PROCESSING OF WAVELET TRANSFORM DATA FOR IMPROVED IMAGE COMPRESSION

Tanzeem Muzaffar and Tae-Sun Choi

Kwangju Institute of Science and Technology,
Kwangju, Republic of Korea

ABSTRACT

This paper presents a novel pre-processing technique of wavelet transform data to link all the significant coefficients together, in order to facilitate the image coding algorithms for increased compression. The proposed algorithm finds the isolated significant coefficients in the wavelet transformed data for current threshold value. All the insignificant coefficients in the wavelet tree that lies above that isolated significant coefficient (i.e its insignificant parents) are changed to significant coefficients and their location is saved, they are then treated just like other significant coefficients. Coding algorithms have been modified to accommodate the processed output. In the end, algorithm indicates converted coordinates that are used to change the value back to original during reconstruction. Noticeably high compression ratio is achieved for most of the images, when the proposed method is used with the modified image codecs.

1. INTRODUCTION

Image compression is of critical importance for efficient data storage and transmission, and has been a subject of study during the last two decades. Many state of the art image codecs[1,2,6] employ wavelet transform in their algorithms. One of its major advantages is that image energy is compacted into a small fraction of the transform coefficients and high compression can be achieved by coding these coefficients efficiently.

It has been observed that most of the wavelet transformed data lies in descending order within a subband tree at different levels. Coefficients at higher level subbands(parents) are generally of larger values compared to coefficients at lower levels(children) of the same orientation. This assumption is exploited and parent-child relationship between subbands is used to achieve high compression. Parent-child relationship among different levels of wavelet subbands is illustrated in fig. 1.

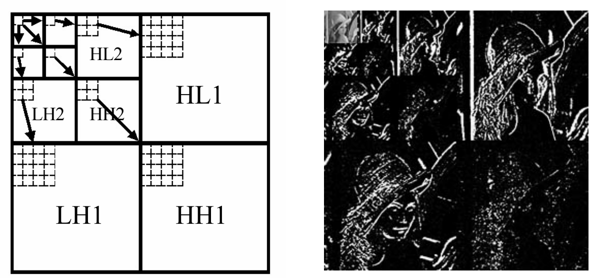


Fig. 1 Parent-child relationship among different levels

Although this assumption is valid in most of the cases, but some coefficients do not follow it. There are some children coefficients that are of larger value than their parents. This type of coefficient i.e. a significant child with insignificant parent is termed as an isolated significant. Extra processing is required to find the isolated significant coefficients, and thus more bits are required to code a given image.

This paper presents a new method to pre-process the wavelet transformed data prior to compression. The method combines with the modified image coders, and in result generates less bits to compress the same image. A simple technique is explained to revert the changed values back to its original state during reconstruction. Afterwards, the paper also proposes modifications to well-known wavelet image coders e.g. EZW, SPIHT for higher coding efficiency.

The organization of this paper is as follows: Section 2 describes the proposed processing algorithm of wavelet transform data. Modifications done in the well known coding algorithms for efficient compression of an image are also explained in detail in this section. Section 3 shows experimental results and discussion. Finally, the paper provides its concluding remarks in section 4.

0-7803-8554-3/04/$20.00 ©2004 IEEE.

2. PROPOSED ALGORITHM

Despite the assumption of descending values in a subband tree at same orientation, there are several isolated significant coefficients present in wavelet transform data of an image. In order to connect and link all the significant coefficients together, a new algorithm is proposed for processing of wavelet data. Isolated significant coefficients are first searched in the transformed data using current threshold value, and their insignificant parents ‘I’ are marked for conversion. The proposed method adds current threshold value to all the insignificant coefficients in the wavelet tree that lies above the isolated significant coefficient.

If the insignificant parent ‘I’ is selected for conversion, the proposed algorithm checks its initial value(sign) and changes its value accordingly. Current threshold value is added to the coefficient ‘I’ when it is positive, whereas threshold is subtracted in case of negative valued coefficients. The addition(or subtraction) of current threshold in the original value makes it significant for that threshold, and as a result all the intermediate coefficients ‘I’ with isolated significant child become significant. This method is repeated for all the threshold values, so that no more isolated significant coefficients exist in the transformed data. Fig 2 below shows conversion of insignificant parent of significant coefficient ‘I’ to significant symbol.

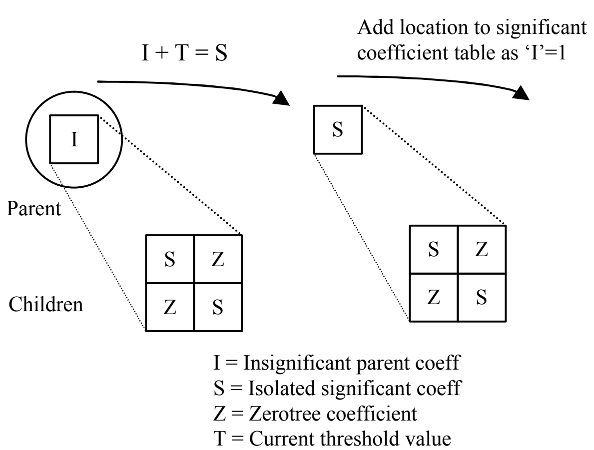


Fig.2 shows insignificant parents of significant coefficient ‘I’ converted to significant symbol

Significant	‘I’
x_a , y_a	0
x_a , y_b	0
$\mathbf{x_b}$ **,** $\mathbf{y_a}$	1
x_b , y_b	0
:	:
:	:
:	:
:	:
$\mathbf{x_n}$ **,** $\mathbf{y_n}$	1

1’s in column ‘I’ shows insignificant parent coeff converted to significant, with their location saved in sig.coeff table in order of occurrence

Fig.3 shows location of converted significant coordinates in significant coefficient table with ‘I’=1.

A table of significant coefficients is maintained along with their location(x,y) and order of occurrence to keep track of converted coefficients. Significant coefficients in lowest level (LH1, HL1, HH1) are not added to the table, since ‘I’ cannot occur in that level. In that table, insignificant parents converted to significant are marked as I=1 for their identification. These locations are later used to revert the values for proper reconstruction during decoding. At decoder, converted significant coefficients ‘I’ are identified using significant coefficient table, and their first nonzero MSB bit(threshold) is removed from the reconstructed data in order to change them back to their original value. This algorithm has overall no effect on reconstruction quality, whereas higher compression is achieved using modified algorithms. Fig.3 shows location of converted significant coordinates in significant coefficient table with I=1.

For example, lets suppose there are two insignificant coefficients (I=5, I=-9) in wavelet block for threshold=32. A significant coefficient must have a value between 32 and 63. Therefore, I=5 is changed to significant with positive sign 5+32=37, and I= -9 is changed to significant with negative sign -9-32= -41. Upon reconstruction, first nonzero msb bit of ‘I’ marked significant coefficients is removed and the resultant value obtained is the original value, i.e. 100101(37) => 101(5). Similarly for negative values, neglecting sign bit and doing the same operation gives -101001(-41) => -1001(-9).

Different wavelet based coding schemes [1,2,6] have been published in literature to compress image data. Embedded zerotree coding (EZW) by Shapiro[1] was the first to apply wavelet compression to images. Since then, many improvements have been proposed in wavelet compression techniques to make it more efficient [2,3,4,5].

SPIHT method [2] was proposed later by Said and Pearlman with improved performance and faster processing. In this paper, we present modifications to well known coding algorithms (EZW and SPIHT) to accommodate the proposed processing of wavelet transformed data, and achieve high compression. They are explained below in detail.

2.1 Embedded Zerotree Wavelet Coder(EZW):

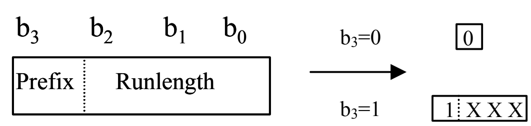
In original embedded zerotree wavelet coder (EZW) [1], four symbols are used to represent the significant and non-significant coefficients in a wavelet-transformed domain. Each of this symbol is coded with two bits each irrespective of their number of occurrence. A modified EZW image coder is proposed for the processed wavelet data, with reduced number of symbols for better compression efficiency. Since all the significant coefficients are linked together within a subband tree after processing with no isolated significant coefficient present, therefore there is no need for isolated zero symbol during encoding. Thus, the modified coder eliminates symbol for isolated-zero coefficients. In this way, we can effectively use only two symbols for coding (zerotree and significant). Each symbol uses one bit indicating either zerotree or significance of coefficient (0=zerotree, 1=significant), along with one bit of sign information for significant coefficients only. Bitplane zerotree encoding of the generated significant map is done next in z-scan fashion starting from highest-level bitplane and then moves on to next lower levels, with each bitplane having separate pass. In the last pass, the algorithm sends stored information of isolated-zero locations to differentiate them from real significant coefficients.

2.2 Set Partitioning in Hierarchical Tree(SPIHT):

SPIHT wavelet compression method is the refined form of zerotree coding with improved performance. It has been modified to use with the proposed processing of wavelet transformed data, for better coding efficiency. A list is maintained of significant pixels(LSP), insignificant pixels(LIP), and insignificant sets(LIS). The algorithm checks significance with current threshold value in the insignificant pixels list(LIP), if there is a significant coefficient(i.e. greater than current threshold) it is moved to significant list(LSP) and outputs '1' else output '0'. Then LIS is scanned to check significance of 4 descendents of those coefficients that are significant. If it finds any of the children as significant, it outputs '1', moves it to LSP and split the set further into 4 (children) sets. If all the children are insignificant, then output a '0' and no operation is done. The modified algorithm no more checks all the descendents D(i,j) for significance, or its children-onward descendants L(i,j) because the absence of isolated significant coefficients from processed wavelet data eliminates the need to check all D(i,j) and/or L(i,j). It only need to check its 4 children O(i,j) since all the significant coefficients within a subband tree are linked together in the processed wavelet data. In the end, locations of converted coefficients 'I' are sent to decoder for proper reconstruction.

2.3 Sending 'I' Locations:

To send locations of converted coefficients 'I' in the end, the algorithm uses difference of data method. Starting with first occurrence of I=1 in significant coefficient table, its order of occurrence is sent to decoder. Next occurrence is subtracted from current and their difference is sent. Because of unequal distribution of 'I' in the transformed image, the modified coding is suggested. For as long as difference is greater than 8, send a '0' and reduce difference by 8, and when the resultant becomes less than or equal to 8, code it in 3 bits (1~8) with '1' as a prefix. With the help of identical significant table maintained at decoder, this technique is used to identify all 'I' symbols among significant symbols, and original data can be recovered by simply eliminating first nonzero MSB bit (threshold value at the time when 'I' was made significant) of these coefficients. Construction of symbol for coding of 'I' positions during significance map encoding is shown below:



Here, prefix b_3=0 means no 'I' in the next 8 values, so increment zeroes by 8 and no extra bit is sent. Otherwise bit b_3=1 indicates 'I' within the next 8 values, and is followed by bits $b_2b_1b_0$ that shows remaining difference of location of 'I'.

3. EXPERIMENTAL RESULTS

The proposed algorithm was implemented in software and experiments were performed to verify the results. Several 512x512 8-bit gray-scale images were used for experimental purposes. The input image was first decomposed into five-level wavelet transform. Proposed processing of transformed output was done before compression to link all significant coefficients in a wavelet tree. Modified wavelet coding methods were applied next on the resultant data. The algorithm used z-scan coding approach to transmit significant pixels starting from highest level and then moved on to next lower level in z-fashion. In the end, locations of converted coefficients 'I' were sent to decoder for proper reconstruction. Finally, reconstruction was done at receiver to decode the coded bitstream data.

When this method was applied to 512x512 gray-scale images, higher compression ratio was achieved for most of the images compared to original algorithms, for similar reconstructed image quality. Comparative results of compressed data size of the proposed EZW method with basic EZW algorithm after first 6 passes are shown in Table 1, whereas results of modified SPIHT compression are compared with original SPIHT and stated in Table 2. These results of table 1 and 2 include extra bytes transmitted to indicate locations of modified coefficients 'I' to decoder. Generally the coefficients 'I' generated for 512x512 images under the given environment are in the range of few 100's. But if sometimes a large number of 'I' are generated for an image, they are truncated at a certain limit during transmission to keep the bitrate under control. This truncation overall produces only marginal effect on output quality, as in this case most of the locations of high-valued 'I' coefficients are sent (first in the order of occurrence) and only few low valued 'I' are neglected. Results show that the proposed processing method along with the modified codecs further increased the compression ratio of an image. Fig. 4 shows reconstructed gray-scale images of Lena and Tiffany after compression using the proposed method with modified EZW and SPIHT codec respectively.

Image (512x512)	*EZW (bytes)*	*Proposed (bytes)*	*SNR (dB)*	*Gain (% age)*
Pepper	3679	2441	24.0	33.6 %
Lena	3216	2141	28.4	33.4 %
Splash	2818	1824	26.8	35.3 %
Goldhill	3147	2095	27.3	33.4 %

Table 1. Comparison of results of proposed algorithm with EZW algorithm for first 6 passes

Image (512x512)	*SPIHT (bytes)*	*Proposed (bytes)*	*SNR (dB)*	*Gain (% age)*
Aerial	2540	2492	21.7	1.9 %
Airplane	3844	3797	28.4	1.2 %
Lake	5206	5163	26.5	0.8 %
Tiffany	3311	3294	27.7	0.5 %

Table 2. Comparison of results of proposed algorithm with SPIHT algorithm for first 6 passes

(a) Lena(28.4 dB) EZW

(b) Tiffany(27.7 dB) SPIHT

Fig. 4 Reconstructed 512x512 images using the proposed method (with modified EZW and modified SPIHT).

4. CONCLUSIONS

A new pre-processing method has been proposed to link all the significant coefficients together in the wavelet transform data, to facilitate the image coding algorithms for increased compression. Coding algorithms have been modified to accommodate the processed output for better results. In this method, a little compromise has been made on the property of embeddedness to achieve its goal. High compression ratio is achieved for most of the images, when the proposed method is used with the modified image codecs.

ACKNOWLEDGEMENT

This work was supported by the Korea Research Foundation Grant (KRF-2003-041-D20470).

REFERENCES

[1] M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients", *IEEE Transactions on Signal Processing*, pp 3445-3462, vol 41, Dec 1993.

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

[3] J. Wu, C. Oliver, C. Chatellier, "Embedded zerotree runlength wavelet image coding" *ICIP Proceedings,* pp 784-787, Oct 2001,

[4] C. Hsieh, Y. Chen, Y. Wu, F. Kuo, "Embedded image compression using the important oriented tree", *ISMSE Proceedings,* pp 432-436, Dec 2000.

[5] T. Muzaffar, T. Choi, "Simplified Wavelet based image compression using fixed length residual value", *IEICE Transaction on Info & Systems*, pp 1828-1831, Dec 2001.

[6] D. Taubman, "High performance scalable image compression with EBCOT", *IEEE Transaction on Image Processing,* pp 1158-1170, vol 7, July 2000.