A New Lossy 3-D Wavelet Transform for High-Quality Compression of Medical Video *

Gregorio Bernabé, José González, José M. García
Dpto. Ingeniería y Tecnología de Computadores
Universidad de Murcia
30071 MURCIA (SPAIN)
{gbernabe, joseg, jmgarcia}@ditec.um.es

José Duato
Dpt. Inf. de Sistemes i Computadors
Universidad Politécnica de Valencia
46071 VALENCIA (SPAIN)
jduato@gap.upv.es

Abstract— **In this paper, we present a new compression scheme based on applying the 3-D Fast Wavelet Transform, to code medical video. This video has special features such as its representation in gray scale, the small amount of interframe variations, and the quality requirements of the reconstructed images. These characteristics as well as the social impact of desired applications deserve the design and implementation of coding schemes especially oriented to exploit its features. We analyze different parameters of the codification process, such as the utilization of different wavelets functions, the number of steps this function is applied, the way the thresholds are chosen, and the selected methods in the quantization and entropy encoder. Our coder achieves a good trade-off between compression ratio and quality of the reconstructed video. These results are better than MPEG-2, without the complexity of motion compensation.**

Keywords— **Telemedicine, medical video, high-quality, lossy compression, 3-D wavelet transform.**

I. Introduction

In the last few years, there has been a considerable increase in the volume of medical images and video generated in hospitals. The medical multimedia information is different from other multimedia data because of its particular properties. There are legal and strict regulations applied to medical multimedia information, since the health of a patient depends on the correctness and accuracy of this information. Moreover, the integrity, confidentiality and security of medical data is crucial to protect it from accidental or malicious alteration during the interchange and storage. Another critical property is that the information related to a patient must be available in a short period of time, whenever or wherever is required, and especially, in the case of an emergency.

Most of the medical history of a patient must be kept and stored since legislation requires all captured healthcare information to be preserved for a certain period of time (typically 5-10 years) before it can be deleted. Thus, hospitals must deal with very high storage requirements. On the other hand, telediagnosis is becoming a popular technique among hospitals. A doctor may ask for the advice of a colleague who works in another hospital, and even another country, by means of real-time transmission of medical images and video. Due to the huge amount of transmitted data, high-bandwidth networks are needed in order to maintain the quality of the video and allow a correct diagnosis. In both cases (storage and transmission), compression techniques are used to drastically reduce the amount of information that needs to be handled. Finally, the quality of the compressed data must be good enough to allow for a correct diagnostic when it is reconstructed.

Nowadays, higher compression ratios can be obtained by means of lossy compression techniques (JPEG [21], MPEG [17]), but radiologists are very reluctant to use them as they might introduce compression artifacts, which could complicate diagnosis. Normally, doctors prefer to use lossless compression techniques (JPEG-LS [10]) so that the quality is preserved. However, lossless compression achieves com-

This work has been partially supported by the Spanish CICYT under grant TIC97-0897. The video sequences have been donated by the Hospital Recoletas (Albacete, Spain).

pression ratios significantly lower than those achieved by lossy techniques. Thus, following the legal rules and keeping medical video for ten years may become prohibitive for most hospitals due to the storage requirements. In addition, the constant increase of the network traffic may difficult the use of tele-diagnosis if images are not sufficiently compressed. All of this makes attractive the research on new lossy compression techniques especially oriented to exploit the behavior of medical video, which is usually encoded in gray scale, using just 1 byte per pixel, and presents very small interframe variations.

Many video compression schemes have been proposed, but the scheme developed by the Moving Pictures Experts Group (MPEG) is the most outstanding compression scheme, and has emerged as a widely accepted industry standard. However, MPEG presents some problems when it is applied to compress medical video, since it cannot fully exploit the aforementioned characteristics. First, it is focused on color video, and the way it represents this, through YUV components, is not appropriate for gray-scale video. Second, MPEG-2 applies the direct cosine transform (DCT) by dividing the image in blocks of 8×8 pixels. This causes the presence of visible artifacts in the border of blocks which influences the quality of the reconstructed medical video. Also, the Gibbs effect [2] is one of the more common artifacts that also afflicts MPEG. It is caused by the DCT used to digitize chrominance and luminance information. Nowadays, a very attractive area of research concerns the proposal and evaluation of different transform functions that may overcome the limitations that the DCT used by MPEG-2 presents for some particular type of video. The Wavelet Transform has emerged as an attractive alternative to the DCT.

Wavelet techniques have recently generated much interest in applied areas such as FBI fingerprint compression [3], modeling the joint statistics of images [18], denoising noisy data [6], etc. Also, the wavelet transform has been mainly applied to image compression. Several coders have been developed using 2-D wavelet transform [11][1][16]. Finally, the 2-D wavelet transform has been used for compressing video [8].

In [14] and [15], Muraki introduced the idea of using 3-D wavelet transform to efficiently approximate 3-D volumetric data. This idea was developed by Ihm and Park [9] for achieving a good trade-off among compression ratio, quality and fast random access. Since one of the three spatial dimensions can be considered similar to time, Chen and Pearlman developed a three-dimensional subband coding to code video sequences [4].

In this work, we present and evaluate a new lossy video compression scheme, based on the use of the Fast Wavelet Transform (FWT) [7][19][20]. This new technique achieves high compression ratios with an excellent quality, so that medical doctors cannot find differences between the original and the reconstructed video. The main differences between our work and previous work are the following:

- Our proposals are mainly focused on medical video, exploiting its particular properties.
- We propose a new coding scheme for three-dimensional wavelet, which exploits both the spatial and the temporal redundancies.
- We evaluate the performance achieved by different wavelet mother functions and the number of steps that this function is applied, in order to find which one provides the best trade-off between quality and compression ratio.
- We propose and evaluate two ways of thresholding: the percentile policy and the discarding of the less significant bits of all wavelet coefficients.
- We propose a quantizer where the number of bits needed by each pixel coefficient to be encoded depends on the layer that this pixel belongs to.
- We use the run-length code and the Huffman code in the entropy encoder.

The rest of this paper is organized as follows. Section 2 summarizes the theory behind wavelets. In Section 3 we describe our proposed codification method using the 3-D wavelet transform. We present the main details of each component of the method. Experimental results with some test medical video are analyzed in Section 4, and conclusions as well as future research directions are outlined in Section 5.

II. The Wavelet Transform Foundations

The basic idea of the wavelet transform is to represent any arbitrary function f as a weighted sum of functions, referred to as wavelets. Each wavelet is obtained from a mother wavelet function by conveniently scaling and translating it. The result is equivalent to decomposing f into different scale levels (or layers), where each level is then further decomposed with a resolution adapted to that level.

In a multiresolution analysis, there are two functions: the mother wavelet and its associated scaling function. Therefore, the wavelet transform can be implemented by quadrature mirror filters (QMF), $G = g(n)$ and $H = h(n)$ $n \epsilon Z$. H corresponds to a low-pass filter, and G is a high-pass filter. The reconstruction filters have impulse response $h^*(n) = h(1-n)$, and $g^*(n) = g(1-n)$. For a more detailed analysis of the relationship between wavelets and QMF see [12].

The filters H and G correspond to one step in the wavelet decomposition. Given a discrete signal, s, with a length of 2^n, at each stage of the wavelet transformation the G and H filters are applied to the signal, and the filter output downsampled by two, generating two bands, G and H. The process is then repeated on the H band to generate the next level of decomposition, and so on. This procedure is referred to as the 1-D Fast Wavelet Transform (1-D FWT).

It is not difficult to generalize the one-dimensional wavelet transform to the multi-dimensional case [12]. The wavelet representation of an image, $f(x, y)$, can be obtained with a pyramid algorithm. It can be achieved by first applying de 1-D FWT to each row of the image and then to each column, that is, the G and H filters are applied to the image in both the horizontal and vertical directions. The process is repeated several times as in the one-dimensional case. This procedure is referred to as the 2-D Fast Wavelet Transform (2-D FWT).

III. The 3-D FWT Encoder

A. Direct 3-D FWT

As in 2-D, we can generalize the one-dimensional wavelet transform for the three-dimensional case. Instead of one image, now there is a sequence of images. Thus a new dimension has emerged, the time (t). The 3-D FWT can be computed by successively applying the 1-D wavelet transform to the value of the pixels in each dimension.

It is important to note the potential of such a transform for medical video. As pointed out before, medical video sequences have little movement. Consequently, applying the 1-D FWT in the time axis to a sequence divides the original signal into two equal sets of frames, where the first one includes the average signal and the second one represents the details. Due to the lack of movement, the coefficients of the second set of frames become very good candidates to be neglected in the thresholding phase (since their values will be close to zero), allowing higher compression ratios.

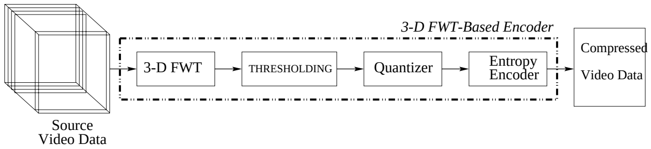

Fig. 1. 3-D FWT-based encoder processing steps

It is common in wavelet compression to recursively transform the average signal. The number of transformations performed in each dimension depends on several factors such as the amount of compression desired, the size of the original video and the mother wavelet function. In general, the higher the desired compression ratio, the more times the transform is performed. Note that applying the wavelet transform too many times may have a significant impact on quality. Hence, this parameter must be carefully chosen.

In this paper, we firstly consider Daubechies W_4 [5] as the mother wavelet function. We have chosen this function because some previous works have proved its effectiveness. However, we have also tried other mother wavelet functions such as Haar, Daubechies W_8, a bathlet wavelet [13] and a hybrid between W_4 (for x and y dimensions) and W_8 (for time dimension), in order to evaluate the impact of the mother wavelet function on final performance.

Figure 1 shows the key processing steps involved in the 3-D FWT-based lossy compression method.

B. Thresholding

After decorrelating the original frames by applying the 3-D FWT, the next step compresses frames by discarding those coefficients whose value does not provide enough information. In order to do that, a threshold (p) is chosen and the next function is applied to the decorrelated coefficients:

$$f_{x,y,t} = \begin{cases} 0 \text{ if } d_{x,y,t} < p \\ d_{x,y,t} \text{ if } d_{x,y,t} \geq p \end{cases}$$

where p is a x-percentile of the set of all wavelet coefficients. For instance, if we want to apply the 90-percentile, p must be chosen so that 90% of the wavelet values are under p, and thus, according to the previous function, they will set to zero. We will refer to p as the threshold. Choosing the adequate threshold has a significant impact on final compression ratio and quality.

C. *Quantization and Entropy Encoding*

Now, we need to quantize the non-zero wavelet coefficients before encoding them. Basically, the task performed in this step involves the transformation of the floating point coefficients into unsigned integer coefficients. To carry out this conversion, it is necessary to specify how many bits are needed to encode each of the wavelet layers. Each frame is independently processed. The number of required bits depends on the number of wavelet transforms applied. Finally, we have carried out another improvement after the quantization step. We have combined the previous thresholding policy with another one that discards the less significant bits of all wavelet coefficients after the quantization step. This yields a higher compression ratio, only slightly decreasing quality.

Finally, an entropy encoding step is performed: a run-length compression is applied to the binary representation of the integer coefficients, exploiting the presence of a large amount of zeros present in the video. After applying the binary run-length, another entropy encoder is applied, the Huffman encoding, in order to increase even more the compression ratio without affecting the video quality.

IV. Performance evaluation

In this section we evaluate the performance of our proposed compression scheme for two different medical scenes.

A. *Measuring the quality of compressed video*

We have used the following measurements to evaluate the quality of the compression process using the 3-D FWT:

a) Compression ratio.

b) Quality of the decoded video sequence.

The compression ratio is computed as the ratio between the original and the compressed data sizes.

The numerical evaluation of the quality of the compressor is achieved by computing the peak signal-to-noise ratio (PSNR) in the reconstructed video. The $PSNR$ of a reconstructed video has been calculated by computing the arithmetic mean of the $PSNR$ for all the frames of the video. In addition to the PSNR value, we have visually tested the quality of reconstructed video.

Finally, and for comparison purpose we have also recorded the execution time of the compressor.

B. *Environment Workbench*

The evaluation has been carried out on a Intel Pentium-III 450 MHz bi-processor with 256 Mbytes of RAM, but we have only used one processor. The operating system was Linux 2.2.12-20smp. The entire video compressor and decompressor have been written in the C programming language.

As input tests, we have compressed and decompressed two medical video sequences coded in gray scale (*heart* and *catheter*), each of them composed of 256 frames with a resolution of 512 by 512 pixels, whose total size is 64 Mbytes. These videos have been divided into four blocks of 64 frames. The *heart* sequence shows the movement of the heart and surrounding regions while the *catheter* sequence shows the introduction of a catheter.

C. *Results*

First of all, we compare the perfomance of the different mother wavelet functions evaluated in this work: Haar, Daub-4 and Daub-8 for the *heart* and *catheter* sequences. Table I and Table II show the results for these evaluations. Each cell in the table shows the compression ratio (top value) and the $PSNR$ (bottom value). The results are presented for different configurations with the following parameters:

- Parameter b represents the number of discarded bits in the quantization process.
- Parameter per-X indicates the X-percentile that has been used for the thresholding step.
- Parameter 2-t or 3-t indicates whether the wavelet transform has been applied 2 or 3 times in each of the axis (time,x,y). We have discarded the results obtained with just one step since the compression ratio obtained is lower than that for the other configurations. Also, applying the wavelet transform

four or more times considerably decreases the quality, and therefore these results are not considered in this work.

Comparing the number of times the 3-D FWT is applied for the *heart* sequence, we can observe that applying the transform 3 times increases the compression ratio but clearly affects the $PSNR$, driving to video sequences with poor quality, so we consider that for *this medical video*, more than 2 applications of the 3-D FWT is not worth. However, for the *catheter* sequence, applying the transform 3 times obtains a better trade-off between compression ratio and quality.

For the *heart* sequence, Daub-4 obtains the best trade-off between compression ratio and quality for all the analyzed configurations. The *catheter* sequence has less interframe variations than the *heart* sequence, and therefore our coding scheme achieves a better trade-off between compression ratio and quality. For the *catheter* sequence, the haar wavelet function slightly outperforms the daub-4, whereas the quality of the Daub-8 is very similar to Daub-4. However, taking into account the execution time, the Daub-8 implementation is much more costly (around twenty percent), since it uses more filters causing an increment in the number of floating point operations.

We have also evaluated the performance of the bathlet mother wavelet function and the hybrid wavelet between Daub-4 and Daub-8. Results for these particular implementations are not shown in this work since they are very similar to the results obtained by the Daub-4 function.

Analyzing the results for the different percentiles and number of bits discarded we can also conclude that the more bits and coefficients are discarded the more compression ratio is achieved. However, as the number of discarded coefficients increases, the $PSNR$ drops significantly, so an optimal configuration must be chosen in order to obtain a reasonable compression ratio without influencing the quality. Up to now, we can see that a per95 or per96 with b=3, is the best choice, but it is necessary to carry out more experiments.

The selection of such an optimal configuration is beyond the scope of this work, and is the subject of ongoing research.

Heart	Haar		Daub-4		Daub-8	
	2-t	3-t	2-t	3-t	2-t	3-t
b=2-per93	7,53	7,68	7,76	8,00	8,14	8,20
	39,77	34,60	39,97	36,63	38,95	39,15
b=2-per95	9,49	9,76	9,68	9,78	9,87	9,84
	38,17	34,06	37,83	35,77	37,28	37,68
b=2-per96	10,93	10,76	11,6	11,08	11,71	11,48
	37,14	33,75	36,61	35,20	36,05	36,72
b=2-per97	14,40	13,20	14,2	13,73	14,10	13,96
	35,25	33,21	34,78	34,30	34,65	35,47
b=3-per93	9,33	9,61	9,49	9,98	9,89	10,17
	39,31	34,41	39,15	36,34	38,52	38,69
b=3-per95	11,72	12,12	11,78	12,10	11,91	12,24
	37,90	33,93	37,52	35,59	37,02	37,38
b=3-per96	13,31	13,48	13,81	13,53	13,82	13,95
	36,95	33,63	36,42	35,06	35,89	36,52
b=3-per97	16,76	16,08	16,57	16,39	16,64	16,54
	35,18	33,13	34,68	34,22	34,09	35,37

TABLE I

Experimental results with Haar, Daub-4 and Daub-8 for heart

D. Comparison with MPEG-2

The *heart* sequence has been compressed using MPEG-2 for comparison purposes, allowing it to achieve the maximum possible compression ratio. This compressor obtains much better compression ratios than our compressor. However, the reconstructed video experiences the different aforementioned problems (blocking effect, softening). In a second experiment, we force MPEG-2 to obtain the same compression ratio as our 3-D FWT, in order to compare the quality of both schemes. MPEG-2 shows $PSNR$ between 28 and 30 (which are consistent with the results presented in [4]), clearly lower than that achieved in our implementation (still appearing some of the previous problems). In addition, the MPEG-2 algorithm is much more complex since, besides applying a transform (the DCT instead of the FWT), it also performs motion compensation, which slows down the compression process (around ten percent) compared to our proposal.

V. Conclussions and future work

In this work, we have presented and evaluated a new compression scheme based on applying the 3-D Fast Wavelet Transform, focused on medical video.

Catheter	Haar		Daub-4		Daub-8	
	2-t	3-t	2-t	3-t	2-t	3-t
b=2-per93	9,55	9,51	9,69	10,07	9,62	10,01
	40,01	40,53	38,88	39,41	38,87	39,43
b=2-per95	11,74	12,06	11,95	12,09	11,84	11,96
	37,95	38,83	36,58	37,65	36,80	37,84
b=2-per96	14,13	13,95	14,15	14,04	14,17	13,94
	36,10	37,76	35,22	36,46	35,28	36,71
b=2-per97	16,58	16,86	16,68	17,18	16,71	17,38
	34,27	36,22	33,66	35,14	33,49	35,25
b=3-per93	11,58	11,75	11,66	12,23	11,61	12,13
	39,58	39,96	38,45	38,98	38,46	39,02
b=3-per95	14,29	14,94	14,47	14,87	14,33	14,69
	37,72	38,50	36,34	37,39	36,57	37,57
b=3-per96	16,60	17,20	16,68	17,30	16,60	17,21
	36,01	37,52	35,09	36,27	35,17	36,52
b=3-per97	19,33	20,11	19,49	20,53	19,42	20,60
	34,23	36,11	33,48	35,05	33,44	35,17

TABLE II

Experimental results with Haar, Daub-4 and Daub-8 for catheter

In order to exploit the scarce presence of movement, we propose to apply the 3-D FWT, performing the wavelet transform first in the time dimension and then in the space dimensions.

We have analyzed the performance of our proposal in terms of both compression ratio and quality ($PSNR$), and we have visually confirmed these results. Compression ratios achieved are quite good, as well as the quality, especially when compared with the compression ratio and quality achieved by the standard MPEG-2, with no extra cost in computation time.

We believe that the significance of medical video deserves research especially oriented to exploit its characteristics, and future work will focus on applying filtering schemes to discard those details useless for doctors, some improvements in the entropy encoder and the evaluation of new mother wavelet functions.

References

[1] M. Antonini and M. Barlaud. Image coding using wavelet transform. *IEEE Transactions on Image Processing*, 1(2):205–220, April 1992.

[2] S. A. Basith and S. R. Done. Digital video, mpeg and associated artifacts. Technical report, Department of Computing and Electrical Engineering, Imperial College, June 1996.

[3] C. M. Brislawn. Fingerprints go digital. *Notices of the AMS*, 42(11), 1994.

[4] Y. Chen and W. A. Pearlman. Three-dimensional subband coding of video using the zero-tree method. *Proc. of SPIE-Visual Communications and Image Processing '96*, pages 1302–1310, March 1996.

[5] I. Daubechies. *Ten Lectures on Wavelets.* Society for Industrial and Applied Mathematics, 1992.

[6] D. Donoho. Nonlinear wavelet methods for recovery of signals, densities, and spectra from indirect and noisy data. *Different Perspectives on Wavelets, Proceeding of Symposia in Applied Mathematics*, 47:173–205, 1993.

[7] A. Graps. An introduction to wavelets. *IEEE Computacional Science and Enginnering*, 2(2), Summer 1995.

[8] M. L. Hilton, B. D. Jawerth, and A. Sengupta. Compressing still and moving images with wavelets. *Multimedia Systems*, 2(3), 1994.

[9] I. Ihm and S. Park. Wavelet-based 3d compression scheme for very large volume data. *Graphics Interface '98*, pages 107–116, June 1998.

[10] I. J. W. (JPEG/JBIG). Fcd 14495, lossless and near-lossless coding of continuous tone still images (jpeg-ls).

[11] A. S. Lewis and G. Knowles. Image compression using the 2-d wavelet transform. *IEEE Transactions on Image Processing*, 1(2):244–256, April 1992.

[12] S. Mallat. A theory for multiresolution signal descomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, July 1989.

[13] D. M. Monro, B. E. Bassil, and G. J. Dickson. Orthonormal wavelets with balanced uncertainty. *School of Electronic and Electrical Engineering, University of Bath, England (ICIP)*, 1996.

[14] S. Muraki. Approximation and rendering of volume data using wavelet transforms. *Proceedings of Visualization '92*, pages 21–28, October 1992.

[15] S. Muraki. Multiscale volume representation by a dog wavelet. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):109–116, June 1995.

[16] J. M. Shapiro. Embedded image coding using zerotrees of wavelets coefficients. *IEEE Transcations on Signal Processing*, 41(12):3445–3462, December 1993.

[17] T. Sikora. *MPEG Digital Video Coding Standars.* McGraw Hill Company, 1997.

[18] E. P. Simoncelli. Modeling the joint statistics of images in the wavelet domain. *Spie 44th Annual Meeting*, 3813, 1999.

[19] E. J. Stollnitz, T. D. DeRose, and D. H. Salesin. Wavelets for computer graphics: A primer, part 1. *IEEE Computer Graphics and Applications*, **15(3)**:76–84, May 1995.

[20] G. Strang and T. Nguyen. *Wavelets and Filter Banks.* Wellesley-Cambridge Press, 1997.

[21] G. K. Wallace. The jpeg still picture compression standard. *Communications of the ACM*, April 1991.