standards (*Journal of Signal Processing*). He is an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA (2004–) and of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (2006–), and he served as a member of the Editorial Board of the *EURASIP Journal of Signal Processing* (2003–2005). Since 2004, he has served as Vice-Chair of the IEEE Multimedia Communications Technical Committee, as a member of the IEEE Multimedia Signal Processing Technical Committee, and of the IEEE Multimedia Systems and Applications Technical Committee. He received the Swiss NSF Professorship Award in 2003, and the IBM Faculty Award in 2005.