MP3D: Highly Scalable Video Coding Scheme Based on Matching Pursuit

Adel Rahmoune, Pierre Vandergheynst and Pascal Frossard
{adel.rahmoune,pierre.vandergheynst,pascal.frossard}@epfl.ch
Swiss Federal Institute of Technology EPFL
Signal Processing Institute ITS
CH- 1015 Lausanne, Switzerland

ITS Technical Report
ITS-TR.06.03. November 2003

Abstract

This paper describes a novel video coding scheme based on a three-dimensional Matching Pursuit algorithm. In addition to good compression performance at low bit rate, the proposed coder allows for flexible spatial, temporal and rate scalability thanks to its progressive coding structure. The Matching Pursuit algorithm generates a sparse decomposition of a video sequence in a series of spatio-temporal atoms, taken from an overcomplete dictionary of three-dimensional basis functions. The dictionary is generated by shifting, scaling and rotating two different mother atoms in order to cover the whole frequency cube. An embedded stream is then produced from the series of atoms. They are first distributed into sets through the set-partitioned position map algorithm (SPPM) to form the index-map, inspired from bit plane encoding. Scalar quantization is then applied to the coefficients which are finally arithmetic coded. A complete MP3D codec has been implemented, and performances are shown to favorably compare to other scalable coders like MPEG-4 FGS and SPIHT-3D. In addition, the MP3D streams offer an incomparable flexibility for multiresolution streaming or adaptive decoding.

I. Introduction

Most successful video compression algorithms are based on the hybrid approach that combines motion compensation between successive frames, and DCT block transform. Such schemes have been quite successful, and represent the core of the current compression standards, like H263 or MPEG-4. While they provide interesting performance in compression, these coders generally lack a increasingly important feature, which is a flexible scalability. The need for adaptive streaming or the possibility to offer different resolutions from a single bitstream is fueled by the continuing development of heterogeneous networks and infrastructure. In streaming applications, for example, a progressive stream allows to adapt to changing network conditions, or to clients with different access bandwidths. Spatio-temporal scalability offers yet additional flexibility since the frame rate, and the size of the decoded frames can be adapted to the client capacities. Due to these recent needs in adaptive coding, scalability is getting a lot of attention and efforts from the research community.

A fine granular scalability (FGS) video coding scheme [1] based on MPEG has recently been proposed to provide SNR scalability. In the same context, Van der Schaar and Hayder [2] proposed MPEG-based video coding scheme with SNR and temporal scalability. A different class of scalable video coding algorithms has been introduced for video streaming applications, based on a 3-D wavelet coding approach. These

This work has been partly supported by the Swiss National Science Foundation.

methods generally use temporal filtering in the direction of motion [3], [4], [5], [6], but interesting results have also been shown in the absence of any motion compensation as in SPIHT-3D [7], which may provide also additional adaptivity and error resilience.

In this paper, a new highly scalable video coding scheme is proposed, based on a three-dimensional Matching Pursuit algorithm (MP3D). The compression performance are shown to compare favorably to SPIHT-3D and MPEG-FGS, especially at low coding rates. Additionally, the stream generated by MP3D provides an increased flexibility in terms of adaptivity. The paper is organized as follows. In Section II, the matching pursuit video coder is presented, and the dictionary construction is detailed. In Section III, the scalability features of MP3D (i.e., SNR, spatial and temporal scalability), are presented. The performance of MP3D are then discussed in Section IV, and Section V finally concludes the paper.

II. MP3D: Matching Pursuit Video Coder

A. Sparse Representations

Most acclaimed technical solutions to both image and video compression, namely the JPEG2000 and MPEGx/H.26x families of standards, rely heavily on transform coding. Moving to the transform domain is usually performed in order to obtain decorrelated sets of coefficients on which scalar quantization and entropy coding is performed, and this drives the choice of the transform. Most techniques use two well controlled orthonormal basis (ONB): DCT and wavelets. Performing the transform by means of an ONB allows the use of well studied data compression results, and in both cases fast algorithms help keeping a low complexity implementation. 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 basically try to derive a coding scheme that preserves pre-defined structures in a sequence of frames. More specifically we consider such a sequence as a 3-D space-time signal $I(x, y, t)$ and try to efficiently encode coherent spatio-temporal structures.

The chosen approach relies on expanding the signal as a linear superposition of generalized waveforms tuned to match the requested structures and selected among a vast library :

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

The only constraint on the collection $\mathcal{D} = \{g_\gamma, \gamma \in \Gamma\}$ is that it is dense in the space of finite energy signals. In the following we refer to g_γ as an atom and to $\mathcal{D}$ as a dictionary. The set Γ in (1) can be chosen as an anonymous set of labels but may also carry important information about the atoms, for example space and frequency localization, as will be the case in this paper. Of course we also wish that the necessary parameters in this expansion, namely the set of coefficients c_i and indexes γ_i yield good compression performances and this leads us to a generic requirement about (1), namely that this expansion is sparse enough.

Without additional constraints on $\mathcal{D}$, and in particular if it is not an ONB, there is generally not a unique sparse expansion. One possible solution can be to look for the sparsest possible exact expansion, that is minimizing the number of coefficients in (1). This unfortunately leads to a daunting combinatorial optimization problem that is NP hard. A close solution may be provided by relaxing this problem and trying to minimize the ℓ^1 norm of the coefficients which leads to the Basis Pursuit algorithm deeply studied by Donoho et al. [8]. Interestingly this algorithm sometimes leads to the optimal sparsest solution of (1) with particular dictionaries [9], [10], [11].

Alternatively, the Matching Pursuit (MP) algorithm [12] provides a interesting generic solution to (1) by iteratively decomposing the signal using a greedy strategy. Starting with $R_0 = I$, the n^{th} iteration reads

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

where the atom g_{γ_n} is the one having maximum correlation with R_n :

$$g_{\gamma_n} = \arg\max_{\mathcal{D}} |\langle R_n, g_\gamma \rangle|\,. \tag{3}$$

After N steps MP yields a sparse approximation :

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

where R_N is a small residual error. Matching Pursuit converges, that is $\|R_N\| \to 0$ when N tends to infinity and converges even exponentially in finite dimension [12]:

$$\|R_N\|^2 \lesssim (1-\beta^2)^N \tag{5}$$

where β is 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 [11] but their description is beyond the scope of this paper. As already shown in [13] MP is intrinsically well suited for compression of visual information because it easily yields scalable streams by simply truncating (4). Moreover 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 very low bit rates.

B. Spatio-temporal dictionary

In order to capture the video signal information, the atoms have to be able to efficiently represent both the spatial image content, and the temporal information within groups of frames. In the same time, the dictionary has also to be designed to permit multiresolution decoding, and provide spatial and temporal scalability with minimal effort. In summary, an effective dictionary should mainly offer the following properties [14]:

- Multiresolution,
- Localization: the atoms are localized in space and frequency,
- Directionality: the atoms can be oriented along image singularities,
- Anisotropy: the atoms can deformed to match signal components.

Based on these requirements, the proposed encoder uses the following dictionary. Firstly, the spatial part of the atoms are generated from two mother functions, that satisfy *the localization* property: a 2-D Gaussian function $g_1(x,y) = \frac{1}{\sqrt{\pi}} e^{-(x^2+y^2)}$ and a wavelet-like function where one of the direction corresponds to the $2^n d$ derivative $g_2(x,y) = \frac{2}{\sqrt{3\pi}}(4x^2-2)e^{-(x^2+y^2)}$ of a gaussian function. The 2-D Gaussian is used to capture the low frequency spatial features, whereas clearly the wavelet-like function, besides nice localization properties and a small number of oscillations, is able to capture image singularities like edges and contours. This function has been shown to yield good approximation performance in natural image representation [15]. The overcomplete spatial dictionary is then generated by shifting, orienting, and scaling the two spatial mother atoms, as follows :

- Shift: $U_{(x_0,y_0)} g = g((x-x_0),(y-y_0))$
- Orientation: $U_\theta g_2 = g_2(r_\theta(x,y))$
- Scaling: $U_a g_1 = g_1(\frac{x}{a}, \frac{y}{a})$, $U_{(a_1,a_2)} g_2 = g_2(\frac{x}{a_1}, \frac{y}{a_2})$

Clearly, the number of translation, rotation and scaling has to be limited to avoid a prohibitive dictionary size, and thus limit the complexity of the search algorithm. In the current implementation, (x_0, y_0) sweeps

the whole image, and $\theta = \frac{i\pi}{16}$ where $i = 0, \ldots, 15$. The scaling factors finally are distributed on a logarithmic scale, as $a = 2^i$, with $i = 0, \ldots, \lfloor \log(\frac{image_{size}}{6}) \rfloor$.

Secondly, the temporal part of the dictionary is built on β-spline $\beta^n(t)$ functions, in order to efficiently capture motion information, and in the same time satisfy the multiresolution and localization properties. The order of $\beta^n(t)$ has to be larger than 2, to have a smooth transition and benefit from a limited support. Experiments have shown that $n = 3$ already offers good performance for group of pictures of a commonly accepted size of 16 frames. The temporal part of the dictionary is finally generated by shifting and scaling the β-spline $T_{t_0,s}\beta^3 = \beta^3(\frac{(t-t_0)}{s})$, similarly to the construction of the spatial part of the dictionary. In the current implementation, translation covers the complete group of frames (i.e., $t_0 \in [0..GOP_{size}]$), and the scaling follows a logarithmic distribution, $s = 2^i$ with $i = 0, \ldots, \lfloor \log(GOP_{size}) \rfloor$. It is noteworthy to notice that in the temporal scale $s = 2^i$, i refers to the number of frames that are filtered in the sequence. For example, $i = 0$ means that only 1 frame is considered, what happens in case of abrupt motion or scene change. It can be noted also that the present implementation does not contain any rotation of the temporal functions, this part is currently under study. Finally, the video dictionary is built on spatio-temporal separable functions, which combine the spatial and temporal sub-dictionaries to yield three dimensional atoms able to match the video signal structures.

C. MP3D encoder

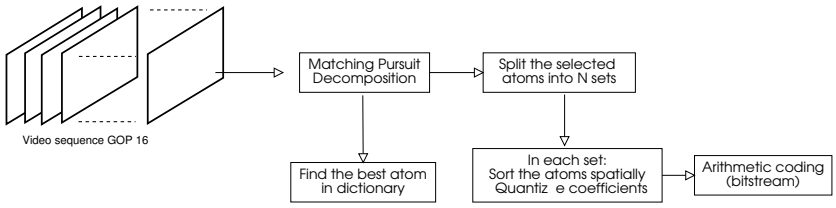

Fig. 1. Block diagram of the MP3D encoder

The complete MP3D encoder can be represented with the block diagram in Figure 1. The original video sequence is first segmented in group of 16 frames (GOP), whose length has been chosen as a good trade-off between encoding complexity, compression efficiency and decoding delay. The Matching Pursuit encoder iteratively selects the 3-D atoms $g_\gamma(x, y, t)$ from the dictionary that best match the residual GOP signal, in terms of the energy of the correlation coefficients, following (3). This iterative process continues until a stopping criteria is reached. Figure 2 (a) shows how the PSNR of the coded video sequence (*foreman* qcif) behaves in terms of iteration number i. Clearly, the rate of increase is very fast at the beginning, due to the nature of MP. The coefficients $c_i = \langle R_i, g_{\gamma_i} \rangle$ indeed decay exponentially with the iteration number i as shown in Figure 2 (b).

(a) PSNR vs iteration

(b) $\log(|c_i|)$ vs iteration

Fig. 2. PSNR and atom coefficient evolution vs iteration number

A classical implementation of the Matching Pursuit search would result in a quite high heavy computation process, since the encoder needs to browse the dictionary and perform the inner product between

each element and the residual signal for every MP iteration i. The current implementation of the MP3D uses a reduced complexity scheme, based on a Fast Fourier Transform.

Coefficients and atoms are then encoded in order to provide a flexible bitstream, but still with a high compression ratio. The embedded coding is achieved through the set-partitioned position map algorithm (SPPM), which is derived from the bit plane encoding. The atoms are first split into l sets according to their energy, where each set contains N_l contiguous atoms, and then spatially sorted to form the index-map. The first sets contain fewer elements than the other sets, but have larger global energies due to the properties of the MP decomposition (see Figure 2 (b)). The number of sets and their size is determined by the energy of the coefficients. The distribution of the coefficients in each set is found to be Laplacian, so uniform quantization is applied since it has been shown to be close to optimal [16]. Finally, the index-map of each set and its quantized coefficients are losslessly coded with an adaptive arithmetic coding scheme.

The decoding process is very simple. It simply consists in decoding the coefficients, and adding the 3-D atoms multiplied by the corresponding coefficients to reconstruct the video signal.

III. Scalability properties

Due to the multiresolution structure of the dictionary, MP3D streams are highly scalable in terms of spatial or temporal (i.e., frame rate) resolution. The geometric properties of the dictionary ensures very easy transcoding operations, such a single bitstream, can with no effort be decoded at any spatial resolution (as long as the re-scaling is isotropic) and various frame rate. For example, a coded video signal f of size $W \times H$ with a frame rate F can be spatially transcoded into a video signal $\tilde{f}$ of spatial resolution $\alpha W \times \alpha H$ at the same frame rate as follows :

$$\tilde{f} = \sum_{i=0}^{N-1} \alpha \; c_i \tilde{g_{\gamma_i}}, \tag{6}$$

where c_i are the atom coefficients and $\tilde{g_{\gamma_i}}$ corresponds to the atom g_{γ_i} after transcoding. Transcoding simply modifies the atom index $(p_x, p_y, p_t, s_x, s_y, s_t)$ which becomes $(\alpha p_x, \alpha p_y, p_t, \alpha s_x, \alpha s_y, s_t)$ where $\vec{p}$ and $\vec{s}$ respectively represents the spatio-temporal position and scale of the atom g_{γ_i}. Figure 3 illustrates an example of the spatial transcoding of the *Foreman* sequence at 200 kbps, scaled with a factor $\frac{1}{2}$.

In addition to spatio-temporal scalability, MP3D intrinsically provides SNR scalability thanks to the properties of the Matching Pursuit algorithm. The energy of the coefficients is exponentially decreasing along the iteration number. Therefore, simple truncation of the embedded bitstream produced by the proposed encoder still ensures that the decoder receives most of the signal energy for the available bandwidth.

(a) Original frame

(b) Decoded frame.

(c) Scaled by 0.5

Fig. 3. The 1^{st} frame in foreman decoded and transcoded

IV. Experimental Results

Performance of MP3D are now compared with state-of-the-art scalable video coding schemes, like MPEG-4 (FGS) and SPIHT-3D. The rate-distortion characteristics are first compared to SPIHT-3D for the video sequence *foreman* (qcif format), with GOP size 16. As shown in Figure 4, the PSNR quality is better for MP3D than for SPIHT-3D at low bit rates $[20 - 250]$ kbps. Note that both schemes offer nice scalability properties, with MP3D being more flexible however. When compared against MPEG-4 with spatial scalability, MP3D outperforms the multi-layer scheme by almost one dB at low bit rates. Finally, Figure 5 proposes a comparison with the state-of-the-art MPEG-4 with FGS scalability having the base layer coded at different bit rates (46, 60, 70) kbps for the same video sequence. When used with a base layer at 46 kbps for increased SNR scalability, MPEG-FGS loses up to 2.6 dB against MP3D at higher bit rates 250 kbps. When the base layer is coded at 60 kbps, FGS is slightly better than MP3D at low bit rates, but it loses a lot of flexibility in terms of scalability, since it obviously cannot serve bit rates lower than the base layer. It also loses its quality advantage at higher bit rates. Finally, visual comparisons also favors MP3D at low bit rates, since it provides less annoying artifacts than ringing in wavelet-based coding, or blocking in DCT-based coding.

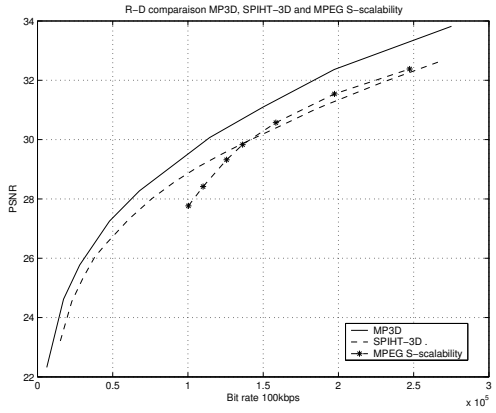

Fig. 4. R-D comparison between MP3D, SPIHT-3D and MPEG with S-scalability for qcif foreman 30fps

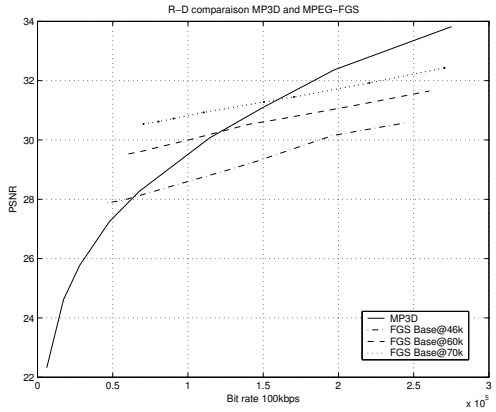

Fig. 5. R-D comparison between MP3D and MPEG-FGS for qcif foreman 30fps

V. Conclusions

This paper has presented a novel video coding scheme based on a Matching Pursuit algorithm. It has been shown to provide a highly flexible scalable bitstream, as a response by an ever increasing demand for adaptive coding structures. In the same time, it still favorably compares with state-of-the-art scalable

coders in terms of rate-distortion characteristics at low bit rates. Even if the current implementation can still be greatly improved, the MP3D structure thus represents a promising alternative for scalable video coding and streaming applications.

REFERENCES

[1] W. Li, "Overview of fine granularity scalability in mpeg-4 video standard," *IEEE Transactions on circuits and systems for video technology*, vol. 11, no. 3, pp. 301–317, March 2001.

[2] M. V. der Schaar and H. Radha, "A hybrid temporal-snr fine granularity scalability for internet video," *IEEE Transactions on circuits and systems for video technology*, vol. 11, no. 3, pp. 318–331, March 2001.

[3] D. Taubman and A. Zakhor, "Multirate 3-d subband coding of video," *IEEE Transactions on image processing*, vol. 3, no. 5, pp. 572–588, Sept 1994.

[4] S.-J. Choi and J. W. Woods, "Motion-compensated 3-d subband coding of video," *IEEE Transactions on image processing*, vol. 8, no. 2, pp. 155–167, Feb 1999.

[5] C. I. Podilchuk, M. S. Jayant, and N. Farvardin, "Three-dimensional subband coding of video," *IEEE Transactions on image processing*, vol. 4, no. 2, pp. 125–139, Feb 1995.

[6] J.-R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Transactions on image processing*, vol. 3, pp. 559–571, Sept 1994.

[7] B.-J. Kim and W. A. Pearlman, "An embedded wavelet video coder using three-dimensional set partitioning in hierarchical trees (3d-spiht)," in *IEEE Data compression conference*, March 1997, pp. 251–259.

[8] S. Chen and D. Donoho, "Atomic decomposition by basis pursuit," in *SPIE International Conference on Wavelets*, San Diego, July 1995.

[9] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decompositions," *IEEE Transactions on Information Theory*, vol. 47, no. 7, pp. 2845–2862, November 2001.

[10] R. Gribonval and M. Nielsen, "Sparse representations in unions of bases," IRISA, Rennes (France), Tech. Rep. 1499, 2003.

[11] A. C. Gilbert and S. Muthukrishnan and M. J. Strauss, "Approximation of functions over redundant dictionaries using coherence," in *Proc. 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2003.

[12] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, December 1993.

[13] P. Frossard, P. Vandergheynst, and R. M. F. i Ventura, "High flexibility scalable image coding," in *Proc. SPIE VCIP03*, Lugano (Switzerland), 2003.

[14] M. N. Do and M. Vetterli, "The contourlet transform:an efficient directional multiresolution image representation," *Submitted to IEEE Transactions on image processing*, 2003.

[15] P. Vandergheynst and P. Frossard, "Efficient image representation by anisotropic refinement in matching pursuit," in *Proceedings of IEEE ICASSP*, Salt Lake City UT, May 2001.

[16] G. J. Sullivan, "Efficient scalar quantization of exponential and laplacian random variables," *IEEE Transactions on information theory*, vol. 42, no. 5, pp. 1365–1374, Sept 1996.