International Journal of Advances in Electronics Engineering
Vol:1 Issue:1 ISSN 2278 - 215X

Unit Embedded Zerotree Coding

Akhilendra Singh Yadav
(Asst. Professor)
Electrical and Electronics Engineering,
College of Engineering Roorkee,
Roorkee, India

Paras Prateek Bhatnagar, Paramjeet Singh Jamwal,
Preeti Kumari and Nisha Rajani
Electrical & Electronics Engineering,
College of Engineering Roorkee
Roorkee, India
paras.bhatnagar89@gmail.com

***Abstract*—The EMBDDED ZEROTREE WAVELET (EZW) algorithm, as presented by J. Shapiro, is a simple yet powerful algorithm, in which bit-streams are generated in order of their significance in containing the image information [1]. The original EZW algorithm scans the entire wavelet decomposed image, at a stroke, during each pass. This work presents a modified method for coding images using EZW method, which works on the principal of fragmentation. The proposed method takes the smallest unit cell, generated from the wavelet decomposed image, to encode at a time .This makes the encoding times independent of the level of wavelet decomposition. Results show that the proposed algorithm is more efficient in performance, in terms of encoding times, as compared to the original algorithm. Moreover, the difference between the encoding times using the original and proposed method, tends to increase with an increase in the dimensions of the image under test.**

***Index Terms*—Compression, EZW, zerotrees.**

I. INTRODUCTION

TRANSFORM coding forms an integral part of the image compression techniques. Transform Coding involves a reversible, linear and unique set of coefficients, which are often quantized and then coded. This work is inspired by the success of the embedded coding algorithm known as EMBEDDED ZEROTREE WAVELET coding. Wavelet decomposition makes the energy distributed throughout the entire image to be clustered in a few sub bands, thereby creating a special parent-child-grandchild relationship (in the wavelet decomposed coefficients) known as TREE. Few such trees having a special relationship tend to become ZEROTREES, as explained in later section. This technique thereby takes advantage of the hierarchical structure of the wavelet decomposed sub images, using the parent-child-grandchild relationships for compression.

Another significant feature is the embedded coding which helps to transmit the image progressively thereby with each new set of next step encoded coefficients improving the details of the already decoded image. Using an embedded code, a coder can terminate the encoding at any desired step taking into account a desired parameter like bit-rate, as against the image quality. Similarly, for a given bit stream, the decoder can stop decoding at any step thereby producing reconstructed image corresponding to a lower-rate encoding. Optimally, for a given bit-rate, the non-embedded code must be more efficient than the embedded code, as it is free from those constraints, which are imposed by the embedded coding. However, overweighting this disadvantage, a large number of advantages like better PSNR and reduced MSE are associated with embedded coders. The EZW algorithm gives a good performance in terms of time taken for encoding, when relatively images of smaller dimensions are coded using it. However, as the image dimensions increases the time taken for encoding goes on increasing with it.

II. EMBEDDED WAVELET ZEROTREE CODING

A. *Wavelet Decomposition*

The first step in embedded wavelet zerotree coding involves decomposition of the original image into wavelet decomposed images using 2-D discrete wavelet transform. The 2-D wavelet transforms the image into four sub-images using special filters which are applied along the rows and then along the columns. This result in four sub-bands named as low-low, high-low, low-high and high-high. These sub-images are named as LL1, HL1, LH1, HH1 .The low-low sub-band may be further decomposed and divided into four sub-bands. These sub-images are obtained by using vertical and horizontal filters.

As a result of this decomposition, the entire image energy gets squeezed into the LL1 sub-band. The HL1, LH1 and the HH1 sub-band contains only the directional information of the original image. The wavelet filters are so designed that the coefficients in the sub-bands are mostly not correlated with each other. Moreover, in case of an image most of the

Figure 1. Figure showing (a) The original woman image (b) The 2-D wavelet decomposed representation of the woman image .The bands are labeled as LL1,HL1,LH1,HH1 following from top-left to bottom-right. The LL1 sub-band contains the most significant image coefficients, while HL1, LH1 and HH1 sub-bands contain only the directional information.

Vol:1 Issue:1 ISSN 2278 - 215X

information exists in the low frequency components while the higher frequency components add only intricate details to the image. The filters used in present work for obtaining the discrete wavelet transform, are based on the *Daubechies* filters as in [2].After the first level of wavelet decomposition the LL1 sub-band is again decomposed by using the same discrete wavelet transform into four another sub-band labeled as LL2,HL2,LH2 and HH2.This process is repeated a desired number of times until a finer resolved image is obtained. The present work is carried out by using 3 levels of wavelet decomposition. The third level of decomposition is obtained from the LL2 sub-band as shown in Figure 2.

B. *Tree Structure of Wavelet Coefficients*

In a hierarchical sub-band structure, each coefficient at a given scale is related to a set of coefficients at the next finer scale of similar orientation. However, the highest frequency sub-bands are exceptions, since there is no existence of a finer scale beyond them. The coefficients at the coarser scale are known as *parent* and the coefficients at the finer scale in similar orientation and same spatial location are known as *children* [9]-[11]. For a given parent, the set of all coefficients, at each finer scale having similar orientation and spatial locations is known as descendants. The wavelet decomposition of each band creates four sub-bands. Thus, each coefficient in the parent band is linked to four coefficients in the child sub-bands (except in LL3). This hierarchical structure is known as *Quad-Tree*. For a three level decomposed images as in Figure2, each coefficient in HL3,LH3 & HH3 is linked to four child coefficients in HL2,LH2 & HH2;which are further linked to sixteen coefficients in HL1,LH1 & HH1.This entire tree structure consisting of one coefficient in HL3,four coefficients in HL2 and sixteen coefficients in HL1 , is known as a Quad-tree as shown in Figure 3.A *zero-tree*, may be defined against a threshold, as a quad-tree having all its children coefficients, including its parent coefficient, lesser than the threshold [6]. The zerotree concept arises from the fact that if a DWT coefficient at a coarse scale is insignificant, then normally all its higher frequency descendants are also likely to be insignificant.

A zerotree must therefore have a root structure, which is insignificant at a threshold. Since in a wavelet decomposed

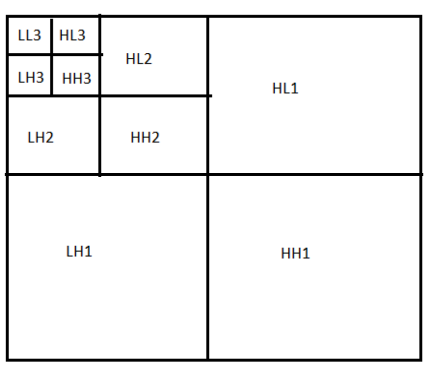

Figure 2. Figure showing the three levels of wavelet decomposition. The third decomposition level is obtained by further decomposition of LL2 sub-band. This decomposition clusters the maximum energy in the LL3 sub-band.

sub-image the maximum energy gets accumulated in the higher order sub-bands, so normally the children coefficients are smaller than the corresponding parent coefficient of the quad-tree, thereby creating a large number of zero-trees. This is illustrated by an example in Figure 4, as against a threshold of 32, here the circled coefficients represents a zero-tree.

C. *Encoding Algorithm*

The EZW algorithm is based on the following observations – When an image, having a low pass spectrum is wavelet decomposed, its energy in the LL sub-band increases with each level of decomposition and higher valued coefficients are more significant as compared to lower valued coefficients. The original algorithm begins encoding by decomposing the image to be encoded using the 2-D discrete wavelet transformation, a desired number of times. In the present work the image has been decomposed for three wavelet levels using filters based on the Daubechies filters as in [2].Next, the algorithm starts by calculating a threshold coefficient (in the power of 2), less than or equal to the maximum magnitude of the coefficients in the image. The algorithm codes using two passes. The first pass is known as the *Dominant Pass*.

This dominant pass creates a list containing coefficients known as Subordinate list. Then, this Subordinate list

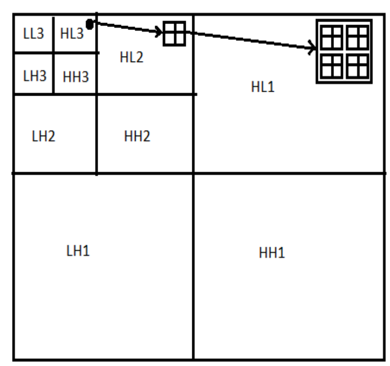

Figure 3. Figure showing a quad-tree obtained after three levels of wavelet decomposition. The arrow points from the parent to child to grand-child. Similar relationship exists for LH3 & HH3 sub-band also. However this relationship does not exist in LL3 sub-band.

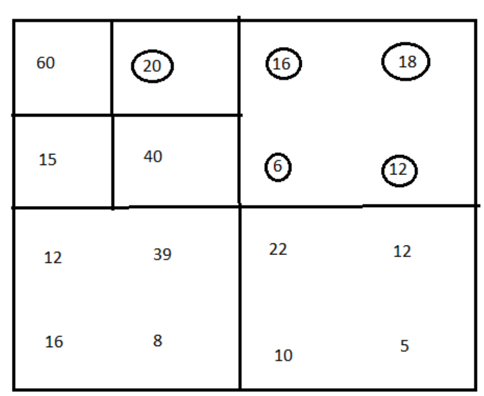

Figure 4. Figure showing an example of zero-tree structure for two level wavelet decomposed image of 4x4(dimensions). The example assumes a threshold value of *32*.The *circled* coefficients represents a zero-tree. The coefficient values of *39* and *40* in the *LH1* and *HH2* sub-bands respectively, prevent the corresponding quad-trees to turn into zero-trees.

undergoes through another pass known as *Subordinate Pass* [4].

The image coefficients are scanned in a prescribed order known as *Morton* scanning order as shown in Figure 5.The lower frequency sub-bands are scanned before the higher frequency sub-bands. Then the scanning continues to the next finer scale. This ensures that the all parent coefficients are scanned before the child coefficients. This process continues till the highest frequency sub-band is covered. This order is to be calculated, depending on the dimensions of the image. The method exploits the fact that a tree once identified to be a zero-tree is exempted from coding and is assigned a symbol once only , thereby all its descendents are prevented from being scanned again. The dominant pass works on the simple concept of significance. A coefficient becomes significant with respect to a given threshold if its magnitude is greater than or equal to the threshold. At the start, each coefficient is assumed to be insignificant and progressively, more and more significant coefficients are detected. In the dominant pass, the magnitude of the coefficient scanned is compared against the threshold. On comparison two conditions arise – either the scanned coefficient is significant with respect to the threshold or it is insignificant. If former condition is satisfied then again the coefficient checked for the sign coding. If this coefficient is positive it is assigned a symbol '*p*' otherwise it is assigned a symbol '*n*'. Moreover, this coefficient is added to a subordinate list, which is to be next scanned in the sub-ordinate pass. If the latter condition is satisfied then the entire tree-structure is considered .If the tree structure is a zero-tree , then it is assigned a symbol't', and is marked a zero-tree. Otherwise, a non zero-tree structure is coded as an isolated zero, and is assigned a symbol '*z*'. The dominant pass thereby creates two sets – one containing the coefficients marked using the four symbols, known as significance map and the other containing a list of coefficients having a significant magnitude than the present threshold, known as subordinate list. This coding reduces the cost of encoding the using the similarity property . Though,2D-DWT essentially decorrelates the coefficients, however the occurrence of insignificant coefficients is not an independent event. Rather,it is easier to predict insignificance, than to predict significant details. Zerotree coding therefore exploits this redundancy among such insignificant coefficients.The following subordinate pass scans all the coefficients of the sub-ordinate list which has been created or concatenated during the present dominant passes. The coefficients in this sub-ordinate list are retained during each pass .After each pass the coefficients found significant during the previous dominant pass are added to this list. This pass is followed by a subordinate pass, during which the coefficients shortlisted during the dominant pass, are scanned again in order to add increased precision to image detail. This is done by splitting the region of uncertainty into two halves - one of them, being greater than half the present threshold and the other one being smaller. For the former half the symbol '1' is assigned while for the latter half the symbol '0' is assigned [3]. The coefficients in the sub-ordinate list are sorted in such an order, as to enable the decoder to carry out the same sorting. The process alternates between both the passes and the present threshold is halved at each pass. In this manner all the coefficients are scanned and the process stops when the present threshold tends to become unity or at a stage, based on any desired image parameter (like the bit-rate or PSNR). The coefficients which were found significant during the present dominant pass are replaced by zero and the process is repeated. The stream of symbols generated from the dominant pass is then compressed using any entropy coding algorithms like the *HUFFMAN* coding or *ARITHMETIC* coding [5], [8], [12]. The present work uses Huffman coding. The symbols generated during the subordinate pass need not to be entropy coded. Actually, EZW encoding reorders the wavelet coefficients in such a way that they can be compressed very efficiently. Therefore the EZW encoder is always followed by an entropy encoder. Since the wavelet coefficients are reordered, in accordance of their importance in determining the image structure, so the decoding algorithm requires the scanning order in which the image was encoded.

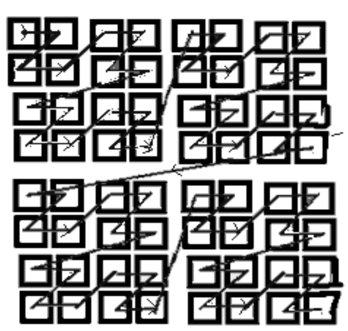

Figure 5. Figure showing scanning order for a 8x8 image .The scanning begins in the order : LL3,HL3,LH3,HH3,HL2,LH2,HH2,HL1,LH1 & HH1.The scanning order continues in the direction of arrow known as *Morton* scanning.

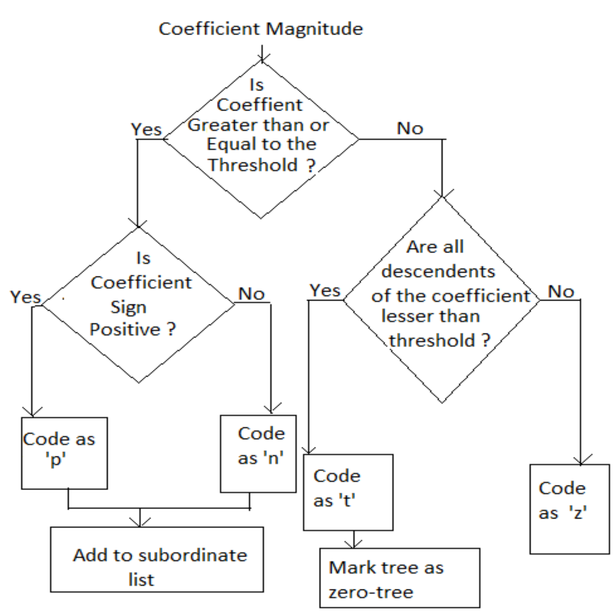

Figure 6. Figure showing flowchart for encoding coefficient during the dominant pass. This pass creates two output sets – one containing the four coded symbols and a sub-ordinate list set.

Vol:1 Issue:1 ISSN 2278 - 215X

III. Proposed Algorithm

A. Unit Cell

The proposed algorithm relies heavily on the concept of a unit cell. A unit cell may be defined as the smallest possible square matrix generated from the wavelet decomposed image, having the same level of wavelet decomposition structure as the original image. For a three level wavelet decomposed sub-image, the unit cell must be a square matrix, having the order of eight. A unit cell is composed of a single coefficient from the lowest frequency & the coarsest parent sub-band, followed by its quad-tree coefficients from the next higher frequency & finer child sub-band. This process continues till the finest scale sub-band has been reached.

B. Encoding Algorithm

Similar to the original algorithm, the proposed algorithm begins the encoding process, by the wavelet decomposition of the image, a desired number of times. The threshold is calculated for the entire image, once only and the same is used throughout the process. From the wavelet decomposed sub-image unit cells are generated. These generated unit cells are then coded using the original EZW coding algorithm. These unit cells are then coded serially. For the embedded coding, the first dominant pass coefficients of all unit cells are combined followed by the next first sub-ordinate pass. The stream generated during dominant pass undergoes through a *RUN-LENGTH* coder. This run-length coding is necessary because the generated bit-stream has a high order of redundancy, which arises due to the fact, that a single threshold value is utilized for each unit-cell. This run-length coding is followed by entropy coding. Similar to the original algorithm the bit-stream generated during the sub-ordinate pass, needs not be run-length or entropy coded. This is followed by the next dominant and sub-ordinate pass and the process is repeated based on any desired image parameter. Once this parameter is obtained the encoder stops. The decoder must be aware of the order in which the unit cells were encoded, in order to be able to reconstruct the original image.

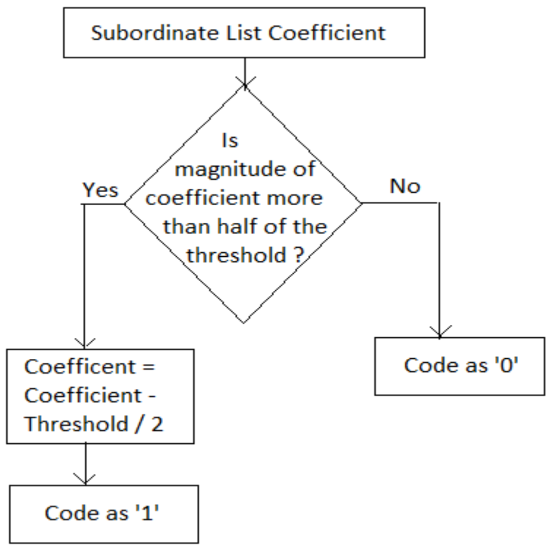

Figure 7. Figure showing flowchart for encoding subordinate list coefficient during the subordinate pass. This pass creates a single set containing the symbols '1' and '0'.

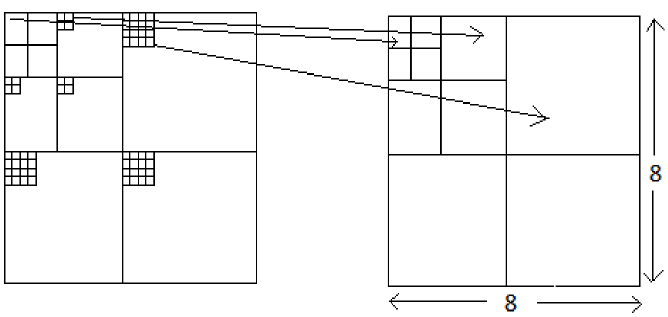

Figure 8. Figure showing formation of a unit cell from a wavelet decomposed sub-image. The arrow points from the sub-image to the unit cell. Similar relationship exists for elements in the LH, HH & HL sub-bands also.

IV. Results

The results have been obtained for different test-images having varying dimensions. The results were obtained on the same machine for three levels of wavelet decomposition [7]. For the original algorithm the entire image was scanned and the coding time was noted, while for the proposed algorithm, unit cells were composed, scanned and coded and again the coding time was noted. Results for the same have been shown in Table I. The results show a considerable improvement in the coding time using the proposed algorithm as compared to original algorithm.

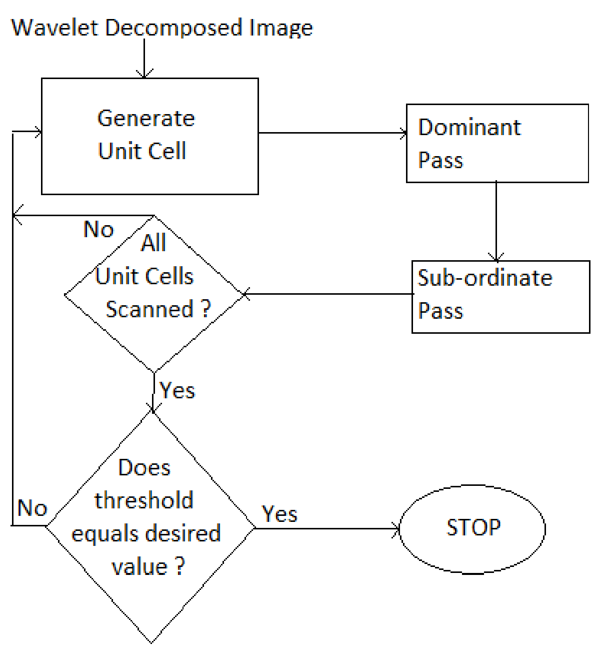

Figure 9. Figure showing the flowchart for the proposed algorithm.

Vol:1 Issue:1 ISSN 2278 - 215X

TABLE I. TABLE SHOWING CODING TIME FOR THE ORIGINAL AND PROPOSED ALGORITHM (SECONDS)

Image	**Original algorithm**					**Proposed algorithm**				
	Dimensions (Pixels × Pixels)	***32 × 32***	***64 × 64***	***128 × 128***	***256 × 256***	***Dimensions (Pixels × Pixels)***	***32 × 32***	***64 × 64***	***128 × 128***	***256 × 256***
LENA		1.092	4.540	20.062	210.617		0.952	3.479	13.120	52.073
BARBARA		1.108	4.477	19.407	213.425		0.983	3.572	13.915	54.632
CAMRAMAN		1.139	4.618	20.639	200.960		0.998	3.588	13.463	53.867
GOLDHILL		1.136	4.680	20.779	216.670		0.996	3.510	13.541	54.226
PEPPERS		1.186	4.524	19.812	214.626		1.030	3.650	14.258	54.632

TABLE II. TABLE SHOWING IMPROVEMENT IN CODING TIME USING PROPOSED ALGORITHM OVER ORIGINAL ALGORITHM (PERCENT)

Image	**Dimensions (Pixels × Pixels)**			
	32 × 32	***64 × 64***	***128 × 128***	***256 × 256***
LENA	12.821	23.370	34.603	75.276
BARBARA	11.282	20.214	28.299	74.402
CAMRAMAN	12.379	22.304	34.769	73.195
GOLDHILL	12.324	25.000	34.833	74.973
PEPPERS	13.153	19.319	28.034	74.545

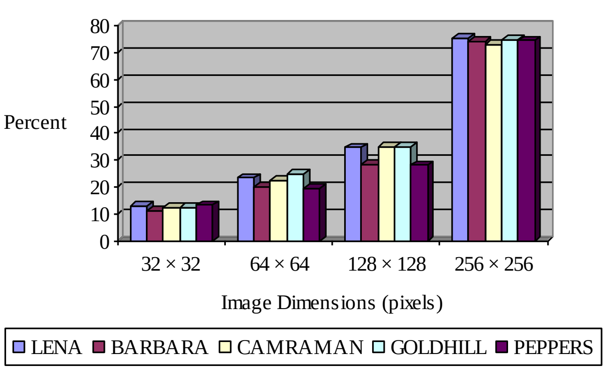

Figure 10. Figure showing improvement in coding time using proposed algorithm over original algorithm.

REFERENCES

[1] Jerome M. Shapiro, "Embedded image coding using zero trees of wavelet coefficients", IEEE Trans. Signal Process. 41 (12), (December 1993) 3445-3462.

[2] Daubechies, I. (1992), ten lectures on wavelets, CBMS-NSF conference series in applied mathematics. SIAM Ed.

[3] Module 5, "Embedded Wavelet Coding", Lesson 13, "Wavelet Approach", Version 2, ECE IIT, Kharagpur.

[4] C. Valens, "EZW Encoding Tutorial" [Online].Available: http://polyvalens.pagesperso-orange.fr/clemens/download/ezwe_26022004.pdf

[5] Vinay U. Kale, Shirish M. Deshmukh, "Visually Improved Image Compression by Combining EZW Encoding with Texture Modeling using Huffman Encoder", IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 3, No 11, May 2010,p. 28-38.

[6] Yushin Cho and William A. Pearlman, "QUANTIFYING THE CODING POWER OF ZEROTREES OFWAVELET COEFFICIENTS: A DEGREE-K ZEROTREE MODEL", Rensselaer Polytechnic Institute, Troy, NY, USA.

[7] Xiangang Li, EZW MatLab Code [Online].Available: http://polyvalens.pagesperso-orange.fr/clemens/download/ezwmatlab2.zip.

[8] Janaki. R and Dr.Tamilarasi.A, "Still Image Compression by Combining EZW Encoding with Huffman Encoder", International Journal of Computer Applications (0975 – 8887), Volume 13– No.7, January 2011, p.1-7.

[9] Arian Munteanu, Jan Cornelis, Geert Van der Auwera and Paul Cristea, "Wavelet Image Compression—The Quadtree Coding Approach", IEEE TRANSACTIONS ON INFORMATION TECHNOLOGY IN BIOMEDICINE, VOL. 3, NO. 3, SEPTEMBER 1999,pp 176-185.

[10] C.-Y. Teng, D. L. Neuhoff, J. A. Storer, and M. Cohn, "Quadtree-guided wavelet image coding," in Proc. IEEE Data Compression Conf., 1996, pp. 406–415.

[11] G. J. Sullivan and R. L. Baker, "Efficient quadtree coding of images and F. Golchin and K. Paliwal, "Quadtree based classification with arithmetic and trellis coded quantization for subband image coding," in Proc. Int. Conf. ASSP, 1997, pp. 2921–2924.

[12] R. L. Joshi, V. J. Crump, and T. R. Fischer, "Image subband coding using arithmetic coded trellis coded quantization," IEEE Trans. Circuits yst. Video Technol., vol. 6, pp. 515–523, 1995.