Real-time Video Compression with 3D Wavelet Transform and SPIHT

Balázs Enyedi, Lajos Konyha, Csaba Szombathy, Kálmán Fazekas
Budapest University of Technology and Economics
Department of Broadband Infocommunications and Electromagnetic Theory
H-1117 Budapest, Goldmann György tér 3, Hungary
enyedi@mht.bme.hu, konyha@mht.bme.hu

Abstract – The following article presents a low-bitrate video comression method using 3 D wavelet transform and SPIHT algorithm. In contrast to conventional motion compensation algorithms, the procedure applies wavelet transform also for time redundancy. A 3 D version of our modified SPIHT algorithm is used for coefficient collection. These solutions can easiliy be adopted to the MPEG-4 compression standard. The possibilities of the Intel Pentium 4 architecture have also fully been exploited to provide real-time operation.

I. INTRODUCTION

The MPEG-4 standard enables the application of wavelet transform for video coding. This has not widely been used yet, because the related computational requirements exceeded the capabilities of the available hardware resources; instead, the DCT algorithm and other procedures included in the MPEG-2 standard have remained in use.

The SPIHT algorithm has revealed the video compression efficiency of wavelet transform. Several versions of the wavelet transform and the SPIHT algorithm have been used for still image compression since then, but the attempts to compress videos have failed due to the large computational requirements, since real time operation is inevitable in the latter case. Due to technological improvements and further development of algorithms, the application of this kind of transform has become reality.

II. THE OPERATION OF THE ALGORITHM

A. Applied procedures

The task is to code the signals arriving from the digitizing card and transfer them via a certain channel. The properties of the coded signal must be adopted to the channel characteristics. The available bandwidth is the most significant bottle neck in general. Only lossy compression methods can fulfil the requirements of low bitrate channels. Different channels with diverse characteristics and requirements are assigned to different applications. A good example is the contrast of digital video broadcasting (large image size, good quality, relatively large bandwidth) to videophone (small image size, medium quality, low bandwidth). The bitstream generated by the algorithm must be scalable in both time and space, as well as in bandwidth to fulfil all these requirements. The simultaneous application of the wavelet transform and the SPIHT algorithm fulfils these requirements. The major steps of the procedure are depicted in "Fig. 1".

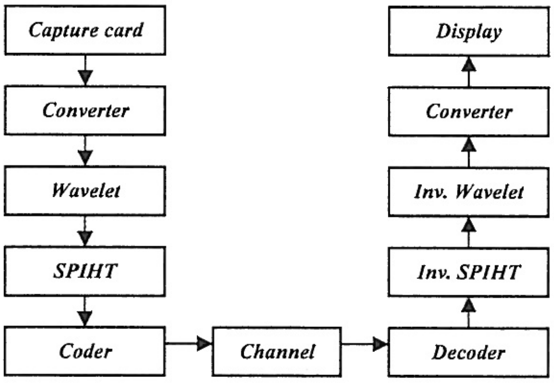

Fig. 1. The system's block diagram

The digital signal to be processed is generated from the arbitrary video signal by the digitizing card. The algorithm also requires preprocessing to obtain the digital signal in proper format (YUV, QCIF). Currently QCIF images are needed for real time signal processing. Wavelet transform is the first step of the actual signal processing. All frames to be simultaneously transformed must be received first to execute the 3 D transform, which can be performed on the basis of several one dimensional transforms. The next step is the extended version of the SPIHT algorithm to 3 D. The resultant bitstream must be transferred in the channel in accordance with the channel characteristics, which is ensured by the channel coder. The decoder on the counter side receives the signal that has passed through the channel and restores the data for the inverse SPIHT. The processes on the receive side are similar to those on the transmitter, with the difference that they are performed in reverse order. Finally, the restored film is dispayed.

B. Wavelet transform

The cosine transform applied in the MPEG-2 standard partially exploits the properties of the HVS, but does not take it into consideration with sufficient precision. The coefficients resulted by the transform split the concerned frequency range into equal subranges, the HVS however senses the individual subranges logarithmically. The wavelet transform, whose base functions can be obtained by shifting and expanding a mother function, helps to solve this issue as well. The frequency and time domain can simultaneously be investigated at an arbitrary specified place with the help of these. The obtained base functions are either low frequency-long duration or high frequency-short duration impulses.

The position of the window along the frequency axis is determined by variable ***a***, while variable ***b*** sets the spatial position. The width of the window is determined also by ***a***. After all, the base functions are as follows:

$$\psi_{a,b}=\frac{1}{\sqrt{a}}\psi(\frac{x-b}{a}), \qquad (1)$$

and the spectrum:

$$\sqrt{a}\Psi(a\omega)\exp(-jb\omega) \qquad a>0; a,b\in\Re, \qquad (2)$$

Continuous wavelet transform is defined as follows:

$$W(a,b)=\int_{-\infty}^{\infty} f(x)\frac{1}{\sqrt{a}}\psi(\frac{x-b}{a})dx, \qquad (3)$$

while the inverse transform:

$$f(x)=\frac{1}{C_\psi}\int_{-\infty}^{\infty}\int_{0}^{\infty}\frac{W(a,b)}{a^2}\frac{1}{\sqrt{a}}\psi\left(\frac{x-b}{a}\right)dadb, \qquad (4)$$

where

$$C_\psi=\int_{0}^{\infty}\frac{|\Psi(\omega)|^2}{\omega}d\omega, \qquad (5)$$

a constant dependant on the base function, ensuring that the energy of the original and the restored signal is identical.

It can be observed that wavelet base functions can be obtained from several kinds of mother functions. The properties of the actual task determine the wavelet base to be applied.

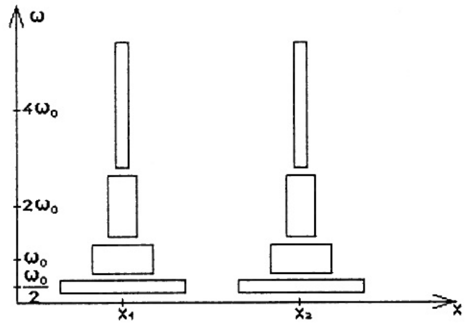

Fig. 2. Covering the time-frequency decomposition with base functions

It can be observed in "Fig. 2" that the sizes and positions of the rectangles covering the time-frequency decomposition continually change as variables ***a*** and ***b*** are modified. The rectangles of different parameters overlap, i.e. the effect of a point of the time-frequency decomposition is represented in several coefficients, which means that the same information is conained in more coefficients, making the transform redundant. Besides of all these, the transform of a one-dimensional signal will be two-dimensional, making the handling of the transformed signal harder. This also requires claculations of higher complexity than in case of one dimension.

Continuous wavelet transform has so far been introduced. As explained above, handling of the transformed function is difficult and the operation is redundant. Redundancy is a consequence of the procedure, while the simultaneous treatment of two continuos variables makes the handling difficult. Both issues can be solved by sampling the parameters of the base functions and using them as discrete variables afterwards. The base functions must fulfil the following criteria:

- They must cover the complete time-frequency decomposition in order to enable the restoration of the original signal using the coefficients resulted by the transform. If the base functions did not cover the entire time-frequency decomposition, there would be spacial points that are not included in any frequency component, their effect would not be represented in the transformed coefficients, i.e. the original signal could not be restored during the inverse operation.
- The redundancy of the sampling that ensures a complete coverage must be as small as possible (or zero, if possible). This means that the overlap of the rectangles representing the coefficients in the time-frequency decomposition must be the smallest (or no overlap, if possible).

Therefore, the first condition is that the rectangles cover the entire time-frequency decomposition, while the second is that they do not overlap. Both conditions are met if the rectangles are tangentially positioned beside each other.

Sampling shall be perfomed as follows:

$$\begin{array}{ll} a=a_0^{-m} & m,n\in Z \\ b=nb_0a_0^{m} & a_0,b_0\in\Re; a_0>0 \end{array}, \qquad (6)$$

The base functions obtained this way are:

$$\psi_{m,n}(x)=a_0^{-\frac{m}{2}}\psi\left(a_0^{-m}x-nb_0\right), \qquad (7)$$

The definition of the wavelet transform:

$$W(m,n)=\frac{1}{a_0^{\frac{m}{2}}}\int_{-\infty}^{\infty}f(x)\psi\left(a_0^{-m}x-nb_0\right)dx, \qquad (8)$$

The definition of the inverse wavelet transform:

$$f(x)=\sum_m\sum_n W(m,n)\psi_{m,n}(x), \qquad (9)$$

In the following it shall be shown how the discrete parameter base functions cover the time-frequency decomposition, with $a_0=2$ and $b_0=1$ in the example (***m*** determines vertical position of the window).

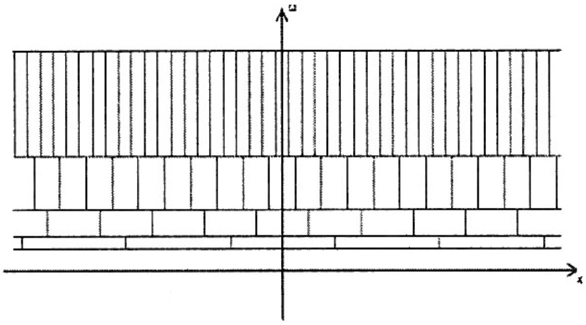

Fig. 3. Covering the time-frequency decomposition with discrete parameter base functions

The base functions obviously cover the complete time-frequency decomposition ("Fig. 2") and are tangential to each other, so the signal can be restored and the transform is free of redundances, i.e. this is an optimal sampling procedure.

The dimensions of the rectangles corresponding to the base functions of the wavelet transform vary along the frequency axis, yielding a worse frequency resolution and better spatial resolution at higher frequencies, while the frequency of the lower frequency components can be determined more accurately but not their spatial position. An analisys of this kind complies well with the properties of the HVS, giving the wavelet transform cruicial importance in image coding.

C. 3D wavelet transform

In order to reduce computational requirements, the transform is perfomed differently from the definition. In the first step the signal is filtered with the highest frequency filter and the corresponding low-pass filter. The obtained signal is decimated (every second sample is discarded), the signal resulted by high-pass filtering is stored, while the previous procedure is repeated on the low-pass foltered signal until the signal length is greater than one. The stored (high-pass filtered) and the final (low-pass filtered) signals represent the wavelet transfomed signal. Instead of extending the filter the signal is "shrinked" during the transform. Symmetric extension is applied at the image edges. The transform is performed on the 3 D signal as follows: the first step is executed on the first row of every frame (like a 1 D function), then the same is repeated on every column of all frames, finally the first step of the transform is performed on the corresponding pixels of every frame. Following this, the transform is continued with the low-pass filtered signals. The transform is permormed along the spacial coordinate axes in the first two cases, while in the last step along the time axis. As a consequence, all frames must be received to execute the transform, requiring a huge memory area that makes live transmission impossible. To overcome this, GOFs (Groups of Frames) are formed of the frames, and the transform is performed independantly on the individual groups. The delay of the system is determined by the number of frames constituing a group: too large groups lead to very high delays, while in case of too small groups the compression will not be efficient enough.

D. The SPIHT algorithm

The coefficients returned by the 3D wavelet transform are quantized according to SPIHT algorithm and are collected.

The SPIHT algorithm is based on the following observations:

- The most significant bits have the greatest influence on the picture quality, therefore these ones must be collected first, followed by the lower significant bits consecutively, in descending order.
- The position and value data of the coefficients must also be stored.
- Coefficients near to each other in a specific subband have similar properties.
- A certain coefficient is similar to the ones in the same position in the following upper subbands. If a coefficient in the low frequency subband has a relatively large amplitude, then the corresponding coefficients in the upper subbands are also expected to be large, therefore it is advisable to collect them one after the other.
- The coefficients in the lower frequency subbands have greater importance from the point of the HVS, therefore these ones must be collected first.

On the basis of all these, the SPIHT algorithm classifies the coefficients into sets. Insignificant sets (LIS), as well as significant (LSP) and insignificant (LIP) coefficients exist. First the list of insignificant sets is filled up with the position of the coefficients of the lowest subband, while the list of significant coefficients is empty. Coefficients in a specific position from the different subbands belong to a certain set. In the next step, the algorithm checks the most significant bits of the coefficients. If a significant coefficient is found in a set (i.e. whose concerned bit has a value of 1), then the corresponding set is split up to subsets, as well as to significant and unsignificant coefficients. The sign of the found significant coefficient is stored. Having investigated every set, the currently checked bits of the significant coefficients are stored. Following this, the complete procedure is repeated with the next most significant bits. The algorithm ends when every bit is stored, or the length of the generated bitstream reaches a maximal value dependant on the selected compression.

The 3D SPIHT algorithm differs from the 2 D one in the sense that the parent-offspring relations are defined differently in its spatial orientation tree. This is depicted in "Fig. 4". The offsprings of the coefficient represented by a little white dice in the corner are the 7 coefficients surrounding it. On lower levels 8 offsprings belong to a coefficient; these relations are indicated by arrows in the figure.

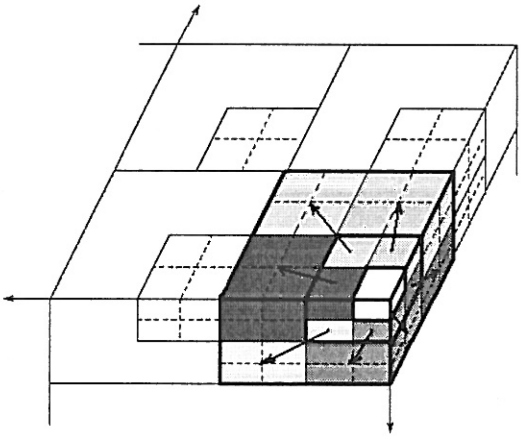

Fig. 4. Relationships in a 3D SPIHT

III. CONCLUSION

The implemented algorithms were tested on an IBM PC featuring the following characteristics:

- Intel Pentium 4 Processor, 2.4GHz
- 1 GB RAM
- Windows XP operating system

C++ programing language has been used for implementation with inserted Assembly routines. The latter one's source code was optimized by the Authors to Intel Pentium 4 architecture (SSE2 instruction set, the XMM registers enable 4 simultaneous floating point operations).

Daubechies 9/3 bases have been used for the wavelet transform in the spacial domain, while Haár bases for the time domain transforms. Symmetric extensions were applied at the edges.

The test patterns included 128 frames, each frame including 352×288 pixels, and the video displayed 30 frames in a second (i.e. the duration of a pattern was 4.27 s, which is the time limit for coding/decoding).

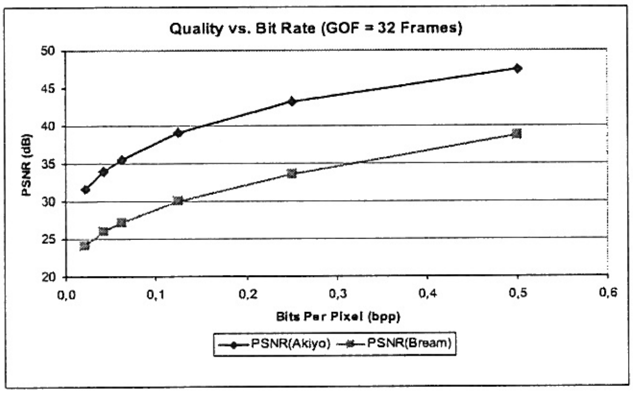

Fig. 5. Quality vs. bitrate

"Fig. 5" shows the picture quality vs. compression rate (0.021 bpp corresponds to 64kbit/s). The curves obviously indicate the improvement of quality as the bitrate increases; on the other hand, the „Bream" series contains more detailed frames. "Fig. 6" shows the computational time required by the individual steps of the algorithm vs. bitrate. The time needed for the wavelet and inverse wavelet transforms is practically independant of the bitrate, in contrast to the SPIHT and inverse SPIHT algorithms, whose required computational time increases with the bitrate. The SPIHT algorithm lasts about 2 s longer, which is equal to the time needed for filling up the structure shown in "Fig. 4" with data.

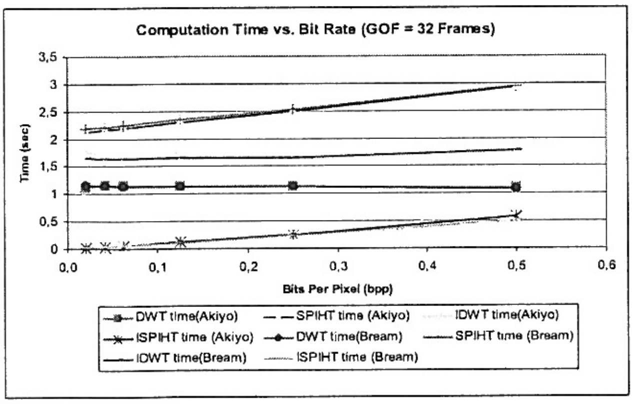

Fig. 6. Computational time vs. bitrate

"Fig. 7" indicates the quality in the function of the GOF size, while "Fig. 8" shows the computational time vs. the GOF size at constant bitrate. The curves indicate that the quality degrades and the time required by the inverse wavelet transform increases if the GOF size is too small.

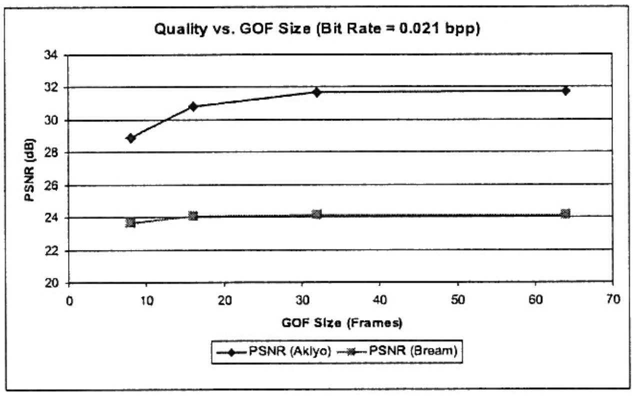

Fig. 7. Quality vs. GOF size

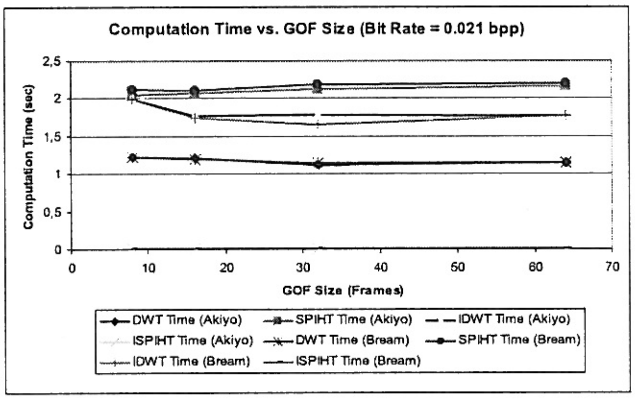

Fig. 8. Computational time vs. GOF size

Quality degradation is caused by the fact that the SPIHT algorithm exhibits higher efficiency for long signals, while small GOFs result in short signals in the time domain. The increase in the wavelet transform's computational time is due to fact that a long signal can be processed faster than in case of splitting it up to short sections and transformed accordingly.

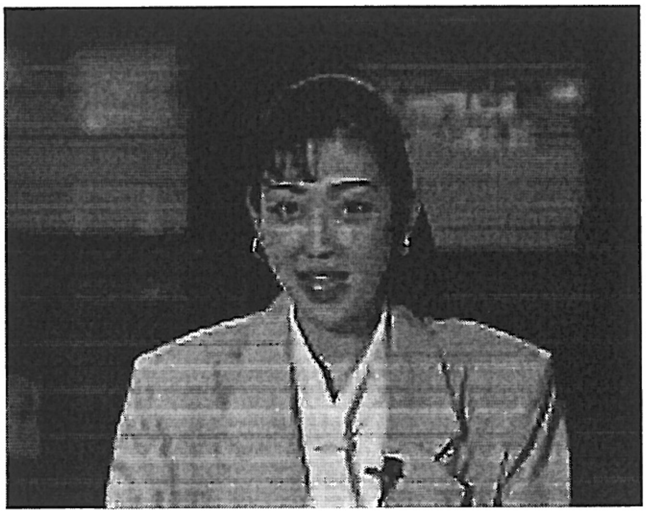

Fig. 9. Akiyo series, 0.021 bpp (64 kbit/s)

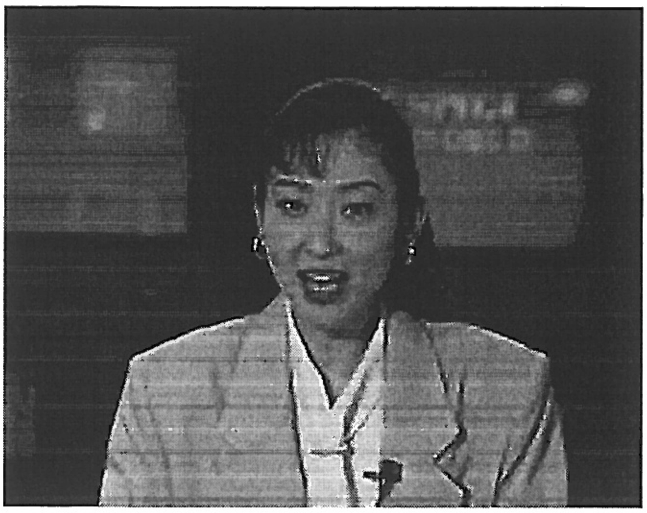

Fig. 10. Akiyo sorozat, 0.042 bpp (128 kbit/s)

"Fig. 9" and "Fig. 10" depict the Akiyo series, while "Fig. 11" and "Fig. 12" the Coastguard test series, compressed at different bitrates. It is obvious that the quality improves at higher rates, and further increasing the speed enables real time coding and decoding as well. ("Fig. 6").

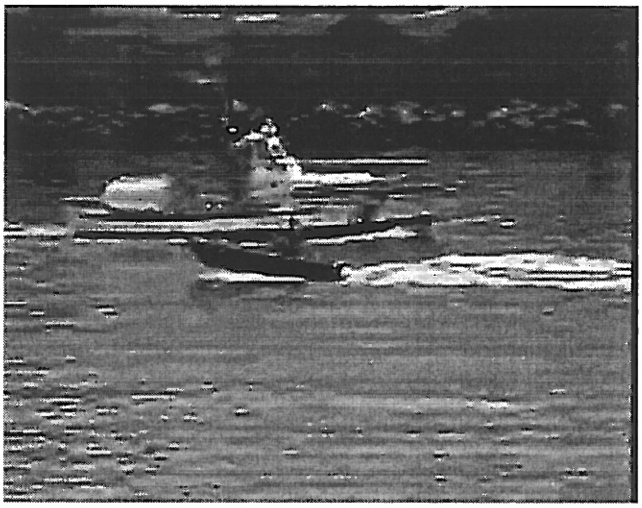

Fig. 11. Coastguard series, 0.0625 bpp (190 kbit/s)

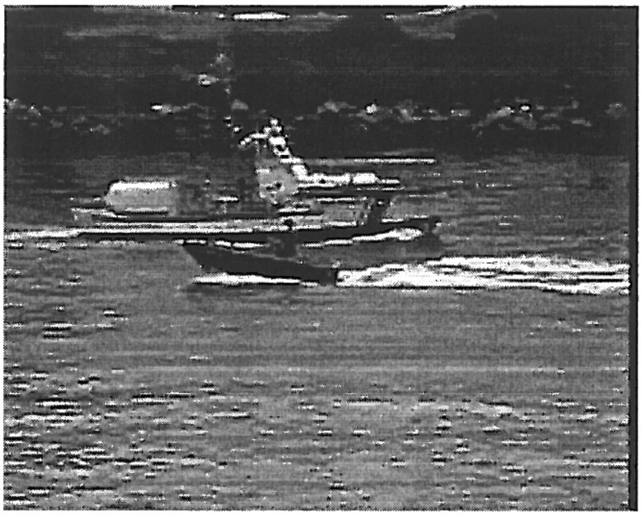

Fig. 12. Coastguard series, 0.125 bpp (380 kbit/s)

IV. REFERENCES

[1] L. Konyha, B. Enyedi, K. Fazekas; "Multimedia Distance Learning – Orthogonal Transformations", *EURASIP Conference on Digital Signal Processing for Multimedia Communications and Services,* Budapest, Hungary, Sept. 2001

[2] B. Enyedi, L. Konyha, K. Fazekas; "Using Wavelet Transform for Guiding Observation Cameras and Efficient Data Storage", *3rd COST #276 Workshop on Information and Knowledge Management for Integrated Media Communication,* Budapest, Hungary, Oct. 2002

[3] J. Turan, "Fast Translation Invariant Transform and Their Applications", *elfa Publ. H.,* Slovakia, 1999

[4] S. G. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation"

[5] Amir Said, William A. Pearlman; "A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees", *IEEE Transaction on Circuit and Systems for Video Technology, Vol.6, June 1996*

[6] V. Bottreau, M. Bénetičre and B. Felts, B. Pesquet-Popescu, "A Fully Scalable 3D Subband Video Codec"