Embedded Coding for 1-D Signals Using Zerotrees of Wavelet Coefficients

Y.S. Mo, W.-S. Lu, and A. Antoniou
Department of Electrical and Computer Engineering
University of Victoria
Victoria, B.C., CANADA, V8W 3P6
Emails: {ymo, wslu, andreas}@ece.uvic.ca

ABSTRACT— Shapiro's concept of embedded zerotree wavelet coding for images is used to develop an embedded coding method for 1-D signals. In this method, the zerotrees of the wavelet coefficients are used to predict insignificant coefficients across scales. The bit stream is generated in the order of importance. The decoder can generate the same zerotrees as the encoder. The embedded property of this method allows precise bit rate and signal quality control. The algorithm is illustrated by an example. Embedded coding is then applied to bathymetric waveforms. Simulation results indicate that the method is quite efficient and produces a fully embedded coding for 1-D signals.

I. INTRODUCTION

Signal compression is of critical importance for efficient data storage and transmission, and has been a subject of study during the last two decades [1]-[6]. An important recent development in signal coding is embedded image coding [3][4] in which the bit stream is generated in the order of importance. In [3], Shapiro proposed an embedded method based on the concept of the zerotrees of the wavelet coefficients for image coding. To date, Shapiro's embedded zerotree wavelet (EZW) method remains one of the most efficient image compression methods [6].

In Shapiro's method, a two-dimensional (2-D) discrete wavelet transform (DWT) is applied to an input image of size $N \times N$ with $N = 2^p$, where p is an integer. This results in a maximum of p sets of subband images. By exploiting the similarity between the subimages at different scales, efficient coding of the decomposed images can be achieved. The encoded subband images are then transmitted and the inverse discrete wavelet transform (IDWT) is used to reconstruct the image.

Since similarity between subsignals at different scales is inherent for most real-world signals, one would expect that efficient signal compression may be possible using a wavelet-based coding method.

In this paper, Shapiro's concept of embedded zerotree wavelet coding for images is applied to develop an embedded coding method for 1-D signals. The main components of the algorithm are as follows [3]:

- A DWT is used to obtain a multiscale representation of the input signal.
- The zerotrees of the decomposed signal are defined and used to identify insignificant coefficients across scales.
- The wavelet coefficients are transmitted in the order of importance.
- During the dominant passes, the decoder generates the same zerotrees as the encoder.
- During the subordinate passes, the decoder uses the mid-point of the uncertainty intervals as the reconstruction coefficients.
- The algorithm runs sequentially and stops when a target distortion metric is met.

The proposed algorithm was used to compress a large number of bathymetric waveforms. Simulation results indicate that this method is quite efficient for the compression of 1-D signals and produces a fully embedded coding.

II. ALGORITHM DESCRIPTION

A. Progressive Transmission

A typical low bit rate transmission system based on wavelets has four components: a DWT, an encoder, a decoder, and an IDWT. The DWT and IDWT used in this paper form a two-channel hierarchical subband system. In a progressive transmission, the decoder initially sets the reconstruction vector to zero and updates its components progressively according to the decoded coefficients. The DWT can be defined by $\mathbf{y} = \mathbf{W}\mathbf{x}$, where $\mathbf{x}$ is the input column vector, $\mathbf{y}$ is vector whose elements are the wavelet coefficients and $\mathbf{W}$ represents an orthogonal hierarchical subband transformation matrix. After receiving the decoded coefficients, the reconstructed signal can be obtained as

$$\mathbf{x}' = \mathbf{W}^{-1}\mathbf{y}' = \mathbf{W}^T\mathbf{y}'$$

The objective in a progressive transmission is to select the most important information to be transmitted first. The mean-squared-error (MSE) distortion measure is defined as

$$MSE = \frac{\| \mathbf{x} - \mathbf{x}' \|^2}{N} = \frac{1}{N}\sum_{n=0}^{N-1}(\mathbf{y}_n - \mathbf{y}'_n)^2$$

where N is the length of $\mathbf{x}$. The coefficients with larger

0-7803-3905-3/97/$10.00

magnitudes should be transmitted first since they have a higher information content.

B. Zerotrees of Wavelet Coefficients

After the DWT, the input signal is characterized by the wavelet coefficients obtained. The number of wavelet coefficients is exactly equal to the length of the original signal. In the run-length coding method, given a threshold ϵ, a wavelet coefficient w_i is defined as *insignificant* if $|w_i| \leq \epsilon$, otherwise the wavelet coefficient is deemed to be *significant.*

Every coefficient at a given scale can be related to a set of coefficients at the finer scales. The coefficient at the coarse scale is called the *parent*, and all coefficients corresponding to the same spatial location at the next finer scale are called *children*. For a given parent, the set of all coefficients at all finer scales corresponding to the same location are called *descendants*. Similarly, for a given child, the set of all coefficients at all coarser scales corresponding to the same location are called *ancestors* [3].

A key observation is that if a wavelet coefficient at a coarse scale is insignificant with respect to threshold ϵ, then the wavelet coefficients in the same spatial location at finer scales are most likely insignificant with respect to the threshold. In other words, those insignificant coefficients in the same spatial location at finer scales can be predicted by a insignificant coefficient at a coarse scale.

Given a threshold level ϵ, a coefficient w_i is defined to be an element of a *zerotree* for threshold ϵ if itself and all of its descendents are insignificant with respect to ϵ. An element of a zerotree for threshold ϵ is a *zerotree root* if it is not the descendant of a previously found zerotree root for threshold ϵ [3]. A zerotree root indicates that the insignificance of the coefficient at finer scales is completely predictable, i.e., if we know one coefficient is a zerotree root, then we can predict that all its descendants at the finer scales are zeros.

A typical zerotree of wavelet coefficients is shown in Fig. 1. If the zerotree root is known at the decoder, the zerotree can be generated automatically.

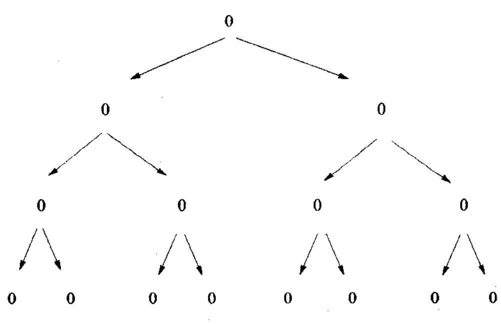

Fig. 1. A typical zerotree of wavelet coefficients.

C. The Significance Map

The scanning of the coefficients is performed in an order such that no children are scanned before their parents. All the positions in a given subband are scanned before the scan moves to the next subband. Such a scan pattern is shown in Fig. 2. For an N-scale transform, the scan begins at the lowest frequency subband, then moves to the scale with the second lowest frequency subband. The

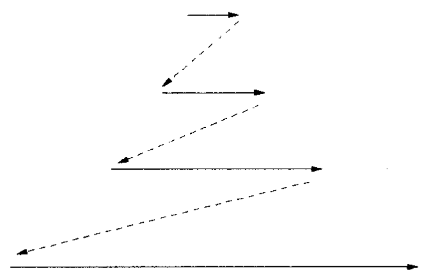

Fig. 2. Scan order for significance map (scale=3).

first threshold ϵ is chosen such that $\epsilon \leq max(|w_i|) < 2\epsilon$ and is usually a power of 2 so that ϵ is halved at the next scan. Although zerotrees usually dominate the coefficients, some coefficients may exist which are insignificant but have some significant descendants. These coefficients are called *isolated zeros.* Those insignificant coefficients in the finest scale that have no children are called simply *zeros* [3].

For a given ϵ, the scanning generates a significance map in which all the insignificant coefficients are replaced by zeros and the significant coefficients are replaced by their signs. The flow chart for a significance map is exactly the same as in Fig. 6 in [3]. The significance map can be efficiently represented in terms of a string of symbols where each symbol can assume one of four values: zerotree root (ZR), isolated zero (IZ), positive significant (POS), and negative significant (NEG).

D. Dominant List and Subordinate List

During encoding and decoding, two separate lists are obtained from the wavelet coefficients. The *dominant* list contains the coordinates of those insignificant coefficients that have been found in each scan. The scan is in such a way that subbands are ordered and within each subband, the coefficients are ordered [3]. Then, using the order of the subbands shown in Fig. 2, the *subordinate* list, which contains the refinement of magnitudes of the significant coefficients that have been found, is obtained. For each threshold, each list is scanned once and then the next scan is undertaken.

During the scanning of the coefficients in a dominant pass, each coefficient is encoded with a symbol from the list: ZR, IZ, POS, or NEG. The string of symbols is then encoded. Each time a coefficient is encoded as significant, its magnitude is appended to the subordinate list.

All the coefficients in the subordinate list are then scanned. For the refinement of the decoded magnitudes, an additional bit of precision is given. For each magnitude, this refinement can be encoded using a binary bit with a '1' indicating that the true value falls in the upper half of the old uncertainty interval and a '0' indicating that the large value falls in the lower half. After the subordinate pass, the magnitudes in the subordinate list are sorted in decreasing magnitude and the decoder can sort its magnitudes in the same decreasing order.

It is crucial that the ordering be known both at the encoder and decoder. At the decoder, the zerotrees generated are exactly the same as at the decoder. The dominant list gives the coordinates of the significant coefficients, and the subordinate list gives the best approximation to each coefficient. Once a coefficient is determined to be significant, its value is set to zero so that it does not prevent a zerotree occurrence in future dominant passes. The process is continued alternately between dominant passes and subordinate passes and the threshold is halved before each dominant passes. Each decoded symbol refines and reduces the width of the uncertainty interval in which the true value of the coefficient may occur. For minimum mean-square-error distortion, the centre in of the uncertainty interval is used to approximate the reconstructed coefficient.

The encoding can be stopped when some target bit rate is met, e.g., the bit budget is exhausted. The encoding can cease at any time when the distortion metric is met.

III. A SIMPLE EXAMPLE

In this section, a simple example is used to demonstrate how the embedded zerotree coding works. Consider a signal of 32 samples as shown in Fig. 3. After a 4-scale wavelet decomposition using a Daubechies filter, the tree-structured wavelet coefficients listed in Fig. 4 are obtained. Notice that one component in detail 1 is 47.

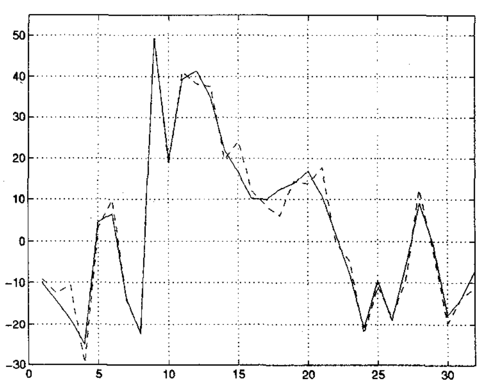

Fig. 3. A test signal (dashed) and its approximation (solid).

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

Fig. 4. Tree-structured wavelet coefficients of test signal.

During the first dominant pass, which uses a threshold $\epsilon = 32$, four significant coefficients are identified and the following results are obtained:

Significant components (1)	[63 34 49 47]
Significant signs (1)	[1 -1 1 1]
Significant coordinates (1)	[1 2 5 20]
Subordinate list (1)	[1 0 1 0]

The subordinate list is based on the intervals illustrated in Fig. 5. Only the threshold, dominant list, i.e., the list of insignificant coordinates, subordinate list, and the signs of the significant components need to be sent to the decoder. The encoder and decoder have the same order of scan. From the subordinate list and the signs of significant components, the decoder can decode the wavelet coefficients as [56 -40 56 40] using Fig. 5. From the dominant list, the decoder generates the same zerotrees as the encoder and puts these coefficients in the appropriate positions. Before the second scan, the decoded coefficients

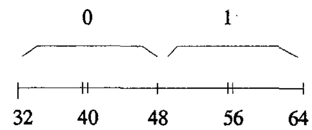

Fig. 5. Intervals used in the first scan.

are reordered according to decreasing magnitude. The encoder does the same to reorder significant signs, significant coordinates, and subordinate list as follows:

Reordered decoded coefficients	[56 56 40 40]
Reordered significant signs	[1 1 -1 1]
Reordered significant coordinates	[1 5 2 20]
Reordered subordinate list	[1 1 0 0]

Those significant components found in the first scan are treated as zeros for the second scan. During the second dominant pass, which uses a threshold $\epsilon = 16$, the following results are obtained:

Significant components (2)	[31 23]
Significant signs (2)	[-1 1]
Significant coordinates (2)	[3 4]

Notice that the second subordinate list is generated from the signal components whose coordinates are [1 5 2 20 3 4] as follows:

Subordinate list (2) [1 0 0 1 1 0]

using the intervals in Fig.6. Again, only the threshold, dominant list, subordinate list, and the signs of significant components are sent to the decoder. Upon receiving the second subordinate list, the decoder generates a *decode matrix*

$$\begin{bmatrix} 1 & 1 & 0 & 0 & * & * \\ 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

where the first row comes from the reordered first subordinate list and the second row is from the second subordinate list. By scanning the columns of this matrix, the decoder can determine in which intervals the decoded coefficients should fall and generates the decoded coefficients using Fig. 6. For example, the third column of the decode matrix has two zeros and hence the decoded value is 36. Thus the decoded coefficients can be determined as [60 52 -36 44 -28 20]. These steps continue until the target bit rate is exceed or the distortion metric is met.

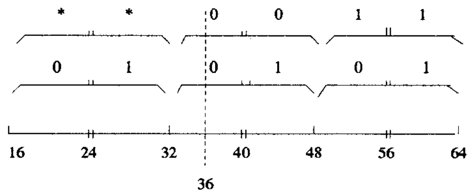

Fig. 6. Intervals used in the second scan.

The reconstructed signal after 3 scans is represented by the solid line in Fig. 4.

IV. SIMULATION RESULTS

The proposed 1-D algorithm was implemented with Matlab and was used to compress a large number of bathymetric waveforms. Fig. 7 (a) shows a typical 256-sample bathymetric waveform. The wavelet filters used were the MIT 9/7 filters proposed by Arias [7]. Eight scales were used in the DWT. The compressed signals using compression ratios 66, 35 and 21 are depicted in Figs. 7 (b), (c), and (d), respectively. The mean-square errors between the original and the compressed signals are given in Table I. For the purpose of depth estimation, the most important features of a waveform are the locations of the peaks. As can be observed from Fig. 8, the peaks in the compressed signal with ratio 35 are well preserved.

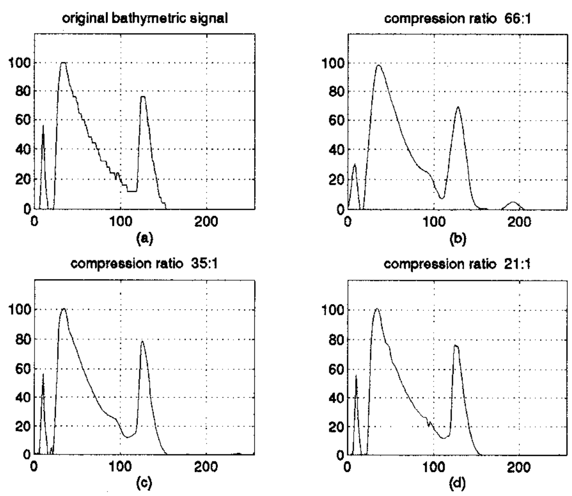

Fig. 7. The original signal and three approximations.

TABLE I

MEAN-SQUARE ERROR OF COMPRESSED SIGNALS

Compression Ratio	Mean-Square Error
66:1	0.4655
35:1	0.1059
21:1	0.0683

V. CONCLUSION

An embedded zerotree wavelet coding for 1-D signals has been described. The algorithm was illustrated by an example. Embedded coding was then applied to bathymetric waveforms. Simulation results indicate that the method is quite efficient and produces a fully embedded coding for 1-D signals.

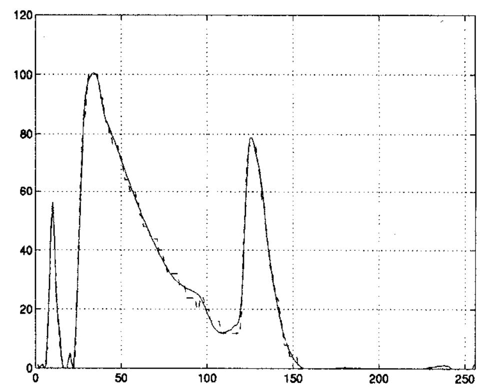

Figure 8: Comparison of the original and compressed waveform original (dashed line); compressed with ratio 35:1 (solid line).

REFERENCES

[1] N. S. Jayant and P. Noll, *Digital Coding of Waveforms,* Englewood Cliffs, NJ.: Prentice-Hall, 1984.

[2] M. Rabbani and P. W. Jones, *Digital Image Compression Techniques,* Bellingham, WA.: SPIE Optical Engineering Press ,1991.

[3] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. on Signal Processing,* vol. 41, pp. 3445-3462, December 1993.

[4] A. Said and W. M. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst., Video Technology* vol. 6, pp. 243-250, June 1996.

[5] B. Tang, A. Shen, A. Alwan, and G. Pottie, "A perceptually based embedded subband speech coder," *IEEE Trans. on Speech and Audio Processing,* vol. 5, pp. 131-139, March 1997.

[6] T. S. Huang, J. W. Stroming, Y. Kang, and R. Lopez, "Advanced in very low bit rate video coding in north America," *IEICE Trans. Communication,* vol. E79-B, pp. 1425-1432, March, 1996.

[7] G. Strang, "Creating and comparing wavelets," in *Numerical Analysis: A. R. Mitchell Birthday Volume,* G.A. Watson and D.F. Griffiths eds., World Scientific, 1996.

[8] G. Strang and T. Nguyen, *Wavelets and Filter Banks,* Wellesley, MA: Wellesley-Cambridge Press, 1996.