Motion Compensated Video Compression with 3D Wavelet Transform and SPIHT

Balázs ENYEDI, Lajos KONYHA, Dr. Kálmán FAZEKAS

Dept. of Broadband Infocommunications and Electromagnetic Theory,
Budapest Univ. of Technology and Economics, Goldmann György tér 3, H-1111 Budapest, Hungary

enyedi@mht.bme.hu, konyha@mht.bme.hu, kfazekas@cyberspace.mht.bme.hu

Abstract. *The following paper introduces a low bitrate video coding method on the basis of 3D motion compensated wavelet transform and SPIHT algorithm. In contrast to the conventional algorithms applying motion compensation and differential coding, here wavelet transform is used to exploit the opportunities of time redundancy. For coefficient collection, the 3D version of SPIHT algorithm was selected from the various procedures developed for wavelet transform. Motion vectors are compressed, too (also by wavelet transform), therefore the time and spatial redundancy of coding is exploited here as well. These procedures result in effective video compressing and can easily be aligned to the MPEG4 standard.*

Keywords

Video coding, 3D wavelet transform, SPIHT, motion compensation, MPEG-4, motion vector compression.

1. Introduction

The MPEG-4 standard enables the application of wavelet transform during video coding. In addition, 3 D transform is also enabled to make use of time redundancy. As a result of time domain transform, conventional 3 D wavelet encoders that do not apply motion compensation generate a huge amount of high amplitude coefficients in the high frequency range in case of intensive movements (e.g. the camera moves). This degrades the efficiency of compression. The fact that in most cases even a simple motion compensation method can highly improve the compression ratio has already been revealed during the development of the MPEG 1-2 standards. On the basis of this principle, the encoder applying 3 D wavelet transform has been improved by a simple motion compensation method. The motion vectors are generated in a structure similar to that of the individual frames, therefore the application of 3 D wavelet transform for the compression of these proves to be sensible.

2. Video Coding Based on Wavelet Transform

2.1 Wavelet Transform

The cosine transform applied in the MPEG-2 standard partially exploits the properties of the HVS, but it does not take it into consideration with sufficient precision. The coefficients resulted by the transform split the concerned frequency range into equal sub-ranges, the HVS however senses the individual sub-ranges 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. Every coefficient resulted by the transform contains information on a certain section of the time and frequency domain, which can be depicted as a window in the time-frequency plane (Fig. 1). The base functions of the transform can be obtained by shifting and stretching a base function (Ψ).

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

$$\psi_{a,b} = \frac{1}{\sqrt{a}}\psi(\frac{x-b}{a}), \quad a>0; a,b \in \Re \tag{1}$$

Continuous wavelet transform is defined as follows:

$$W(a,b) = \int_{-\infty}^{\infty} f(x)\frac{1}{\sqrt{a}}\psi(\frac{x-b}{a})dx \cdot \tag{2}$$

Wavelet base functions can be generated from various base functions. The nature of a certain task determines the wavelet base to be selected. It is also the selected base that determines the subdivision of the time-frequency domain

(Fig. 1.). The transform is reversible if the base functions cover the complete phase plane, and is free of redundancies if there are no overlaps. Accordingly, the base functions of the transform must be chosen in a way that they seamlessly cover the plain but do not overlap, i.e. the signal can be restored while the transform shall be free of redundancies.

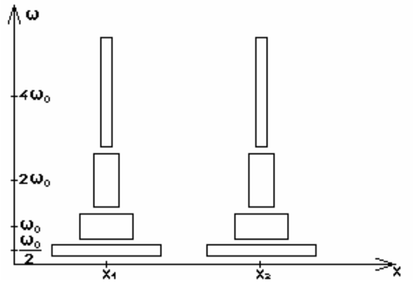

Fig. 1. Location of some base functions in the time-frequency plane.

In case of multiple dimension transform, the base functions will also be multi-dimensional. However, it is worth to choose such bases that are constituted by the product of 1 dimensional functions. In this case the transform can be performed by consecutively executing several 1 dimensional transforms, considerably reducing also the computational requirements this way. Continuous 3 dimensional wavelet transform is defined as follows:

$$W(a,b,c,d,e,f)=\iiint_V f(x,y,t)\frac{1}{\sqrt{ade}}\psi\left(\frac{x-b}{a},\frac{y-c}{d},\frac{t-d}{e}\right)dxdydt=$$

$$=\int\left(\int\left(\int f(x,y,t)\frac{1}{\sqrt{a}}\psi\left(\frac{x-b}{a}\right)dx\right)\frac{1}{\sqrt{d}}\psi\left(\frac{y-c}{d}\right)dy\right)\frac{1}{\sqrt{e}}\psi\left(\frac{t-d}{e}\right)dt \tag{3}$$

In case of discrete signals discrete wavelet transform is applied, which is obtained from the continuous transform by sampling the base functions.

2.2 SPIHT Algorithm

Coefficients returned by the 3D wavelet transform are quantized and collected according to SPIHT [9] algorithm.

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 sub-band have similar properties.
- A certain coefficient is similar to the ones in the same position in the following upper sub-bands. If a coefficient in the low frequency sub-band has a relatively large amplitude, then the corresponding coefficients in upper sub-bands are also expected to be large, hence it's advisable collecting them one after the other.
- The coefficients in the lower frequency sub-bands 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 sub-band, while the list of significant coefficients is empty. Coefficients in a specific position from the different sub-bands 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. 2. 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.

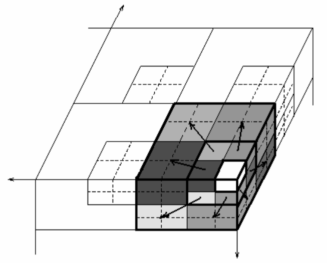

Fig.2. Relationships in a 3D SPIHT

3. The Operation of the Algorithm

3.1 Motion Compensated 3D Wavelet Transform

The motion compensation method complementing the original 3 D wavelet transform [1,4,6] must be inserted into the time domain steps of the transform. Fig. 3. depicts the block diagram of the algorithm, with a GOF (Group of Frames) size of 8 frames. Regarding the time domain transform, Haar base [8] has been chosen. Prior to the time domain transform, an attempt is made to increase the similarity between the input frames using motion compensation. The number of frames used in a filtering phase is equivalent to the length of the filters of the given base, i.e. this is the frame count that must be made similar to each other by motion compensation (the required computational performance and the number of motion vectors to be transferred is proportional to this). As a result, the application of a filter as short as possible is recommended. This induced our decision for Haar base which features an impulse response of a length of 2.

The blocks (LP, HP) below the motion compensation (MC_{ij}) in the block diagram correspond to the low pass and high pass filters of the time domain Haar wavelet. In case of exact motion compensation the high frequency components resulted by the high pass filtering will feature small amplitude, increasing the efficiency of the entire compression. An arbitrary base independent of the time domain can be chosen for the spatial transform. The procedure is continued after this (motion compensation, time and spatial domain transform) for the low frequency components. Having completed every step of the transform, a bitstream is generated from the obtained components by SPIHT algorithm [1, 5], which is also used for the quantization of the bitstream. The beginning of the stream comprises the most significant bits of the individual coefficients, followed by the lower significant bits in decreasing order. If this stream is interrupted somewhere, than the lower significant bits are rejected, i.e. the coefficients are quantized. Depending on the place of interruption, either constant bitrate or constant quality (varying bitrate) coding can be set. Finally, the quantized bitstream is lossless compressed by an entropy encoder.

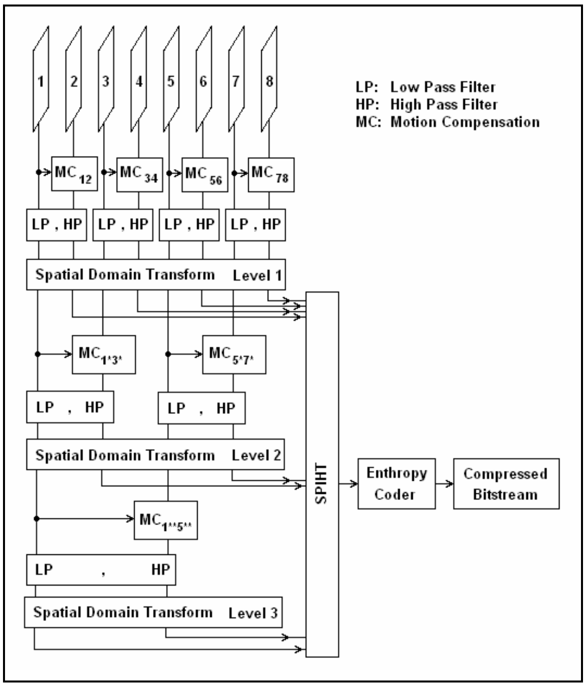

Fig.3. Motion Compensated 3D Wavelet Encoder

3.2 Motion Compensation

For the sake of simplicity, block based motion compensation method has been chosen from the several possibilities available (e.g. block based, mesh based, etc.) [2, 3]. This method works well for linear movements, but is unsuitable for handling more complex (e.g. rotating) movements. A motion estimation algorithm required for these movements is complicated, giving a further reason for deciding for the block based solution. The selected motion compensation method permits the selection of the block size as well as the dimensions of the search window. Also the less consistent movements can accurately be described if the block sizes are set to smaller, but the increased number of blocks resulted by this approach leads to an increased number of motion vectors, i.e. increases the amount of side information to be transferred.

The amount of side information also depends on the extent of the search domain. Bigger search domains may result longer motion vectors, enabling the handling of faster movements, though leading to more bits in the description of the vectors (in case of an image in CIF format, assuming a block and search domain size of 8x8 and a frame rate of 30 fps, the information content of the vectors exceeds 285kbps). By doubling the size of the search domain, the amount of side information increases by approximately 100 kbps (in the case of the above example), while the computation requirement of motion estimation quadruples. This example reveals that also the motion vectors must be compressed in case of low bitrate coding, because the decisive amount of the bandwidth is occupied by the side information, while at very low data rates the transmission of even these becomes impossible. If longer bases were used for time domain transform, more frames would have to be made similar to each other, i.e. the number of motion vectors would multiply.

Fig. 4. Foreman and Coastguard series.

3.3 Compression of Motion Vectors

The motion vectors are generated in a structure similar to that of the individual frames, therefore the application of 3 D wavelet transform for the compression of these proves to be sensible. If the motion field is consistent, the motion vectors corresponding to the blocks that are close to each other have similar values (just like the pixels of the frames). The number of the motion vectors is much smaller than the pixel count, therefore the compression rate for the motion vectors is much worse. This ratio is further degraded by the requirement that the vectors must be transferred without losses. Haar base was selected for the wavelet transform of the motion vectors, and one vector was stored by less than 2 bits (depending on the contents of the image). The operation of the SPIHT algorithm is stopped when the last important bit has been coded, too. The bitstream is cut upon achieving a constant quality, not a constant transmission speed. The more movements occur in a video and the more dynamic it is, the less consistent the motion field will be, therefore more bits will have to be used for coding the motion vectors.

4. Conclusion

Random access has also been taken into consideration during the implementation, which sets up requirements related to the GOF size. The frames of a GOF can simultaneously be coded or decoded, therefore, in case of random access the beginning of a GOF must definitely be waited for. If a GOF includes a huge number of frames, the access time increases proportionally with the frame count. The GOF size has therefore been chosen as 16, which corresponds to about 0.5 sec in case of 30 fps, yielding an expected random access time of 0.25 sec.

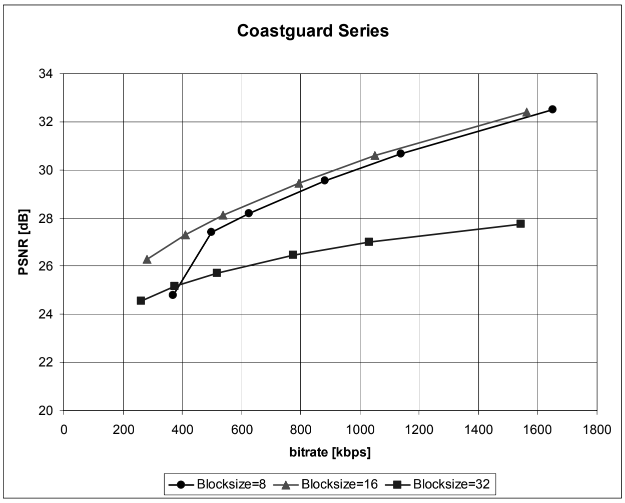

Fig. 5. Quality vs. Bitrate.

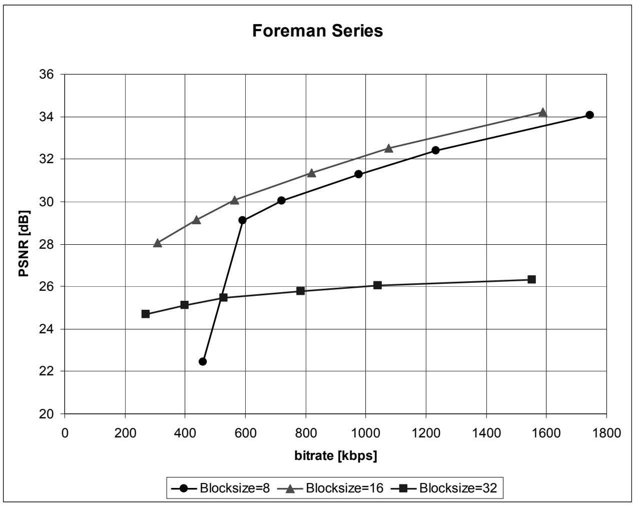

Fig. 6. Quality vs. Bitrate.

The results shown (Fig. 5. and Fig. 6.) were obtained by the Coastguard and Foreman test sequences (Fig. 4.) in CIF format (352x288 pixels/frame, 30fps). A Daubechies 9/3 [7] base was used for the spatial wavelet transform, while Haar base for the time domain transform. Symmetrical extension was applied at the edges. The size of the search domain of the motion compensation was 8x8 pixels, while the block size was varied between sensible limits.

At the introduction of the results the quality is investigated in the function of the bitrate for different block sizes. Quality is characterized by the PSNR (Peak Signal to Noise Ratio) value, here given in dB. The horizontal axis shows the bitrate in kbps. Experiments were performed on various block sizes (8x8, 16x16, 32x32 pixels).

The figures indicate that the coding resulted bad quality in case of 32 pixel blocks. The reason of this is that the chosen block size is too large for handling the fine movements, therefore motion compensation does not operate correctly. The individual movements can already be well described by blocks of 8 pixels, but considerably huge amount of motion vectors are generated in this case, requiring a relatively large transmission bandwidth. In case low bitrates this bandwidth is not available, therefore the motion vectors cannot be properly transferred, leading to the abrupt degradation of the picture quality. The application of blocks of 16 pixels provided optimal results with both series, because the movements can already be described with sufficient accuracy in this case, while the number of the generated motion vectors still remains acceptable.

In the Coastguard sequence, most of the image is constituted by the background, while the two ships occur in a smaller area. The motion vectors of the background are mostly identical as a result of the movement of the camera, yielding a consistent motion field that can be stored by a relatively small amount of bits. There is little time and spatial redundancy on the waving surface of the water, therefore high bitrate is required to achieve good quality.

Considering the Foreman sequence, also the man’s head is moving in every direction besides the camera. The motion vectors therefore exhibit considerable variation, and the quality degrades significantly in case of low bitrate and a block size of 8 x 8 pixels. This is because the data speed is not sufficient even for the transmission of the motion vectors. The best results were obtained for a block size of 16 x 16 pixels in case of both sequences, because these blocks are already suitable for handling the movements adequately but do not yield too many motion vectors yet.

Only the innovative elements were implemented and investigated in the procedures introduced in this paper, the charts show these results. According to Fig.3, the last step of compression is entropy coding, which is a simple lossless compressing method (e.g. Huffman, arithmetic coding). The efficiency of compression can be further improved by applying the entropy coding mentioned.

References

[1] ENYEDI, B., KONYHA, L., FAZEKAS, K. Fast video compression based on 3D wavelet transform and SPIHT. In 7th *COST 276 Workshop on Information and Knowledge Management for Integrated Media Communication*. Ankara (Turkey), 4-5 November 2004.

[2] TURAN, J. *Fast Translation Invariant Transform and Their Applications*. Kosice, Slovakia: elfa Publ. H. ISBN-80-88964-19-9, pp.156, 1999.

[3] TURAN, J., FAZEKAS, K., GAMEC, J., KOVESI, L. Railway station crowd motion estimation using invertible rapid transform. *Image Processing & Communications, International Journal*, 1997, vol.3, no.1-2, pp.12-23.

[4] MALLAT, S. G. A Theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Pattern Anal. Machine Intell.*, 1989, vol. 11, pp. 674-693.

[5] AMIR SAID, PEARLMAN; W. A. A New fast and efficient image codec based on set Partitioning in Hierarchical Trees. *IEEE Transaction on Circuit and Systems for Video Technology*, vol.5, June 1996, pp 243-250.

[6] BOTTREAU, V., BENETICRE, M., FELTS, B., PESQUET-POPESCU, B. A fully scalable 3D subband video codec. *Image Processing*, vol.2, 2001, pp. 1017-1020.

[7] DAUBECHIES, I. Ten lectures on wavelets. *CBMS-NSF Lecture Notes nr. 61*, SIAM, 1992.

[8] http://www.bearcave.com/misl/misl_tech/wavelets/haar.html

[9] AMIR SAID, PEARLMAN, W. A. A new fast and efficient image codec based on set partitioning in hierarchical trees. *IEEE Transactions on Circuits and Systems for Video Technology*, vol.6, 1996, pp. 243-250.

About Authors...

Lajos KONYHA was born in Budapest, Hungary, on February 2, 1978. He received the M.Sc. degree in electrical engineering from the Budapest University of Technology and Economics, Faculty of Electrical Engineering and Informatics, Dept. of Microwave Telecommunications, in 2001. Since 2001 he has been a Ph.D. student at the Budapest University of Technology and Economics, Dept. of Broadband Infocommunications and Electromagnetic Theory, Media Technology Laboratory and Rohde & Schwarz Reference Laboratory. As a Ph.D. student, he deals with one and more dimensional signal processing, transform image coding, video and image compression.

Balázs ENYEDI was born in Budapest, Hungary, on January 13, 1978. He received the M.Sc. degree in electrical engineering from the Budapest University of Technology and Economics, Faculty of Electrical Engineering and Informatics, Dept. of Electric Power Engineering, in 2001. He worked for Matáv Rt. (Hungarian Telecommunications Company Limited), IT Directorate, Customer Care Systems Dept. as IT Project Manager from 2001 to 2003. He dealt with CRM and OSZTR (Nationwide Computerized Information Bureau System) systems. Since 2001 he has been a Ph.D. student at the Budapest University of Technology and Economics, Dept. of Broadband Infocommunications and Electromagnetic Theory, Media Technology Laboratory and Rohde & Schwarz Reference Laboratory. As a Ph.D student, he deals with image and signal processing methods, segmentation, video and still image compression.

Kálmán FAZEKAS (Prof., PhD. Dr.) studied electronics at TU Budapest, now is working as professor at TU Budapest. His research interests include multimedia signal processing, teleeducation and image coding. Research activities: digital image processing/coding: multiresolution decomposition methods, VLBR coding, object-based coding/processing (MPEG-4, H.264), motion estimation; image communication, DVB, multimedia technology, distance education.

RADIOENGINEERING REVIEWERS

April 2006, Volume 15, Number 1