Asian Journal of Information Technology 5(8) :893-900, 2006

A New Approach Based on Shapiro's Embedded Zerotree Wavelet (EZW) Algorithm for Image Compression

[1]Ouafi, A., [1]Z.E. Baarir, [2]A. Taleb Ahmed, [3]N. Doghmane,
[1]LESIA Laboratory of Research, Electronic Department, University of Biskra, Algeria
[2]LAMIH Laboratory of Research, University of Valenciennes, France
[3]Electronic Department , University of Annaba, Algeria

Abstract: In this study, we propose a new study to image compression based on the principle of Shapiro's Embedded Zerotree Wavelet (EZW) algorithm. Our study, the modified EZW (MEZW), distributes entropy differently than Shapiro's and also optimizes the coding. This study can produce results that are a significant improvement on the PSNR and compression ratio obtained by Shapiro, without affecting the computing time. These results are also comparable with those obtained using the SPIHT and SPECK algorithms.

Key words: Image compression, shapiro's EZW algorithm, MEZW, entropy, coding, PSNR, compression ratio, SPIHT and SPECK algorithms

INTRODUCTION

Todays massive use of digital images generates increasingly significant volumes of data. Compressing these digital images is thus necessary in order to store them and simplify their transmission. Two classes of image compression techniques can be defined: conservative, or lossless, techniques and non-conservative, or lossy techniques[1]. The lossless techniques guarantee an exact copy of the data after the compression/decompression cycle, but tend to generate rather low compression ratios. The lossy techniques, on the other hand, offer high compression ratios and though they do allow a degree of information loss, the quality of the image perceived by the user is not affected. This second type of compression, however, requires significant computing time.

Over the years, many compression algorithms have been proposed and standarized, such as the JPEG standard[2] for still images and the MPEG standard[3] for videos images. These standards are based on the Discrete Cosine Transform (DCT)[4]. Recently, however, an interesting alternative transform has been developed, called the Discrete Wavelets Transform (DWT). This transform is able to attain significant compression ratios without producing the artefacts (block effects) observed in images compressed using DCT[5-8].

Several compression algorithms using wavelets have been proposed, with the most used being the EZW (Embedded Zero-trees Wavelet)[9], the SPIHT (Set Partitioning in Hierarchical Trees)[10] and the SPECK (Set Partitioning Embedded Block)[11]. These algorithms which rely on embedded coding, create an embedded binary flow, a progressive data transmission that allows the image to be reconstructed using various compression ratios. Thus, these algorithms can be used for either conservative or non-conservative compression[1,12].

In this study, we propose a modification of the EZW coding algorithm for coding wavelet coefficients. Our modification is called the Modified EZW (MEZW). This modified algorithm has two specificities: it distributes entropy differently than the original Shapiro EZW algorithm and it optimizes the coding. In addition, the robustness of the MEZW compares favorably with the original EZW algorithm and both the SPIHT and SPECK algorithms.

Shapiro's EZW algorithm: The objective of Shapiro's encoder is to exploit possible dependence protocols between the wavelet coefficients of different sub-bands in order to successfully create zero-trees[9,13,14] (Fig. 1). A “zerotree” is composed of a parent (ancestor) and its descendants. In this

Corresponding Author: Ouafi, A., LESIA Laboratory of Research, Electronic Department, University of Biskra, Algeria

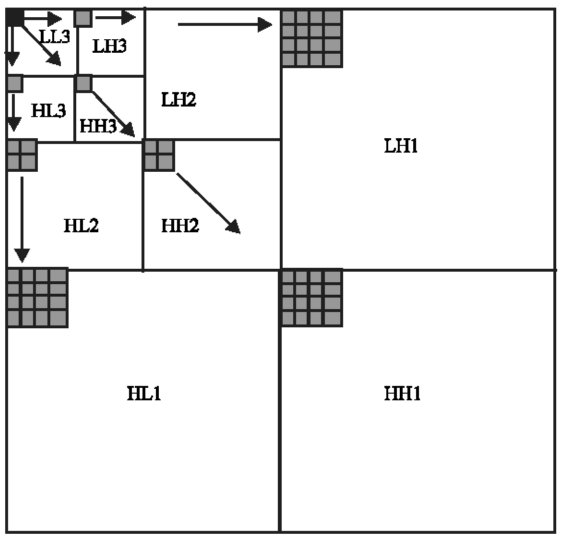


Fig. 1: Parent-descendant dependencies between sub-bands

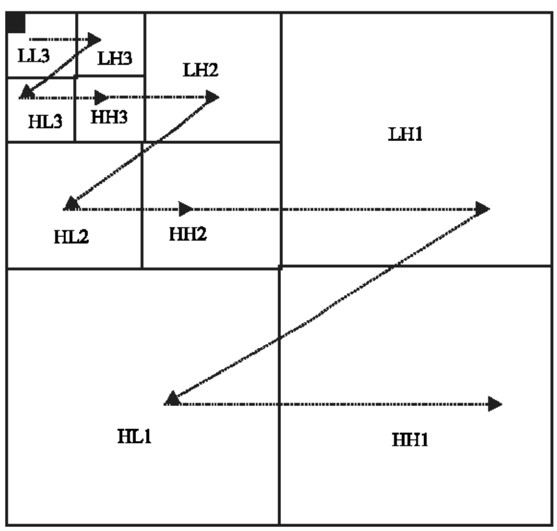


Fig. 2: Scanning order of the sub-bands for encoding a significance map

tree, each parent on the scale j has four descendants on the scale j-1 (Fig. 1).

The wavelet coefficients are scanned for the path presented in Fig. 2, in order to make the "zerotree" as effective as possible. If a parent and all its descendants are insignificant, then the ancestor is coded "zerotree" and the descendants are not coded. Not coding the descendants will save memory space, thus improving the rate of compression.

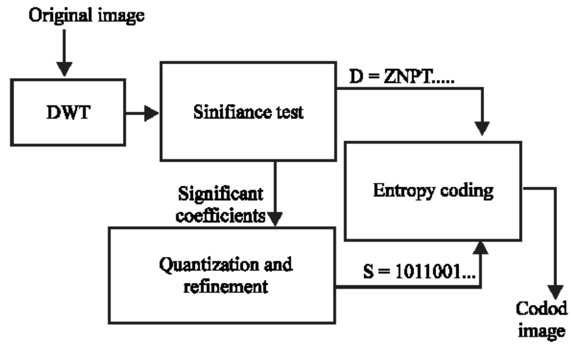


Fig. 3: Principle of Shapiro's EZW algorithm for a compression cycle

63	-34	49	10	7	-13	12	7
-31	23	-14	-13	3	4	6	1
15	14	3	-12	5	-7	3	9
-9	-7	-14	8	4	-2	3	9
-5	9	-1	47	4	-6	-2	2
3	0	-3	2	2	-2	0	4
2	-3	6	4	3	6	3	6
5	11	5	6	0	3	-4	4

Fig.4: Example of decomposition to three resolutions for an 8x8 matrix

Theoretically, the insignificance of an ancestor does not mean that all of its descendants are insignificant. In practice, however, the probability of this phenomenon occurring in real images is high. The structure of the "zerotree" makes it possible to code the significant coefficients using a very low number of bits. Coding the wavelet coefficients is performed by determining two lists of coefficients (Fig. 3):

- The *dominant list D* contains information concerning the significant coefficients, which will be coded using entropy coding.
- The *significant list S* contains the amplitude values of the significant coefficients, which will undergo uniform scalar quantization followed by entropy coding.

Let us consider the matrix test shown in Fig. 4, in terms of the steps of Shapiro's algorithm[9,8,13,15]:

Initialization: The wavelet transform is applied to the image and then the threshold T_0 is determined so that so that: $T_0 = 2^{\log_2(|C_{max}|)}$ where C_{max} is the largest wavelet coefficient[13]. For the test matrix in question (Fig. 4), T_0=32 for C_{max} = 63.

Significance test: The wavelet coefficients are scanned for the path presented in Fig. 3. Each coefficient is assigned a significance symbol (P, N, Z or T), by comparing each coefficient with the actual threshold T_j ($T_j = T_0/2^j$, where j is the iteration count):

- P (significant and positive): if the absolute value of the coefficient is higher than the threshold T_j and is positive. This is the case for the coefficients {63, 49 and 47} in the matrix test (Fig. 4).
- N (significant and negative): if the absolute value of the coefficient is higher than the threshold T_j and is negative. This is the case for the coefficient {-34} in the matrix test (Fig. 4).
- T (zerotree): if the value of the coefficient is lower than the threshold T_j and has only insignificant descendants. Like coefficient {23} in the matrix test (Fig. 4), the descendants of this type of coefficient will not be coded.
- Z (isolated zero): if the absolute value of the coefficient is lower than the threshold and has one or more significant descendants with respect to T_j. This is the case for the coefficients {-31 and 14} (Fig. 4).

The insignificant coefficients of the last sub-bands, which do not accept descendants and are not themselves descendants of a zero-tree, are also considered to be isolated zeros. This is the study for the coefficients {7,-13, 3 and 4}. The significance symbols of the image coefficients are then placed in list D, which is subjected to entropy coding before being transmitted (Fig. 3). The amplitudes of the significant coefficients are placed in the list *S*. Their values in the transformed image are set to zero in order to not undergo the next step.

Quantization and refinement: A bit corresponding to 2^{j-1} is emitted for all the significant values in the list S in order to increase the precision of those values transmitted[9,13]. The significant values {63, -34, 49 and 47} from the matrix test shown in Fig. 4 are quantified respectively by the bits " 1 0 1 0 "[9]. Then, step B of the algorithm is repeated on the image residue by incrementing j by one and by dividing the threshold T_j by two. This process is reiterated until the desired quality of the reconstructed image is reached or until the number of transferable bits required is exceeded. Let us consider the matrix test shown in Fig. 4 for a first iteration (T_0 = 32). After covering the matrix coefficients according to the Shapiro algorithm, the following results are obtained:

D: P N Z T P T T T T Z T T Z Z Z Z Z P Z Z
S: 1 0 1 0

The above summary of the Shapiro algorithm was provided in order to facilitate comparison of our algorithm to Shapiro's. In general, we have modified the original algorithm in the following manner:

- Symbols were added to the significance test stage to allow a better redistribution of the entropy .
- The coding of the list *D* elements and the list S quantization bits was optimised by the use of block coding.

Proposed algorithm (modified EZW): The difference between the MEZW algorithm that we propose and the Shapiro algorithm lies in the significance test process used for the wavelet coefficients and the coding procedure used for the significance symbols.

Coefficient significance test: If a coefficient is tested and found to be significant, its descendants must also be tested. If at least one descendant is significant, then the coefficients are coded according to the doing rules of the Shapiro's algorithm, which is the case for the coefficients {63, -34 and 47}.

However, if all the descendants are judged insignificant, the coefficients are coded according to our MEZW algorithm's coding rules, using the symbols P_t for positive coefficients and N_t for negative coefficients. Thus, coefficient 49 is coded P_t. In this situation, it is no longer useful to code this coefficient's descendants {7,-13, 3 and 4}, whereas in the EZW algorithm, they would be coded "ZZZZ".

Our study requires about the same amount of computing t ime as the EZW approach. In fact, the

63	-30	9	10	7	-13	12	7
-31	23	-14	-13	3	4	6	1
15	14	3	-12	5	-7	3	9
-9	-7	-14	8	4	-2	3	9
-5	9	-1	7	4	-6	-2	2
3	0	-3	2	2	-2	0	4
2	-3	6	4	3	6	3	6
5	11	5	6	0	3	-4	4

Fig. 5: Example of decomposition to three resolutions for an 8x8 matrix in which all the descendants of the root (63) are insignificant compared to the threshold $T_0 = 32$. The matrix is coded '*PTTT*' with the EZW algorithm and by the symbol P_t alone with the MEZW algorithm

Table 1: Appearance frequency, probability and entropy of the four symbols of the EZW algorithm applied to the Lena image 512*512 for a threshold T=13 (PSNR = 36.71 dB, CR=0.57 bpp)

Symbol	Appearance frequency	Probability	Entropy (bits)
T	86278	0.7111	0.4918
Z	15340	0.1264	2.9835
P	9935	0.0819	3.6102
N	9775	0.0806	3.6337

Total information = 159590 bits

computing time difference can be supposed negligible because the same number of test operations is performed.

If the significant coefficient is in the root of the matrix representing the parent and its descendants, then a symbol P_t (or N_t) represents four symbols "PTTT" (or "NTTT"). (Though this does not occur in the example in Fig. 4, it does occur in the example given in Fig. 5.)

If the significant coefficient is not in the root, P_t (or N_t) represents five symbols "PTTTT" (or "NTTTT"), which is the case for coefficient 49 in the matrix test shown in Fig. 4.

By introducing the two symbols, P_t and N_t , the probabilities of the symbols in the dominant list D can be redistributed. In fact, the symbol T is generally the most probable in both Shapiro's algorithm and ours, but this probability is lower in our MEZW algorithm (Table 2).

Thus, the entropy of the symbol T in the MEZW algorithm is close to 1 bit, making the coding more

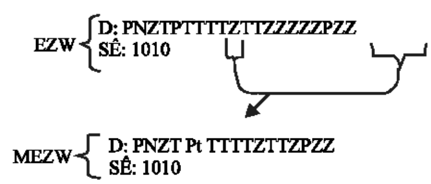
Table 2: Appearance frequency, probability and entropy of the six symbols of our MEZW algorithm applied to the Lena image 512*512 for a threshold T=13 (PSNR=36.71 dB, CR=0.47 bpp)

Symbol	Appearance Frequency	Probability	Entropy (bits)
T	38486	0.5234	0.9341
Z	15340	0.2086	2.2612
N_t	6174	0.0840	3.5742
P_t	5838	0.0794	3.6549
N	3937	0.0535	4.2233
P	3761	0.0511	4.2893

Total information =146800 bits

effective. Thus, the total amount of information contained in the symbols (Table 2) of the MEZW algorithm is less than in the symbols of the EZW algorithm (Table 1). This redistribution of probabilities makes it possible to obtain an optimal entropy coding[1].

By applying the MEZW to the matrix in Fig. 4 for the first iteration, we obtain the following results:



The code for coefficient 49 and its four descendants (Fig. 4) is the symbol 'P_t', rather than the "P, Z, Z, Z and Z" used in Shapiro's algorithm.

Coding procedure used for the significance symbols: In Shapiro's EZW algorithm, the dominant list D is composed of four symbols {P, N, Z and T}, each one coded into binary on two bits; these symbols are coded arithmetically before transmission. Practically, we find that such entropy coding becomes more effective when the size of the symbols is significant. This significant size is obtained by binary regrouping of several juxtaposed symbols. In our MEZW algorithm, since the six symbols in list D (P, N, Z, T, P_t and Z_t) are coded on three bits, entropy coding is no longer effective for these symbols. Our method regroups the list D elements on 9 bits (a binary regrouping with three juxtaposed symbols) before performing entropy coding. Then, the elements on the significant list S are regrouped on 8 bits (bits obtained by refining the significant coefficients); this regrouping is followed by an entropy coding.

The example detailed in Tables 1 and 2 shows that the total amount of information calculated for the MEZW algorithm is less than the amount calculated for the EZW algorithm.

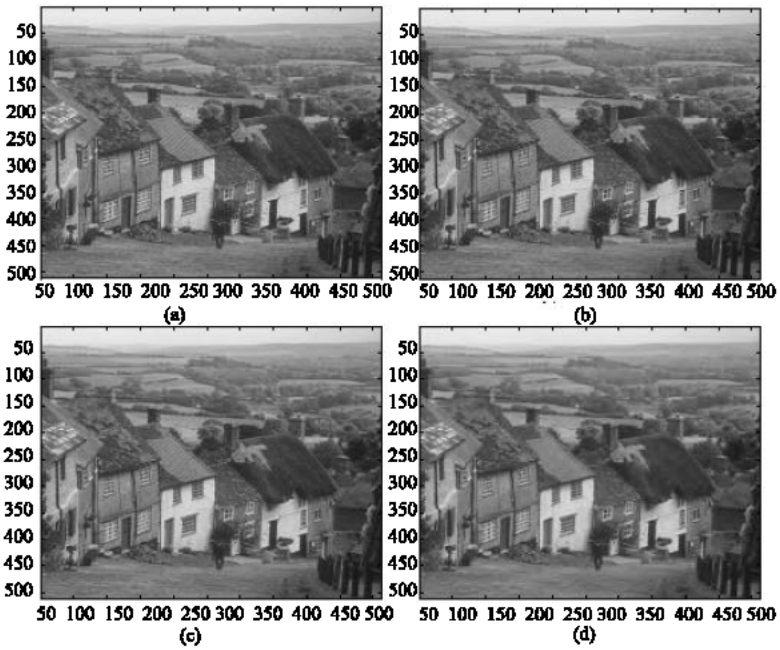


Fig. 6: Results for the Lena image reconstructed using the MEZW algorithm : •) the original Lena image 512x512, •) the Lena Image reconstructed with PSNR = 40.64db and TC=1 bpp, •) the Lena Image reconstructed with PSNR=36.93db and TC= 0.50 bpp and d) the Lena Image reconstructed with PSNR = 33.20db and TC= 0.25 bpp

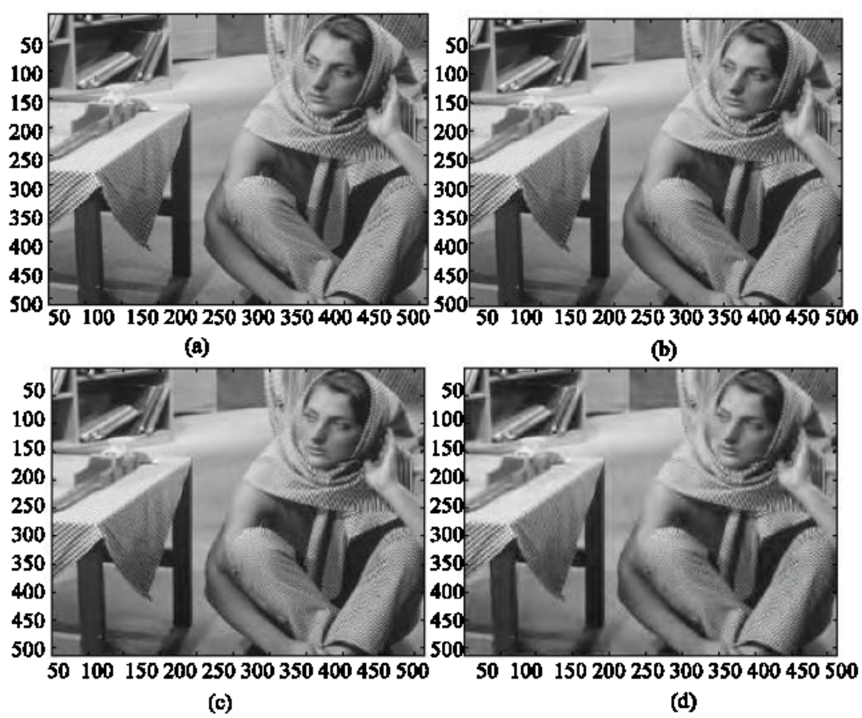


Fig. 7: Results for the Barbara image reconstructed using the MEZW algorithm: •) the original image Barbara 512x512, •) the Barbara Image reconstructed with PSNR = 36,77 dB and TC=1 bpp, •) the Barbara Image reconstructed with PSNR = 31,41 dB and TC = 0.50 bpp and d) the Barbara Image reconstructed with PSNR = 27,23 dB and TC = 0.25 bpp

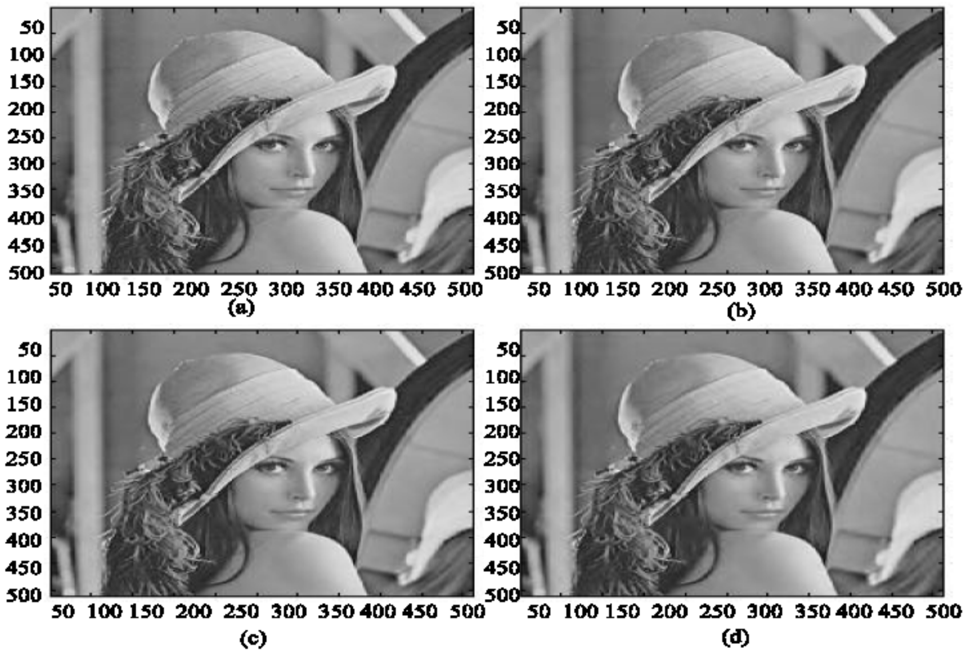
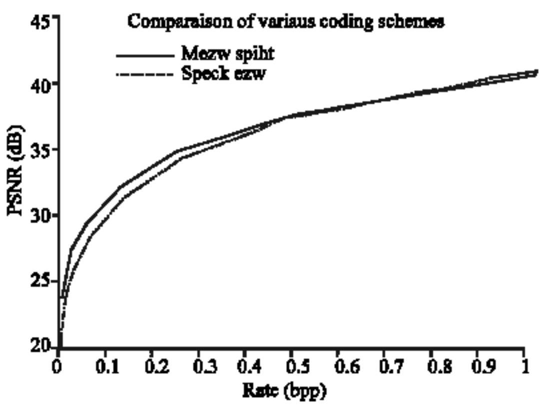


Fig. 8: Results for the Goldhill image reconstructed using the MEZW algorithm: •) the original Goldhill image 512x512, •) the Goldhill Image reconstructed with PSNR = 36,86 dB and TC=1 bpp, •) the Goldhill Image reconstructed with PSNR=32,92 dB and TC = 0.50 bpp and d) the Goldhill Image reconstructed with PSNR = 29,91 dB and TC = 0.25 bpp



.9: Comparison of different compression methods applied to the Lena image

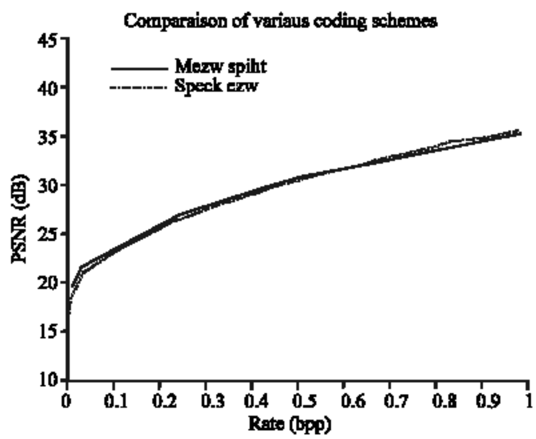


Fig.10: Comparison of different compression methods applied to the Barbara image

RESULTS AND DISCUSSION

The MEZW algorithm was performed using Matlab on an INTEL Pentium 4 PC (3 Ghz; RAM 512 Mo).

We tested our algorithm on three different still images (Lena, Barbara and Goldhill 8 bpp, 512x512), according to a five-level wavelet decomposition using biorthogonal filters 9/7 [16,17] and arithmetic coding.

The PSNR (dB) performance and compression ratio CR (bpp) of our MEZW algorithm were compared to those for the EZW algorithm[9] well as to those obtained with the SPIHT and SPECK algorithms[10,11]. These parameters are expressed by the following relations[10,8]:

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

$$\mathrm{MSE} = \frac{1}{n \times m}\sum_{i=1}^{n}\sum_{j=1}^{m}(x_i - y_j)^2$$

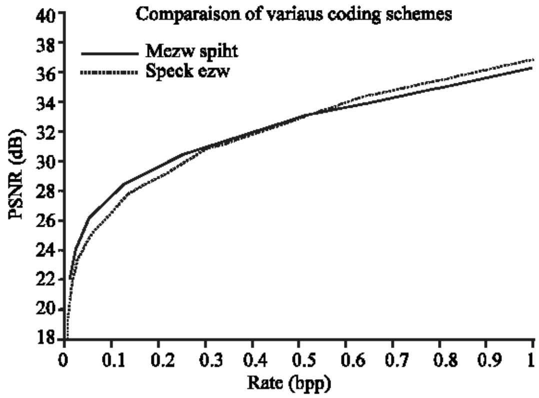


Fig.11: Comparison of different compression methods applied to the Goldhill image

Table 3: Results of the various algorithms applied to the three test images (Lena, Barbara and Goldhill)

Image	Coding algorithm	PSNR (db) 0.25 bpp	0.5 bpp	1 bpp
Lena (512x512)	MEZW	33,20	36,93	40.64
	EZW	33,17	36,28	39,55
	SPIHT	34,11	37,21	40,44
	SPECK	34,03	37,10	40,25
Barbara (512x512)	MEZW	27,23	31,41	36,77
	EZW	26,77	30,53	35,14
	SPIHT	27,58	31,40	36,41
	SPECK	27,76	31,54	36,49
Goldhill (512x512)	MEZW	29,91	32,92	36,86
	EZW	30,31	32,87	36,20
	SPIHT	30,56	33,13	36,55
	SPECK	30,50	33,03	36,36

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

Where n, m is the image size, x_i the initial image and y_j the reconstructed image

In the majority of studies, the results obtained by the MEZW are better than those obtained by Shapiro's algorithm (Table 3). For rates higher than 0.70 bpp, the MEZW performs better than the SPIHT and SPECK algorithms (Fig. 9, 10 and 11). Even for lower rates, the MEZW performance is still very close to that of the SPIHT and SPECK algorithms and the results start getting better around 0.60 bpp for the Barbara image, 0.55 bpp for the Goldhill images and 0.70 bpp for the Lena image.

The Lena, Barbara and Goldhill images reconstructed by the MEZW algorithm for compression ratios 0.25, 0.50 and 1 bpp (Fig. 6, 7 and 8) are shown as examples below. The comparison of the different compression methods applied to the three test images is shown in Fig. 9, 10 and 11.

CONCLUSION

In this study, we developed an image compression algorithm (MEZW) based on the same principle as Shapiro's EZW algorithm. This algorithm is able to improve the performance of the EZW algorithm because •) using six significance symbols instead of four better optimizes the entropy and •) the binary regrouping of these symbols on 9 bits better optimizes the coding. The proposed algorithm is able to accomplish this without increasing computation time. In addition, this algorithm performed comparably with the SPIHT and SPECK algorithms, which could be very interesting for the field of hierarchical coding.

REFERENCES

1. Angeles Losada, M., Gulay Tohumoglu, David Fraile and Antonio Artès, 2000. Multi-itération wavelet zerotree coding for image compression. Elsevier Sci. Signal Processing, pp: 1281-1287.
2. Wallace, G.K., 1991. The JPEG still-picture compression standard, commun.ACM, pp: 30-44 .
3. ITU-T, 1995. Recommendation H.262-ISO/IEC 13818-2.
4. Rao, K.R. and P. Yip, 1990. Discrete Cosine Transform: Algorithms, advantages, applications, academic press.
5. Xiong, Z., K. Ramchandran, M.T. Orchard and Ya-Qin Zhang, 1999. A Comparative study of DCT-and wavelet-based image coding. IEEE transctions on circuits and systems for video Tech., pp: 692-695.
6. Karczmarczuk, J., 1997. Images: Techniques de Synthèse, Outils de Traitement et de Compression, et Standards de Stockage, mémoire de DESS d'Informatique, Université de Caen.
7. Oliver, J. and M.P. Malumbres, 2001. A new fast lower-tree wavelet image encoder. International Conference on Image Processing, pp: 780-783.
8. Low, Y. and R. Besa, 2003. Wavelet-based Medical Image Compression Using EZW, 4th National Conference on Telecommunication Tech. Proceedings, Malaysia.
9. Shapiro, J.M., 1993. Embedded image coding using zerotres of wavelet coefficients, IEEE Trans. on Signal Processing, pp: 3445–3462.
10. Said, A. and W.A. Pearlman, 1996. A new, fast and efficient image codec based on set partitioning in hierarchical trees, IEEE Trans. Circuits and Systems for Video Tech., pp: 243-250.
11. Islam, A. and W.A. Pearlman, 1999. An embedded and efficient low-complexity hierarchical image coder, Visual communications and image processing '99, proceedings of SPIE, pp: 294-305.

12. Adams, M.D. and A. Antoniou, 2000. Reversible EZW-Based image compression using best-transform selection and selective partial embedding. IEEE Transactions on circuits and systems-II: Analog and digital processing, pp: 1119-1122.
13. RODET Thomas, 2002. Algorithmes rapides de reconstruction en tomographie par compression des calculs. Application à la tomofluoroscopie 3D. Thèse de doctorat, institut national polytechnique de GRENOBLE.
14. Woe, J., C. Olivier and C.C. Tellier, 2001. Embedded zerotree runlength wavelet image coding, Intl. Conference on image processing, pp: 784-787.
15. Ramaswamyt, V.N., K.R. Vamuduri and N. Ranganathan, 1999. Context-Modeling of wavelet coefficients in EZW-Based lossless image coding, acoustics, speech and signal processing. ICASSP '99 proceedings, pp: 3165-3167.
16. Antonini, M., M. Barlaud, P. Mathieu and I. Daubechies, 1992. Image coding using wavelet transform. IEEE transaction, image processing, 1: 205-220.
17. Villasenor, J.D., B. Belzer and J. Liano, 1997. Wavelet filter evaluation for image compresion, proc. In IEEE international conference on Image Processing ICIP'97, pp: 624-627.