Lossless Image Compression using Tuned Degree-K Zerotree Wavelet Coding

Li Wern Chew*, Li-Minn Ang and Kah Phooi Seng

***Abstract* - This paper presents a compression technique called the Tuned Degree-K Zerotree Wavelet (TDKZW) coding. The proposed algorithm uses a set-partitioning approach and the degree of zerotree tested is tuned at each encoding pass to achieve optimal compression performance. TDKZW coding is an embedded coding algorithm that provides a progressive transmission as well as lossless compression. Simulation results show that our proposed algorithm gives a better compression efficiency compared to many existing state-of-the-art lossless image compression algorithms.**

***Index Terms*—Degree-k zerotree coding, embedded coding lossless compression, set-partitioning coding.**

I. INTRODUCTION

Image compression has become increasingly important with the continuous development of Internet, remote sensing and satellite communication techniques. Due to the high cost of providing a large transmission bandwidth and a huge amount of storage space, many fast and efficient image compression engines have been introduced.

In image processing applications such as web browsing, photography, image editing and printing, a lossy coding such as JPEG [1] is sufficient as an image compression tool. Although some information loss can be tolerated in most of these applications, there are certain image processing applications that demand no pixel difference between the original and the reconstructed image. Such applications include medical imaging, remote sensing, satellite imaging and forensic analysis where a lossless compression is extremely important.

An image compression engine like the context-based, adaptive, lossless image codec (CALIC) [2] provides a lossless and efficient image coding technique. However, CALIC is a non-embedded coder. This makes other compression algorithms such as the progressive partitioning binary wavelet-tree coder (PPBWC) [3], progressive lossless/near-lossless image coding [4] [5], JPEG2000 [6] and set-partitioning in hierarchical trees (SPIHT) [7] which provide a lossy up to a lossless compression performance preferable over CALIC.

The embedded property of such coders allows progressive transmission. Since the bit-stream is generated in accordance to the order of importance, the end user can choose to stop decoding as soon as the targeted bit-rates are achieved and yet, able to obtain a fully reconstructed image of sufficiently good quality. For a lossless reconstruction, the whole bit-stream is retrieved and a reconstructed image which is identical to the original image is obtained. The advantage of embedded coding is that both the lossy and lossless reconstructions can be performed using a single compression file.

In this paper, a Tuned Degree-K Zerotree Wavelet (TDKZW) coding is presented. In our proposed algorithm, the degree of zerotree tested is tuned at each encoding pass to achieve optimal compression performance.

The remaining sections of this paper is organized as follows: Section II gives an overview of some existing lossless image coding techniques - PPBWC, progressive lossless/near-lossless image coding, JPEG2000, SPIHT and CALIC. Section III presents our proposed TDKZW coding scheme. Section IV discusses the performance evaluation of TDKZW coder and Section V concludes this paper.

II. OVERVIEW OF LOSSLESS IMAGE CODING TECHNIQUES

A review on some of the existing state-of-the-art lossless image coding techniques is presented in this section. Five lossless image compression algorithms that will be presented are the PPBWC, progressive lossless/near-lossless image coding, JPEG2000, SPIHT and CALIC coding schemes.

A. *Progressive Partitioning Binary Wavelet-Tree Coder*

The progressive partitioning binary wavelet-tree coder (PPBWC) [3] uses a binary wavelet transform (BWT) to convert the image into a bi-level format prior to encoding. Following the BWT, an entropy coding that uses a joint bit scanning method and a non-causal adaptive context modeling is applied to encode the wavelet-transformed coefficients.

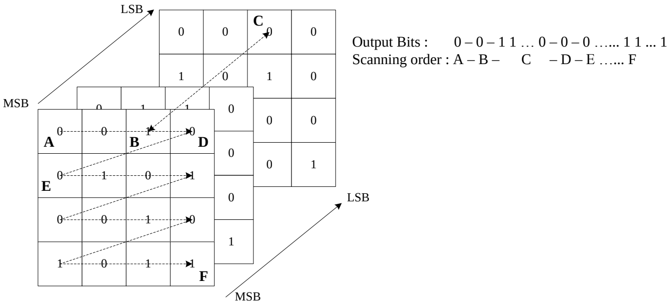

Figure 1. Zig-zag and joint bit scanning of PPBWC.

Manuscript received September 12, 2008; revised January 11, 2009. Manuscript submitted to the IAENG International Conference on Imaging Engineering, ICIE'09.

The authors are with the School of Electrical and Electronics Engineering, The University of Nottingham, 43500 Semenyih, Selangor, Malaysia. Tel: +603 8924 8350, Fax: +603 8924 8017, Email: (eyx6clw, kezklma, kezkps)@nottingham.edu.my.

For instance, if we consider an 8-bit grayscale image where a BWT is first applied to each of the eight bit-planes separately, we will get eight bi-level planes as shown in Fig. 1. Starting from the Most-Significant-Bit (MSB) plane, the coefficients are scanned in a zig-zag order. The binary values of the coefficients are output to the encoder as they are scanned. When a coefficient is found to be significant, that is when a '1' is output, the rest of the bits from the current bit-plane to the Least-Significant-Bit (LSB) are output to the encoder. This coefficient is then marked as 'significant' and will not be scanned again in the following passes.

PPBWC adopts the properties of progressive transmission, embedded coding and eliminates the need of zerotree analysis. Although it gives an efficient lossless compression efficiency, the rate-distortion of this approach is not as good compared to other embedded coders such as SPIHT and JPEG2000 [3].

B. *Progressive Lossless/Near-Lossless Image Coding*

Recently, a progressive lossless/near-lossless image coding has been proposed in [4] and [5]. This coding scheme exploits the image context information by estimating the conditional probability density function (pdf) of a pixel given its neighboring pixels information.

Given that ε is the difference between the original and reconstructed pixel value, a near-lossless image coding algorithm is able to provide a decompression quality with a very small value of ε, typically in the range of 0 to 7 [4]. For lossless compression, a ε value of zero is needed, i.e. there is no difference in pixel value between the original and the reconstructed pixel.

The compression technique using the progressive lossless/near-lossless coding is summarized as follows:

i At each encoding pass, the pdf of each pixel is first estimated.
ii With this estimated pdf, a prediction of the actual value of the pixel is carried out. Assume that the error between the predicted value and actual value is ε.
iii With the knowledge of ε, the pdf of the pixel is revised and the reconstructed pixel value is redefined.
iv This process is repeated until $\varepsilon \leq T$ for near-lossless coding where T is the predefined error tolerance and $\varepsilon = 0$ for lossless coding.

C. *Lossless JPEG2000*

Popular image compression standard JPEG2000 [6] adopts the embedded block coding with optimized truncation (EBCOT) image compression technique. In EBCOT, the information coding and the information ordering is separated and is referred to as two tiers coding. The process carried out in each tier is described below.

- Tier 1: Embedded block coding
  The wavelet-transformed image is first partitioned into code-blocks of size 32 x 32 pixels or 64 x 64 pixels. Each of these code-blocks is then encoded independently with a context-based bit-plane adaptive arithmetic coding. An embedded bit-stream is then generated for every code-block.
- Tier 2: Block contributions to each quality layer
  The independently generated bit-streams from tier 1 are subdivided into small 'chunks'. These chucks are then interleaved and packed into different quality layers depending on their contributions to the layers. A compressed bit-stream is then output for transmission.

In lossy mode, JPEG2000 uses the CDF 9/7 irreversible DWT filter whereas in lossless mode, a reversible Le Gall 5/3 DWT filter is used. Although JPEG2000 provides many features such as embedded coding, progressive transmission, resolution scalability and random access features, its multi-layer coding procedures are very complex and computationally intensive. Thus, it is currently not generally used in imaging applications.

D. *Set-Partitioning in Hierarchical Trees Coding*

The set-partitioning in hierarchical trees (SPIHT) [7] coding operates by exploiting the relationships among the wavelet coefficients across the different scales at the same spatial location in the wavelet subbands.

In general, SPIHT coding involves the coding of the position of wavelet coefficients that are found to be significant as well as the coding of the position of zerotrees in the wavelet subbands. Three lists are maintained during the set-partitioning approach, i.e. the list of significant pixels (LSP), list of insignificant pixels (LIP) and list of insignificant sets (LIS). These lists are used to keep track of the order and the position of the coefficient/set on which the significant test is to be carried out.

As the wavelet coefficients are transmitted in the order of magnitude, an embedded bit stream is generated. Thus, SPIHT coding allows a variable bit rate and rate distortion control as well as progressive transmission.

For lossless coding, SPIHT uses an integer multi-resolution transformation which is referred to as the S+P transform [8]. During the S+P transformation calculation, the number of bits required to represent the transformed image is kept small through careful scaling and truncations. Furthermore, the S+P transformation can be computed with only integer addition and bit-shift operations resulting in a low complexity implementation of image transformation.

E. *Context-Based, Adaptive, Lossless Image Coding*

In context-based, adaptive, lossless image coding (CALIC) [2], the image pixels are encoded in a raster scan format. This coding uses an image data modeling technique with nonlinear prediction in the spatial domain. With the feedback mechanism, CALIC coder enhances the coding performance by learning from its previous mistakes.

Although CALIC coder gives a very good compression efficiency as compared to many of the wavelet-based image compression techniques, it has the disadvantage of being a non-embedded coder and hence does not support signal-to-noise ratio (SNR) scalability and progressive transmission.

III. TUNED DEGREE-K ZEROTREE WAVELET CODING

A degree-k zerotree coder codes the significance of a spatial orientation tree up to k levels. A degree-k zerotree refers to a spatial orientation tree where all the tree nodes are found to be insignificant with respect to a predefined threshold except for the nodes in the top k levels.

In [9], it is shown that a SPIHT coding [7] which is a degree-2 zerotree coding gives a much better coding performance compared to embedded zerotree wavelet (EZW) coding [10] which is a degree-0 zerotree coding. This is because a higher degree-k zerotree coding provides more levels of descendant information for each coefficient tested. Thus, in the case where a descendant test on a tree node (i, j) at level k has an insignificant result '0', this zerotree with root (i, j) is encoded using a single symbol which indicates that all the nodes in the tree are insignificant. Hence, the four direct offsprings of (i, j) which are at level (k+1) are not partitioned into subtrees. In this case, four encoding bits can be saved.

A demonstration of this with a degree-3 zerotree coding is shown in Fig. 2. The notations used are summarized as follows:

- SIG(i, j) is the significance of node (i, j);
- DESC(i, j) is the significance of descendant of node (i, j);
- GDESC(i, j) is the significance of grand descendant of node (i, j);
- GGDESC(i, j) is the significance of great grand descendant of node (i, j).

In a degree-3 zerotree coding, a significance test is also carried out on the great grand descendants of the coefficient (i, j), i.e. GGDESC(i, j) in addition to the significance tests on SIG(i, j), DESC(i, j) and GDESC(i, j). If GGDESC(i, j) = 0 at the n^{th} bit-plane coding, significance tests on the grand descendants for all offsprings of (i, j), i.e. GDESC(k, l) where $(k, l) \in O(i, j)$, can be omitted. Therefore, when compared to a degree-2 zerotree coding, with just one extra GGDESC(i, j) bit transmitted in a degree-3 zerotree coding, four GDESC(k, l) bits can be saved in the best case scenario if GGDESC(i, j) is shown to be insignificant.

From this analysis, it is anticipated that the more the levels of descendant information that can be obtained for each coefficient tested, the more there will be the number of encoding bits that can be saved. However, this only applies if the extra significance test that is carried out has an insignificant result '0'. If the result of significance test is '1', then one bit will be wasted since the decoder has to search the lower level of the tree.

From our studies carried out on a higher degree-k zerotree coding, it was found that at lower bit-rates where most of the coefficients are insignificant, a higher degree zerotree coding gives a better coding performance. On the other hand, at higher bit-rates, coding with a higher degree zerotree is less efficient since the wavelet coefficients are more likely to be significant as the number of planes encoded is increased.

Thus, the degree of zerotree tested is tuned in each encoding pass in our proposed TDKZW coding scheme. At an earlier encoding pass, a higher degree-k zerotree coding is carried out and as the number of encoding planes is increased, the degree of zerotree tested is reduced. Our proposed TDKZW coding uses the set-partitioning approach similar to the SPIHT coding. It has all the properties of SPIHT coding including embedded coding and it also allows progressive transmission.

In terms of complexity, our proposed algorithm did not significantly increase the coder complexity as the fast technique used to identify zerotrees which is proposed in [11] is incorporated into our proposed TDKZW coding scheme. Since the wavelet coefficients are generated from lower subbands to higher subbands, this technique [11] involves determining the zerotree information at the (k-1) level by performing a bitwise-OR operation on the zerotree information at the k level with the parent node at (k-1) level. By applying this operation before encoding, a recursive check on the degree-k zerotree is eliminated during the sorting stage thereby increasing the TDKZW coding processing speed.

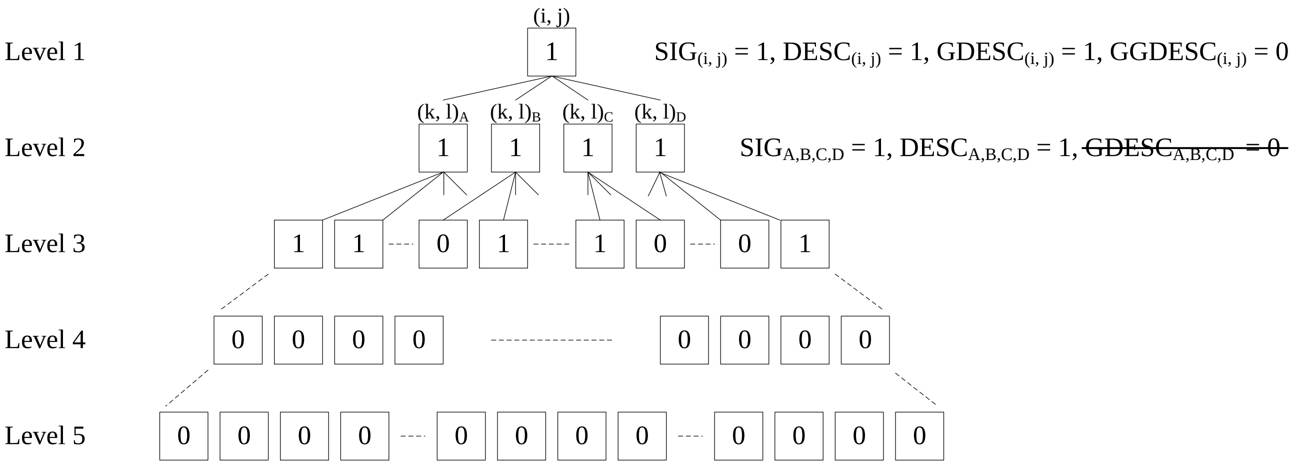

Figure 2. Degree-3 zerotree coding.

IV. PERFORMANCE EVALUATION

The lossless compression efficiency of our proposed TDKZW coding is compared with SPIHT [7], PPBWC [3], JPEG2000 [6], progressive lossless/near-lossless image coding (PDF 1-Pass) [4] [5] and CALIC [2]. All of these are embedded coders except the CALIC.

Natural grayscale images of size 512 x 512 pixels were used in our simulation. For lossless compression, our proposed TDKZW with a context-based arithmetic coding uses the reversible Le Gall 5/3 wavelet transform. A six-scale wavelet decomposition and a seven-scale spatial orientation tree decomposition were performed on the test images. The tuning table used is given in Table I.

TABLE I
TUNING TABLE FOR LOSSLESS TDKZW CODING

	(MSB)													(LSB)
Bit-Plane:	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Degree-K:	7	7	7	7	7	7	7	6	5	2	2	1	1	1

Table II shows the lossless compression efficiency obtained from the simulations carried out in terms of bit-per-pixel (bpp). The results for SPIHT, PPBWC and CALIC are extracted from [3] and the results of JPEG2000 and PDF 1-Pass are extracted from [4]. The bold text shows the lowest bpp and results that are not available are indicated by a dash line.

From the simulation results, it is shown that our proposed TDKZW coding gives a better lossless compression efficiency for four out of the seven images tested. Among the five embedded coders, TDKZW coding is found to require the fewest encoding bits except for image Barbara. It is also shown that the TDKZW coding has the highest average lossless compression efficiency of 4.13 bpp compared to the other embedded coding techniques, with a maximum difference of 0.55 bpp.

Although CALIC gives a slightly better average lossless coding performance compared to TDKZW coding with an average of 4.11 bpp and 4.13 bpp respectively, it is a non-embedded coder. In comparison, our proposed TDKZW coder not only provides embedded coding, it also allows progressive transmission. This makes our TDKZW coding applicable to both lossless and lossy image compression.

V. CONCLUSION

Our proposed Tuned Degree-K Zerotree Wavelet (TDKZW) coding gives a better lossless coding performance than many state-of-the-art lossless image compression techniques. Since the degree of zerotree tested in tuned in each encoding pass, the proposed algorithm gives optimal compression efficiency. In addition, TDKZW coding which uses the set-partitioning approach can be applied to both lossless and lossy image compression.

REFERENCES

[1] C. Christopoulos, A. Skodras, T. Ebrahimi, “The JPEG sill image coding system: an overview,” *IEEE Trans. Cons. Elect.*, vol. 46, pp. 1103-1127, 2000.

[2] X. Wu and N. Memon, “Context-based, adaptive, lossless image coding,” *IEEE Trans. Comm.*, vol. 45, no. 4, pp. 437-444, Apr. 1997.

[3] H. Pan, W. -C. Siu, N. -F. Law, “Lossless image compression using binary wavelet transform,” *IET Image Process.*, vol. 1, no. 4, pp. 353-362, Dec. 2007.

[4] I. Avcıbas, N. Memon, *B.* Sankur, *K.* Sayood, “A Successively Refinable Lossless Image-Coding Algorithm,” *IEEE Trans. Comm.*, vol. 53, no. 3, Mar. 2005.

[5] I. Avcıbas, N. Memon, *B.* Sankur, *K.* Sayood, “A Progressive Lossless/Near-Lossless Image-Compression Algorithm ,” *IEEE Signal Process. Lett.*, vol. 9, no. 10, Oct. 2002.

[6] D. Taubman and M. W. Marcellin, *JPEG 2000, Image Compression, Fundamentals, Standards and Practice*, Kluwer Academic Publishers, 2002.

[7] A. Said and W. A. Pearlman, “A new fast/efficient image codec based on set partitioning in hierarchical trees,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 12, pp. 243-250, Jun. 1996.

[8] A. Said and W. A. Pearlman, “An image multiresolution representation for lossless and lossy compression,” *IEEE Trans Image Process.*, vol. 5, no. 9, pp. 1303-1310, Sep. 1996.

[9] Y. Cho and W. A. Pearlman, “Quantifying the coding performance of zerotrees of wavelet coefficients: degree-k zerotree,” *IEEE Trans. Signal Process.*, vol. 55, no. 6, pp. 2425-2431, Jun. 2007.

[10] J. M. Shapiro, “Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3445-3462, Dec. 1993.

[11] J. M. Shapiro, “A fast technique for identifying zerotrees in the EZW algorithm”, *ICASSP-96*, vol. 3, pp. 1455-1458, May. 1996.

TABLE II
PERFORMANCE COMPARISON OF TUNED DEGREE-K ZEROTREE WAVELET (TDKZW) CODER WITH EXISTING CODERS ON NATURAL IMAGES IN TERMS OF LOSSLESS COMPRESSION EFFICIENCY REPRESENTED BY BIT-PER-PIXEL (BPP)

	Embedded Coders					Non-Embedded Coder
Images	TDKZW	SPIHT [7]	PPBWC [3]	JPEG2000 [6]	PDF 1-Pass [4]	CALIC [2]
Airplane	**3.28**	3.97	3.92	-	-	3.83
Barbara	4.50	4.82	4.68	4.69	**4.21**	4.49
Boat	4.01	4.29	4.08	4.43	4.08	**3.78**
Fingerprint	**5.15**	-	-	5.69	5.40	-
Goldhill	4.51	4.63	4.54	4.87	4.69	**4.39**
Lenna	**3.91**	4.18	4.15	4.35	4.07	4.04
Zelda	**3.58**	-	-	4.02	3.79	-
Average:	4.13	4.38	4.27	4.68	4.37	**4.11**