Embedded Zero Tree as Image Coding

Xiaoyan Xu
School of Engineering, University of Guelph
xux@uoguelph

***Abstract*— In this paper, I implemented the embedded zerotree wavelet algorithm(EZW), which is a simple, yet remarkably effective, image compression algorithm. The experiment is done on a set of standard images and the results show the good performance of this algorithm compared to some other compression scheme. EZW has proven to be a very effective image compression method based on the mean-square error(MSE) distortion measure. However, the MSE does not guarantee preservation of good perceptual qualities in the decoded image, especially at low bit rates. Therefore, the improvement is done to the EZW by implementing a perceptually-tuned embedded zerotree image codec(PEZ) which introduces a perceptual weighting to the wavelet transform coefficients prior to EZW encoding. The perceptual weights for all subbands are computed based on the just noticealbe distortion(JND) thresholds for uniform noise. Coding results shown in this paper illustrates the performance of this improevment.**

***Index Terms*— wavelet transform, embedded zerotree coding, peak signal to noise ratio(PSNR), perceptual-tuned zerotree image codec**

I. Introduction

A. Image compression using wavelet

Uncompressed multimedia(graphics, audio and video) data requires considerable storage capacity and transmission bandwidth. Despite rapid progress in mass-storage density, processor speeds, and digital communication system performance, demand for data storage capacity and data-transmission bandwidth continues to outstrip the capabilities of available technologies. The only solution is to compress multimedia data before its storage and transmission, and decompress it at the receiver for play back. There are two ways of classifying compression techniques: (a) Loeeless vs. Lossy compression; (b) Predictive(DPCM for example) vs. Transform coding. Over the past several years, the wavelet transform has gained widespread acceptance in signal processing in general, and in image compression research in particular. The principle of the wavelet transform is to hierarchically decompose an input signal into a series of successively lower resolution reference signals ans their assiciated detail signals. At each level, the reference signal and the detail signal contain the information needed to reconstruct the reference signal at the next higher resolution level. Efficient image coding is enabled by allocating bandwidth accoeding to the relative importance of information in the reference and detail signals and then applying scalar or vector quantization to the transformed data values.

There are several ways wavelet transforms can decompose a signal into various subbands. These include uniform decomposition, octave-band decomposition, and adaptive or wavelet-packet decomposition. Out of these, octave-band decomposition is the most widely used. This is a non-uniform band splitting method that decomposes the lower frequency part into narrower bands and the high-pass output at each level is left without any further decomposition. Figure 1 shows the various subband images of a 3-level octave-band decomposed Lena using the 9/7 biorthogonal wavelet. Over the past few years, a variety of novel and sophisticated wavelet-based image coding schemes have been developed. These include EZW, SPIHT, SFQ, CREW, EPWIC, SP, PEZ, Second generation image coding, Image Coding using Wavelet Packets, Wavelet Image Coding using VQ, and Lossless Image Compression using integer Lifting. More and more such innovative techniques are still being developed. In this paper, I only implemented the most original algorithm EZW to do the image compression and then explored the improvement scheme PEZ. The result and the performance are shown in the following section.

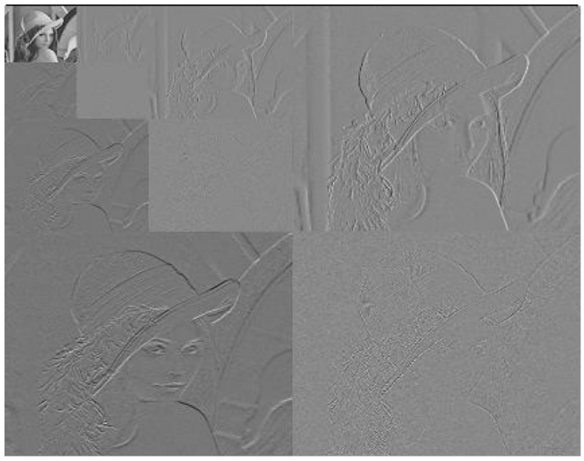

Fig. 1. 3 level wavelet decomposition of Lenna

B. Embedded Zerotree Wavelet(EZW) Compression

In octave-band wavelet decomposition, shown in Figure 2, each coefficient in the high-pass bands of the wavelet transform has four coefficients corresponding to its spatial position in the octave band above in frequency. Because of this very structure of the decomposition, Lewis and Knowles in 1992 were the first to introduce a tree-like data structure to represent the coefficients of the octave decomposition. EZW is based on 2 hypothesises: the first is that if a wavelet coefficient at the location of the coarse level is insignificant with respect to a given threshold T, then all the wavelets coefficients at the same location of higher level are most likely to be insignificant with respect to the same threshold. This hypothesis determine the *ZeroTree* structure and this hypothesis is always correct in practical images. Another hypothesis is the larger wavelet coefficient is more important than the smaller one. From Figure 1 we see that the wavelet transform tends to concentrate the mose energy to the coarsest level, thus the second hypothesis determine the scanning scheme in EZW algorithm which is zig-zag scanning(Figure 2).

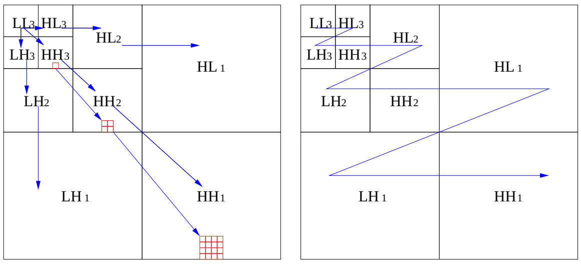

Fig. 2. Structure of ZeroTree and scanning scheme in EZW algorithm

Later, in 1993 Shapiro[1] built an elegant algorithm for entropy encoding called *Embedded Zerotree Wavelet(EZW) algorithm*. The zerotree is based on the hypothesis that if a wavelet coefficient at a coarse scale is insignificant with respect to a given threshold T, then all wavelet coefficients of the same orientation in the same spatial location at a finer scales are likely to be insignificant with respect to T. Many insignificant coefficients at higher frequency subbands(finner resolutions) can be discarded. This results in bits that are generated in order of importance, yielding a fully embedded code. The main advantage of this encoding is that the encoder can terminate the encoding at any point, thereby allowing a target bit rate to be met exactly. Similarly, the decoder can also stop decoding at any point resulting in the image that would have been produced at the rate of the truncated bit stream. The algorithm produces excellent results without any pre-stored tables or codebooks, training, or prior knowledge of the image source.

C. Perceptual-tuned Embedded Zerotree codec(PEZ)[2]

For an image the ultimate reciever is the human visual system, and image perception is affected by its sensitivity and masking properties. However, most of the existing methods for image coding are designed to minimize tractable distortion criteria such as the MSE between the images at the input and output of the codind system. Minimizing such distortion measures does not necessarily guarantee preservation of good perceptual quality of the reconstructed images and may result in visually annoying artifacts despite the potential for a good *signal-to-noise ratio(SNR)*. The block diagram of PEZ method is shown in Figure 3. After the wavelet pyramid decomposition, the coefficients in each band are multiplied by the perceptual weighting factor derived for this band. The resulting set of coefficients is then encoded using EZW. At the decoder, the perceptually-weighted transform coefficients are first decoded followed by the inverse of the perceptual weighting operation. Subsequently, the reconstructed image is produced by the inverse wavelet pyramid transform.

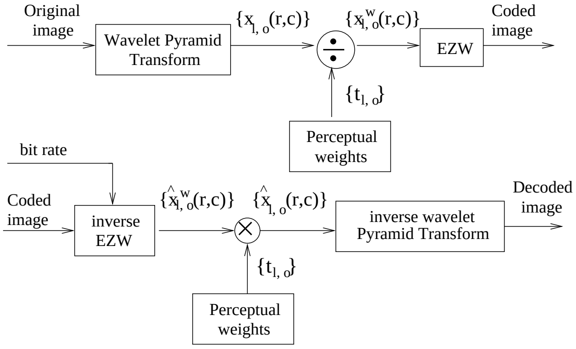

Fig. 3. the block diagram of perceptually-tuned EZW

A measure commonly used to quantify perceptual distortion is the Minkowsky metric:

$$M_\beta = \left(\frac{1}{N}\sum_{l,o}\sum_{r,c}\left(\frac{x_{l,o}(r,c)-\hat{x}_{l,o}(r,c)}{t_{l,o}(r,c)}\right)^\beta\right)^{\frac{1}{\beta}} \quad (1)$$

Where $x_{l,o}(r,c)$ is the subband coefficient located at position (r,c) in band (l,o)(l represents the level, o represents the orientation), $\hat{x}_{l,o}(r,c)$ is the corresponding coefficient in the wavelet pyramid representation of the decoded image, $t_{l,o}(r,c)$ denotes the just noticeable detection threshold for a distortion at the location under consideration, and N denotes the number of pixels in the image.

There is a variety of models to compute $t_{l,o}(r,c)$. In this project, I just chose $t_{l,o}(r,c) = t_{l,o}$ which are the just noticeable distortion(JND) thresholds for uniform noise injected in subband l,o of a newtral gray level image. The JND thresholds $t_{l,o}$ for the 3 level pyramid are shown in Table I.

Orientation	level		
	1	2	3
0			0.33
1	8.33	1.24	0.50
2	10.11	3.50	0.66
3	6.57	1.39	0.50

TABLE I

JND THRESHOLDS $t_{l,o}$ FOR THE 3 LEVEL PYRAMID

D. Paper Organization

Section II discusses the main method implemented in this project which includes the wavelet basis, the procedure of EZW and the pseudo code of the algorithm, the procedure of PEZ. Section III presents the experimental results for various rates and for various standard test images. A comparison between EZW and PEZ is done and a discussion follows each results is also presented in this section. The paper concludes with Section IV.

II. METHODS

A. wavelet basis choosing

Many issues relating to the choice of filter bank for image compression remain unresolved. Constraints on filter bank include perfect reconstruction, finite-length, and the regularity requirement that the iterated lowpass filters involved converge to continuous functions. According to [3], it shows that the 9/7 biothogonal wavelet filter banks has a very good performance for wavelet image compression.They have good localization properties as well as their summetriy allows for simple edge treatments. They also produce good results empirically since the original paper[1] on EZW is using this wavelet basis. Moreover, using properly scaled coefficients, the transformation matrix for a discrete wavelet transform obtained using these filters is so close to unitary that it can be treated as unitary for the purpose of lossy compression. Therefore, I implemented this wavelet basis too in this project.

B. Procedure of EZW

The EZW algorithm is based on four key concepts: 1)a discrete wavelet transform or hierarchical subband decomposition, 2)prediction of the absence of significant information across scales by exploiting the self-similarity inherent in images, 3)entropy-coded successive-approximation quantization, and 4)universal lossless data compression which is achieved via adaptive arithmetic coding.

The procedure of EZW is:
1. discrete wavelet transform to a given image
2. encoding a coefficient of the significance map(DominantPass).

- A wavelet coefficient x on the Dominant List is said to be insignificant with respect to a given threshold T if$|x| < T$.
- 4 simbols are used: ZTR(zerotree root);IZ(isolated zero);NEG(negative significant);POS(positive significant).
- For each coefficient coded as significant(POS or NEG), put its magnitude on the Subordinate List and remove it from the Dominant List.

The flow chart of this procedure is shown in Figure4. 3.

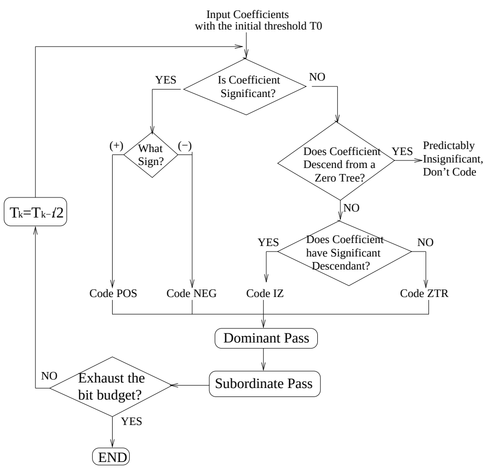

Fig. 4. Flow chart for encoding a coefficient of the signifcant map

subordinate pass(also called refinement pass).

- Provide one more bit on the magnitudes on the Subordinate List as follows:
  - Halve the quantizer cells
  - If magnitude is in upper half of old cell, provide ”1”
  - If magnitude is in lower half of old cell, provide ”0”
- Entropy code sequence of 1’s and 0’s

Stop when bit budget is exhausted. Encoded stream has embedded in it all lower-rate encoded versions. Thus, encoding/decoding can be terminated prior to reaching the full-rate version.

C. EZW Encoder Pseudocode

The pseudocode implemented in this project for the embedded zerotree coding is shown in Table II.

TABLE II
PSEUDOCODE FOR EZW CODING

Initialization
$T_0 = 2^{floor(log2(max(coeffs)))}$
k=0
Dominant List=All coefficients
Subortinate List=[]

SignificantMap
for each coefficient$x$ in the Dominant List
  if $|x| \geq T_k$
    if $x > 0$
      set symbol POS
    else
      set symbol NEG
  else if x is non-root part of a zerotree
      set symbol ZTD(ZeroTree Descendant)
    if x is zerotree root
      set symbol ZTR
    otherwise
      set symbol IZ

DominantPass
if symbol(x) is POS or NEG(it is significant)
  put symbol(x) on the Subordinate List
  Remove x from the Dominant List

SubordinatePass
for each entry symbol(x) in Subordinate List
  if value(x)$\in$ Bottom Half of [T_k,$2T_k$]
    output ”0”
  else
    output ”1”

Update
$T_{k+1} = T_k/2$
$k = k + 1$
Go to SignificanceMap

D. Perceptual-tunned embedded zerotree codec

Just to implement the algorithm mention in section I and I try to get the result as the auther in [2] mentioned. However, from the result later, we can hardly tell there is any improvement with this method.

III. EXPERIMENTS AND RESULTS

The encoded bit file include a 12-byte header which contains:1) the number of wavelet scales; 2) the dimension of image; 3) the maximum histogram count for the models in the arithmetic coder; 4) the image mean and 5) the initial threshold. After that, the entire bit stream is arithmetically encoded using a single arithmetic coder with an adaptive model.

The EZW algorithm in this project is applied to the standard black and white 8 bpp test image, 512×512"Lena".

A. *performance of EZW algorithm*

Discussion: As mentioned previously, one of the advantages of EZW is that it encode the image from lossy to lossless in one algorithm. People at the receiver can choose the quality of the image by control the bit budget. As the bit rate increases, you will get more detailed information and of course the image quality becomes better and better. Figure 5 shows this procedure. You can clearly observe some block effect at the lower bit budget, this is due to my implementation in MATLAB which is very slow doing large number of *For loop*. Figure 6 shows the decoded PSNR(Peak Signal-to-Noise Ratio) vs. bit rate curve. This is quite consist with what we suppose to be.

Fig. 5. decoded image given different bit budget. Left top: lowest bit rate(bpp), right bottom: highest bit rate(bpp)

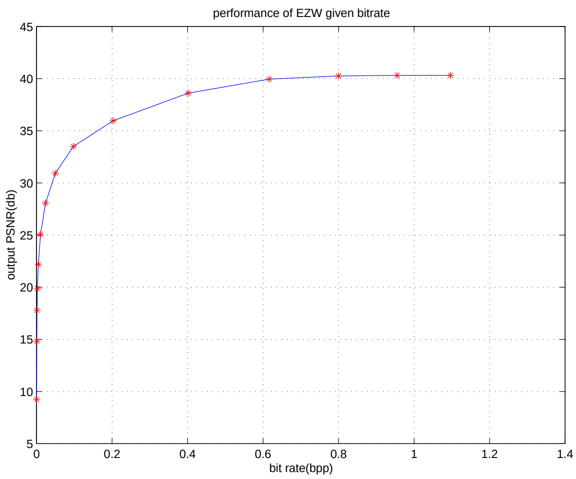

Fig. 6. The performance of EZW algorithm

B. *comparison with PEZ algorithm*

Discussion: It is reported[2] that the PEZ removes many artifacts especially at lower bit rate and the result is shown in Figure 7. We can see that PEZ provides a lower PSNR and when I look at the image, I can hardly tell the improvement of PEZ claimed in[2]. Also I plot the comparison between EZW and PEZ in Figure 8, which shows that the PEZ always provide lower PSNR than EZW. Figure 9 shows the decoded images of these two methods at bpp. The only thing I can tell is the PEZ decoded image is a little smoother than EZW decoded image, which is due to the effects of weighting the wavelet coefficients something like the low pass filter. Therefore, my conclusion is the performance of PEZ is not that good as claimed.

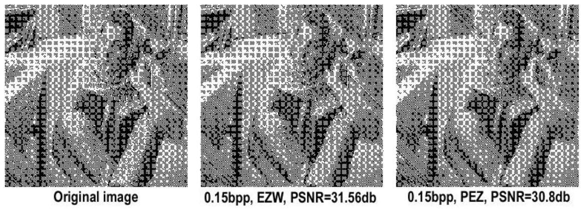

Fig. 7. result obtain from [2]

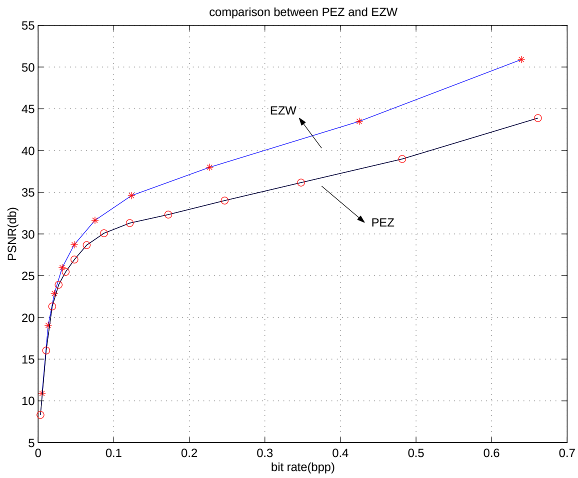

Fig. 8. comparison between PEZ and EZW

C. *comparison with baseline JPEG*

Firstly, I should point out that the baseline JPEG coding results that I use as the performance benchmark are far from the best that JPEG offers. Much better performance can be obtained with JPEg by optimal quantization matrix design, and coefficient thresholding, while being compatible with the JPEG syntax.

Figure 10 shows the comparison between the EZW algorithm and baseline JPEG, from which we can see that the performance of EZW is a little better than the baseline JPEG.

Discussion: It is reported that EZW is a little bettern than baseline JPEG and my result also shows that. However, as I said, people always use their best wavelet-based coding

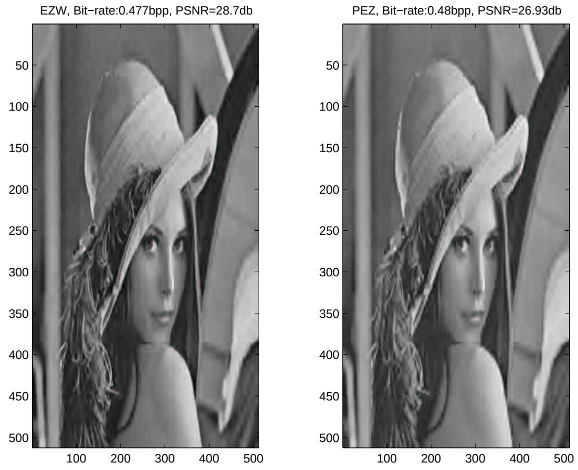

Fig. 9. coding example of EZW and PEZ

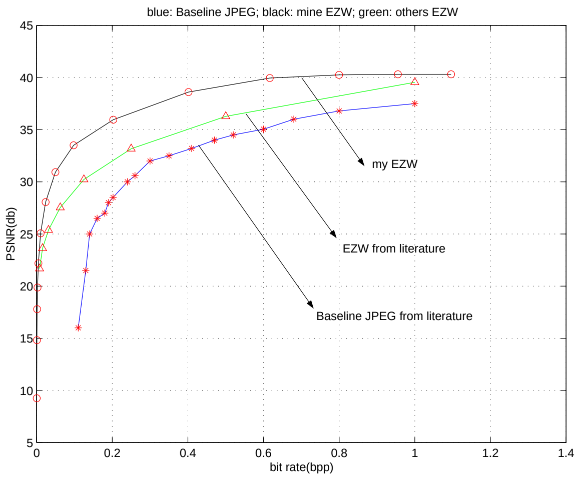

Fig. 10. comparison between EZW and baseline JPEG

scheme with the worst DCT-based scheme(baseline JPEG). This often gives reader a distorted perspective of the issues involved in image coding. The reason for people still tend to replace DCT by EZW is not only because the performance of EZW is a little better, but also due to the many other advantages of EZW:

- Having all lower bit rate codes of the same image embedded at the beginning of the bit stream.
- Bits are generated in order of importance
- Encoder can terminate encoding ar any point, allowing a target rate to be met exactly.
- Suitable for applications with scalability.

In March 1997 a new call for contributions were launched for the development of a new standard for the compression of still images, the JPEG 2000. With the release of Final Committee Draft(FCD) for Part I from ISO/IEC/JTCI/SC29/WGI in April 2000, this new standard emerges into our life. Comparing to the most popular existing standard JPEG, JPEG 2000 provides the following advantages:

- ROI(Region of Interest)
- Error resilience
- Progression orders
- Lossy and lossless in one system
- Better compression at low bit-rates
- Better ar compund images and graphics

Two wavelet filters are used in JPEG2000 Part I. The daub97 wavelet which contain floating filter coefficients is used for lossy coding. The integer wavelet 5/3 is used for both lossless and lossy.

JPEG2000 offers performance superior to the current standards at low bit-rates(e.g. below 0.25bpp). Figure 11 compare the performance between JPEG and JPEG2000 at the same bit-rate on the same image. As you can see, the JPEG compressed image is visually unacceptable(obvious block effect) while the JPEG2000 compressed image is pretty good.

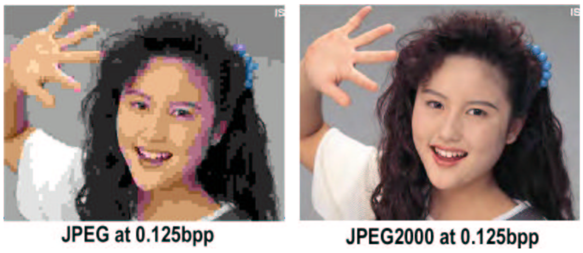

Fig. 11. comparison between JPEG and JPEG2000 at the same bit-rate

IV. Conclusion

From the results above we can see that the performance of the original EZW is a little better than the baseline JPEG, which is the worst one of JPEG. This oftengives people a distortive perspective of the issues involved in image coding. The main factors in image coding are the quantizer and entropy coder rather than the difference between the wavelet transform and the DCT. However, people still intend to replace the DCT with the EZW is not only because of the superiority of the wavelet transform, but also for its other features: 1) Zerotree structure, which provides substantial coding gains over the first-order entropy for significance maps; 2) Successive-approximation, which allows the encoding or decoding to stop at any point; 3)Adaptive arithmetic coding, which allows the entropy coder to incorporate learning into the bitstream itself. User can choose a bit rate and encode the image to exactly the desired bit rate. Furthermore, since no training of any kind is required, the algorithm is fairly general and performs remarkably well with most types of images.

Acknowledgment

The author would like to thank to Professor Dony who offered the Advanced digital signal processing course that helps me understand much better about the background of this work. Professor Dony's lots of useful suggestion is greatly appreciated. I would also thank to my husband and my parents who give the strongest support in my life.

REFERENCES

[1] Jerome M. Shapiro, *Embedded Image Coding Using Zerotrees of Wavelet Coefficients*, IEEE Transactions on Signal Processing. Vol.41, NO.12, December 1993, pp3445-3462.

[2] Ingo Höntsch, Lina J. Karam, Robert J. Safranek, *A Perceptually Tuned Embedded Zerotree Image Coder*

[3] John D. Villasenor, Benjamin Belzer, and Judy Liao, *Wavelet filter Evaluation for Image Compression*, IEEE Transaction on Image Processing, Vol 4, NO 8, August 1995. pp1053-1060.

[4] Marc Antonini, Michael Barlaud, Member IEEE, Pierre Mathieu, and Ingrid Daubechies, Member IEEE, *Image Coding Using Wavelet Transform*, IEEE TRANSACTION ON IMAGE PROCESSING, VOL.1, NO.2, April 1992, pp 205-220.

[5] Zixiang Xiong, Kannan Ramchandran, Michael T. Orchard, and Ya-Qin Zhang, *A Comparative Study of DCT- and Wavelet-Based Image Coding*, IEEE TRANSACTION ON CITCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, VOL.9, NO.5, August 1999, pp 692-695.

[6] V. N. Ramaswamy, K. R. Namuduri and N. Ranganathan, senior Member IEEE, *Performance Analysis of Wavelets in Embedded Zerotree-based Lossless Image Coding Schemes*, IEEE TRANSACTIONS ON SIGNAL PROCESSING, pp100-105.

[7] Xiaobo Li, Jason Knipe, Howard Cheng, *Image Compression and Encryption Using Tree Structures*, Pattern Recognition Letters, 18(1997), 1253-1259.

[8] Charles D. Creusere, *A New Method of Robust Image Compression Based on the Embedded Zerotree Wavelet Algorithm*, IEEE TRANSACTION ON IMAGE PROCESSING, VOL.6, NO.10, October 1997, pp1436-1442.

[9] D. M. Monro and G. J. Dickson, *Zerotree Coding of DCT Coefficients*, http://dmsun4.bath.ac.uk/icip97ztdct.pdf.

[10] Christos Chrysafis, David Taubman, Alex Drukarev, *Overview of JPEG2000*, http://sipi.usc.edu/ ortega/students/chrusafi/doc/christoschrysafis_IS_n_T_1999.pdf.