Bekkouche, Barret and Oksman: Adapted lifting schemes for lossless image coding

Adapted lifting schemes for lossless image coding

Hocine Bekkouche, Michel Barret and Jacques Oksman

H. Bekkouche and M. Barret are with SUPELEC, Équipe Signaux et Systèmes Électroniques, 2 rue É. Belin, 57000 Metz, France.
J. Oksman is with SUPELEC, Équipe Signaux et Systèmes Électroniques, Plateau de Moulon, 91192 Gif-sur-Yvette, France.
e-mail: `FirstName.Name@supelec.fr`
This work was supported in part by the Lorraine Region.

Abstract

In this paper we introduce adapted lifting schemes with truncated linear mean square estimation. The idea behind this work is to design subband coders capable of universal performance, in the progressive lossless compression framework, for all type of images. Predictors can be adapted to the local features of the image (method LAE), or to the whole image (method GAE). An experimental study and a comparison of compression results with other well-known methods are presented. The proposed compression system gives, on average, smaller compression bit rates than other codecs permitting progressive coding, with the trade-off of an increase in the computational complexity.

EDICS Categories: 1-STIL (Still Image Coding), 2-WAVP (Wavelets and Multiresolution Processing)

I. Introduction

The lossless image compression finds applications in satellite and medical imaging, where a lossy or near lossless coding is not satisfactory. In the well-known wavelet multiresolution decompositions, the coefficients of the filters are fixed. They are not calculated in order to fit the image the best. Moreover, most of the lossless image compression algorithms used today, like CALIC [26] and LOCO-I [25], are not associated with multi-resolution decompositions. On the other hand, they adapt to the image using context-based predictors. Bi-orthogonal wavelet decompositions are efficient for lossy and near lossless image compression, hence they are used in the ISO JPEG2000 standard. The lifting scheme was introduced by W. Sweldens [24] in order to construct wavelet decompositions by a simple, reversible and fast process. It found quickly its main application in lossless image compression. In this case, a nonlinear filter bank with critical sampling and perfect reconstruction is obtained, with nonlinearities which are limited to truncations. I. Daubechies and W. Sweldens show that any bi-orthogonal wavelet decomposition with FIR (Finite Impulse Response) filters can be represented by a lifting scheme [13], and hence that all the well-known wavelets used in lossy image codecs can be closely approximated by integer-to-integer wavelets [11]. A performance evaluation of lossy and lossless image compression using integer-to-integer wavelets and the $S+P$ transform of Said and Pearlman [21] [22] is given in [5]. F. J. Hampson and J. C. Pesquet [17] propose a structure which is more general than the lifting scheme with an arbitrary number of channels and arbitrary nonlinear filters. It is interesting to note the simplicity of this structure and the way in which the perfect reconstruction is ensured in an inherent manner by a synthesis filter bank "mirror" of the analysis filter bank, as in the lifting scheme. That structure, with nonlinear prediction filters based on image segmentations has been applied for still image and video coding by I. Amonou and P. Duhamel [6].

About ten years ago, adaptive linear predictions using least mean square (LMS) algorithms were tested for image compression (see [12] and its bibliography), but they were not associated with hierarchical pyramidal decomposition. Hence, they were not suitable for progressive coding in resolution. Ö. N. Gerek and A. E. Çetin [16] use the lifting scheme with adaptive prediction filters. The filter coefficients are updated for each pixel of the image with a conventional stochastic gradient algorithm in order to minimize the variance of the detail signal. In [10], N. V. Boulgouris, D. Tzovaras and M. G. Strintzis express each filter of the optimal M-subband analysis filter bank as a function of the power spectral density (PSD) of the input image. They assume the entire input image is a wide sense stationary (WSS) signal. The optimum is achieved by minimizing the mean square error of prediction for each of the $M-1$ detail signals. Two kinds of parameterized models of the input image PSD are assumed, i.e., the adaptation is optimum if the PSD of the image belongs to a set of two models. The update step is calculated using the equivalence between a conventional filter bank and its lifting implementation, so that the resulting low-pass image is smooth and thus suitable for coding. The update filters do not adapt to the image. They are identical to the ones encountered in the lifting scheme of well-known wavelets. In order to improve the prediction whenever the global WSS assumption is invalid, the linear predictors are enhanced by nonlinear means, namely by directional post-processing in the quincunx case, and by adaptive-length post-processing in the row-column case.

An important functionality in image compression is the possibility of embedded coding [23], because it gives the user a total control of the precision in which the image pixels are represented. This coding allows to reconstruct from a truncated bit flow a decompressed image, which is a rough version of the encoded one. As the data are received, the user is capable of enhancing the image quality or resolution, until reaching the original quality and resolution. These techniques find applications for example in the transmission on a network with a limited or variable bandwidth, and in the consultations of data bases (a minimal version of an image is enough initially for a fast visualization). Hierarchical decompositions will suit naturally to progressive coding, more especially when the successive compressed versions of the image forming the hierarchy are embedded, without redundancy. However, the embedded progressive coding in resolution or in quality is not always allowed by a hierarchical decomposition.

This paper completes the presentations given in [7] and [9]. The two adapted decompositions presented here do not assume any specific condition satisfied by the PSD of the image, in order to achieve the optimum. One decomposition, called globally adapted estimation (GAE), assumes the entire input image is a wide sense stationary signal. On the other hand, the other one, called locally adapted estimation (LAE), assumes that the input image is a locally WSS signal. The reason that led us to carry out this study lies in the fact that the image models are not fully appropriate for entire images, they are better justified for well-chosen parts of the images, taken separately. Those parts are the textured regions that can be found in the most kinds of images. Our intent is to form multiresolution decomposition transforms based on the lifting scheme [24] capable of universally exceptional performance, in the lossless compression framework, for all type of images. In order to achieve this, we present adapted multiresolution decomposition techniques based on LMS estimations.

The methods introduced in this paper are both suitable for an embedded progressive coding. The locally adapted method (LAE) allows only progressive coding in resolution, because the coefficients of the optimal predictor are not transmitted and cannot be recovered at the decoder from a rough version of the lowpass subimage. On the other hand, in the globally adapted method (GAE), the same optimal filter is applied to all the pixels of the subimages. Its coefficients are transmitted to the decoder, and hence progressive coding in resolution and in quality are both possible.

In the next section we present the adapted lifting scheme framework which is shared by both locally and globally adapted estimation methods. In Sections III and IV, the LAE and GAE methods are shown with details. These methods are compared with well-known codecs (SPIHT [21], LOCO I [25], CALIC [26] and JASPER [1]) in Section V. In the following, $\mathbb{Z}$ denotes the set of all integers. For a matrix $\mathbf{A}$, its transpose is written $\mathbf{A}^T$. Underlined lower case letters denote vectors, which are identified with the column matrix of their coordinates.

II. Adapted lifting schemes with truncated LMS prediction

The general principle of the subband decompositions is shown in the block diagram of Fig. 2. The input image I is splitted by decimation into two subimages I_1 and I_2, as in a polyphase transform. The down-sampling can be in quincunx or separable. In the quincunx case, the image I is rotated of $\pi/4$ before the down-sampling, this gives two diamond-shape subimages which are completed with zeros in order to be represented by the matrices I_1 and I_2 (see Fig. 1). In the separable case, the down-sampling is alternatively applied to the rows and to the columns. Each pixel $I_2(i,j)$ is estimated from its semi-causal neighborhood and a non causal one of the pixel $I_1(i,j)$ using the following expression

$$\hat{I}_2(i,j) = \sum_{(h,k)\in\Delta_1} a_{hk} I_1(i-h, j-k) + \sum_{(h,k)\in\Delta_2} b_{hk} I_2(i-h, j-k), \tag{1}$$

where the finite subset $\Delta_1 \subset \mathbb{Z}^2$ is either a rectangle centered in $(0,0)$ (GAE method):

$$\Delta_1 = \left\{(h,k) \in \mathbb{Z}^2 \ : \ |h| \leq p \text{ and } |k| \leq q\right\} \tag{2}$$

or a subset of such a rectangle (LAE method) and the subset $\Delta_2 \subset \mathbb{Z}^2$ is either a finite non-symmetric half-plane (GAE method):

$$\Delta_2 = \left\{ (h,k) \in \mathbb{Z}^2 \ : \ \begin{array}{c} (h = 0 \text{ and } 1 \leq k \leq q) \\ \text{or} \\ (|k| \leq q \text{ and } 1 \leq h \leq p) \end{array} \right\} \tag{3}$$

or a subset of such a half-plane (LAE method, see Fig. 4).

Initially the original image I is replaced by two subimages I_ℓ and I_h defined by the equations:

$$\begin{aligned} I_\ell &= I_1 \\ I_h(i,j) &= I_2(i,j) - \left\lfloor \hat{I}_2(i,j) + 1/2 \right\rfloor, \end{aligned} \tag{4}$$

I_1 I_2

Fig. 1. Polyphase decomposition with a quincunx decimation.

where for a real number x, $\lfloor x \rfloor$ represents the greatest integer smaller or equal to x. Thereafter the same process is applied (exchanging rows and columns in the case of separable decimation) on the subimage I_ℓ in order to obtain the two subimages $I_{\ell\ell}$ and $I_{\ell h}$. The subimage I_h is only subject to the polyphase transform, giving the subimages $I_{h\ell}$ and I_{hh} (see Fig. 2). One level of decomposition consists in the transformation that maps the input image I into the four subimages $I_{\ell\ell}$, $I_{\ell h}$, $I_{h\ell}$ and I_{hh}. By iterating the procedure described above to the lowpass component $I_{\ell\ell}$ in a hierarchical pyramidal scheme, a multiresolution decomposition is formed, consisting of detail coefficients at different scales and a lowpass subimage of very small dimension. In the next two sections, we describe two techniques of nonlinear multiresolution decompositions which adapt to the input image and which are based on truncated linear mean square estimation (LMSE) associated with a lifting scheme [24]. In these methods the filter coefficients a_{hk} and b_{hk} are calculated in order to adapt the best (in the L^2 sense) to the input image. The adaptation is either global [7], i.e., minimizes the total energy, or local [9], i.e., minimizes the current energy in the high-pass image. The proposed decomposition structure given in Fig. 2 is naturally reversible, and no conditions on filters are needed to have a perfect reconstruction. The reconstruction filter bank is straightforwardly found from the analysis filter bank, as it can be seen on Fig. 3.

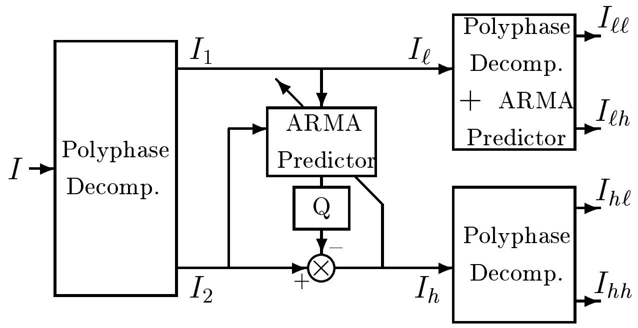

Fig. 2. One level of decomposition of the adapted lifting scheme.

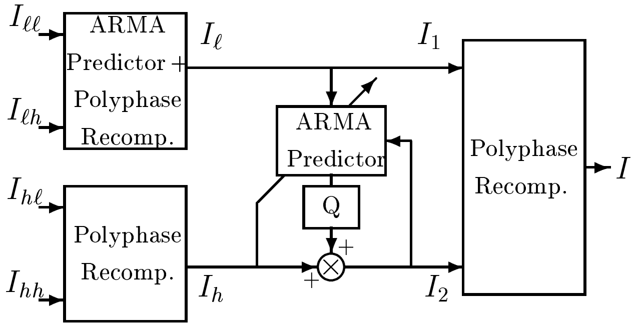

Fig. 3. The perfect reconstruction structure associated with the decomposition described in Fig. 2.

The input image I is an array with M rows and N columns. In the following, M_2 and N_2 denote respectively the numbers of rows and columns of I_2.

III. Locally adapted estimation (LAE) method

In the LAE method, the coefficients a_{hk} and b_{hk} of (1) are updated during the decomposition by the Recursive Least Square (RLS) method. They minimize the current energy

$$W(m,n) = \sum_{i=0}^{m} \sum_{j=0}^{n} \left(I_2(i,j) - \hat{I}_2(i,j) \right)^2 \alpha^{m-i+n-j},$$

with $\alpha = 0.9995$ a forgetting factor, $(0 \leq m < M_2)$ and $(0 \leq n < N_2)$. The estimated image I_2 is scanned in a raster manner, the energy $W(m,n)$ can be written $W(u)$ with $u = mN_2 + n$ varying from 0 to $M_2N_2 - 1$. Hence a one-dimensional RLS algorithm is applied. Let us introduce the vector $\underline{y}_1(i,j)$ (resp. $\underline{y}_2(i,j)$) of dimension r_1 (resp. r_2)

containing the pixels $I_1(i-h,j-k)|_{(h,k)\in\Delta_1}$ (resp. $I_2(i-h,j-k)|_{(h,k)\in\Delta_2}$), reordered as indicated in Fig. 4(a) (resp. in Fig. 4(b)) and where only the r_1 (resp. r_2) first elements are retained. With this reordering, only the r_1+r_2 closest pixels to $I(i,j)$ appear in the observation vector $\underline{y}(i,j)=(\underline{y}_1(i,j)^T,\underline{y}_2(i,j^T))^T$.

Using the coefficients a_{hk} and b_{hk}, the vector $\underline{c}(i,j)$ of the estimation filter coefficients is built in the same manner as the observation vector. In terms of these matrices definitions, the following equation

$$\hat{I}_2(i,j)=\underline{c}(i,j)^T\underline{y}(i,j)$$

is equivalent to (1), where the subsets Δ_1 and Δ_2 are shown in Fig. 4. The subset Δ_1 (resp. Δ_2) contains the r_1 pairs (h,k) of integers which are the closest to $(0,0)$ in $\mathbb{Z}^2$ (resp. the r_2 pairs (h,k) of integer which are the closest of $(0,0)$ in a non-symmetric half plane of $\mathbb{Z}^2$). The optimal filter $\underline{c}(i,j)$, from the least squares point of view, is the solution of the Yule-Walker equations [18] and is given by:

$$\underline{c}(i,j)=\mathbf{\Phi}^{-1}(i,j)\underline{\theta}$$

where $\mathbf{\Phi}$ is the autocorrelation matrix of the observation vector and $\underline{\theta}$ is the cross-correlation vector between the observation vector and the desired response $I_2(i,j)$. This direct solution needs the inversion of the autocorrelation matrix. To avoid performing such an operation we calculate for each (i,j) the least square estimate $\underline{c}(i,j)$ recursively. The adaptive decomposition algorithm is summarized in Table I.

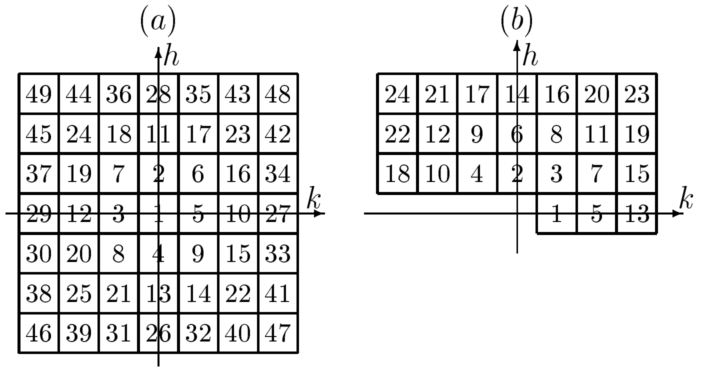

Fig. 4. Order of appearance of the pixels for $p=q\leq 3$ in (a) Δ_1 and (b) Δ_2 (LAE method only).

TABLE I

ESTIMATION AND ADAPTATION ALGORITHM (LAE METHOD).

`Initialisation`

$$\begin{aligned}
\underline{c} &= (1,0,\ldots,0)^T, \quad \textit{with } \underline{c} \in \mathbb{R}^{r_1+r_2} \\
\mathbf{K} &= \delta^{-1}\mathbf{I}, \ \textit{with } \mathbf{I} \textit{ the } r_1+r_2 \textit{ order identity} \\
& \quad \textit{matrix and } \delta \textit{ a small positive constant.} \\
\alpha &= 0.9995, \quad \textit{forgetting factor.}
\end{aligned}$$

`for` $i = 1,2,\ldots,M_2$

`for` $j = 1,2,\ldots,N_2$

$$\begin{aligned}
I_\ell(i,j) &= I_1(i,j) \\
\hat{I}_2(i,j) &= \lfloor \underline{c}^T \underline{y}(i,j) + 1/2 \rfloor \\
I_h(i,j) &= I_2(i,j) - \hat{I}_2(i,j) \\
\underline{u} &= \mathbf{K}\underline{y}(i,j) \\
\mathbf{K} &= \Big(\mathbf{K} - \frac{\underline{u}\,\underline{u}^T}{\alpha + \underline{y}(i,j)^T \underline{u}}\Big)/\alpha \\
\underline{c} &= \underline{c} + \mathbf{K}\underline{y}(i,j) I_h(i,j)
\end{aligned}$$

`end` loop j

`end` loop i

In the LAE method, the use of a forgetting factor α (see Table I) equal to 1 suits well to stationary environments, for which the best steady-state results are obtained. However, when the RLS algorithm operates in a nonstationary environment such as images, the algorithm must use finite memory. A value of α less than 1 is used. By doing so, the algorithm attains the capability to track statistical variations in the environment in which it operates. An experimental value of the forgetting factor α equal to 0.9995 is a good compromise. This value is used in all our simulations. The results of lossless compression of five images are given in Table II. It is clear that the case $\alpha = 0.9995$ performs significantly better than the case $\alpha = 1$ for each of the five images.

When the pixel $I_2(i, j)$ is closed to the boundaries of the subimage I_2, there are three possibilities to complete the observation vector outside of the image: with zeros, or with a symmetric extension or with a periodic extension of the image I_1. A zero completion gives the worst compression results. It is intuitive and it has also been verified in our experiments that the symmetrical extension gives better compression results than periodic extension. This is due to the fact that the symmetric extension does not introduce jump discontinuities in the extended signal. Discontinuities are undesirable, because they increase the amount of high frequency energy in the signal. Lossless compression results for the same five test images are shown in Table II. In general, the results obtained with a symmetric extension are comparable to those obtained with a periodic extension. In Fig. 5, the lossless compression bit rates are plotted versus r_1 and r_2. It can be clearly seen that the use of an ARMA (auto-regressive and moving average) prediction filter ($r_2 \neq 0$) gives a better compression bit rate than an MA (moving average) one ($r_2 = 0$). It can be noticed in Fig. 5 that the compression bit rate decreases when r_1 or r_2 increases. This reduction vanishes when r_1 and r_2 reach their optimal values. These optimal values change according to the type of the image. For natural images, $(r_1, r_2) = (8, 4)$ is a satisfactory choice, while for MRI images, which are generally smooth, a small number of pixels is sufficient to have a good estimation and tests have shown that the value $(r_1, r_2) = (6, 2)$ performs well. The choice of r_1 and r_2 is limited by the computing time and we have found that for each class of images, the value $(r_1, r_2) = (6, 2)$ gives a good compromise. The computational complexity of the LAE method we have implemented is, with $r = r_1 + r_2$,

$(\frac{11r^2+13r}{2})M_2N_2 + o(M_2N_2)$. However, its mathematical formulation and therefore the implementation is relatively simple. This computational complexity can be reduced, without loss of performance, by implementing a fast version of RLS [18], [20]. In LAE method no overhead information is transmitted, and the filter coefficients are calculated one more time by the decoder, hence, the coding and decoding times are roughly the same.

TABLE II

LOSSLESS COMPRESSION RESULTS (IN BIT PER PIXEL bpp) FOR PERIODIC (P.E.) AND SYMMETRIC (S.E.) EXTENSION, AND FOR THREE VALUES OF THE FORGETTING FACTOR α (LAE METHOD).

	P.E.	S.E.		
image \ α	0.9995	0.999	0.9995	1
Lena	4.21	4.22	4.21	4.22
Goldhill	4.72	4.72	4.71	4.74
Barbara	4.54	4.53	4.54	4.65
Irm1	2.89	2.87	2.88	3.18
Finger1	4.23	4.23	4.23	4.29

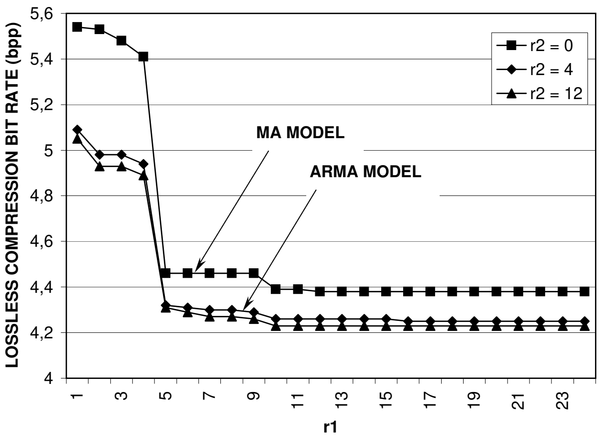

Fig. 5. Lossless compression bit rate versus filter order, obtained with the image *Finger1* of dimension 768×768 (LAE method).

IV. Globally adapted estimation (GAE) method

In the GAE method, the filter coefficients a_{hk} and b_{hk} of (1) are computed in order to minimize the mean square error

$$W = \sum_{i=0}^{M_2-p-1} \sum_{j=0}^{N_2-q-1} \left(I_2(i,j) - \hat{I}_2(i,j) \right)^2, \tag{5}$$

where p and q are defined in relations (2) and (3). In (1) and (5), the subimages I_1 and I_2 are completed by zeros.

Let us introduce the z-transforms $I_2(z_1, z_2)$, $I_1(z_1, z_2)$ and $I_h(z_1, z_2)$ of respectively $I_2(i,j)$, $I_1(i,j)$ and $I_h(i,j)$ and the transfer functions

$$A(z_1, z_2) = \sum_{(h,k)\in\Delta_1} a_{hk} z_1^h z_2^k$$
$$B(z_1, z_2) = \sum_{(h,k)\in\Delta_2} b_{hk} z_1^h z_2^k.$$

If we neglect the nonlinearity (i.e., the function $\lfloor \quad \rfloor$ in relation (4)) we obtain

$$I_h(z_1, z_2) = -A(z_1, z_2) I_1(z_1, z_2) + [1 - B(z_1, z_2)] I_2(z_1, z_2)$$

and hence

$$I_2(z_1, z_2) = \frac{I_h(z_1, z_2) + A(z_1, z_2) I_1(z_1, z_2)}{1 - B(z_1, z_2)}.$$

In order to reconstruct the subimage I_2 from I_1 and I_h, the subset Δ_2 must be chosen in such a way that the filter $\frac{1}{1-B(z_1,z_2)}$ is causal or semi-causal. Moreover, this filter can be unstable and therefore a test of stability is required (see [8]). When the coefficients b_{hk} are adapted to the input image, the filters stability must be tested for each input image and this is rather time consuming. The problem of stability tests disappears by introducing the rounding in relation (4). Indeed, small perturbations do not lead to the divergence of the algorithm, because they vanish for each new sample, due to the rounding.

For image coding application the filter coefficients must be transmitted to the decoder. In order to simplify their transmission, it is better to truncate the digits of their fractional part according to the relation

$$\hat{a}_{hk} = \left\lfloor S * a_{hk} + 1/2 \right\rfloor / S$$

(and a similar relation with b_{hk}), where the scale S is a natural number (in our tests $S = 10^6$). Let $\tilde{I}_2(i,j)$ be the left side of equality (1), when the filter coefficients in the right side are replaced by $\hat{a}_{hk}$ and $\hat{b}_{hk}$. Therefore $\hat{I}_2$ must be replaced by $\tilde{I}_2$ in relation (4) in order to permit perfect reconstruction of I_2 from I_ℓ and I_h:

$$I_2(i,j) = I_h(i,j) + \left\lfloor \tilde{I}_2(i,j) + 1/2 \right\rfloor .$$

Moreover, it is better to apply the decomposition to a zero mean image I, because under this condition the correlation between the pixel $I_h(i,j)$ and the pixels $I_\ell(i-h, j-k)$ in the neighborhood $(h,k) \in \Delta_1$ of $I_\ell(i,j)$ vanishes, when the effects of the non linearity in relation (4) and the difference between $\hat{I}_2$ and $\tilde{I}_2$ are neglected.

We have also implemented nonlinear decompositions nearly identical to the aforementioned one. The difference lies in the introduction of the S-transform [22] in the process, just after the down-sampling and before the filtering: $I_1(i,j)$ and $I_2(i,j)$ are then respectively replaced in (1) and (5), by $I'_1(i,j)$ and $I'_2(i,j)$ such that

$$\begin{aligned} I'_1(i,j) &= \left\lfloor (I_1(i,j) + I_2(i,j))/2 \right\rfloor \\ I'_2(i,j) &= I_1(i,j) - I_2(i,j). \end{aligned}$$

Since the sum and the difference of two integers give either two odd or two even integers, the S-transform is reversible [22]:

$$\begin{aligned} I_1(i,j) &= I'_1(i,j) + \left\lfloor \frac{I'_2(i,j) + 1}{2} \right\rfloor \\ I_2(i,j) &= I_1(i,j) - I'_2(i,j). \end{aligned}$$

In order to perfectly reconstruct the image I from the four subimages $I_{\ell\ell}$, $I_{\ell h}$, $I_{h\ell}$ and I_{hh}, the transmission of a few parameters is required. These parameters are the dimensions M, N of I, the filters orders, the filters coefficients $\hat{a}_{hk}$, $\hat{b}_{hk}$, the type of decimation (separable or quincunx), a Boolean indicating whether the S-transform has been applied or not, the mean of I and the level L of decomposition. These parameters fill an insignificant part of the bit stream.

The step of the GAE method that requires the highest number of operations is the calculation of the Yule-Walker equations associated with the minimization of W in (5).

Hence it is important to reduce the complexity of this step by taking into account the large redundancy between the coefficients of the Yule-Walker equations. We present the detail of the proposed method in Appendix VI. It is an extension of the well-known prewindowing method [15] of linear prediction.

When all the filter coefficients are reordered in a vector $\underline{\alpha}$, the Yule-Walker equations can be written

$$\mathbf{C}\begin{pmatrix} 1 \\ -\underline{\alpha} \end{pmatrix} = \begin{pmatrix} W \\ \underline{0} \end{pmatrix} \quad \text{and} \quad \mathbf{C} = \mathcal{X}^T\mathcal{X}.$$

If no close-to-a-Toeplitz-matrix property is used for matrix $\mathcal{X}$, the computation of the symmetric matrix $\mathbf{C}$ requires

$$\sharp_1 = \omega(\omega+1)M_2N_2 + o(M_2N_2)$$

arithmetic operations, where $\omega = 6pq + 3p + 3q + 2$ is the order of $\mathbf{C}$. On the other hand, this complexity can be reduced to

$$\sharp_2 = 2(24pq + 6p + 8q + 3)M_2N_2 + o(M_2N_2)$$

operations (see Table III), by using the small displacement rank of $\mathcal{X}$.

TABLE III

COMPLEXITY OF THE COMPUTATION OF YULE-WALKER EQUATIONS, WHEN: ($\sharp_1$) NO PROPERTY OF THE SYSTEM IS USED, ($\sharp_2$) SMALL DISTANCE TO TOEPLITZ IS USED. FOR $i = 1, 2$, THE NUMBER OF OPERATIONS REQUIRED IS $\sharp_i M_2N_2 + o(M_2N_2)$ (GAE METHOD).

$(\sharp_1, \sharp_2)$	$q = 1$	$q = 2$	$q = 3$
$p = 1$	$(210, 82)$	$(552, 146)$	$(1056, 210))$
$p = 2$	$(552, 142)$	$(1482, 254)$	$(2862, 366)$
$p = 3$	$(1056, 202)$	$(2862, 362)$	$(5550, 522)$

In order to better fit the orders (p, q) of the filters (a_{hk}) and (b_{hk}), a number of tests on MRI medical images of dimension $512 \times 512 \times 8$ and on rectangular and square satellite images of various dimensions were performed. The best results have been achieved by adapting the orders (p, q) to the dimension of the low resolution subimage and to the class of the image (either satellite or MRI or natural). As it could be expected, we noticed that the more the level of decomposition increases, the less the pixels of the low resolution subimage are correlated, and adapting the filters coefficients becomes less efficient.

Moreover, we have noticed that the S transform is not efficient on low resolution images of dimension greater than 256×256. For low resolution subimages of small dimension, the estimate $\hat{I}_2(i,j)$ in the relation (1) is replaced by a nonlinear approximation of $I_1(i,j)$ close to the transformed image of the first encoder in the LAR method introduced by Déforges and Ronsin [14].

V. Comparative evaluations in lossless image compression

In this section we compare the performance of the above mentioned adapted lifting schemes with other well-known lossless codecs, which are LOCO-I [2] (of the ISO standard JPEG-LS), CALIC [3], SPIHT [4] and JASPER [1] (compatible with ISO standard JPEG2000). LOCO-I and CALIC adapt each to the input image using context-based prediction, however they do not allow embedded (progressive) coding. On the other hand, SPIHT and JASPER do not adapt to the input image, nevertheless they allow embedded coding, both in resolution and quality, like GAE. On the other hand, LAE permits embedded coding only in resolution.

In order to obtain a complete lossless image compression system, the adapted lifting schemes LAE and GAE have been combined with a SPIHT-like coder. In Sections III and IV, we have indicated that the efficiency of the proposed methods depends on several parameters like decimation (separable or quincunx), length of prediction filters, extension of image around the boundaries, etc. The optimal values of these parameters are used in all our simulations. Moreover, the bit rates obtained with the SPIHT codec are the smallest among the two possibilities (smooth image or not) offered by the software of Said and Pearlman [4].

The principal drawback of the adapted lifting scheme with truncated LMS is its computational complexity (see Table V). In LAE the coding and decoding times are roughly the same, but in GAE the decoding time is significantly smaller than the coding one. As expected, LOCO-I (which has been designed with the aim of obtaining a maximal coding and decoding speed) is the faster. However, among those performing progressive coding, JASPER and SPIHT have coding times close to each other, and about ten times smaller than LAE and GAE. In decoding, GAE is nearly as fast as JASPER and SPIHT, which are roughly ten times faster than the implemented version of LAE. Actually, no computation optimization has been performed on the present version of LAE. The straightforward implementation of a fast RLS [18] [20] would significantly lower the computational burden.

Using the average of the compression ratios over a large number of images, with various types and sizes, as a performance measure, we could not escape the too famous rule which says: "All lossless compression techniques have more or less similar performance, and there is never one which is distinctly better than the others for all images". In order to give an accurate assessment to the proposed methods, we have tested the codecs on various classes of images : photographic, medical MRI, satellite[1] and textures. All the images are coded with 8 bits per pixel. Table IV shows compression results obtained with software implementations of the above described algorithms. The "Average" row associated with photographic images has been obtained with a set of 14 images of dimensions 512×512 and 1024×1024. The one associated with medical MRI images has been obtained with a set of 16 images of dimension 512×512. For satellite images, the average has been computed from a set of 14 square or rectangular images of various dimensions. For fingerprints and textures, the average has been calculated on a set of 11 images of dimensions 512×512 and 768×768. For the set of photographic images it can be noticed that CALIC gives the smallest average bit rate and that the LAE method gives an average bit rate slightly smaller than GAE, SPIHT and LOCO-I. JASPER is behind the other coders with an increase of ~ 0.1 bpp for the bit rate. On the other hand, it achieves a significantly smaller bit rate for MRI images. This is due to the smoothness of the MRI images, and the $(5, 3)$ wavelet transform used in JASPER, because having more vanishing analyzing moments performs better when applied to smooth images [11]. Its performance suffers for images containing many edges, which is the case of the three other images classes. For satellite images, CALIC provides better results in comparison with the other methods. However, among the coders performing progressive coding, LAE comes in first position followed by GAE. CALIC and proposed methods, applied to texture and fingerprint images, have very close results to each other, with better compression results against the other coders.

[1]Satellite images have been given as a favor by the French National Center of Spatial Studies (CNES) and by the society named SPOT Image.

TABLE IV

COMPARATIVE EVALUATION OF LOSSLESS COMPRESSION RESULTS (IN bpp) OBTAINED WITH SPIHT, LOCO-I, CALIC, JASPER AND THE PROPOSED METHODS LAE AND GAE. THE "AVERAGE" ROWS ARE OBTAINED BY CLASSES OF IMAGES.

Image	LOCO	CALIC	LAE	JASP	SPIHT	GAE
Lena	4.24	4.13	4.21	4.32	4.17	4.25
Goldhill	4.71	4.65	4.75	4.84	4.75	4.78
Barbara	4.74	4.51	4.59	4.66	4.53	4.60
Mandrill	6.04	5.90	5.98	6.11	5.93	5.99
Peppers	4.49	4.39	4.56	4.62	4.54	4.57
Airplane	4.61	4.47	4.46	4.62	4.50	4.45
Airport	5.32	5.22	5.24	5.48	5.32	5.27
Average	4.82	4.70	4.77	4.91	4.78	4.78
Image	LOCO	CALIC	LAE	JASP	SPIHT	GAE
MRI1	2.27	2.20	2.40	2.58	2.41	2.41
MRI2	2.54	2.34	2.13	1.69	2.59	2.24
MRI3	5.27	5.09	5.17	5.22	5.04	5.02
Average	2.81	2.65	2.54	2.40	2.86	2.62
Image	LOCO	CALIC	LAE	JASP	SPIHT	GAE
Sat1	3.81	3.72	3.85	4.01	3.89	3.88
Sat2	4.26	4.03	4.01	4.24	3.85	3.76
Sat3	4.41	4.28	4.28	4.44	4.31	4.26
Average	4.70	4.53	4.61	4.85	4.69	4.67
Image	LOCO	CALIC	LAE	JASP	SPIHT	GAE
Fing.1	4.57	4.44	4.24	4.46	4.33	4.25
Text1	6.71	6.62	6.46	6.79	6.53	6.48
Text2	5.97	5.88	5.93	6.18	5.95	5.91
Average	4.69	4.58	4.57	4.75	4.62	4.59

TABLE V

AVERAGE TIME AT THE CODING AND THE DECODING STEP EXPRESSED IN SECONDS. TIMING ON PIII 700 MHz, 256 Mb RAM FOR IMAGES OF DIMENSION $512 \times 512 \times 8$.

time	LOCO	CALIC	BBA	JASP	SPIHT	BBG
coding	0.12	0.21	3.27	0.40	0.36	3.19
decoding	0.11	0.26	3.20	0.37	0.35	0.46

VI. Conclusion

In this paper we have introduced two adapted lifting schemes with truncated linear mean square estimation. The idea behind this work is to design subband coders capable of universally performance, in the progressive lossless compression framework, for all type of images. Predictors can be adapted to the local features of the image (method LAE), or to the whole image (method GAE). An experimental study and a comparison of compression results with other well known methods have been presented. The proposed methods fulfill the requirement of universality for progressive lossless compression. They lead to an average slightly better compression bit rate than other codecs permitting progressive coding, with the trade-off of an increase in the computational complexity.

Appendix A

For two vectors $\underline{u} \in \mathbb{R}^\alpha$ and $\underline{v} \in \mathbb{R}^\beta$, let $\mathbf{T}[\underline{u}, \underline{v}]$ be the Toeplitz matrix of dimension $\alpha \times \beta$ whose first column (resp. row) is equal to $\underline{u}$ (resp. $\underline{v}^T$). For a matrix $\mathbf{A}$ of dimension $m \times n$, let $[\mathbf{A}]_{i,j}$ be the coefficient on the $(i+1)$th row and the $(j+1)$th column ($0 \leq i < m$ and $0 \leq j < n$). Let us introduce the vector $\underline{a}_0 = (0, \ldots, 0, -1, a_{0,1}, a_{0,2}, \ldots, a_{0,q})^T$ of dimension $2q+1$, the p vectors $\underline{a}_k = (a_{k,-q}, a_{k,-q+1}, \ldots, a_{k,q})^T$ ($1 \leq k \leq p$) of dimension $2q+1$, the $2p+1$ vectors $\underline{b}_k = (b_{k,-q}, b_{k,-q+1}, \ldots, b_{k,q})^T$ ($-p \leq k \leq p$) of dimension $2q+1$ and the vectors $\underline{a} = (\underline{a}_0^T, \underline{a}_1^T, \ldots, \underline{a}_p^T)^T$ and $\underline{b} = (\underline{b}_{-p}^T, \underline{b}_{-p+1}^T, \ldots, \underline{b}_p^T)^T$.

With the $(m+1)$th row of the subimage I_2, we associate the $p+1$ Toeplitz matrices ($0 \leq k \leq p$)

$$\begin{aligned}\mathbf{X}_k(m) = \mathbf{T}[&(I_2(m-k, q), \ldots, I_2(m-k, N_2-1))^T, \\ &(I_2(m-k, q), \ldots, I_2(m-k, 0), 0, \ldots, 0)^T]\end{aligned} \tag{6}$$

of dimension $(N_2 - q) \times (2q+1)$ and the block matrix

$$\mathbf{X}(m) = [\ \mathbf{X}_0(m) \quad \mathbf{X}_1(m) \quad \cdots \quad \mathbf{X}_p(m)\ ] \tag{7}$$

of dimension $(N_2 - q) \times (2q+1)(p+1)$. In a similar way, with the $(m+1)$th row of the subimage I_1 we associate the $2p+1$ Toeplitz matrices $(-p \leq k \leq p)$

$$\begin{aligned}\mathbf{Y}_k(m) \;=\; & \mathbf{T}[(I_1(m-k,q),\ldots,I_1(m-k,N_2-1))^T, \\ & (I_1(m-k,q),\ldots,I_1(m-k,0),0,\ldots,0)^T]\end{aligned} \tag{8}$$

of dimension $(N_2-q)\times(2q+1)$ and the block matrix

$$\mathbf{Y}(m) = [\;\mathbf{Y}_{-p}(m) \quad \mathbf{Y}_{-p+1}(m) \quad \cdots \quad \mathbf{Y}_p(m)\;]$$

of dimension $(N_2-q)\times(2p+1)(2q+1)$. With these notations, the estimation error

$$\begin{aligned}\underline{e}(m) \;=\; & (I_2(m,0)-\hat{I}_2(m,0),\ldots, \\ & I_2(m,N_2-q-1)-\hat{I}_2(m,N_2-q-1))^T\end{aligned}$$

associated with the $(m+1)$th row of I_2 meets the relation $-\underline{e}(m) = \mathbf{X}(m)\,\underline{a} + \mathbf{Y}(m)\,\underline{b}$. Moreover, it is clear from (6) and (8) that $\mathbf{Y}_k(m) = \mathbf{Y}_{k-1}(m-1) = \mathbf{Y}_0(m-k)$, $\mathbf{X}_k(m) = \mathbf{Y}_k(m) = \mathbf{0}$ (if $m<k$) and, for $0<k\leq p$ and $0\leq m< M_2-p-1$,

$$\mathbf{X}_k(m) = \mathbf{X}_{k-1}(m-1) = \mathbf{X}_0(m-k), \tag{9}$$

Hence the estimation equations give

$$-(\underline{e}(0)^T,\underline{e}(1)^T,\ldots,\underline{e}(M_2-p-1)^T)^T = \mathbf{X}\,\underline{a} + \mathbf{Y}\,\underline{b}, \tag{10}$$

with the block Toeplitz matrices $\mathbf{X}$ and $\mathbf{Y}$, whose first columns are respectively $(\mathbf{X}_0(0)^T$, $\mathbf{X}_0(1)^T,\ldots,\mathbf{X}_0(M_2-p-1)^T)^T$ and $(\mathbf{Y}_0(p)^T,\mathbf{Y}_0(p+1)^T,\ldots,\mathbf{Y}_0(M_2-1)^T)^T$ and first rows are respectively $(\mathbf{X}_0(0),\mathbf{0},\ldots,\mathbf{0})$ and $(\mathbf{Y}_0(p),\mathbf{Y}_0(p-1),\ldots,\mathbf{Y}_0(0),\mathbf{0},\ldots,\mathbf{0})$. Since any block $\mathbf{X}_0(k)$ or $\mathbf{Y}_0(k)$ is Toeplitz, let us remark that $\mathbf{X}$ and $\mathbf{Y}$ are each Toeplitz and block Toeplitz at once. The filter coefficients that minimize the energy W in (5) are solutions of the Yule-Walker equations [18] which can be written

$$(\underline{a}^T,\underline{b}^T)\mathbf{C} = (\underline{0}^T,-W,\underline{0}^T),$$

where

$$\mathbf{C} = \begin{bmatrix}\mathbf{X} & \mathbf{Y}\end{bmatrix}^T\begin{bmatrix}\mathbf{X} & \mathbf{Y}\end{bmatrix} = \begin{bmatrix}\mathbf{X}^T\mathbf{X} & \mathbf{X}^T\mathbf{Y} \\ \mathbf{Y}^T\mathbf{X} & \mathbf{Y}^T\mathbf{Y}\end{bmatrix} \tag{11}$$

is a symmetric matrix of order $\omega' = (2q+1)(3p+2)+1$. Since the $p+1$ first components of $\underline{a}$ are known, the real dimension of the system is $\omega = 6pq+3p+3q+2$. Nevertheless it is faster to compute first the entire matrix $\mathbf{C}$ and then extract the system matrix than to compute it straight. The relation (10) differs from the well-known prewindowing prediction equations [15] [18] because of the added term $\mathbf{Y}\underline{b}$. Therefore the displacement rank of the matrix $\mathbf{C}$ is not equal to 1 [15]. However, the coefficients of $\mathbf{C}$ have a great redundancy and the reasoning of [19] can be adapted.

In order to be brief, we just give the relations that permit the computation of the block $\mathbf{X}^T\mathbf{X}$ in $\mathbf{C}$. The same reasoning can be done with the other blocks. It results from (10) and (7) that

$$\mathbf{X}^T\mathbf{X} = \sum_{m=0}^{M_2-p-1} \mathbf{X}(m)^T\mathbf{X}(m) = \left[\sum_{m=0}^{M_2-p-1} \mathbf{X}_k(m)^T\mathbf{X}_\ell(m)\right]$$

(for $0 \le k, \ell < p$), where the last expression is a block representation of $\mathbf{X}^T\mathbf{X}$. Let

$$\mathbf{T}_{k,\ell} = \sum_{m=0}^{M_2-p-1} \mathbf{X}_k(m)^T\mathbf{X}_\ell(m) \qquad (0 \le k, \ell < p)$$

be the block of dimension $(2q+1)\times(2q+1)$. One has $\mathbf{T}_{k,\ell}^T = \mathbf{T}_{\ell,k}$. The relation (9) leads to

$$\mathbf{T}_{k,\ell} = \mathbf{T}_{k-1,\ell-1} - \mathbf{X}_0(M_2-p-k)^T\mathbf{X}_0(M_2-p-\ell)$$

for $1 \le k \le \ell \le p$. It can be easily deduced from (6) that $\mathbf{X}_0(m-k)^T\mathbf{X}_0(m-\ell)$ ($0 \le k \le \ell \le p$ and $\ell \le m < M_2-p$) has a displacement rank of 2. hence, for $0 \le \ell \le p$, it comes

$$\begin{aligned}[\mathbf{T}_{0,\ell}]_{i,j} &= [\mathbf{T}_{0,\ell}]_{i-1,j-1} + \sum_{m=\ell}^{M_2-p-1} I_2(m,q-i)I_2(m-\ell,q-j) \\ &\quad - \sum_{m=\ell}^{M_2-p-1} I_2(m,N_2-i)I_2(m-\ell,N_2-j).\end{aligned}$$

Therefore the calculus of a block requires (in additions and multiplications):

- $O(N_2)$ operations for $\mathbf{T}_{k,\ell}$ with $0 \le k \le \ell \le p$,
- $2(4q+1)M_2N_2 + o(M_2N_2)$ operations for $\mathbf{T}_{0,\ell}$ with $p \ge \ell \ge 1$,
- $2(2q+1)M_2N_2 + o(M_2N_2)$ operations for $\mathbf{T}_{0,0}$,

and the computation of $\mathbf{X}^T\mathbf{X}$ requires $2[(4q+1)p+2q+1]M_2N_2 + o(M_2N_2)$ operations. In a same way, the computation of $\mathbf{X}^T\mathbf{Y}$ requires $2(4q+1)(3p+1)+o(M_2N_2)$ operations and the one of $\mathbf{Y}^T\mathbf{Y}$ requires $2[(4q+1)2p+2q+1]M_2N_2+o(M_2N_2)$ operations. Therefore the computation of $\mathbf{C}$ requires $2[2(4q+1)(3p+1)+1]M_2N_2+o(M_2N_2)$ operations.

References

[1] `http://www.ece.ubc.ca/~mdadams/jasper/`

[2] `http://www.hpl.hp.com/loco/software.htm`

[3] `http://www.csd.uwo.ca/faculty/wu/`

[4] `http://www.cipr.rpi.edu/research/SPIHT/`

[5] M. D. Adams and F. Kossentini, "Reversible Integer-to-Integer Wavelet Transforms for Image Compression: Performance Evaluation and Analysis", *IEEE Trans. Image Processing*, vol. 9, no. 6, pp. 1010-1024, Jun. 2000.

[6] I. Amonou and P. Duhamel, "Nonredundant representation of images allowing object- based and multi-resolution scalable coding", in *Proc. Int. Conf. Visual Communications and Image Processing*, pp. 598–608, Perth, Australia, Jun. 2000.

[7] M. Barret and H. Bekkouche, "Adapted nonlinear multiresolution decomposition with applications in progressive lossless image coding", in *Proc. Int. Symp. Image and Signal Processing and Analysis*, pp. 609–613, Pula, Croatia, Jun. 2001.

[8] M. Barret and M. Benidir, "Behaviour of stability tests for two-dimensional digital recursive filters when faced with rounding error", *IEEE Trans. Circuits and Systems (II)*, vol. 44, no. 4, pp. 319–323, Apr. 1997.

[9] H. Bekkouche and M. Barret, "Adaptive multiresolution decomposition: application to lossless image compression", in *Proc. Int. Conf. Acoustic, Speech and Signal Processing*, Orlando, Florida, USA, May 2002.

[10] N. V. Boulgouris, D. Tzovaras and M. G. Strintzis, "Lossless image compression based on optimal prediction, adaptive lifting, and conditional arithmetic coding", *IEEE Trans. Image Processing*, vol. 10, no. 1, pp. 1–14, Jan. 2001.

[11] A. R. Calderbank, I. Daubechies, W. Sweldens and B. L. Yeo, "Wavelet transforms that map integers to integers", *Appl. Comput. Harmon. Anal.*, vol. 3, no. 2, pp. 186–200, 1996. *IEEE Trans. Image Processing*, vol. 10, no. 1, pp. 1–14, Jan. 2001.

[12] Y.-S. Chung and M. Kanefsky, "On 2D recursive LMS algorithms using ARMA prediction for ADPCM encoding of Images", *IEEE Trans. Image Processing*, vol. 1, no. 3, pp. 416–422, Jul. 1992.

[13] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps", *J. Fourier Anal. Appl.*, vol. 4, no. 3, pp. 247–269, 1998.

[14] O. Déforges and J. Ronsin, "Locally Adaptive Resolution method for progressive still image coding", in *Proc. Int. Symp. Signal Processing and its Applications*, Brisbane, Australia, 22-25 Aug. 1999.

[15] B. Friedlander, M. Morf, T. Kailath and L. Ljung, "New inversion formulas for matrices classified in terms of their distance from Toeplitz matrices", *Linear Algebra and its Applications*, vol. 27, pp. 31–60, 1979.

[16] Ö. N. Gerek, A. E. Çetin, "Adaptive polyphase subband decomposition structures for image compression", *IEEE Trans. Image Processing*, vol. 9, no. 10, pp. 1649–1660, Oct. 2000.

[17] F. J. Hampson and J. C. Pesquet, "M-band nonlinear subband decomposition with perfect decomposition", *IEEE Trans. Image Processing*, vol. 7, no. 11, pp. 1547–1560, Nov. 1998.

[18] Haykin S., *Adaptive filter theory*, Engelwood Cliffs, NJ: Prentice-Hall, second edition 1991.

[19] T. Kailath, S. Y. Kung and M. Morf, "Displacement ranks of a matrix", *Bull. Amer. Math. Soc.*, vol. 1, no. 5, pp. 769–773, Sep. 1979.

[20] X. Liu and M. Najim, "A two-dimensional fast lattice recursive least squares algorithm", *IEEE Trans. Signal Processing*, vol. 44, no. 10, Oct. 1996.

[21] A. Said and W. A. Pearlman, "A new, fast and efficient image codec based on set partitioning in hierarchical trees", *IEEE Trans. Circuit and System for Video Technomogy*, vol. 6, no. 3, pp. 243–250, Jun. 1996.

[22] A. Said and W. A. Pearlman, "An image multiresolution representation for lossless and lossy compression", *IEEE Trans. Image Processing*, vol. 5, pp. 1303–1310, Sep. 1996.

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

[24] W. Sweldens, "The lifting Scheme: A new phylosophy in biorthogonal wavelet construction", in *Proc. SPIE*, vol 2569, pp. 68–78, Sept. 1995.

[25] M. J. Weinberger, G. Seroussi and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS", *IEEE Trans. Image Processing*, vol 9, pp. 1309–1324, Aug. 2000.

[26] X. Wu and N. Memon, "Context-based, Adaptive, Lossless Image Coding", *IEEE Trans. Communication*, vol. 45, no. 4, pp. 437–444, Apr. 1997.