

Wavelet-Based Lossless Compression Scheme with Progressive Transmission Capability

Adrian Munteanu,¹ Jan Cornelis,¹ Geert Van der Auwera,¹ Paul Cristea²

¹ ETRO Department, IRIS Research Group, Vrije Universiteit Brussel, Pleinlaan 2, Brussels B-1050, Belgium

² Digital Signal Processing Department, Politehnica University of Bucharest, Splaiul Independentei 205, Bucharest 77206, Romania

ABSTRACT: Lossless image compression with progressive transmission capabilities plays a key role in measurement applications, requiring quantitative analysis and involving large sets of images. This work proposes a wavelet-based compression scheme that is able to operate in the lossless mode. The quantization module implements a new technique for the coding of the wavelet coefficients that is more effective than the classical zerotree coding. The experimental results obtained on a set of multimodal medical images show that the proposed algorithm outperforms the embedded zerotree coder combined with the integer wavelet transform by 0.28 bpp, the set-partitioning coder by 0.1 bpp, and the lossless JPEG coder by 0.6 bpp. The scheme produces a losslessly compressed embedded data stream; hence, it supports progressive refinement of the decompressed images. Therefore, it is a good candidate for telematics applications requiring fast user interaction with the image data, retaining the option of lossless transmission and archiving of the images. © 1999 John Wiley & Sons, Inc. *Int J Imaging Syst Technol*, 10, 76–85, 1999

I. INTRODUCTION

Various methods both for lossy (irreversible) and lossless (reversible) image compression are proposed in the literature. The recent advances in the lossy compression techniques include different methods such as vector quantization (Cosman et al., 1996), wavelet coding (Shapiro, 1993; Antonini et al., 1992), neural networks (Dony and Haykin, 1995), and fractal coding (Fisher, 1994). Although these methods can achieve high compression ratios (of the order of 50:1 or more), they do not allow one to reconstruct exactly the original version of the input data. Lossless compression techniques permit perfect reconstruction of the original image, but the achievable compression ratios are only of the order of 2:1 up to 4:1. The current article addresses the following combined problem: 1) obtaining high compression ratios for the lossless coding of images comparable to those obtained by currently used state-of-the-art schemes; and 2) achieving this in an embedded fashion, i.e., in such a way that all encodings of the same image at lower bit rates are embedded in the beginning of the bit stream needed for the lossless coding.

Solving this problem is important for many application domains. In medical imaging it is not acceptable to lose information because

of the coding, since this could cause severe legal problems and lead to the destruction of small image detail which might be an indication of pathology. In several medical applications such as coronary angiography, where one has to measure submillimeter blood vessel diameters at the location of a stenosis, lossy coding methods are not accepted (Lewis, 1996). In satellite imaging huge amounts of image data (e.g., METEOSAT multispectral images of the full earth disc represent 37 MB every 30 min, increased to 266 Mb every 15 min, for METEOSAT Second Generation) are disseminated for scientific and professional usage. Some users might be interested in following the time behavior of the single pixel corresponding with their local coordinates; others might exploit the spatial structure of the images at the level of a country or continent. More generally, for measurement image data, one usually cannot afford information loss due to compact coding.

A. Lossless Coding and Progressive Data Transmission.

In recent years, several methods have been proposed for lossless image compression, such as context-based, predictive coding (CALIC) (Wu, 1997), universal context modeling-based coding (Weinberger et al., 1996), two-dimensional multiplicative autoregressive model-based coding (Das and Burgett, 1993), and segmentation-based image coding (Shen and Rangayyan, 1997). Although these coders are among the best performing ones as far as the compression ratio is concerned, the lack of a progressive transmission capability makes them unsuitable for interactive applications expanded over large networks. Embedded coding permits the progressive transmission of the compressed data by starting with an economical initial transmission of a rough image version, followed by gradual transmissions of the refinement details, without adding overhead in the bit rate needed for the final reconstruction. The concept of progressive data transmission is of particular importance in an environment including applications such as fast telebrowsing through large data sets (Tzou, 1987), because as distance interactivity grows, the available network bandwidth has the tendency to collapse. For economical data transmission over large networks, it is therefore desirable to send the compressed data progressively and refine the images gradually at the destination site, retaining the option of perfect reconstruction.

B. Implementation of Lossless Embedded Coders. Among the possible alternatives for a lossless compression scheme with

Correspondence to: A. Munteanu

progressive transmission capabilities, the transform-based techniques are the most attractive. The wavelet transform can be implemented in compression schemes featuring progressive transmission, in which one transmits the coarser version of the image first, followed by successive transmissions of the refinement details. This article presents a wavelet-based compression method that is able to operate in the lossless mode. There is a significant difference between this method and the standard wavelet compression techniques (Shapiro, 1993; Antonini et al., 1992), which cannot reconstruct the lossless version of the original image because of the rounding to integers of the wavelet coefficients. Our alternative is the use of the lifting scheme (Sweldens, 1996) to compute an integer wavelet representation. Using lifting, it is particularly easy to build nonlinear wavelet transforms that map integers to integers and which permit a lossless representation of the image pixels (Dewitte and Cornelis, 1997; Calderbank et al., 1996). Moreover, factorization of the wavelet transform into lifting steps reduces the computational complexity by a factor of two compared with the standard algorithm implementing the fast wavelet transform (Daubechies and Sweldens, 1996).

C. Structure of the Article. Section II briefly describes the integer wavelet transform. The quantization module described in Section III implements a new technique for the coding of the wavelet coefficients that is more effective than the classical zerotree coding (Shapiro, 1993). Section IV reports the performances obtained by the proposed compression algorithm and other lossless compression algorithms applied on 1) a set of multimodal medical images, and 2) a heterogeneous set of general images. The rate-distortion performances of our algorithm are compared with those obtained with other techniques. An example is given to illustrate the progressive transmission capabilities of the proposed algorithm, and a visual comparison to a JPEG block transform scheme shows the superior performance of our method.

II. INTEGER WAVELET TRANSFORM

In most cases, the filters used in wavelet transforms have floating-point coefficients. Since the input images are matrices of integer values, the filtered output no longer consists of integers and losses will result from rounding. For lossless coding it is necessary to make a reversible mapping between an integer image input and an integer wavelet representation.

The solution was first offered by the sequential transform (S-transform) introduced in (Heer and Reinfeldt, 1990). The analysis and synthesis high-pass filters corresponding to this transform are of low orders (they have only one vanishing moment), yielding poor compression results, particularly for “natural” images. Higher-order filters have been obtained with two similar generalizations of the S-transform: the TS transform (Zandi et al., 1995) and the S + P transform (Said and Pearlman, 1996). Filters of arbitrarily high orders have been derived in (Dewitte and Cornelis, 1997) starting from the S-transform and using the lifting scheme (Sweldens, 1996).

A unified framework for designing integer wavelet transforms is given in (Calderbank et al., 1996), where it is shown that the S, TS, and S + P transforms can all be seen as special cases of the lifting scheme, and that in general, an integer version of every wavelet transform employing finite filters can be built with a finite number of lifting steps.

The lifting scheme is used in (Sweldens, 1996) to construct symmetric, biorthogonal wavelet transforms starting from interpolating Deslauriers–Dubuc scaling functions (Deslauriers and Dubuc, 1987). A family of (M, \tilde{M}) symmetric biorthogonal wavelets is

derived, where M is the number of vanishing moments of the analysis high-pass filter and \tilde{M} is the number of vanishing moments of the synthesis high-pass filter. An instance of this family of transforms is the (4,2) interpolating transform. The integer version of it, given in (Calderbank et al., 1996), is implemented in the first stage of our coding algorithm. In this case, the integer wavelet representation of a one dimensional signal $A^0(n)$ having N nonzero samples is given by:

$$\forall n: D^{i+1}(n) = A^i(2n + 1) - \left[\sum_k p_k A^i(2(n - k)) + \frac{1}{2} \right],$$

$$0 \leq i < J, \quad 0 \leq n < 2^{-(i+1)}N, \quad -2 \leq k \leq 1$$

$$\forall n: A^{i+1}(n) = A^i(2n) + \left[\sum_k u_k D^{i+1}(n - k) + \frac{1}{2} \right],$$

$$0 \leq i < J, \quad 0 \leq n < 2^{-(i+1)}N, \quad 0 \leq k \leq 1$$

where $[x]$ represents the integer part of x , J is the number of scales, and $A^{i+1}(n)$ and $D^{i+1}(n)$ denote the approximation and the detail, respectively, of the original signal calculated at the scales $(i + 1)$, $0 \leq i < J$; the filters coefficients are given in (Calderbank et al., 1996):

$$p_{-2} = p_1 = -\frac{1}{16}, \quad p_{-1} = p_0 = \frac{9}{16}$$

$$u_0 = u_1 = \frac{1}{4}$$

The extension of the integer wavelet representation of a one-dimensional signal to the two-dimensional case consists of the successive application of the one-dimensional wavelet transform, as described in (Mallat, 1989). This results in the representation of any arbitrary input image f via the set:

$$(A_j, WX_k, WY_k, WXY_k)_{1 \leq k \leq J}$$

where A_j is the approximation of f at the coarser resolution 2^{-j} , and WX_k , WY_k , and WXY_k , are the details of the image in horizontal, vertical, and diagonal orientation, respectively, at the resolutions 2^{-k} , $1 \leq k \leq J$.

III. CODING OF THE WAVELET COEFFICIENTS

A. Significance Maps Coding. Embedded zerotree coding of the wavelet coefficients (EZW) (Shapiro, 1993) uses a simple and general model to describe the distribution of the coefficients in the wavelet image. This model is based on the “zerotree” hypothesis (Shapiro, 1993), which assumes that if a wavelet coefficient w at a certain scale is insignificant with respect to a given threshold T , i.e., $|w| < T$, then all the coefficients of the same orientation in the same spatial location at finer scales are also insignificant with respect to the threshold T .

Successive approximation quantization (SAQ) is applied in (Shapiro, 1993) to determine the significance of the wavelet coefficients with respect to a set of thresholds T_i . The positions of the significant and insignificant coefficients, respectively, are indicated in binary maps (called significance maps), for each threshold T_i . It is proven in (Shapiro, 1993) that no matter how optimal the wavelet transform,

quantizer, or entropy coder are, the cost of encoding the significance maps represents an important portion of the total encoding cost. Hence, an improved coding of the significance maps results in a large coding gain.

A method to encode the significance maps is given by the zerotree representation (Shapiro, 1993) which allows an efficient prediction of insignificant information across the scales. With this technique, the cost of encoding the significance maps is reduced by grouping the insignificant coefficients in trees growing exponentially across the scales, and by coding them with zerotree symbols. The efficiency of such a coding technique relies on the assumption that there is a high probability for the coefficients in the wavelet image to satisfy the zerotree hypothesis.

A more complex technique, set partitioning in hierarchical trees (SPIHT), used for the coding of the significance maps is described in (Said and Pearlman, 1996b). This method uses the principles of partial ordering by magnitude of the wavelet coefficients (resulting from SAQ), set partitioning in hierarchical trees (i.e., sorting of the trees, based on their significance, at every applied threshold), ordered bitplane transmission of the refinement bits (i.e., the magnitude of each significant coefficient is progressively refined), and self-similarity across scales as in EZW. The essential difference of the SPIHT coding process with respect to EZW is the way trees of coefficients are partitioned and how significance information is conveyed.

An alternative method for the significance maps coding is described in this section. This method makes use of quadtree data structures to encode the positions of the significant coefficients in the wavelet image.

The quadtree (QT) decomposition for image and video coding has been considered by several researchers. QT decomposition divides an image into two-dimensional homogeneous (in the property of interest) blocks of variable size. Gray-scale image coding based on a quadtree data structure and a linear model for the luminance function inside the blocks of a variable size has been presented in (Strobach, 1991). Improved QT decomposition algorithms were designed in (Shusterman and Feder, 1994) by using a near optimal choice of the thresholds used for the QT decomposition and optimal bit allocation procedures. Algorithms developed to build optimal quadtree structures, given some distortion measures, such as squared error or absolute difference were reported in (Sullivan and Baker, 1994); these algorithms were used to design a video coding system implementing optimal/near optimal quadtrees and vector quantization. Generally, in terms of compression performances these techniques are similar to the DCT-based techniques. From a comparison of several QT techniques given in (Strobach, 1991), it appears that they are appropriate for image coding at high to medium rates (the algorithm reported in (Sullivan and Baker, 1994) is an exception, since it reports good compression performances for low-bit-rate video coding). In contrast to QT decomposition applied in the spatial domain, as in (Strobach, 1991; Shusterman and Feder, 1994; Sullivan and Baker, 1994), in this article we investigate the use of quadtree decomposition in the wavelet domain, and more specifically, for significance maps coding. We show in the following that the positions of the significant coefficients in the wavelet image can be encoded efficiently using a hierarchical structure of squares that group the insignificant coefficients in blocks of variable width.

Successive approximation quantization is applied to determine the significance of the wavelet coefficients with respect to a monotonically decreasing series of thresholds. In the following, we refer only to one of the quantization steps in which one determines the

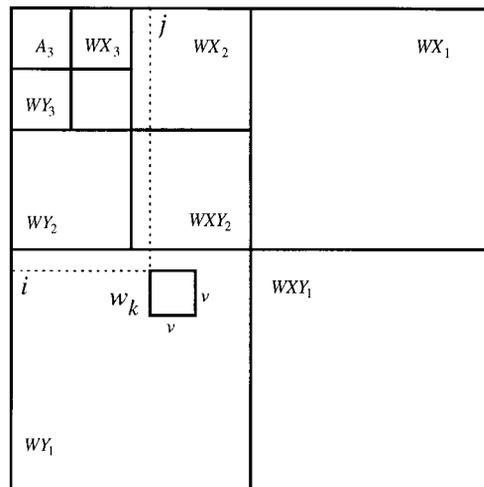


Figure 1. Schematic representation of a three-level wavelet decomposition ($J = 3$); the square w_k with the top left corner coordinates (i, j) delimits v^2 coefficients from the wavelet image.

significance of the coefficients from the wavelet image with respect to a given threshold T .

Denote by (p, q) the spatial location in the wavelet image of an arbitrary coefficient $w(p, q)$, where p and q stand for the row, respectively, column index. The significance of $w(p, q)$ with respect to the threshold T is recorded in the location (p, q) of a binary matrix (significance map). The positions of the significant coefficients are indicated with values of "1," while the values of "0" indicate the positions of the insignificant coefficients.

Define a square w_k of width v delimiting a set of wavelet coefficients located in an arbitrary wavelet subband $W_k \in \{WX_k, WY_k, WXY_k, 1 \leq k \leq J\}$. Make the assumption that the width v of the square has the form:

$$v = 2^\alpha, \quad \alpha \in N \quad (1)$$

where N denotes the set of natural numbers.

Also, denote by i and j the indices corresponding to the row and column location in the wavelet image of the top left corner of w_k , as indicated in the example in Figure 1.

Define a matrix of $v \times v$ elements containing the wavelet coefficients $w(p, q) \in W_k$ delimited by the square w_k :

$$SQ(i, j, v) = (w(p, q))_{i \leq p < i+v, j \leq q < j+v} \quad (2)$$

The significance with respect to the threshold T of the coefficients from $SQ(i, j, v)$ is determined by a function f which maps any value $w(p, q)$ from $SQ(i, j, v)$ to the set $\{0, 1\}$:

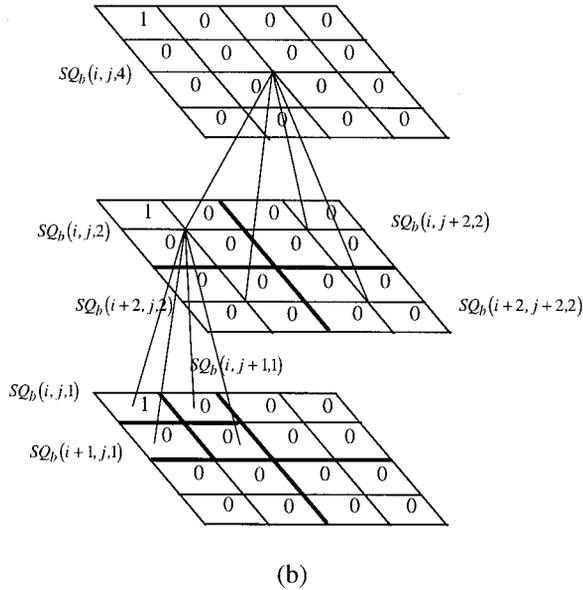
$$\forall w(p, q) \in SQ(i, j, v), f(w(p, q)) = \begin{cases} 0, & \text{if } |w(p, q)| < T \\ 1, & \text{if } |w(p, q)| \geq T \end{cases} \quad (3)$$

The binary matrix which corresponds to $SQ(i, j, v)$ and indicates the significance of the coefficients with respect to the applied threshold T is defined as:

$$SQ_b(i, j, v) = (f(w(p, q)))_{i \leq p < i+v, j \leq q < j+v} \quad (4)$$

1	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

(a)



(b)

Figure 2. Hierarchical tree structure of matrices corresponding to $SQ_b(i, j, 4)$. (a) Example of a 4×4 binary matrix $SQ_b(i, j, 4)$. (b) Partition rule P is applied recurrently only on the binary matrices containing the nonzero element (thick lines indicate the successive partitioning in minors).

Define the partition rule P which divides the matrix $SQ_b(i, j, v)$ into four adjacent minors:

$$P: SQ_b(i, j, v) = \begin{pmatrix} SQ_b(i, j, v/2) & SQ_b(i, j + v/2, v/2) \\ SQ_b(i + v/2, j, v/2) & SQ_b(i + v/2, j + v/2, v/2) \end{pmatrix} \quad (5)$$

Partition the matrix $SQ_b(i, j, v)$ using the rule P given in Equation (5), and continue to apply it recursively, only on the minors that contain nonzero elements. The partitioning stops when all the nonzero values from $SQ_b(i, j, v)$ are localized, i.e., when all the minors containing the nonzero elements have the form $SQ_b(m, n, 1)$. Using the rule P , the locations (m, n) of the nonzero elements in the significance map are uniquely identified.

The partition rule P decomposes the matrix $SQ_b(i, j, v)$ into a hierarchical structure of matrices. Therefore, one can define a tree structure having nodes corresponding to every matrix that is to be partitioned with P , and four child-nodes for the corresponding minors. An example with a 4×4 binary matrix $SQ_b(i, j, 4)$ and the corresponding hierarchical tree structure of matrices are illustrated in Figure 2(a), respectively Figure 2(b).

Encoding the positions of the significant coefficients in the significance map translates in encoding the entire tree structure of matrices. To do this, both the encoder and the decoder have to make the distinction between the matrices that contain only "0" elements and the ones that have to be partitioned. Each matrix containing only "0" elements is identified with a nonsignificant symbol (NSG), which indicates that the insignificance of the corresponding coefficients is completely predictable. On the contrary, any matrix that contains at least a nonzero value is identified with a significant symbol (SGN), showing that at least one coefficient is expected to be significant.

The tree structure of matrices can be represented by the corresponding tree structure of symbols. For the example given in Figure 2(a), the corresponding three-dimensional structure of symbols is given in Figure 3.

A scanning procedure is necessary to map at the encoder site the three-dimensional structure of symbols to a one-dimensional set of symbols (significance list). In the procedure described in the following, the breadth-first search technique (Parsaye and Chignell, 1988) is adapted to the three-dimensional structure of symbols.

1. Scanning Procedure A.

1. *Initialization:* start scanning from the root symbol of the tree (situated at the highest level, $i = Q$) and add it to the significance list;
 - 1.1. If the symbol is "NSG" then stop scanning since there are no symbols at the next level;
 - 1.2. If the symbol is "SGN" then scan in raster order the corresponding four symbols from the level below and add them to the significance list.
2. *Recurrent module:*
 - 2.1. Move to the next level in the tree, $i \rightarrow i - 1$ (replace "i" by "i - 1");
 - 2.2. If the current level i is the lowest level in the tree (i.e. $i = 1$), then stop scanning; else, go to step 2.3;
 - 2.3. Scan all the symbols situated at the level i in the tree in the same order they were previously added in the significance list;
 - 2.3.1 If the current symbol is "SGN" then:

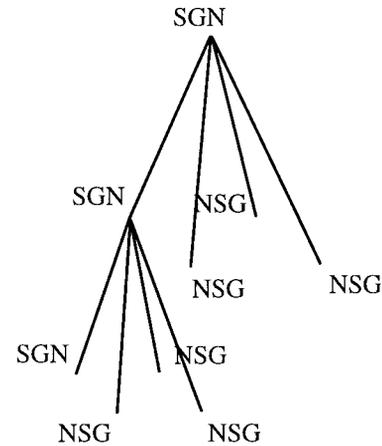


Figure 3. Tree structure of symbols associated to the tree structure of matrices presented in Figure 2(b); note that for the SGN symbols there are four corresponding symbols at the level below, while for the NSG symbols there are no correspondents.

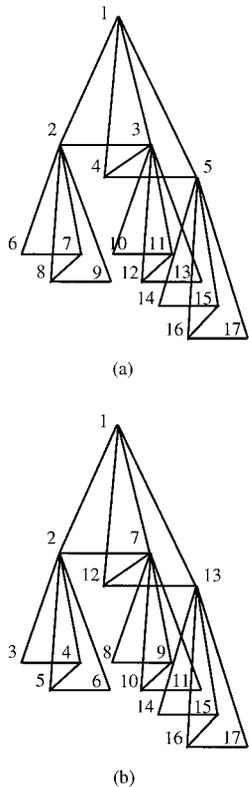


Figure 4. (a) Example of a tree structure of symbols scanned by using Procedure A; the numbers represent the scanning order. (b) Results obtained with scanning Procedure B applied on the same structure of symbols.

- Scan in raster order the corresponding four symbols from the level below and append them to the significance list;

2.3.2. Else, continue;

2.4. Go to step 2.1.

A simple example of this scanning procedure applied to a three-dimensional structure of symbols is given in Figure 4(a). The numbers corresponding to each node indicate the order in which the symbols in the tree are added in the significance list.

To describe the second scanning procedure, we introduce a few definitions. For a given SGN symbol in the tree (named the parent symbol), we find four corresponding symbols at the level below, which are called descendant symbols. For an arbitrary descendant symbol, the other three descendant symbols are called neighbor symbols. In this procedure, the depth-first search technique (Parsaye and Chignell, 1988) is adapted to the three-dimensional structure of symbols.

2. Scanning Procedure B.

1. *Initialization:* start scanning from the root symbol of the tree (highest level, $i = Q$);
 - 1.1. If the symbol is “NSG” then add it to the significance list, and stop scanning since there are no symbols at the next level;
 - 1.2. If the symbol is “SGN” then add it to the significance list and go to step 2.1.

2. Recurrent module:

- 2.1. Select from the level $(i - 1)$ the four descendant symbols corresponding to the current symbol in level i , and do $i \rightarrow i - 1$;
- 2.2. Start scanning in raster order the symbols selected in step 2.1. or 2.2.2.3.;
 - 2.2.1. For every scanned symbol from level i which was not appended to the significance list, do:
 - 2.2.1.1. Add the current symbol in the significance list;
 - 2.2.1.2. If the current symbol is “SGN” then:
 - If $i \neq 1$ (the current level is not the lowest level), go to 2.1.;
 - Else, continue;
 - 2.2.1.3. Else, (current symbol is “NSG”), continue;
 - 2.2.2. If all four coefficients from level i were appended to the significance list, then:
 - 2.2.2.1. Consider the parent of these four symbols as being the current symbol;
 - 2.2.2.1. Do $i \rightarrow i + 1$;
 - 2.2.2.2. If $i = Q$ then stop scanning;
 - 2.2.2.3. Else, select the set of symbols that includes the current symbol and the corresponding neighbors and go to 2.2.

We applied this scanning procedure to the same three-dimensional structure given in Figure 4(a). The order in which the symbols are added to the significance list is shown in Figure 4(b).

Whether to choose the first scanning procedure or the second one is a matter of experimental assessment. In the experimental results reported in Section IV we prefer the second alternative, since the coding algorithm implementing it, applied to various types of images, yields on average the best compression performances.

We remark that by using the proposed algorithm the regions in the significance map containing “1” elements are gradually localized by grouping the “0” elements in square matrices of different size. If the significant coefficients are clustered in certain areas in the significance map, then the large areas containing insignificant coefficients can be encoded with few bits. In this situation the proposed algorithm is expected to yield good encoding performances, since it can isolate interesting nonzero details by immediately eliminating large insignificant regions from further consideration. On the contrary, if we have a random distribution of “0” and “1” values in the binary map, the cost of encoding the positions of the nonzero elements increases tremendously since a high number of partitioning steps is needed. However, practical situations reveal that the nonzero values in the significance map (i.e., significant coefficients) are mostly clustered around the edges, while the zero values (i.e., insignificant coefficients) correspond mostly to the homogeneous areas in the image. This simple heuristic model is applied to characterize the distribution of the significant coefficients in the wavelet image and is exploited by the significance map encoding algorithm previously described.

B. Coding Algorithm. Successive approximation quantization is applied in the coding algorithm to determine the significance of the coefficients with respect to a set of predefined thresholds $(T_i)_{1 \leq i \leq N}$. The thresholds are chosen so that $T_{i+1} = T_i/2$, for every value i , $1 \leq i < N$. The maximum threshold is $T_1 = 2[\log_2 w_{max}]$, where w_{max} denotes the maximum absolute value of the coefficients from

Table I. Lossless compression bit rates (bpp) obtained with different algorithms on several 8-bpp gray-value images.

Image	LJPG	EZW&IWT	SQP	BTPC	SPIHT	CALIC
“Aerial” 256 × 256 pixels	4.40	4.22	4.21	4.56	4.11	3.86
“Barbara” 512 × 512 pixels	5.53	4.75	4.65	5.40	4.58	4.45
“Cameraman” 256 × 256 pixels	5.03	4.73	4.65	4.98	4.51	4.19
“Lena” 512 × 512 pixels	4.70	4.31	4.31	4.51	4.20	4.11
“Baboon” 512 × 512 pixels	6.58	6.09	6.05	6.51	5.96	5.87
“Peppers” 512 × 512 pixels	5.19	4.68	4.68	4.84	4.62	4.41
Average bit rate	5.24	4.80	4.76	5.13	4.66	4.48

The last row indicates the average bit rate (bpp).

the wavelet image, and $[x]$ is the integer part of x . The minimum threshold is $T_N = 1$, where $N = \lceil \log_2 w_{max} \rceil + 1$.

If w is an arbitrary wavelet coefficient, the following inequality is satisfied:

$$|w| \leq w_{max} = 2^{\log_2 w_{max}} < 2^{\lceil \log_2 w_{max} \rceil + 1} = 2^N$$

which shows that any coefficient from the wavelet image can be binary represented with N bits (excluding the sign bit). We label the most significant bit from the binary representation of any coefficient with the index l , and the least significant bit with the index N .

Consider that the wavelet image is a square matrix of $L \times L$ elements, where $L = 2^\beta$, $\beta \in N$. This assumption does not restrict the generality, because (if necessary) a padding with zeros of the wavelet image can be done. Since L is an integer power of two, according to the relations (1) and (2) one can write the entire wavelet image as $SQ(0, 0, L)$.

The previously described significance maps coding algorithm is used to encode the positions of the significant coefficients in $SQ(0, 0, L)$. The necessary inputs are the wavelet image $[SQ(0, 0, L)]$ and the currently applied threshold T_i .

Observe that the applied thresholds are monotonically decreasing from T_1 to T_N . We deduce that if a coefficient is significant with respect to a threshold T_i , then it is significant also with respect to the whole set of thresholds T_j , $j > i$. This statement can be extended to matrices: if at a certain coding step, an SGN symbol is assigned to a matrix (owing to a significant coefficient belonging to it), then at the next coding steps the matrix is labeled again with an SGN symbol, and so on. This implies that any SGN symbol assigned to a matrix by the significance map encoding algorithm has to be appended only once in the significance list.

Suppose that an arbitrary coefficient is insignificant with respect to the set of thresholds T_j , $1 \leq j < i$, and significant with respect to T_i . As soon as its position is encoded in the coding step i , the following refinement step is triggered. At every coding step j , $j > i$, the value corresponding to the bit plane j from the binary representation of the coefficient is appended into a refinement list. This is equivalent to a progressive refinement at the encoder side of the coefficient’s magnitude at every coding step j , $j > i$.

In addition to the significance list and refinement list, a third binary list retaining the sign information is maintained. The updating of the sign list is done for every significant coefficient after the encoding of the corresponding position in the significance list.

The performances of this coding algorithm can be improved by entropy coding its output, but at the cost of larger coding/decoding times. The significance and the refinement lists are entropy coded with the adaptive arithmetic coder described in (Moffat et al., 1995). In our algorithm, arithmetic coding of the significance and refine-

ment lists is restricted to a binary alphabet. These lists are encoded independently of each other, using separate models in the arithmetic encoder. The significance list is entropy coded using an adaptive fixed-context model. From the input stream, one symbol is taken at a time, and the correct context is chosen based on the previous m symbols. The symbol and corresponding context are passed into the statistics module (Moffat et al., 1995) which calculates the cumulative frequencies of the symbols in the alphabet, and then adjusts these values to account for the occurrence of the new symbol. Good compression results are obtained if m is large (corresponding to higher-order entropy coding), but the coder runs slowly. Empirical experiments reveal that $m = 6$ (equivalent to a number of 64 contexts) produces a good trade off between fast execution and compression efficiency.

The arithmetic coding of the refinement list is based on a classical adaptive zero-order model (Moffat et al., 1995; Witten et al., 1987). Also, there is no need to entropy code the sign list since there is almost no gain in the compression ratio.

Observe that with this coding scheme the coefficients with larger magnitudes are encoded first. This partial ordering by magnitude of the coefficients corresponds to the progressive transmission method proposed in (DeVore et al., 1992). Moreover, the bit-plane ordering of the bits transmitted in the refinement step corresponds to the bit-plane method for progressive transmission described in (Rabbani and Jones, 1991). Another ordering of the coefficients is made in terms of their position: Procedure B starts by scanning the coefficients located in the highest levels of the wavelet pyramid and ends with coefficients situated in the lowest levels. This means that at every quantization step, the algorithm starts by coding the positions of the coefficients situated in the lowest frequency subband and ends by coding the positions of the coefficients located in the highest frequency subbands. The resulting bit stream is completely embedded so that all the versions of the encoded image at lower bit rates are embedded in the beginning of the bit stream needed for the lossless coding.

In the next section, this coding algorithm is identified as the SQP (square partitioning) coder.

IV. EXPERIMENTAL RESULTS

We evaluate the lossless compression performances obtained with the proposed algorithm on two sets of images: The first set contains 8-bpp photographic monochrome images, while the second set includes 20 medical images obtained with different imaging modalities such as angiograms, magnetic resonance (MR), and computed tomographic (CT) images. The coding results are compared with those obtained with other lossless compression techniques: context-based, predictive coding (CALIC) (Wu, 1997), binary tree predictive coding (BTPC) (Robinson, 1997), lossless JPEG (LJPG) (Wallace,

1991; Konji and Smith, 1994), the EZW coder (Shapiro, 1993) combined with the integer wavelet transform (EZW & IWT), and the set partitioning into hierarchical trees coder (SPIHT) (Said and Paerlman, 1996b). To allow both lossless and lossy compression, the SPIHT coder is used with the S + P transform, as proposed by its authors (Said and Pearlman, 1996a). Also, to achieve perfect reconstruction with EZW, we used the same integer wavelet transform as the one proposed in our coder (we refer to this algorithm as EZW&IWT).

The lossless compression performances obtained with the tested algorithms on the set of photographic images are reported in Table I. Compared with the SQP method, on average the bit rate is lower for CALIC and SPIHT coders, but is higher for LJPEG, BTPC, and EZW&IWT.

Figure 5 shows the bit rate values achieved with the algorithms applied on the set of medical images. For most of the images from the test set, the curve corresponding to the SQP algorithm is below the curves corresponding to the SPIHT coder, BTPC, EZW&IWT, and LJPEG coders.

The average compression ratio and the differences in the average bit rates with respect to the SQP coder are presented in Table II. Our algorithm outperforms by 0.1 bpp the set partitioning coder (SPIHT), by 0.21 bpp the BTPC coder, by 0.28 bpp the EZW&IWT coder, and by 0.6 bpp the lossless JPEG coder.

Compared with the SQP coder, CALIC provides a better compression performance both for the photographic set of images (an average gain of 0.28 bpp) and for the tested set of medical images (an average gain of 0.12 bpp), but it cannot be used for progressive image transmission.

The progressive transmission capability of the proposed algorithm is assessed by using the standard 8-bpp test images, 512×512 "Lena" and 512×512 "Barbara." The bit rates reported for SQP are determined from the actual size of the compressed files and are not entropy estimates. Moreover, since the proposed method features progressive transmission, one can measure the distortion corresponding to any rate by using the file generated in lossless compression: The decoder reads the file up to the desired rate, decodes the image, calculates the inverse wavelet transform, and then compares the reconstructed image with the original. The distortion measure is the PSNR, given by:

$$\text{PSNR} = 10 \log_{10} \left(\frac{255^2}{\text{MSE}} \right)$$

where *MSE* denotes the mean squared error between the original and the reconstructed images.

Table II. Average compression ratio, average bit rate (bpp), and difference in average bit rate (bpp) (with respect to our algorithm) obtained in the lossless compression of medical images test set.

Algorithm	Compression Ratio	Average Bit Rate (bpp)	Difference (bpp)
LJPEG	3.0608	2.6137	+0.6077
BTPC	3.6024	2.2207	+0.2147
EZW&IWT	3.4933	2.2901	+0.2841
SPIHT	3.7880	2.1119	+0.1059
SQP	4.2026	2.0060	0.0000
CALIC	4.2072	1.8820	-0.1240

Table III. Rate distortion performances for 512×512 "Lena" obtained with EZW, SPIHT, and SQP.

Bit Rate (bpp)	SPIHT (dB)	EZW (dB)	SQP (dB)	SQP Without Entropy Coding (dB)
0.5	36.51	36.28	36.17	35.98
0.25	33.62	33.17	33.35	33.17
0.125	30.80	30.23	30.55	30.36
0.0625	28.19	27.54	27.86	27.74
0.03125	25.95	25.38	25.72	25.63

The rate-versus-PSNR results obtained for the test images with SQP are compared against those obtained with EZW, SPIHT, BTPC, and JPEG. Coding results for "Lena" and "Barbara" are summarized in Tables III and IV, respectively. The figures given for EZW are reported in (Shapiro, 1993), in which the author used nine-tap symmetric quadrature mirror filters. These filters have floating-point coefficients; therefore, they do not allow lossless compression. As shown in the tables, the PSNR figures obtained for "Lena" with our method are higher relative to EZW at all the rates except 0.5 bpp; also, the coding results for "Barbara" (which is often considered to be a very difficult test image) show that our method outperforms both SPIHT and EZW at every bit rate.

The last columns in Tables III and IV indicate the PSNRs achieved with SQP without entropy coding the significance and refinement lists. We observe that the gain obtained by using entropy coding is relatively small and varies between 0.09 and 0.19 dB for "Lena" and 0.03 and 0.17 dB for "Barbara." These figures show the efficiency of the proposed significance maps coding technique and indicate that for applications requiring the fastest execution, one could eliminate arithmetic coding and allow a small reduction in PSNR. Table V shows the CPU times spent to encode and decode 512×512 "Lena" on a Pentium II computer; these figures include the time spent for the calculation of the wavelet transform. The programs are not optimized, so the data are given just to indicate the speed of the proposed coding algorithm and to illustrate the effect of entropy coding.

Figure 6 shows the PSNRs in decibels for test images "Lena" and "Barbara" as a function of the bit rate. The rate distortion performances depicted for EZW were obtained by replacing the nine-tap symmetric quadrature mirror filters used in (Shapiro, 1993) with the integer wavelet transform used in our coder.

In these graphs the PSNR values achieved with SQP are higher than those obtained with EZW&IWT: At rates between 0.1 and 0.8 bpp, the average difference relative to EZW&IWT is 0.44 dB for "Lena" and 0.85 dB for "Barbara." Compared with BTPC, the

Table IV. Rate distortion performances for 512×512 "Barbara" obtained with EZW, SPIHT, and SQP.

Bit Rate (bpp)	SPIHT (dB)	EZW (dB)	SQP (dB)	SQP Without Entropy Coding (dB)
0.5	31.26	30.53	31.40	31.23
0.25	27.29	26.77	27.41	27.31
0.125	24.96	24.03	25.00	24.93
0.0625	23.54	23.10	23.55	23.52
0.03125	22.51	21.94	22.70	22.67

Table V. SQP encoding/decoding times (s) obtained for 512×512 “Lena” with and without entropy coding.

Bit Rate (bpp)	SQP Without Entropy Coding (Encoding)	SQP Without Entropy Coding (Decoding)	SQP With Entropy Coding (Encoding)	SQP With Entropy Coding (Decoding)
0.25	0.41	0.28	0.56	0.46
0.5	0.48	0.32	0.73	0.55
1	0.56	0.40	0.91	0.77

proposed method significantly outperforms it with an average of 2.02 dB for “Lena” and 3.50 dB for “Barbara.” The comparison with the standard JPEG reveals that our coder brings an overall gain of 3.95 and 3.05 dB for “Lena” and “Barbara,” respectively.

To illustrate the progressive transmission capability of our algorithm, the gradual refinement of the “Lena” image is shown in Figure 7 both for SQP and JPEG. Although we cannot claim that in lossy compression mode the SQP coder is optimal in a rate distortion sense, we remark that our technique preserves better visual quality at high compression ratios, because typical blocking artifacts are present in JPEG compression.

From the coding results reported in this section, we note that for all the tested images, the proposed algorithm outperforms both in lossless and lossy compression the classical EZW technique combined with IWT. Also, for “Barbara” the use of the new method SQP results in a higher PSNR relative to SPIHT, while similar results are reported for “Lena.” These results are achieved without exploiting interband redundancies. EZW and SPIHT are zerotree-based coders, and their efficiency comes from exploiting the self-similarity within the wavelet transform. Coefficients from different subbands located in the same spatial locations obey with high probability the zerotree hypothesis (Shapiro, 1993). Therefore, organizing them in tree structures permits an efficient exploitation of the interband redundancies. However, the zerotree representation has its disadvantage—that is, the zero regions in the significance maps are approximated as a highly constrained set of tree-structured regions; the consequence is that certain zero regions which are not aligned with the tree structure may be expensive to code, and some portions of zero regions may not be included in zerotrees at all. The SQP coder abandons the tree structure constraint, and this prevents it from exploiting cross-subband redundancies. However, intraband redundancies are ex-

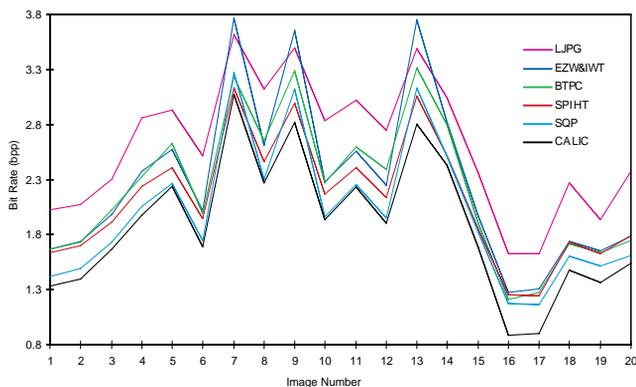


Figure 5. Bit rates achieved for the lossless compression of 20 medical images with various algorithms.

ploited to a larger extent. The zero regions in the significance maps are represented by a union of independent square zero regions of different sizes. The coding gain resulting from an efficient exploitation of intraband redundancies compensates (at least) for the losses incurred by not using an interband decorrelation technique.

The coding results given here seem to indicate that exploiting intraband redundancies can offer a better coding gain than exploiting interband redundancies. However, a technique able to exploit both types of redundancy in an efficient way would certainly provide better coding results. The embedded conditional entropy coding of wavelet coefficients (ECECOW) (Wu, 1997) uses the same successive approximation quantization (SAQ) as EZW and SPIHT, but the gain in performance comes from the exploitation of much higher order statistics than EZW and SPIHT for the coding of the significance maps. This is done by a more efficient interband and intraband dependency model than the classical tree structures of its predecessors (EZW and SPIHT). In the future, performance comparisons will certainly have to include ECECOW.

V. CONCLUSIONS

With telematics applications expanding over large networks, and telemultimedia applications—in which the users are both consumers and producers of image data—becoming more popular, there is a market-driven need for lossless compression schemes with efficient progressive transmission capabilities.

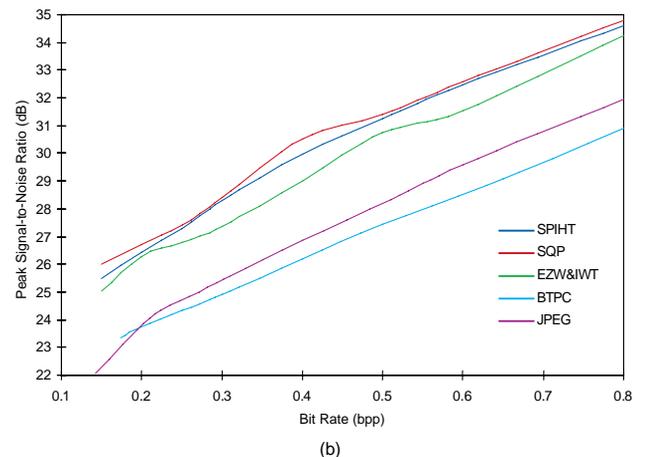
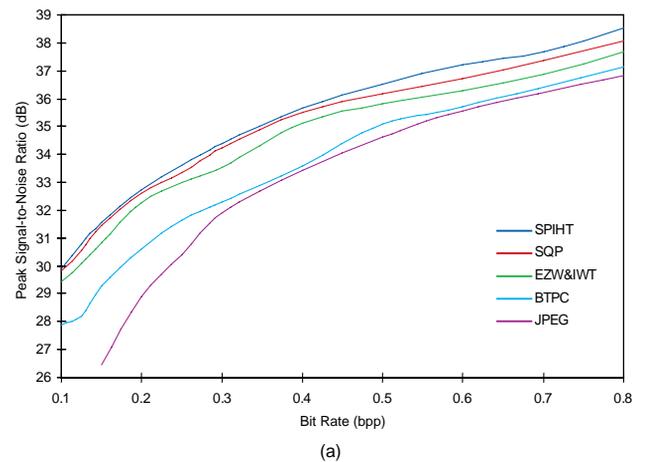


Figure 6. PSNR (in decibels) as a function of bit rate (in bpp) for five different coders applied to (a) “Lena” and (b) “Barbara.”



Figure 7. The “Lena” image was compressed at 60:1, 45:1, and 25:1 with JPEG, respectively SQP; the image is progressively refined up to the lossless version depicted on the right.

Considering the reported results (lossless compression ratios close to the state-of-the-art CALIC technique) and the good rate-distortion performances, we conclude that the proposed compression scheme is a good candidate for applications involving images which have to be treated as measurements for archiving, for image transfer over limited bandwidth channels, and for fast interaction with coarser versions of the image data. The progressive transmission capability of the algorithm makes it attractive for compression schemes in which a subset or all of the received images have to be gradually refined up to the lossless version of the input data.

REFERENCES

M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, Image coding using wavelet transform, *IEEE Trans Image Process* 1 (1992), 205–220.

A.R. Calderbank, I. Daubechies, W. Sweldens, and B. Yeo, Wavelet transforms that map integers to integers, Tech Rep, Department of Mathematics, Princeton University, 1996.

P.C. Cosman, R.M. Gray, and M. Vetterli, Vector quantization of image subbands: A survey, *IEEE Trans Image Process* 5 (1996), 202–225.

M. Das and S. Burgett, Lossless compression of medical images using two-dimensional multiplicative autoregressive models, *IEEE Trans Med Imaging* 12 (1993), 721–726.

I. Daubechies and W. Sweldens, Factoring wavelet transforms into lifting steps, preprint, Bell Laboratories, Lucent Technologies, 1996.

G. Deslauriers and S. Dubuc, “Interpolation dyadique,” *Fractals, dimensions non entières et applications*, Masson, Paris, 1987, pp. 44–55.

R.A. DeVore, B. Jawerth, and B.J. Lucier, Image compression through wavelet transform coding, *IEEE Trans Inform Theory* 38 (1992), 719–746.

S. Dewitte and J. Cornelis, Lossless integer wavelet transform, *IEEE Signal Process Lett* 4 (1997), 158–160.

R.D. Dony and S. Haykin, Neural networks approaches to image compression, *Proc IEEE* 83 (1995), 288–303.

Y. Fisher, *Fractal compression: Theory and application to digital images*, Springer-Verlag, New York, 1994.

V.K. Heer and H.-E. Reinfelder, A comparison of reversible methods for data compression, *Proc SPIE Med Imaging IV* 1233 (1990), 354–365.

H. Kongji and B. Smith, Lossless JPEG codec, version 1.0, June 1994, <http://www.cs.cornell.edu/Info/Projects/multimedia/Projects/LJPG.html>.

R.P. Lewis, President’s page: Developing standards for the digital age: The Dicom project, *J Am Coll Cardiol* 28 (1996), 1631–1632.

S. Mallat, A theory for multiresolution signal decomposition: The wavelet representation, *IEEE Trans Pattern Anal Mach Intell* 11 (1989), 674–693.

A. Moffat, R.M. Neal, and I.H. Witten, Arithmetic coding revisited, *Proc IEEE Data Compression Conf, Snowbird, UT*, IEEE Computer Society Press, 1995.

K. Parsaye and M. Chignell, *Expert systems for experts*, Wiley, New York, 1988.

M. Rabbani and P.W. Jones, *Digital image compression techniques*, SPIE Opt. Eng. Press, Washington, 1991.

J.A. Robinson, Efficient general-purpose image compression with binary tree predictive coding, *IEEE Trans Image Process* 6 (1997), 601–608.

A. Said and W.A. Pearlman, An image multiresolution representation for lossless and lossy compression, *IEEE Trans Image Process* 5 (1996a), 1303–1310.

A. Said and W. Pearlman, A new fast and efficient image codec based on set partitioning in hierarchical trees, *IEEE Trans Circuits Syst Video Technol* 6 (1996b), 243–250.

J.M. Shapiro, Embedded image coding using zerotrees of wavelet coefficients, *IEEE Trans Signal Process* 41 (1993), 3445–3462.

L. Shen and R.M. Rangayyan, A segmentation-based lossless image coding method for high-resolution medical image compression, *IEEE Trans Med Imaging* 16 (1997), 301–307.

- E. Shusterman and M. Feder, Image compression via improved quadtree decomposition algorithms, *IEEE Trans Image Process* 3 (1994), 207–215.
- P. Strobach, Quadtree-structured recursive plane decomposition coding of images, *IEEE Trans Signal Process* 39 (1991), 1380–1397.
- G.J. Sullivan and R.L. Baker, Efficient quadtree coding of images and video, *IEEE Trans Image Process* 3 (1994), 327–331.
- W. Sweldens, The lifting scheme: A custom design construction of biorthogonal wavelets, *J Appl Comput Harmonic Anal* 3 (1996), 186–200.
- K.H. Tzou, Progressive image transmission: A review and comparison of techniques, *Opt Eng* 26 (1987), 581–589.
- G.K. Wallace, The JPEG still picture compression standard, *Commun ACM* 34 (1991), 30–44.
- M.J. Weinberger, J.J. Rissanen, and R.B. Arps, Applications of universal context modelling to lossless compression of gray-scale images, *IEEE Trans Image Process* 5 (1996), 575–586.
- I.H. Witten, R. Neal, and J.G. Cleary, Arithmetic coding for data compression, *Commun ACM* 30 (1987), 520–540.
- X. Wu, Lossless compression of continuous-tone images via context selection, quantization, and modeling, *IEEE Trans Image Process* 6 (1997), 656–664.
- X. Wu, High-order context modelling and embedded conditional entropy coding of wavelet coefficients for image compression, *Proc 31st Asilomar Conf Signals Systems Computers*, Nov. 1997.
- A. Zandi, J.D. Allen, E.L. Schwartz, and M. Boliek, CREW: Compression by reversible embedded wavelets, *Proc Data Compression Conf* (1995), pp. 212–221.