PAPER

Flexible Zerotree Coding of Wavelet Coefficients

Sanghyun JOO†, **Hisakazu KIKUCHI**†, **Shigenobu SASAKI**†, ***and*** **Jaeho SHIN**††, ***Members***

SUMMARY We introduce an extended EZW coder that uses flexible zerotree coding of wavelet coefficients. A flexible parent-child relationship is defined so as to exploit spatial dependencies within a subband as well as hierarchical dependencies among multi-scale subbands. The new relationship is based on a particular statistics that a large coefficient is more likely to have large coefficients in its neighborhood in terms of space and scale. In the flexible relationship, a parent coefficient in a subband relates to four child coefficients in the next finer subband in the same orientation. If each of the children is larger than a given threshold, the parent extends its parentship to the neighbors close to its conventional children. A probing bit is introduced to indicate whether a significant parent has significant children to be scanned. This enables us to avoid excessive scan of insignificant coefficients. Also, produced symbols are re-symbolized into simple variable-length binary codes to remove some redundancy according to a pre-defined rule. As a result, the wavelet coefficients can be described with a small number of binary symbols. This binary symbol stream gives a competitive performance without an additional entropy coding and thus a fast encoding/decoding is possible. Moreover, the binary symbols can be more compressed by an adaptive arithmetic coding. Our experimental results are given in both binary-coded mode and arithmetic-coded mode. Also, these results are compared with those of the EZW coder.
key words: *image compression, wavelet transform, zerotree coding*

1. Introduction

Recently, Shapiro developed the EZW coder [1], which utilizes dependencies among subbands decomposed by wavelets [2]–[6]. This coder outperforms today's JPEG standard, ranging from low bit-rates to high bit-rates. It can be also applied for lossless coding by using integer wavelet transforms [7]–[12]. In particular, the properties of progressive transmission and encoding/decoding ceasing option can be very important, depending on the characteristics of available channels and application scenarios.

Since the EZW coder was published, there have been many developments in the field of image compression [11]–[26]. Some of them have been applied to three dimensional images [13]–[15] as well as even to medical images that usually require a high bit-rate coding [14], [15]. Some coders [16]–[18] offer improved results by the modification of the EZW coder.

In the EZW coder, zerotrees are used to describe many wavelet coefficients that are not so important to reconstruct an image. The insignificant coefficients can be represented as a few zerotree roots, and thus significant coefficients are efficiently described. The zerotree is structured in such a way that a parent coefficient in a subband connects to four child coefficients in the next finer subband. This tree structure is suitable to exploit the hierarchical dependencies among subbands.

Wavelet coefficients are described by four symbols of zerotree root, isolated zero, positive and negative in EZW. Symbols produced by the description are entropy-coded to remove some redundancies that exist among them. The efficiency of this entropy coding can be more improved, if the spatial and hierarchical dependencies are considered. That is, the produced symbols can be entropy-coded as a higher-order Markov source or context-based models [27]–[29]. In general, the higher the order is, the lower the entropy is. On the other hand, the memory and execution time increase exponentially as the order.

In this paper, a new tree structure is introduced to simultaneously exploit spatial dependencies within a subband as well as hierarchical dependencies among subbands. In addition, two effective techniques, probing bits and re-symbolization, are proposed to remove some redundancies among the symbols produced by the new treeing. In the new tree structure, a parent coefficient in a subband basically relates to four child coefficients in the next finer subband in the same orientation, as is the same in the EZW. Then, the parent flexibly extends its parentship onto at most five adjacent coefficients that exist at the right and bottom of the basic four children. The extension depends on the distribution pattern of significant coefficients.

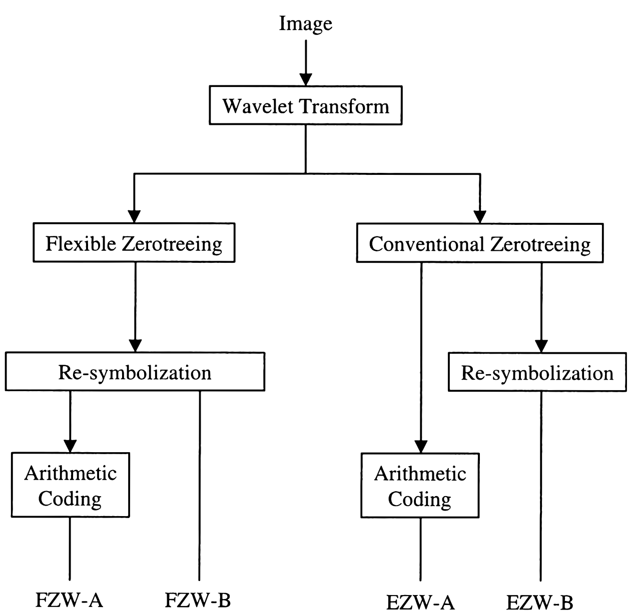
By using this flexible treeing, insignificant coefficients are more likely to be related with zerotree roots and significant coefficients are more likely to be related with significant parents. On the contrary, the total number of produced symbols can increase by as many as the extended parent-child relationship would have. A probing bit is introduced to reduce the number of symbols to be produced. It indicates whether a parent has significant children to be scanned. Also, since

Manuscript received September 3, 1998.
Manuscript revised January 11, 1999.
†The authors are with the Department of Electrical Engineering, Niigata University, Niigata-shi, 950–2181 Japan.
††The author is with the Department of Electronic Engineering, Dongguk University, 3–26 Phildong, Joonggu, Seoul 100–715, Korea.

all subbands do not always require the same kinds of symbol alphabets, produced symbols for parents and children are re-symbolized into binary codes according to a pre-defined procedure.

Due to those efforts, the proposed coder outputs a binary symbol stream, and the stream gives a good performance without any additional entropy coding such as adaptive arithmetic coding. Moreover, the performance can be improved, when the stream is entropy-coded by an adaptive arithmetic coding. Our coder is compared with the EZW coder in two modes: a binary-coded mode and an arithmetic-coded mode. For comparison, four coders are prepared in Fig. 1, where produced symbols can be fed to an adaptive arithmetic coder as a *zero*th order Markov source (memoryless source).

This paper is organized as follows. The EZW coder is reviewed in Sect. 2. Section 3 introduces a new coder and shows how to produce a binary symbol stream. The probing bit and the re-symbolization process are explained in detail. In Sect. 4, the proposed FZW coders are compared with the EZW coder in three ways: FZW-B to EZW-B, FZW-A to EZW-A, and FZW-B to EZW-A. (For these abbreviations, refer to Fig. 1.) Conclusions follow in Sect. 5 and several points are suggested for further improvements.



FZW: Flexible zerotree coding of wavelet coefficients
FZW-A: Arithmetic-coded FZW coder
FZW-B: Binary-coded FZW coder
EZW-A: Arithmetic-coded EZW coder
EZW-B: Binary-coded EZW coder

Fig. 1 The FZW and EZW coders.

2. Embedded Image Coding Using Zerotrees of Wavelet Coefficients

The EZW coder encodes images in an "embedded" fashion from their dyadic wavelet representations. The goal of the embedded coding is to generate a single encoded bit-stream that permits to be truncated to achieve any desired bit-rate, while giving the best possible reproduction at that rate. This coder encodes wavelet expansion coefficients in order of importance with respect to a sequence of thresholds. The initial threshold is set as a maximum power of two smaller than the largest coefficient. The threshold value is halved as the significance map has been generated at each bit plane. For every threshold, two passes are performed: a dominant pass and a subordinate pass. All significant coefficients with respect to a given threshold are found in the corresponding dominant pass, where four symbols are used for signaling the dominant pass to the decoder.

In this coder, trees are structured according to a rule such that a parent coefficient in a subband relates to four child coefficients in the next finer subband. Note that a parent coefficient only in the DC subband relates to three child coefficients in the coarsest three AC subbands (See Fig. 2). A zerotree root (ZTR) symbol is generated for a coefficient such that it is insignificant and has no significant descendants. An isolated zero (IZ) symbol is generated, if a coefficient itself is insignificant but it has significant descendants. The other two symbols, POS and NEG, are used for a significant coefficient to describe its sign. All coefficients are scanned in such a way that no children are scanned before their parents. Owing to the scanning order, a few ZTR sym-

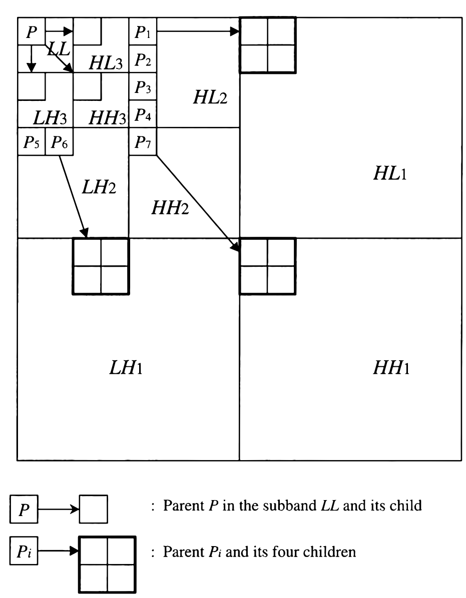


Fig. 2 Parent-child relationship for 3-scale decomposition.

bols can describe many insignificant coefficients.

Significant coefficients found in previous dominant passes are refined in the subsequent subordinate pass. A subordinate pass makes the lower or higher decisions on the approximation value with respect to a given threshold to minimize errors between the reconstructed and original coefficients. That is, a coefficient in the upper half of the uncertainty interval is coded with the symbol UPPER, while a coefficient in the lower half is coded with the symbol LOWER. By reading the subordinate symbols corresponding to significant coefficients and knowing the threshold, the decoder can determine the interval and can approximately reconstruct the significant coefficients. Therefore, from the decoder's viewpoint, the rough estimates of significant coefficients are more refined as more subordinate passes are developed.

Four symbols are needed for the dominant passes, except for the finest three AC subbands that have no IZ symbols. Two symbols are needed for subordinate passes. Those symbols are output as a symbol stream and they are entropy-coded by an adaptive arithmetic coder [30].

3. Flexible Zerotree Coding of Wavelet Coefficients (FZW)

The parent-child relationship used in EZW is drawn in Fig. 2 for three-scale wavelet decomposition. For instance, each parent coefficient in subband HL_2 relates to four child coefficients at the same spatial locations in subband HL_1. This relationship is modified to give flexibility between a parent and its children.

3.1 Extension in the Parent-Child Relationship

The basic four children mentioned before are referred to as leading children. To explain the flexible parent-child relationship, a part of Fig. 2 is enlarged in Fig. 3. Unlike EZW, P_1 relates to the leading children, C_{11}, C_{21}, C_{31}, and C_{41}, and flexibly extends its parentship onto at most five adjacent coefficients, C_{51}, C_{61}, C_{71}, C_{81}, and C_{91}, that exist at the right and bottom of the leading children. Note that C_{ij} denotes the ith child for the parent P_j.

Every nine children of a parent, say P_1, are classified into four groups of G_{11}, G_{21}, G_{31}, and G_{41} as shown in the figure, where G_{ij} means the ith group for the parent P_j. First, P_1 connects to G_{11} that consists of the leading children. Then, the connection is flexibly extended to G_{21}, G_{31}, and G_{41} depending on the significance of C_{21}, C_{31}, and C_{41}. For example, assume that C_{21} and C_{41} are significant. The children of P_1 are defined as G_{11} and then the parent-child relationship is extended to G_{21} and G_{41}, because some of the leading children, C_{21} and C_{41}, are significant. Hence, P_1 has eight children except C_{71}.

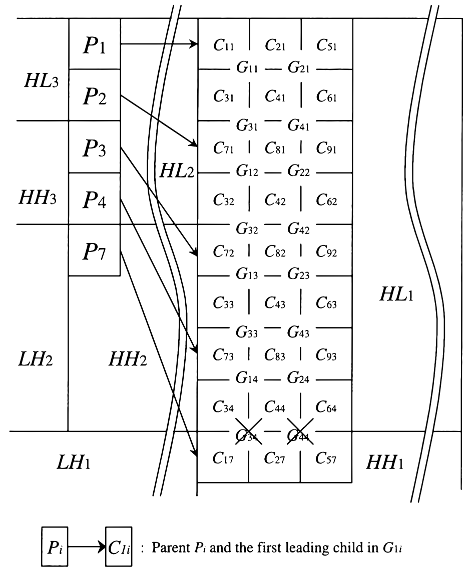


Fig. 3 Parents, children, and groups in the flexible relationship. A group consists of four children and its name is written in the center of them.

In this way, every parent can take care of at most nine children, and actual parent-child relationship varies in a flexible manner depending on the distribution pattern of significant coefficients. P_2 also has four child groups of G_{12}, G_{22}, G_{32}, and G_{42}. Three children, C_{71}, C_{81}, and C_{91}, of P_1 can be read as the top three children, C_{12}, C_{22}, and C_{52}, of P_2. Note that P_4 does not define G_{34} and G_{44} as its child groups, because G_{34} and G_{44} cover the children, C_{17}, C_{27}, and C_{57}, in another different subband.

Every parent in all subbands except LL, HL_3, LH_3, and HH_3, flexibly relates to its children. A parent in subband LL relates to three children in the coarsest AC subbands, HL_3, LH_3, and HH_3 (See Fig. 2).

3.2 A Significance Map

Once a group has been activated, children in that group are scanned until they are found as significant. This scanning is processed before a dominant pass and is referred to as a pre-dominant pass. Unlike EZW, significant coefficients with respect to a given threshold are searched through the two passes: pre-dominant pass and dominant pass.

A particular significance map is prepared for these passes. The significance map has the same dimension as an original image, and all significant coefficients are found by referring this map. This map is filled with symbols that represent the significance information about all pixels. Also, pixels that have children include probing bits information to indicate scanning of their children.

Significant pixels to be found are described as POS and NEG. The significant pixels found in the earlier passes are described as SE to differentiate from pixels to be found in current passes. In the significance map, following four symbols are used.

SE: Significant coefficient already found in an earlier pre-dominant or dominant pass. This symbol is not transmitted.

SC: Significant coefficient to be newly found in the current pre-dominant or dominant pass. This symbol is changed with POS or NEG, when it is transmitted.

ZTR: Insignificant coefficient that has no significant descendants

IZ: Insignificant coefficient that has significant descendants

Our purpose is to find SC symbols from the significance map, and the significance map for n-scale decomposition is made by the following procedure.

Step 1. Initialize all pixels with ZTR and set all probing bits for pixels that have children to OFF.

Step 2. Mark significant pixels found in the earlier passes with SE.

Step 3. Mark significant pixels to be found in the current pre-dominant and dominant passes with SC.

Step 4. Set the scale m to 2.

Step 5. For each parent P_e at (i, j) in subbands HL_m, LH_m, and HH_m,

- Inspect the significance of four children $(2i, 2j)$, $(2i+1, 2j)$, $(2i, 2j+1)$, and $(2i+1, 2j+1)$ in G_{1e}.
- If one of them is either SC or IZ and if it is not included in any other active groups, then
  - switch G_{1e} and the probing bit of P_e to ON .
  - if the pixel (i, j) has been marked with ZTR, replace the pixel by IZ.
  - if each pixel of $(2i+1, 2j)$, $(2i, 2j+1)$, and $(2i+1, 2j+1)$ is SE or SC, switch the corresponding group among G_{2e}, G_{3e}, and G_{4e} to ON.
- Else if the child group G_{1e} has been already ON, then
  - if each pixel of $(2i+1, 2j)$, $(2i, 2j+1)$, and $(2i+1, 2j+1)$ is SE or SC, switch the corresponding group among G_{2e}, G_{3e}, and G_{4e} to ON.
- Else do nothing.

Step 6. If m is smaller than n, increase m by 1 and go to Step 5.

Step 7. For each parent (i, j) in subband LL,

- If one of the pixels, $(i + p, j)$, $(i, j + q)$, and $(i + p, j + q)$, is SC or IZ, then
  - if the pixel (i, j) has been marked with ZTR, replace the pixel by IZ, where p and q represent the numbers of horizontal and vertical pixels in the LL subband.

Step 8. Stop.

Some coefficients are found as significant by scanning active groups in the pre-dominant pass, where only the pixels that has been not yet marked with SE are scanned. Since these significant coefficients are described through this pass, many zerotree roots in the subsequent dominant pass can be produced from coarser subbands. The significant coefficients that have been not found in the pre-dominant pass are described in the subsequent dominant pass. The bit-saving process described in the previous sub-section can be performed in this dominant pass.

3.3 Re-Symbolization Process

The pre-dominant and dominant passes are executed with the help of the significance map. SC symbol is changed into POS or NEG to indicate the sign of a coefficient and SE symbol is neglected. Therefore, four symbols of POS, NEG, IZ, and ZTR are generated in these two passes. All subbands do not always need the same number of symbol alphabets. Note that the DC subband, LL, needs no NEG symbols, and the finest AC subbands, HL_1, LH_1, and HH_1, need no IZ symbols. In particular, the area of the finest subbands is three quarters of the total area. It is wasteful to use 2-bit symbols throughout the entire set of all subbands.

Therefore, all symbols produced from pre-dominant and dominant passes are re-symbolized according to the definition given in Table 1. In the table, one bit is assigned for differentiating ZTR symbol from the other symbols. One more bit is assigned for differentiating IZ symbol from SC symbol, but this augmented bit is not used for the finest subbands. SC symbol is encoded into POS or NEG by augmenting another bit, and again this augmentation does not apply for the DC subband.

3.4 One-Bit Probe and Bit-Saving Operation

In FZW, the probing bit is generated to avoid redun-

Table 1 Zerotree symbols and binary symbols in the re-symbolization of a significance map.

Subbands	ZTR	IZ	SC		SE
			POS	NEG	
LL	0	10	11	N/A	Don't Code
HL_1, LH_1, HH_1	0	N/A	10	11	Don't Code
Others	0	10	110	111	Don't Code

dant scans for insignificant descendants. It has two states between ON and OFF. The role of probing bits is to indicate whether a parent has a significant child of SC and IZ that should be scanned because of its significance.

Then how and when are the probing bits generated? Assume that the encoder is now going to scan a particular subband with a significance map and probing bits information. That means all coarser subbands have been already coded. All parents in the parent subband have been already coded with SE, SC, ZTR, and IZ, and have their own probing bits. When one of the parents is SE or SC and its leading child group is inactive, its probing bit is generated to inform whether the leading child group is going to be scanned. If the probing bit is ON, the leading child group will be scanned and then a flexible extension will be followed. Otherwise, do nothing for the children. If the parent is IZ, the probing bit must be ON and the leading child group must be inactive. In this case, a probing bit is not generated, because it is predictable. Then the leading group scan and the flexible extension are followed. In the case of ZTR, the probing bit is OFF. This is also predictable and no children are scanned.

Once the children scan for the parent has been finished, the process moves to the next parent. All parents have their own probing bits in the entire process of encoding. Among the probing bits, a part of them are output: only the probing bits belonging to SE and SC parents that have inactive leading child groups are output to inform the scan of children.

This probing bit works well over a wide range of bit rates: low, middle, and high bit rates. There has been such kind of trials to avoid redundant scans in ZTE [19]. In ZTE, the use of zerotrees has been modified by defining a new set of symbols designed for very low bit rate coding of video. Since many insignificant coefficients are likely to appear particularly in very low bit rate coding, an efficient handling of them is strongly required. The new symbol set defined in ZTE reflects well this requirement and is composed of zerotree root (ZR), valued zerotree root (VZR), and value (V).

- VZR is a symbol for a significant coefficient that has no significant descendants. In FZW, VZR can be considered as a significant symbol of POS and NEG when the probing bit of its significant symbol is OFF.
- V is a symbol for a coefficient that has significant descendants. In FZW, V can be considered as a symbol among IZ, POS, and NEG when the probing bit of the symbol is ON.

It can be said that the probing bit in FZW plays a role in differentiating VZR from V. However, two schemes have different sets of symbols and different use. Moreover, ZTE finds significant coefficients through one-pass, and then scalar quantization follows. In contrast, FZW finds significant bits rather than significant coefficients through multi-pass (pre-dominant passes and dominant passes), and then successive approximation quantization follows in subordinate passes.

A significant parent P_s connects with the leading children in G_{1s} by outputting a probing bit. Since a significant parent usually has many children to be scanned, this probe enables us to avoid excessive scan of insignificant children. After the first group is activated, the other groups, G_{2s}, G_{3s}, and G_{4s}, can be activated depending on the significance of C_{2s}, C_{3s}, and C_{4s} among the leading children.

Assume that the probing bit of a parent coefficient is ON. It means that there exists at least one child to be newly found as significant among the leading children. Since it implies that at least one child is not coded as ZTR, we can save the coding budget for that child. This bit-saving is applied to significant parents and those parents that have been coded with IZ.

3.5 FZW Coder

The algorithm of FZW is summarized as follows.

Algorithm of the FZW coder

Step 1. Decompose an image into a collection of multiscale subbands by using a wavelet transform.
Step 2. Find the maximum coefficient among all coefficients.
Step 3. Decide an initial threshold from the maximum coefficient.
Step 4. Make a significance map with respect to a given threshold.
Step 5. Execute a pre-dominant pass.
Step 6. Execute a dominant pass.
Step 7. Execute a subordinate pass.
Step 8. Divide the threshold by two.
Step 9. If the threshold is smaller than 1, then stop. Else go to Step 4.

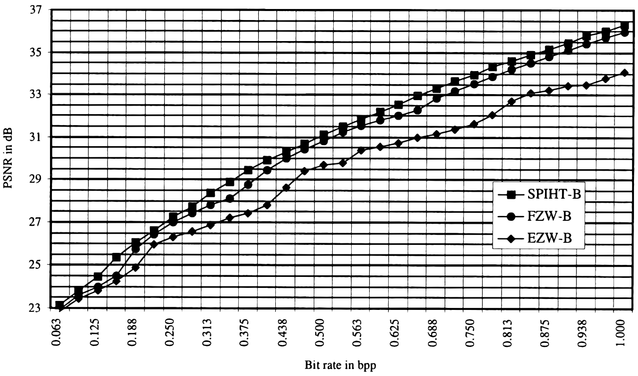
This encoding algorithm can be terminated at any time, and the decoder can reproduce an image at any bit rate allowed by the extent of the received information.

4. Experimental Results and Discussions

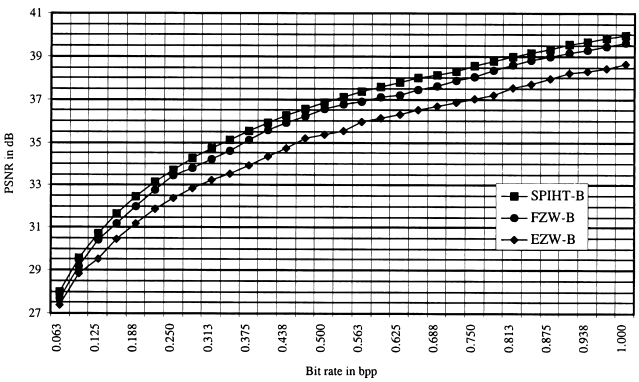
To compare the performance between the FZW and EZW coders, we have simulated on two test images of Lena and Barbara (512×512 in grey scale). All simulations have been performed with a 6-scale biorthogonal wavelet transform by using a 9/7-tap filter bank [31]. (See also Table 2) A reflection extension has been implemented at the image border. To reproduce an image from received binary symbols, the output bit stream includes seven bytes of header information: four bytes for

Table 2 9/7-tap filter bank coefficients.

n	Analysis	Synthesis
0	0.852699	0.788486
1	0.377402	0.418092
2	−0.110624	−0.040689
3	−0.023849	−0.064539
4	0.037828	



(a) Barbara



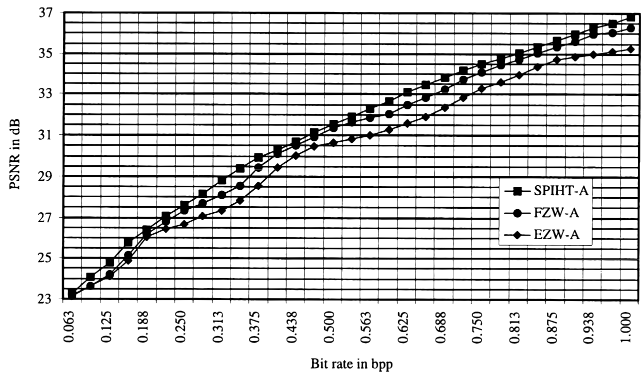
(b) Lena

Fig. 4 Performance comparisons among FZW-B, EZW-B, and SPIHT-B. Compressed size varies from 2,048 to 32,768 bytes.

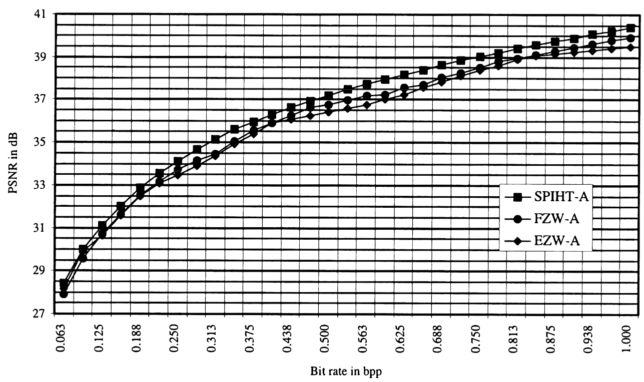
the horizontal and vertical dimensions of the image and one byte each for the filter bank, the level of the wavelet decomposition, and the initial threshold, respectively.

First, we compared the performances of FZW-B and EZW-B. Both coders produce binary symbol streams through the re-symbolization procedure, and they have been compared in terms of the length of bit streams. In FZW-B, one probing bit for a significant parent is output to indicate whether the parent needs to connect with its children, before they are scanned. Although the probing bit enables us to save 1-bit budget for a child that is not ZTR, the bit-saving operation has not been implemented in our simulations. The bit-saving can be combined to obtain better performances, but the programming will be more complicated.

As can be seen in Fig. 4, FZW-B shows remarkable performances comparing with EZW-B. FZW-B owes its preformance to the two points as follows.



(a) Barbara

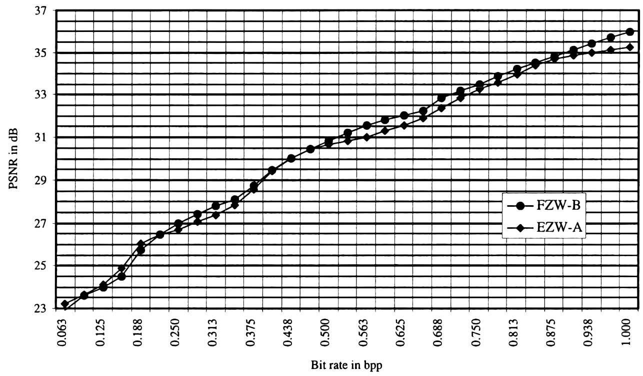


(b) Lena

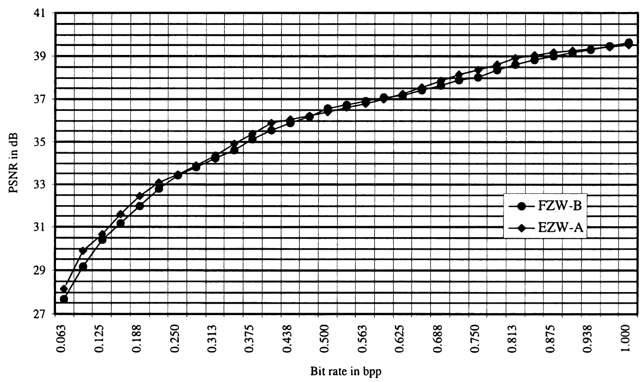
Fig. 5 Performance comparisons among FZW-A, EZW-A, and SPIHT-A. Compressed size varies from 2,048 to 32,768 bytes.

- One-bit probe in conjunction with the significance map avoid excessively many scans for insignificant coefficients and ease the task for efficient extension in the flexible treeing.
- Flexible parent-child relationship allows us to find efficient description in which significant children can be connected with fewer parents and thus many zerotree roots are produced from coarser subbands.

In the next experiments, entropy coding is applied to get better performances. In FZW-A, the symbol streams produced by FZW-B are coded by an arithmetic coding. On the other hand, EZW-A excludes the re-symbolization process to keep the original EZW algorithm as it is. Entropy coding has been performed as *zero*th order Markov sources with a maximum frequency count of 256 by Jones' adaptive arithmetic coder [32]. Although we use the *zero*th order entropy coding in this experiment, a higher-order entropy coding [28], [29] will offer better performances. The performance curves are given in Fig. 5. The performance differences between FZW-A and EZW-A are not as large as those between FZW-B and EZW-B. This means that the frequencies of symbols produced by FZW-A are strongly affected by the probing bits.



(a) Barbara



(b) Lena

Fig. 6 Performance comparisons between FZW-B and EZW-A. Compressed size varies from 2,048 to 32,768 bytes.

Our main purpose is to show some effectiveness of our proposed techniques: flexible treeing, a probing bit, and re-symbolization. They are applied to the EZW coder that is referred to as the most well-known zerotree coder. Our improvements in performance can be evaluated as effectiveness of the proposed techniques. Just as the reference purpose for broad readership, we give the plots for SPIHT [16] in Figs. 4 and 5. As well known for its good performances in the literature, SPIHT outperforms FZW in both arithmetic coding and binary modes. If the proposed techniques are adjusted to the SPIHT coder with adequate modifications, they may cause different result.

It is also interesting to compare the performances between FZW-B and EZW-A. Figure 6 gives the comparisons between FZW-B and EZW-A. While FZW-B shows better performances for the Barbara image, EZW-A offers better performances for the Lena image.

Let us consider the different results. When the images are reconstructed only with the coefficients that are greater than 8 in magnitude, the reproduced images are of 39.59 dB and 39.16 dB in PSNR for Barbara and Lena, respectively. This suggests that a reasonable image can be obtained at thresholds greater than 8. Table 3 shows the numbers of coefficients to be found as significant with respect to several thresholds. From

Table 3 The number of significant coefficients with respect to a given threshold in different subbands.

Threshold	N (LL)	N (6)	N (5)	$N(4)$	$N(3)$	$N(2)$	$N(1)$
8192	27	0	0	0	0	0	0
4096	35	0	0	0	0	0	0
2048	2	1	0	0	0	0	0
1024	0	32	5	0	0	0	0
512	0	47	69	17	0	0	0
256	0	53	139	134	60	0	0
128	0	30	170	336	355	296	86
64	0	10	144	489	858	1838	2528
32	0	6	103	569	1466	3903	5197
16	0	8	68	529	1948	5370	9244
8	0	3	36	400	2126	6710	15427
4	0	0	17	289	2041	7680	26365
2	0	1	7	155	1489	8708	41364
1	0	0	2	78	954	6736	41011

(a) Barbara

Threshold	N (LL)	N (6)	N (5)	$N(4)$	$N(3)$	$N(2)$	$N(1)$
8192	26	0	0	0	0	0	0
4096	37	0	0	0	0	0	0
2048	1	5	0	0	0	0	0
1024	0	25	9	0	0	0	0
512	0	44	74	16	0	0	0
256	0	36	117	121	29	0	0
128	0	35	117	334	331	42	0
64	0	17	137	416	794	584	4
32	0	14	111	503	1230	1808	499
16	0	8	103	469	1489	3397	2631
8	0	3	50	439	1848	5478	8470
4	0	2	24	356	2254	9651	28305
2	0	3	12	202	1950	11247	51940
1	0	0	10	102	1139	7952	45918

(b) Lena

$N(m)$ represents the total number of significant coefficients with respect to a given threshold in different subbands, HL_m, LH_m, and HH_m, except that $N(LL)$ is the number in subband LL.

the comparison between the two tables, we know that Barbara image involves more significant coefficients in finer subbands for the thresholds above 8 than the Lena image does. That is, it can be interpreted that the flexible tree is especially effective to exploit coefficients in the higher frequency subbands.

Comparing the 1-to-4 parent-child relationship used in the EZW coder with the flexible parent-child relationship, the latter allows more significant children to belong to significant parents. As a result, significant children are connected with fewer significant coefficients, and thus more zerotree roots can be produced in coarser subbands. In other words, efficient zerotree roots are made. This is likely to appear, when there are many significant children in higher subbands. This analysis for the flexible treeing can be supported by the comparisons in Fig. 6.

We have described two coders: FZW-A and FZW-B. Especially FZW-B has an advantage that gives

Table 4 Execution time for encoding/decoding.

	Thresholds*	FZW-A	EZW	FZW-B
Encoder	8	4.2	4.0	3.2
	16	3.6	3.4	2.8
Decoder	8	3.0	2.6	1.9
	16	2.5	2.1	1.5

(in second)

* Thresholds are given to indicate stopping points of encoding/decoding; where encoding/decoding is processed up to the corresponding subordinate pass with respect to the threshold.

fast encoding/decoding by replacing the adaptive arithmetic coding with re-symbolization. Table 4 has been added to compare the execution time among FZW-A, FZW-B and EZW. This was done with a DOS/V computer equipped with a Pentium-II 266 MHz processor.

5. Conclusions

We have proposed a modified EZW coder that implements a flexible zerotree coding of wavelet coefficients. In the proposed coder, a flexible parent-child relationship is defined so as to exploit spatial dependencies within a subband as well as hierarchical dependencies across subbands. The produced symbols are re-symbolized into binary codes.

The binary symbol streams produced by the FZW and EZW coders are compared in length. In this simulation, the FZW coder showed remarkable improvements. The improvements have been resulted from the flexible parent-child relationship, the probing bit and re-symbolization. In arithmetic-coded mode, FZW-A offered better performances than EZW-A. Moreover the performances of FZW-B are competitive with those of EZW-A incorporated with arithmetic coding.

Performances of the FZW coder can be more improved by using the bit saving algorithm that was not implemented in the experiments. Also, further improvements can be achieved by a higher-order adaptive arithmetic coding.

Acknowledgement

This work was in part supported by the Grant-in-Aid for Scientific Research, No. 07650419, from the Ministry of Education, Science and Culture of Japan and the research grant for sophisticated telecommunication systems for sharing advanced medical resources given by Telecommunications Advancement Organization of Japan. We appreciate the reviewers for their helpful comments and thorough reading.

References

[1] J.M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," IEEE Trans. Signal Processing, vol.41, no.12, pp.3445–3462, Dec. 1993.
[2] I. Daubechies, "Orthonormal bases of compactly supported wavelets," Communications on Pure and Applied Mathematics, vol.41, pp.909–996, Nov. 1988.
[3] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," IEEE Trans. Pattern Anal. & Mach. Intell., vol.11, pp.674–693, July 1989.
[4] I. Daubechies, *Ten Lectures on Wavelets*, SIAM, Philadelphia, PA, 1992.
[5] J.D. Villasenor, B. Belzer, and J. Liao, "Wavelet filter evaluation for image compression," IEEE Trans. Image Processing, vol.4, no.8, pp.1053–1060, Aug. 1995.
[6] G. Strang and T. Nguyen, "Wavelets and Filter Banks," Wellesley-Cambridge Press, Wellesley, MA, 1996.
[7] D. Le Gall and A. Tabatabai, "Subband coding of digital images using short kernel filters and arithmetic coding techniques," Proc. Int. Conference on Acoustics Speech and Signal Processing, pp.761–764, April 1988.
[8] A.N. Akansu and R.A. Haddad, "Multiresolution Signal Decomposition," Academic Press, San Diego, CA, 1992.
[9] K. Komatsu, K. Sezaki, and Y. Yasuda, "Reversible subband coding of images," IEICE Trans., vol.J78-D-II, no.3, pp.429–436, March 1995.
[10] S. Dewitte and J. Cornelis, "Lossless integer wavelet transform," IEEE Signal Processing Letters, vol.4, no.6, pp.158–160, June 1997.
[11] A. Zandi, J.D. Allen, E.L. Schwartz, and M. Boliek, "CREW: Compression with reversible embedded wavelets," IEEE Data Compression Conference, pp.212–221, Snowbird, March 1995.
[12] A. Said and W.A. Pearlman, "An image multi-resolution representation for lossless and lossy compression," IEEE Trans. Image Processing, vol.5, no.9, pp.1303–1310, Sept. 1996.
[13] Y. Chen and W.A. Pearlman, "Three-dimensional subband coding of video using the zerotree method," Proc. SPIE, Visual Communications and Image Processing, pp.1302–1309, Orlando, March 1996.
[14] J. Luo, X. Wang, C.W. Chen, and K.J. Parker, "Volumetric medical image compression with three-dimensional wavelet transform and octave zerotree coding," Proc. SPIE, Visual Communications and Image Processing '96, pp.579–590, Orlando, March 1996.
[15] J. Wang and H.K. Huang, "Medical image compression by using three-dimensional wavelet transformation," IEEE Trans. Med. Imaging, vol.15, pp.547–554, Aug. 1996.
[16] A. Said and W.A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," IEEE Trans. Circuits and Systems for Video Technology, vol.6, no.3, pp.243–250, June 1996.
[17] Z. Xiong, K. Ramchandran, and M.T. Orchard, "Space-frequency quantization for wavelet image coding," IEEE Trans. Image Processing, vol.6, no.5, pp.677–693, May 1997.
[18] Z. Xiong, K. Ramchandran, and M.T. Orchard, "Wavelet packets image coding using space-frequency quantization," IEEE Trans. Image Processing, vol.7, pp.892–898, June 1998.
[19] S.A. Martucci, I. Sodagar, T.H. Chiang, and Y.Q. Zhang, "A zerotree wavelet video coder," IEEE Trans. Circuits and System for Video Technology, vol.7, no.1, pp.109–118, Feb. 1997.
[20] J. Li, P. Cheng, and C. Kuo, "On the improvements of embedded zerotree wavelet (EZW) coding," Proc. SPIE, Visual Communications and Image Processing, pp.1490–1501, Orlando, April 1995.
[21] C.S. Barreto and G.V. Mendonca, "Enhanced zerotree wavelet transform image coding exploiting similarities in-

side subbands," Proc. Int. Conference on Image Processing, vol.2, pp.549–551, Lausanne, Switzerland, Sept. 1996.

[22] S.H. Joo, H. Kikuchi, S. Sasaki, and J. Shin, "A flexible zerotree coding with low entropy," Proc. Int. Conference on Acoustics Speech and Signal Processing, vol.5, pp.2685–2688, Seattle, May 1998.

[23] P.G. Sherwood and K. Zeger, "Progressive image coding for noisy channels," IEEE Signal Processing Letters, vol.4, no.7, pp.189–191, July 1997.

[24] H. Man, F. Kossentini, and M. Smith, "Robust EZW image coding for noisy channels," IEEE Signal Processing Letters, vol.4, no.8, pp.227–229, Aug. 1997.

[25] C.D. Creusere, "A new method of robust image compression based on the embedded zerotree wavelet algorithm," IEEE Trans. Image Processing, vol.6, no.10, pp.1436–1442, Oct. 1997.

[26] J.K. Rogers and P.C. Cosman, "Wavelet zerotree image compression with packetization," IEEE Signal Processing Letters, vol.5, no.5, pp.105–107, May 1998.

[27] T. Cover and J. Thomas, "Elements of Information Theory," Wiley, New York, 1991.

[28] S.S. Yu and N.P. Galatsanos, "Binary decompositions for high-order entropy coding of grayscale images," IEEE Trans. Circuits and Systems for Video Technology, vol.6, no.1, pp.21–31, Feb. 1996.

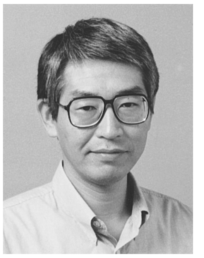
[29] M.J. Weinberger, J.J. Rissanen, and R.B. Arps, "Applications of universal context modeling to lossless compression of gray-scale images," IEEE Trans. Image Processing, vol.5, no.4, pp.575–586, April 1996.

[30] H. Witten, R. Neal, and J.G. Cleary, "Arithmetic coding for data compression," Comm. ACM, vol.30, no.6, pp.520–540, June 1987.

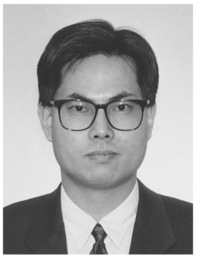
[31] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," IEEE Trans. Image Processing, vol.1, no.2, pp.205–220, April 1992.

[32] C.B. Jones, "An efficient coding system for long source sequences," IEEE Trans. Inf. Theory, vol.27, no.3, pp.280–291, May 1981.

Sanghyun Joo received B.E. and M.E. degrees from Dongguk University, Korea, in 1989 and 1994, respectively, and Dr.Eng. degree from Niigata University, Niigata, Japan, in 1999. From 1994 to 1996 he worked at Korea Academy of Industrial Technology (KAITech). Since 1999 he has been with Niigata University, where he is now a Research Associate. His research interests include digital signal processing, image processing and wavelets. He is a member of IEEE and IEEK.



Hisakazu Kikuchi was born in Niigata in 1952. He received B.E. and M.E. degrees from Niigata University, Niigata, in 1974 and 1976, respectively, and Dr.Eng. degree in electrical and electronic engineering from Tokyo Institute of Technology, Tokyo, in 1988. From 1976 to 1979 he worked at Systems Laboratories, Fujitsu Ltd. Since 1979 he has been with Niigata University, where he is a Professor in electrical engineering. During a year of 1992 to 1993, he was a visiting scientist at the Electrical Engineering Department, University of California, Los Angeles. His research interests include digital signal processing, image processing, wavelets, and mobile telecommunications. He is a member of IEEE, Inst. of Image Information and Television Engineers, and Japan SIAM.



Shigenobu Sasaki was born in Nara, Japan on April 18, 1964. He received B.E., M.E. and Dr.Eng. degrees from Nagaoka University of Technology, Nagaoka, Japan, in 1987, 1989 and 1993, respectively. Since 1992, he has been with the Department of Electrical and Electronics Engineering, Faculty of Engineering, Niigata University, Niigata, Japan, where he is now Assistant Professor. His research interests are in communication systems and signal processing, especially spread spectrum communication systems and mobile communications. Dr. Sasaki is a member of IEEE and Society of Information Theory and its Applications (SITA), Japan.

Jaeho Shin received the B.E., M.E. and Ph.D. degrees in electronics engineering from Seoul National University, Seoul, Korea, in 1979, 1982 and 1987 respectively. He is a Professor in the Department of Electronic Engineering, Dongguk University in Seoul, Korea. His research interests include digital signal processing, digital system design, information security. He is a member of IEEE, IEEK, KICS, KISSC, and KISS.