NEW VIDEO COMPRESSION USING MSPIHT3D

A. ZITOUNI(1), A. TALEB AHMED(2), A. OUAFI(1), Z. BAARIR(1), A. KHELIFA(1)

(1)LESIA Laboratory of Research, Electronic Department University of Biskra, Algeria
(2)Université of Valenciennes and Hainaut Cambrésis FRE CNRS 3304 UVHC LAMIH le mont houy 59313 Valenciennes Cedex 9 ,France

ABSTRACT

In this paper, we propose a new approach to video compression based on the principle of Set Partitioning In Hierarchical Tree algorithm (SPIHT). Our approach, the modified SPIHT3D (MSPIHT3D), distributes entropy differently than SPIHT3D and also optimizes the coding. This approach can produce results that are a significant improvement on the Peak Signal-to-Noise Ratio (PSNR) and compression ratio obtained by SPIHT3D algorithm, without affecting the computing time.

KEYWORDS: video compression,, MSPIHT3D, arithmetic Coding, PSNR, Compression ratio.

1 INTRODUCTION

In our application for compression of moving image, we use three-dimensional wavelet transforms to decompose the image sequence into sub-bands in pyramid architecture. This allows us to then have uncorrelated entered they sub-bands, which allows for the best coding steps. Was used MSPIHT3D new coding method that is an extension of the coding algorithm MSPIHT already mentioned in [1,2]. In addition, we used a arithmetic coding for improving the compression ratio.

The general structure of the compression-decompression chain (analysis and synthesis) based on which this work is shown in Figure 1.

A lot of algorithms were proposed in the literature like MPEG-4 Advanced Video Coding (H.264/AVC) is newest video coding standard of the ITU-T Video Coding Experts Group and the ISO/IEC Moving Picture Experts Group [2].

The main goals of the H.264/AVC standardization effort have been enhanced compression performance and provision of a “network-friendly” video representation addressing “conversational” (video telephony) and “non conversational” (storage, broadcast, or streaming) applications. H.264/AVC has achieved a significant improvement in rate-distortion efficiency relative to existing standards. New compression method called dynamic 3D meshes MCGV (Multi-Chart Geometry Video) which extends the GV (Geometry Videos) introduced in [15] approach.

The proposed approach combines the compression strategy prediction model deskinning with a representation of the residual motion errors as dune sequence of 2D images.. The zerotree based compression techniques in conjunction with signed-binary representations and arithmetic coding, particularly in the context of 3D image encoding. [3,4].

In this paper, we propose a modification of the SPIHT3D [4] coding algorithm for coding wavelet coefficients. Our modification is called the Modified SPIHT3D (MSPIHT3D) and has two specificities: it distributes entropy differently than the original SPIHT3D algorithm and it optimizes the coding. In addition, the robustness of the MSPIHT3D compare favorably with the original SPIHT algorithm. arithmetic coding as the compression rate is used for improvement.

Following this introduction the Three-dimensional wavelet decomposition (section.2). In section.3, the proposed MSPIHT3D algorithm is described in detail. In section.4, the Difference between the MSPIHT3D. In section.5, the results obtained with the MSPIHT3D algorithm are analyzed and compared with result from SPIHT3D algorithm and the results MSPIHT3D , SPIHT plus arithmetic coding.

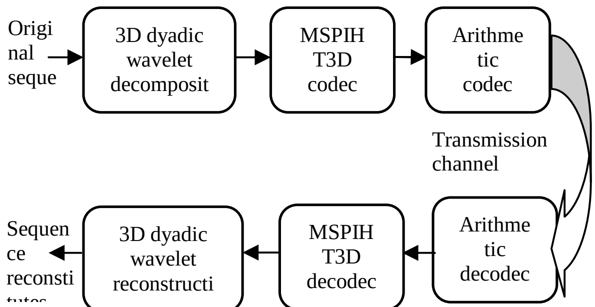


Figure 1: général structure of the compression-décompres

2 THREE-DIMENSIONAL WAVELET DECOMPOSITION

As the two-dimensional transform, the 3D transform can be obtained by a separable decomposition based 1D transform applied in the three directions (horizontal, vertical and temporal). The latter is responsible for the design of video coding schemes that do not require motion compensation, and which exploit temporal redundancy in the same manner that the spatial redundancy whiles considering that the motion is relatively slow as a function of time. Thus, a significant portion of signal energy is concentrated primarily in the subband of the lowest frequencies in the spatial-temporal transform domain.

There are two different types of 3D wavelet decomposition: the dyadic decomposition used in our work and wavelet packet decomposition [1, 2]. In the dyadic case, a temporal decomposition is followed by a spatial decomposition and the process is iterated for the spatio-temporal sub-band lower frequencies until you get a certain level of decomposition. In this way, the number of decomposition levels in the spatial or temporal directions is the same, and the number of sub-bands in this case is 17N (where N is the number of levels of spatial and temporal decompositions).

Figure.2 shows the structure of 3D dyadic wavelet decomposition in two spatial and temporal scales where 'Ht' and 'Bt' represent the temporal subbands high frequencies and low frequencies, respectively, and 'Hh', 'BH', 'Hv "and" Bv 'are high spatial frequency subbands in horizontal, horizontal low frequency, high frequency and low frequency vertical upright respectively.

In the wavelet transformation packet, the number of spatial and temporal decompositions peuvt be different. In this case, the wavelet transform is applied successively 1D following the time direction for the desired number of levels of decomposition. Then all the images in the sequence are separately decomposed in the horizontal and vertical directions. The number of sub-bands we can have is (Nt +1) (3Ns+1) where Nt and Ns are the levels of temporal and spatial decomposition, respectively. (Figure 2 shows the structure of a 3D wavelet packet decomposition).

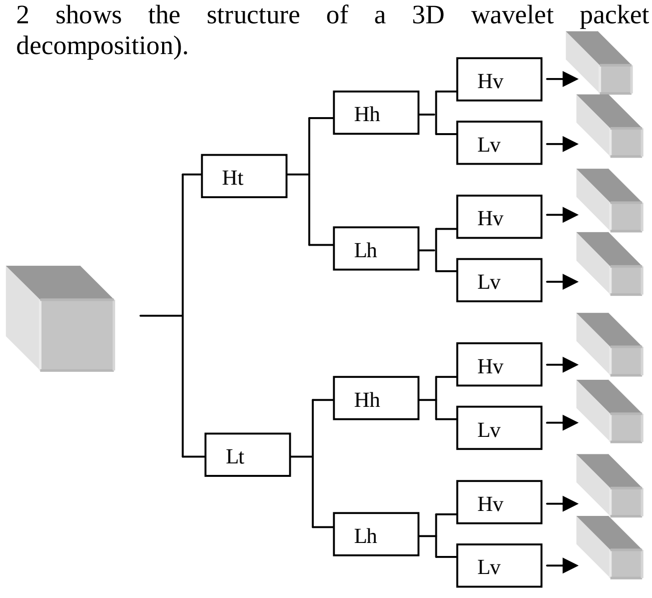


Figure 2: Structure of 3D dyadic wavelet decomposition

3 PROPOSED ALGORITHM (MODIFIED MSPIHT3D)

MSPIHT3D algorithm is an extension of MSPIHT algorithm [1, 2, 11, 12, 13] which is a modification of the SPIHT algorithm, with similar characteristics: Arrangement amplitude coefficients, transmission of the most significant bits in the password refinement and exploitation of self-similarity across spatio-temporal regions of the tree structure of the decomposed signal.

In this way, the bit stream remains perfectly fitted, and progressive video quality is guaranteed. The coding phase may be stopped at any time for a bit rate target. Assuming that a certain distortion in the reconstruction, it continues processing until all information is transmitted in the case of reconstruction without losing what is sometimes desired in some applications such as high definition television HDTV.

In MSPIHT3D pass sorting algorithm of the coefficients is performed in the same Manner as in MSPIHT algorithm; the only difference is the tree structure defined in the transform domain through the subbands. Once sorted coefficients (Within the meaning of meaning), the refinement pass unchanged.

In the 3D structure of the sub-bands, a new tree of spatio-temporal orientation with its own parent-child relationship has been introduced [5] [6] . It is defined such that each node has eight child (Figure 2and 3). MSPIHT3D for the pixels in the sub-band at lower frequencies are grouped into $2 \times 2 \times 2$ adjacent pixels, one of them has no children. DSL list contains only insignificant overall coefficients that have descendants(Figure2).

With the exception of sub-bands of the lowest levels of the pyramid, the children of a pixel with coordinates (i, j, k) in a dyadic decomposition is the set

$$O(i,j,k): O(i,j,\quad k)=\{(2i,2j,2k),\ (2i,2j+1,2k),\ (2i+1,2j,2k),\ (2i+1,2j+1,2k),\ (2i,2j,2k+1),\ (2i+1,2j,2k+1),\ (2i,2j+1,2k+1),\ (2i+1,2j+1,2k+1)\}.$$

The following sets can represent the corresponding tree representations:

- O(i,j,k) is the set of coordinates of all offspring of node (i,j,k).
- D(i,j,k) is the set of all coordinates that are descendants (all nodes that are below) of the node (i,j,k).
- L(i,j,k) is the set of all coordinates that are descendants but not offspring of node (i,j,k).

The lists that will be used to keep track of important pixels are:

- LIS : List of Insignificant Sets, this list is one that shows us that we are saving work by not accounting for all coordinates but just the relative ones.

- LIP: List of Insignificant Pixels, this list keeps track of pixels to be evaluated.
- LSP: List of Significant Pixels, this list keeps track of pixels already evaluated and need not be evaluated again.

A general procedure for the code is as follows (see Fig 5):

3.1 Initialization: output n, n can be chosen by user or predefined for maximum efficiency

$$n = \left|\log_2(\max_{(i,j,k)\in cube})\left|coeff_{(i,j,k)}\right|)\right|.$$

LSP is empty, add starting root coordinates to LIP and LIS.

3.2 Sorting pass: (new n value)

3.2.1 for entries in LIP: (stop if the rest are all going to be insignificant)

- Decide if it is significant and output the decision result .
- If it is significant, move the coordinate to LSP and output the sign of the coordinate.

3.2.2 for entries in LIS: (stop if the rest are all going to be insignificant)

- if the entry in LIS represents D(i,j,k) (every thing below node on tree).
- Decide if there will be any more significant pixels further down the tree and output the decision result.
- If it is significant, decide if all of its eigth children (O(i,j,k)) are significant and output decision results .
- If significant, add it to LSP, and output sign.
- If insignificant, add it to LIP.
- If the entry in LIS represent L(i, j,k) (not children but all others).
- If the four children (O(i,j,k)) are insignificant, their coding is performed by one bit “0” in the outbit set instead of eight bits “00000000” encoding in the SPIHT algorithm.
- Dcide if there will be any more significant pixels in L(i,j,k) further down the tree and output the decision result.
- If there will be one, add each child to LIS of type D(i,j,k) and remove it from LIS.

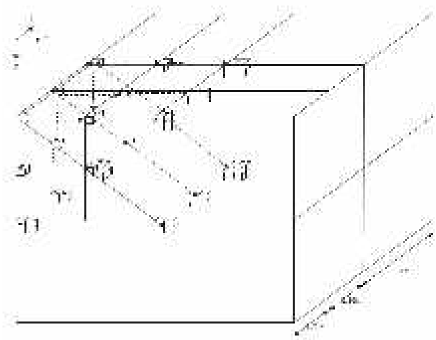
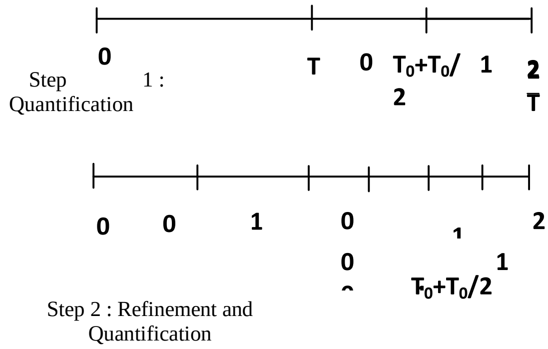


Figure 3: Parent-offspring dependencies in the 3D orientation tree.

3.2.3 Refinement Pass: all values in LSP are now $2^n \leq |C_{ij}$

For all pixels in LSP, output the nth most significant bit (see Fig 4)



3.2.4 Quantization-step Update

A bit corresponding to 2 j-1 is emitted for all the significant values in the list LSP in order to increase the precision of those values transmitted [5, 13]. The significant values {63, -45, 61 and........ 49} from the cube test (cf. fig 6) are quantified respectively by the bits " 1 0 1 11111" [7,8]. Then, step B of the algorithm is repeated on the cube residue by incrementing j by one. This process is reiterated until the desired quality of the reconstructed image is reached or until the number of transferable bits required is exceeded.

4 DIFFERENCE BETWEEN THE MSPIHT3D

The difference between the MSPIHT3D algorithm that we propose and the SPIHT3D algorithm lies in the insignificance test process used for the set of coordinates of all offspring of node (i,j,k) and the coding procedure used for the outbit symbols.

4.1 Insignificance test process

Let us consider the cube test (see Fig 6) for a first iteration (initial threshold T0 = 32).

Result of the algorithms SPIHT3D and MSPIHT3D applied to example1 cube test (cf. fig 6).

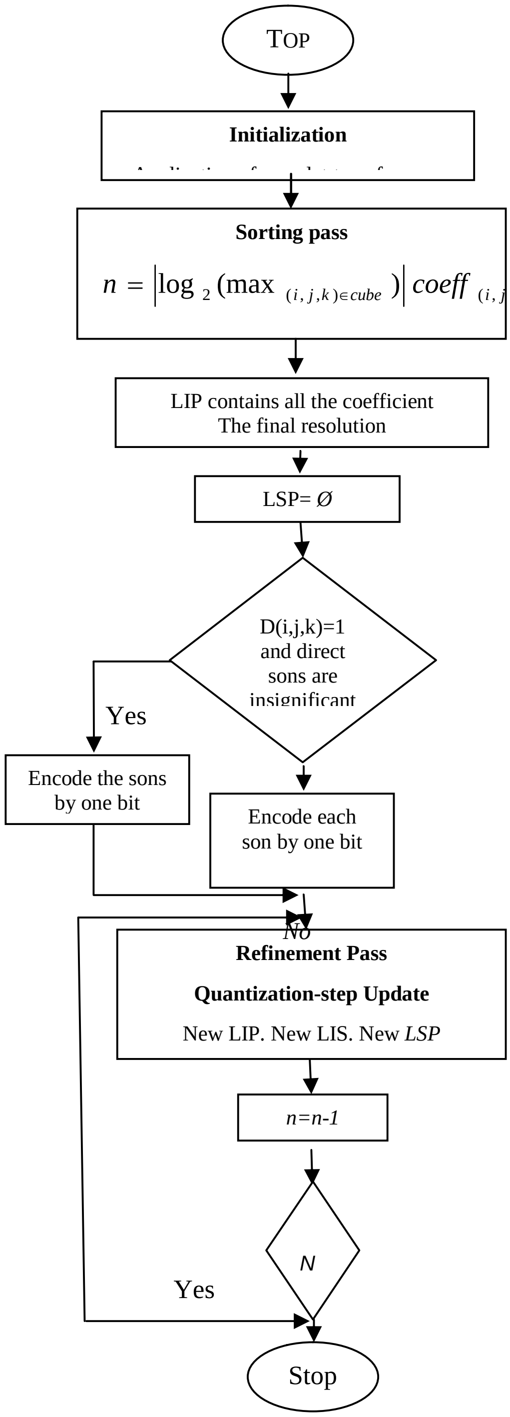


Figure 5: Structure of the MSPIHT3D algorithm

Example

SPIHT : Out Bit : +1 0 0 0000+1 1 00000 000 1 00000000 1 0000 0000 1 000000 00 1 00000000 1 000 000001 000 000010 0 0 00001 00000 0 0 +1 10 0 0 00001 00000 0 0 +110 0 0 00001 00000 0 0 +110 0 0 00001 00000 0 0 +110 0 0 00001 00000 0 0 +110 0 0 00001 00000 0 0 +110 0 0 00001 00000 0 0 +1
MSPIHT : Out Bit +1 0 0 0000+1 1 00 1 00 1 001 00 1 0 1 001 0010 0 0 00001 00000 0 0 +1 10 0 0 00001 00000 0 0 +110 0 0 00001 00000 0 0 +110 0 0 00001 00000 0 0 +110 0 0 00001 00000 0 0 +110 0 0 00001 00000 0 0 +110 0 0 00001 00000 0 0 +1

Lsp : 1 0 1 11111

As in Example cubes (cf. fig 6) the offspring of (-31) are encoded in MSPIHT3D with one single Symbol “0” in set outbit instead of eight Symbol “00000000” in the SPIHT3D algorithm.

These are the initial MSPIHT3D settings. The initial threshold is set to 32 Example : cubes (see Fig 6) The notation (i,j,k)A or (i,j,k)B, indicates that an LIS entry is of type ‘A’ or ‘B’, respectively. Note the duplication of co-ordinates in the lists, as the sets in the LIS are trees without the roots. The coefficient (0, 0, 0) is not considered a root.

MSPIHT3D begins coding the significance of the individual pixels in the LIP. When a coefficient is found to be significant it is moved to the LSP, and its sign is also coded. We used the notation +1 and -1 to indicate when a bit 1 is immediately followed by a sign bit.

After testing pixels it begins to test sets, following the entries in the LIS (active entry indicated by bold letters). In this example D(1, 0,0) is the set of 63 coefficients {(2,0,0), (3,0,0), (2,1,0), (3,1,0), (4,0,0),(5,0,0), (6,0,0), (7,0), (4,1), (5,1), (6,1), ... (6,3,3), (7,3,3)}.

Because D(1,0,0) is significant MSPIHT3D next tests the insignificance of the eight offspring {2,0,0), (3,0,0),(2,1,0), ...(3,3,1)}.

After all offspring are tested they are coded by only one bit zero in the outbit set using MSPIHT3D algorithm insted of eight bits zero when SPIHT3D algorithm is employed, (1, 0,0) is moved to the end of the LIS, and its type changes from ‘A’ to ‘B’, meaning that the new LIS entry meaning changed from D(1, 0,0) to L(1, 0,0) (i.e., from set of all descendants to set of all descendants minus offspring). Same procedure as in comments and applies to set D (0, 0,1) and ....D (1, 1,1).

4.2 Coding the outbit symbols

- If D (i,j,k)=0 the sons and their sons are all coded by one bit zero.
- If D (i,j,k)=1 and one or more sons is significant each direct son is coded by one bit .
- If D(i,j,k)=1 and direct sons are insignificant the set of the eight sons is coded by one bit zero .

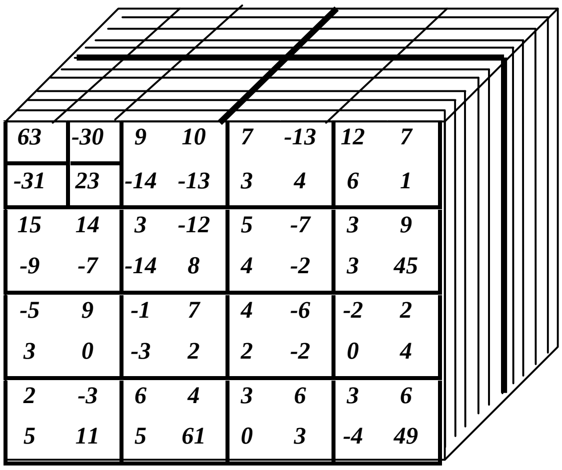


Figure 6: Example of decomposition to three resolutions for an 8x8x8 cube.

The code of insignificance of the eight offspring for coefficient (0,0,1),(0,1,0), (1,0,0),(0,1,1), (1,1,0),(1,0,1), and (1,1.1) its one zero in set out bit in bold in figure 5 is the symbol ‘0’, rather than the “00000000” used in SPIHT3D algorithm.

Number of symbols to calculate the Outbit list coefficients for both Example (see fig.6).

Example

	SPIHT3D	MSPIHT3D
Number of Symbols	**1 89**	**147**

Without using any other form of entropy coding the SPIHT3D algorithm used 189 bits in this first pass but in the MSPIHT3D entropy coding used 147 bits in this first pass.

5 RESULTS AND DISCUSSION

The MSPIHT3D algorithm was performed using Matlab on an INTEL Pentium Core duo (2.2 Ghz, RAM 4G).

We tested our algorithm on sequences images (trees 232x232x16, 8 bpp), according to a tree-level wavelet decomposition using biorthogonal filters 9/7 [9,10].

The PSNR (dB) performance and compression ratio CR (bpp) of our MSPIHT3D algorithm were compared to those for the SPIHT3D algorithm. These parameters are expressed by the following relations (1-3) [14]:

$$PSNR\ (db) = 10 \log_{10}\left[\frac{(255)^2}{MSE}\right] \qquad (1)$$

$$MSE = \frac{1}{N \times M \times K}\sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{k=1}^{K}\left(x_{i,j,k} - y_{i,j,k}\right)^2 \qquad (2)$$

$$CR(bpp) = \frac{number \text{ of coded bits}}{number \text{ of initial bits}} \qquad (3)$$

Where N, M, K is the cub size, $x_{i,j,k}$ the initial cub and $y_{i,j,k}$ the reconstructed cub.

Referring to Tables 1-2, it is clearly seen that, regardless the threshold value, the total information required for the coding of the moving image using the MSPIHT3D is much smaller than the one required using the SPIHT3D algorithm.

Referring to Tables 3-4, it is clearly seen that, regardless the threshold value, the total information required for the coding of the moving image using the MSPIHT3D plus arithmetic coding is much smaller than the one required using the SPIHT3D plus arithmetic coding algorithm.

In the total of cases, the results obtained by the MSPIHT3D are better than those obtained by SPIHT3D (Table5).. Even for lower rates, the MSPIHT3D performance is still very close to that of the SPIHT3D.

TThe MSPIHT3D method is not a simple extension of traditional methods for moving image compression, and represents an important advance in the field. The method deserves special attention because it provides the following: moving image quality, progressive, image transmission, optimized Embedded coding, lossless compression , lossy compression and rate or distortion specification.

Table 1: OUTBIT and LSP of the MSPIHT3D algorithm applied to the trees sequence 232×232x16 for threshold Th =32.

MSPIHT3D			
Seuil	OUTBIT	LSP	TOTAL INFORMATION
seuil /16	419229	80248	499477
seuil /32	856477	213540	1070017
seuil /64	1237630	391064	1628694
seuil /128	1501111	557997	2059108(bits)

Table 2: OUTBIT and LSP of the SPIHT3D algorithm applied to the trees sequence 232×232x16 for threshold Th =32.

MSPIHT3D			
Seuil	OUTBIT	LSP	TOTAL INFORMATION
seuil /16	429706	80248	509954
seuil /32	866580	213540	1080120
seuil /64	1244933	391064	1635997
seuil /128	1505328	557997	2063325(bits)

Table 3: MSPIHT3D plus arithmetic coding applied to the trees sequence 232×232x16 for threshold Th =32.

mspiht3D + ARITHMETIC CODING			
Seuil	DECOMPOSITION : **2 BITS**	DECOMPOSITION : **4BITS**	DECOMPOSITION : **8 BITS**
seuil /16	390544	388152	389504
seuil /32	898080	894040	893560
seuil /64	1420104	1396520	1389344
seuil /128	1729520	1675288	1660208(bits)

Table 4: SPIHT3D plus arithmetic coding applied to the trees sequence 232×232x16 for threshold Th =32.

MSPIHT3D +ARITHMETIC CODING			
Seuil	DECOMPOSITION: **2 BITS**	DECOMPOSITION: **4 BITS**	DECOMPOSITION: **8 BITS**
seuil /16	396352	393640	395136
seuil /32	904520	899640	898904
seuil /64	1425008	1401688	1393904
seuil /128	1732040	1677704	1662384(bits)

Table 5: Results of different algorithms applied to the image sequences (trees).

Sequences images	algorithm coding	PSNR (db)
		0.5 0.75 1 5 2(**bpp)**
Sequences trees (236x236x16)	MSPIHT3D	**32.2 33.9 37.7 43.4 52.0**
	SPIHT3D	31.3 32.7 36.40 41.9 48.9

6. CONCLUSION

In this paper, we developed an video compression algorithm (MSPIHT3D) based on the same principle as Set Partitioning In Hierarchical Tree algorithm. This algorithm is able to improve the performance of the SPIHT3D algorithm because 1) using one insignificance symbols instead of eight better optimizes the entropy and 2) the binary regrouping of these symbols on outbit set better optimizes the coding. The proposed algorithm is able to accomplish this without increasing computation time. In addition, this algorithm performed comparably with SPIHT3D algorithm, which could be very interesting for the field of hierarchical coding.

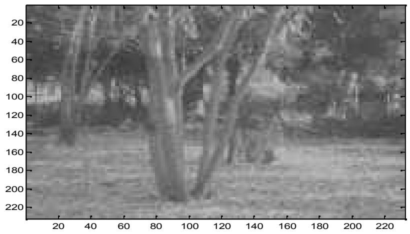


Figure 7: Image 1 of the original trees sequences

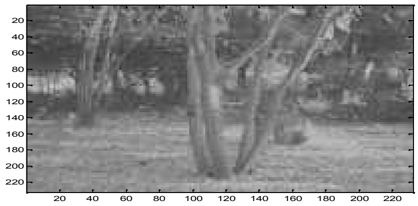


Figure 8: Image 15 of the original trees sequences

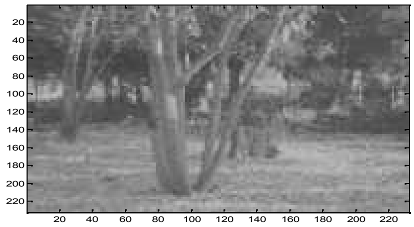


Figure 9: Image 1 reconstructed of the trees sequences th/32 psnr 37.81

Figure 10: Image 15 reconstructed of the trees sequences th/32 psnr 37

REFERENCES

[1] Zitouni Athmane.. Baarir Zine Eddine, Ouafi Abd Elkarim and Taleb-Ahmed Abdelmalik. « A New Application of MSPIHT for Medical Imaging»,Journal of Applied Computer Science &Mathematics, no. 13 (6) /2012, Suceava.

[2] Ouafi Abd Elkarim,, Taleb-Ahmed Abdelmalik, Baarir Zine Eddine and Zitouni Athmane A Modified Embedded Zerotree Wavelet (MEZW) Algorithm for Image Compression, Journal of Mathematical Imaging and Vision 01/2008; :298-307.

[3] H. M. Brice˜ no, P. V. Sander, L. McMillan, S. Gortler, et H. Hoppe. Geometry videos a new representation for 3D animations. InEurographics/SIGGRAPH Symposium onComputer Animation, pages 136–146, San Diego,Etats-Unis, 2003.

[4] Emmanuel Christoph, Pierre Duhamel and Corinne Mailhes /SIGNED BINARY DIGIT REPRESENTATION TO SIMPLIFY 3D-EZW/ IEEE, ICASSP 2007.

[5] X. Tang and W. A. Pearlman, "Three-Dimensional Wavelet-BasedCompression of Hyperspectral Images", Chapter in Hyperspectral DataCompression, Kluwer Academic Publishers 2005.

[6] X. Tang, "Wavelet Based Multi-Dimensional Image Coding Algorithms",Ph.D thesis, Rensselaer Polytechnic Institute, 2005.

[7] M. Shapiro, “Embedded image coding using zerotres of wavelet coefficients,” IEEE Trans. on Signal Processing, vol. 41, pp. 3445–3462, Dec. 1993.

[8] A. Islam and W. A. Pearlman, “An embedded and efficient low-complexity hierarchical image coder,” Visual Communications and Image Processing ’99, Proceedings of SPIE, vol. 3653, pp. 294–305, Jan. 1999.

[9] A. Vlaicu, S. Lungu, N. Crisan, and S. Persa, \New compression techniques for storage and transmission of 2D and 3D medical images", in Proc. SPIE Advanced Image and Video Communications and Storage Technologies, vol.2451, pp. 370-377, Feb. 1995.

[10] P. Schelkens, J. Barbarien, and J. Cornelis, \Compression of volumetric medical data based on cube-splitting", in Applications of Digital Image Processing XXIII, Proc. of SPIE 4115, pp. 91-101, San Diego, CA, July 2000.

[11] R.Viswanath, Reddy1, T.Sreenivasulu Reddy2, and Dr.Govind Sharma31,3 IIT Kanpur, Dept. of EEE, Kanpur, Indi, “Efficient Coding of Image Subbands using Blockbased Modified SPIHT”, International Journal of Recent Trends in Engineering, Vol 2, N° 5, November 2009.

[12] Nicola Sprljana,Sonja Grgicb,Mislay Grgicb, “Modified Spiht algorithm for wevlet packet image coding”, real-time Imaging Vol 11 pp 378-388 , 2005.

[13] R.Suudhakar, R.Karthing, and S.Jayaraman, “Fingerprint Compression Using Contourlet Transform with Modified SPIHT Algorithm”. Iranian journal of electrical and Computer Engineering Vol 5, N°,1,Winter-spring 2006.

[14] Analysis of optical near-field images by Karhunen—Loève transformation Daniel Charraut, Daniel Courjon, Claudine Bainier, and Laurent Moulinier, Applied Optics, Vol. 35, Issue 20, pp. 3853-3861 (1996

[15] D. Marpe, H. Schwarz, and T. Wiegand,“Context-adaptive binaryarithmetic coding for H.264/AVC,”IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 7, pp. 620–636, Jul. 2003.