Fast Adaptive Wavelet Packet Image Compression

François G. Meyer, *Member, IEEE*, Amir Z. Averbuch, and Jan-Olov Strömberg

***Abstract*—Wavelets are ill-suited to represent oscillatory patterns: rapid variations of intensity can only be described by the small scale wavelet coefficients, which are often quantized to zero, even at high bit rates. Our goal in this paper is to provide a fast numerical implementation of the best wavelet packet algorithm [1] in order to demonstrate that an advantage can be gained by constructing a basis adapted to a target image. Emphasis in this paper has been placed on developing algorithms that are computationally efficient. We developed a new fast two-dimensional (2-D) convolution-decimation algorithm with factorized nonseparable 2-D filters. The algorithm is four times faster than a standard convolution-decimation. An extensive evaluation of the algorithm was performed on a large class of textured images. Because of its ability to reproduce textures so well, the wavelet packet coder significantly out performs one of the best wavelet coder [2] on images such as Barbara and fingerprints, both visually and in term of PSNR.**

***Index Terms*—Adaptive transform, best basis, image compression, ladder structure, wavelet packet.**

I. Introduction

WAVELETS with many vanishing moments are very effective for coding piece-wise smooth images [2]–[4]. Wavelets, however, are ill suited to represent oscillatory patterns. Oscillatory variations of intensity can only be described by the small scale wavelet coefficients. Unfortunately, those small scale coefficients carry very little energy, and are often quantized to zero, even at high bit rates. Fingerprints, or seismic signals are few examples of non-wavelet-friendly signals. Much larger libraries of functions, called wavelet packets, have been constructed [1] to address this problem. Because the collection of wavelet packets is overcomplete (there are many more basis functions than the dimension of the input space) one can construct a basis that is fitted for a target image (or for a class of images). In this work, a good basis should be able to describe the target image with a very small number of basis vectors. The metric proposed by Coifman and Wickerhauser [1] to compare bases is of little practical use. A more meaningful measure considers the number of bits needed to approximate the image with a given error. Ramchandran and Vetterli [5] followed this path, and wedded the bit allocation algorithm of Shoham and Gersho [6] to the best basis algorithm [1]. Unfortunately this approach (and its variation [7]) is extremely computationally intensive (as explained in Section III-B, the problem in [5] involves three layers of nonlinear approximations, only one of which lends itself to a fast algorithm). Very little work has been expended beyond the papers [1], [5], and as a result adapted wavelet packet bases remain a theoretical curiosity, with no clear practical advantage, and that cannot be computed within a reasonable amount of time. Our goal in this paper is to provide a fast numerical implementation of the best wavelet packet algorithm, in order to demonstrate that an advantage can be gained by constructing a basis adapted to a target image. Emphasis in this paper has been placed on developing algorithms that are computationally efficient. Three original contributions result from this work:

1) a new fast 2-D convolution-decimation algorithm with factorized nonseparable 2-D filters; the number of operations is reduced by a factor 4 in comparison to a standard implementation, and the image need not be transposed;
2) a cost function that takes into account the cost of coding the output levels of the quantizers, and the cost of coding the significance map;
3) a context-based entropy coder that conditions the probability of significance of a given pixel on the probability of its neighbors using a space filling curve.

The compression algorithm is divided into three parts. In the first part, we select that wavelet packet basis which is best adapted to encode the image. During the second part, the wavelet packet coefficients of the image are quantized; we assume a Laplacian distribution and we use an efficient near optimal scalar quantizer [8]. Finally, the significance map is entropy coded using a higher order arithmetic coder [9] that relies on a context consisting of pixels in a causal neighborhood. This paper is organized as follows. In Section II, we review the wavelet packet library. In Section III, we explain how to select, among a large collection of bases, that basis which is best adapted to encode a given image. The factorization of the conjugate quadrature filters is described in Section IV. In Section V, we describe the quantization, and the context-based entropy coding. Results of experiments are presented in Section VI.

II. Wavelet Packet Library

Let $\boldsymbol{h} = \{h_n\}$ and $\boldsymbol{g} = \{g_n\}$ be biorthogonal filters. Let ψ^0 be the scaling function and ψ^1 be the wavelet associated with $\{h_n\}$, $\{g_n\}$. Let us define the basic wavelet packets

$$\begin{aligned}\psi^{2n}(x) &= \sum_k h_k \psi^n(2x-k)\\ \psi^{2n+1}(x) &= \sum_k g_k \psi^n(2x-k). \qquad (1)\end{aligned}$$

Manuscript received April 17, 1998; revised August 17, 1999. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Christine Podilchuk.

F. G. Meyer is with the Department of Electrical Engineering, University of Colorado, Boulder, CO 80309-0425 USA, on leave from Yale University, New Haven, CT 06520 USA (e-mail: francois.meyer@colorado.edu).

A. Z. Averbuch is with the Department of Computer Science, School of Mathematical Sciences, Tel Aviv University, Tel Aviv, Israel.

J.-O. Strömberg is with the Department of Mathematics, Royal Institute of Technology, Stockholm, Sweden.

Publisher Item Identifier S 1057-7149(00)03862-8.

1057–7149/00$10.00 © 2000 IEEE

A multiscale wavelet packet: $\psi^n_{j,\,l} = \psi^n(2^j x - k)$ is located at $l2^{-j}$, has a support of size 2^{-j}, and roughly n oscillations (a frequency $2^j n$). Let us associate the dyadic frequency interval $[2^j n,\, 2^j(n+1))$ to the wavelet packet $\psi^n_{j,\,l}$. If a collection of intervals $\{[2^j n,\, 2^j(n+1))\}$ provides a cover of the time-frequency plane, then the set $\{\psi^n_{j,\,l}\}$ is an orthonormal basis [1]. If we consider dyadic subdivisions at each level, we get 2^{4^L} bases for L levels. Clearly, we have an extremely large amount of freedom for the construction of orthogonal bases from the wavelet packet library $\{\psi^n_{j,\,l}\}$. This greater flexibility is exploited to increase the efficiency of the representation.

III. Best Basis Algorithm

A. Fast Dynamic Programming Approach

Coifman and Wickerhauser [1] suggested to use a fast dynamic programming algorithm with order $N\,\log(N)$ (N is the number of pixels in the image) to search for that *best basis* which is optimal according to a given cost function $\mathcal{M}$. A key criterion must be met in order to invoke a dynamic programming strategy [1], [10]. The cost-function should be separable: $\mathcal{M}(\boldsymbol{x}) = \sum_k \mu(x_k)$ where $\{x_k,\, k = 0, \cdots, N-1\}$ are the wavelet packet coefficients of $\boldsymbol{x}$, and μ is a positive function such that $\mu(0) = 0$.

B. Choice of a Cost Function $\mathcal{M}$

Initially, Coifman and Wickerhauser [1] used $h(\boldsymbol{x}) = -\sum_k |x_k|^2/\|\boldsymbol{x}\|^2 \log |x_k|^2/\|\boldsymbol{x}\|^2$ as a cost function. It is important to realize that $h(\boldsymbol{x})$ bears no connection with the entropy $\mathcal{H}(\boldsymbol{x})$ of the probability distribution of the $\{x_k\}$. For instance, if all x_k are equal, then $h(\boldsymbol{x})$ is maximal, but the entropy of the distribution $\mathcal{H}(\boldsymbol{x})$ is minimal. More importantly, we have noticed that in practice $h(\boldsymbol{x})$ is usually not capable of discovering any meaningful basis. Ramchandran and Vetterli [5] used the optimal bit allocation algorithm of Shoham and Gersho [6] to distribute the budget across the nodes of the wavelet packet tree, and they select the best basis B according to the rate distortion criterion

$$\mathcal{M}(\boldsymbol{x},\, Q,\, \lambda) = D(\boldsymbol{x},\, Q) + \lambda R(\boldsymbol{x},\, Q). \tag{2}$$

Given a set of quantizers Q, the rate $R(\boldsymbol{x},\, Q)$ is estimated with the first order approximation of the entropy $\mathcal{H}(Q(\boldsymbol{x}))$, and the distortion $R(\boldsymbol{x},\, Q)$ is defined as the mean square error. The selection of the best basis involves three embedded nonlinear optimization problems. The overall complexity of the approach in [5] is therefore extremely high. Incidentally, a theoretical problem with the cost function (2) is that it is not additive: the mean square error is not additive (the l^2 error is), and the entropy $\mathcal{H}(\boldsymbol{x})$ is not additive: if $\boldsymbol{x}$, and $\boldsymbol{y} = \{y_k\}$ are the children of a node in the wavelet packet tree, then we have [11]: $\mathcal{H}(\boldsymbol{x};\, \boldsymbol{y}) = \mathcal{H}(\boldsymbol{x}) + \mathcal{H}(\boldsymbol{y}) - \mathcal{I}(\boldsymbol{x};\, \boldsymbol{y})$ where $\mathcal{I}(\boldsymbol{x};\, \boldsymbol{y})$ is the mutual information [11], a measure of the amount of information that $\boldsymbol{x}$ contains about $\boldsymbol{y}$. Because the subband $\boldsymbol{x}$, $\boldsymbol{y}$ are not independent (this is in fact the tenet of the zero-tree based coding algorithms), $\mathcal{I}(\boldsymbol{x};\, \boldsymbol{y})$ is usually not zero. Finally, we note that the results published in [5] correspond to hypothetical compression rates, since the first order entropy was chosen to measure the rate.

Instead of using the rate distortion framework, we designed a cost function that mimics the actual scalar quantization and entropy coding (see Section V). It is composed of two complementary terms:

- $c_1(\boldsymbol{x})$, the cost of coding the sign and the magnitude of the nonzero output levels of the scalar quantizer, Q;
- $c_2(\boldsymbol{x})$, the cost of coding the locations of the nonzero output levels (significance map).

A first order approximation of the cost $c_1(\boldsymbol{x})$ of coding the magnitude of the output levels $\{|Q(x_k)|\}$ is given by

$$c_1(\boldsymbol{x}) = \sum_{k/Q(x_k)\neq 0} \max(\log_2 |Q(x_k)|,\, 0). \tag{3}$$

A fast implementation of c_1 can be devised using the standard representation of floating numbers. The second term, c_2, is given by the first order entropy of a Bernoulli process

$$c_2(\boldsymbol{x}) = -N\,(p\,\log_2(p) + (1-p)\,\log_2(1-p)). \tag{4}$$

Each coefficient x_k is significant with a probability p, and the significance of the coefficients are independent events (a more accurate model would take into account the local spatial correlations, as well as the correlations across subbands). The computation of the cost function requires to quantize the coefficients, and we use the scalar quantizer described in Section V with an initial estimate of the quantization step. After a first compression, this estimate can be refined in order to improve the result.

IV. Fast Convolution-Decimation

In any practical situation, the best basis algorithm is only applicable if the computation of the coefficients at each node of the wavelet packet tree does not require an absurdly large amount of time. This section presents an efficient numerical algorithm for computing multidimensional convolution-decimation that divides the number of operations by a factor 2 in 1-D, and a factor 4 in 2-D.

A. Factoring the Biorthogonal Filters

Several methods [12] have been proposed in the literature for the fast implementation of convolution and decimation. FFT implementations [12] are useful for large filters (of length 64 or 128), but do not bring any improvement over a straightforward implementation for short and medium size filter [12]. Recently, several authors [13]–[16] have proposed efficient implementations of 1-D biorthogonal filters using a factorization of the filters into smaller filters. In [15], the authors show that all biorthogonal filters can be factored into a sequence of *ladder* steps. A ladder step maps every couple of even and odd samples $(x_{2k},\, x_{2k+1})$ into $(y_{2k},\, y_{2k+1})$

$$\begin{aligned} y_{2k} &= x_{2k} + L(x_{2k+1}) \\ y_{2k+1} &= x_{2k+1}. \end{aligned} \tag{5}$$

The proof of the factorization of 1-D biorthogonal filters into ladder steps [15] relies on the use of the Euclidean algorithm to

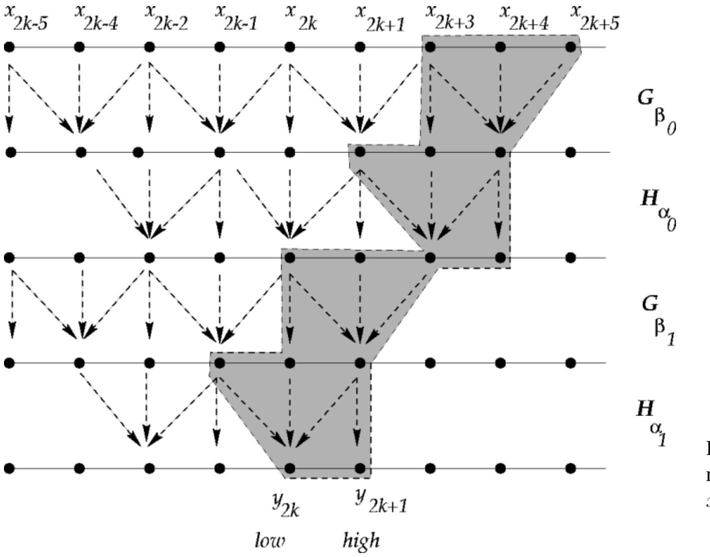


Fig. 1. Four ladder steps (shown in the shaded regions) are needed to compute the lowpass and highpass filtered coefficients $y_{2k},\, y_{2k+1}$.

factor polynomials. In [13] the ladder step is a *lifting step*, and similar results were derived using the same methods. A more direct proof, based on matrix factorization, was obtained in [17], [16]. We now introduce the notations and recall the key results. Let $\boldsymbol{h}$, $\boldsymbol{g}$ be any symmetric biorthogonal filters of length $2m+1$. We merge $\boldsymbol{h}$ and $\boldsymbol{g}$ into an orthogonal band diagonal operator $\boldsymbol{W}$ where the only nonzero coefficients are

$$W_{2k,\,2k+1+j} = h_j, \quad W_{2k+1,\,2k+1+j} = g_j \qquad j = -m, \cdots, m. \tag{6}$$

Let $\boldsymbol{H}_\alpha$ be the lowpass ladder step: $\boldsymbol{H}_\alpha$ is a band diagonal matrix, with nonzero elements given by

$$H_{2i,\,2i-1} = \alpha \quad H_{2i,\,2i} = 1 \quad H_{2i,\,2i+1} = \alpha \tag{7}$$

and let $\boldsymbol{G}_\beta$ be the highpass ladder step: $\boldsymbol{G}_\beta$ is a band diagonal matrix, with nonzero elements given by

$$G_{2i,\,2i+2} = \alpha \quad G_{2i+1,\,2i} = \alpha \quad G_{2i+1,\,2i+1} = 1. \tag{8}$$

One has the following result [13], [15], [17].

Lemma 1: $\boldsymbol{W}$ can be factored into at most m operators

$$\boldsymbol{W} = \boldsymbol{H}_{\alpha_m}\boldsymbol{G}_{\beta_m}\boldsymbol{H}_{\alpha_{m-1}}\boldsymbol{G}_{\beta_{m-1}} \cdots \boldsymbol{H}_{\alpha_0}\boldsymbol{G}_{\beta_0}. \tag{9}$$

The result extends to orthogonal filters [13], [17]. We note that the computational complexity of each ladder step (7) or (8) is three operations per two samples.

1) Fast Convolution-Decimation: A fast algorithm for convolution-decimation implements the ladder steps (7) and (8). After the calculation of the first lowpass and highpass coefficients $(y_0,\, y_1)^T = (\boldsymbol{h},\, \boldsymbol{g})(x_0,\, x_1)^t$, each new lowpass and highpass coefficients $(y_{2k},\, y_{2k+1})$ only require a cascade of m ladder steps (7) and (8) (see Fig. 1). The complexity of the computation of $(y_{2k},\, y_{2k+1})$ is therefore $3m$. We compare this result with a standard implementation that requires $8m+2$ operations. The fast algorithm divides the number of operations by a factor $((8m+2)/3m) > 2.67$.

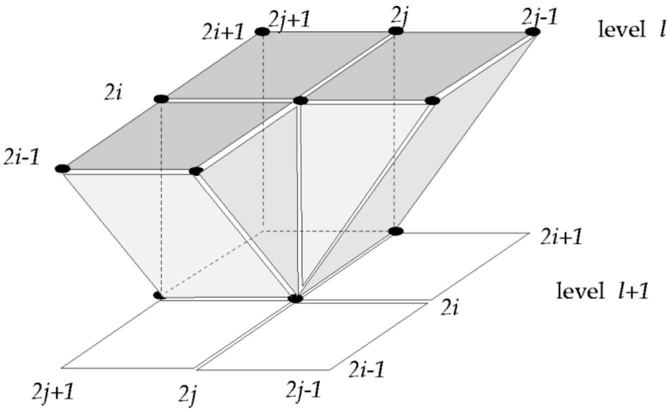


Fig. 2. Two-dimensional ladder step $\boldsymbol{H}_{2,\,\alpha_l}$. Points in the shaded regions at level l are the input to the computation of $x^{l+1}_{2i,\,2j}$, $x^{l+1}_{2i+1,\,2j}$, $x^{l+1}_{2i,\,2j+1}$, $x^{l+1}_{2i+1,\,2j+1}$.

B. Two-Dimensional Case

In 2-D, one could use the 1-D factorization to process independently rows and columns. Since a regular convolution-decimation requires $4(4m+1)(m+1)$ operations to obtain the output of the four filters: $\boldsymbol{hh}$, $\boldsymbol{hg}$, $\boldsymbol{gh}$, $\boldsymbol{gg}$ for any input sample $(x_i,\, x_j)$, this approach would yield a speed up factor of $(4(4m+1)(m+1))/(3m(2m+3))$. For a filter of size 9, the speed up factor is at most 2.57. Fortunately one can significantly improve this result by merging the horizontal, and vertical factorizations. This new algorithm provides a gain of a factor 4, and our numerical experiments validated this result. This algorithm is a new and original contribution of the present work.

Let $\boldsymbol{W}_2 = \boldsymbol{W}_x\boldsymbol{W}_y$ be the 2-D convolution-decimation operator, where $\boldsymbol{W}_x$, and $\boldsymbol{W}_y$ are the biorthogonal filters that are applied along the rows an columns, respectively. We factorize $\boldsymbol{W}_x$ and $\boldsymbol{W}_y$ as in (9)

$$\boldsymbol{W}_2 = \prod_{l=1}^{m}\prod_{k=1}^{m} \boldsymbol{H}_{y,\,\alpha_l}\boldsymbol{G}_{y,\,\beta_l}\boldsymbol{H}_{x,\,\alpha_k}\boldsymbol{G}_{x,\,\beta_k}. \tag{10}$$

Because the terms commute, we combine the lowpass [highpass] filters for both directions into a 2-D filter

$$\boldsymbol{H}_{2,\,\alpha_l} = \boldsymbol{H}_{y,\,\alpha_l}\boldsymbol{H}_{x,\,\alpha_l} \quad \boldsymbol{G}_{2,\,\beta_l} = \boldsymbol{G}_{y,\,\beta_l}\boldsymbol{G}_{x,\,\beta_l}. \tag{11}$$

We obtain a factorization of $\boldsymbol{W}_2$ similar to (9)

$$\boldsymbol{W}_2 = \prod_{l=1}^{m} \boldsymbol{H}_{2,\,\alpha_l}\boldsymbol{G}_{2,\,\beta_l}. \tag{12}$$

This decomposition suggests a 2-D ladder structure. The lowpass ladder step $\boldsymbol{H}_{2,\,\alpha_l}$ is given by

$$\begin{aligned} x^{l+1}_{2i+1,\,2j+1} &= x^{l}_{2i+1,\,2j+1} \\ x^{l+1}_{2i,\,2j+1} &= x^{l}_{2i,\,2j+1} + \alpha_l(x^{l}_{2i-1,\,2j+1} + x^{l}_{2i+1,\,2j+1}) \\ x^{l+1}_{2i+1,\,2j} &= x^{l}_{2i+1,\,2j} + \alpha_l(x^{l}_{2i+1,\,2j-1} + x^{l}_{2i+1,\,2j+1}) \\ x^{l+1}_{2i,2j} &= x^{l}_{2i,\,2j} + \alpha_l(x^{l}_{2i,\,2j-1} + x^{l}_{2i,\,2j+1} \\ &\qquad + x^{l+1}_{2i-1,\,2j} + x^{l+1}_{2i+1,\,2j}) \end{aligned} \tag{13}$$

where x^l come from the grid at level l, and $x^{l+1}_{2i-1,2j}$, $x^{l+1}_{2i+1,\,2j}$ have already been calculated and stored (see Fig. 2). Similarly, the highpass ladder step $\boldsymbol{G}_{2,\beta_l}$ is given by

$$\begin{aligned} x^{l+1}_{2i,2j} &= x^l_{2i,2j} \\ x^{l+1}_{2i,2j+1} &= x^l_{2i,2j+1} + \beta_l(x^l_{2i,2j} + x^l_{2i,2j+2}) \\ x^{l+1}_{2i+1,2j} &= x^l_{2i+1,2j} + \beta_l(x^l_{2i,2j} + x^l_{2i+2,2j}) \\ x^{l+1}_{2i+1,2j+1} &= x^l_{2i+1,2j+1} + \beta_l(x^l_{2i+1,2j} + x^l_{2i+1,2j+2} \\ &\quad + x^{l+1}_{2i,2j+1} + x^{l+1}_{2i+2,2j+1}). \end{aligned} \tag{14}$$

The number of operations necessary to compute a 2-D ladder step (13), or (14), is eight additions and three multiplications. After the first coefficients, $\boldsymbol{hh}(x_0, y_0)^T$, $\boldsymbol{hg}(x_0, y_0)^T$, $\boldsymbol{gh}(x_0, y_0)^T$, $\boldsymbol{gg}(x_0, y_0)^T$, have been computed, the computation of each new set of 4 coefficients only requires to apply at a small number of positions a sequence of m alternating lowpass and highpass ladder steps (see Fig. 2). It is easy to verify that the total number of operations needed to compute the output of the four filters: $\boldsymbol{hh}$, $\boldsymbol{hg}$, $\boldsymbol{gh}$, $\boldsymbol{gg}$ for any input sample (x_i, x_j) is $(11m(m/2+1)/2)$ if m is even, and $11(m+1)^2/4$ otherwise. In the case of a 9-7 filter we get a theoretical computational gain of 4.54. Another major advantage of the 2-D ladder structure is that it does not require to transpose the image; a benefit that is even more important in 3-D.

1) Experimental Validation: We report in Table I the average processor time needed for computing the convolution-decimation of an entire image using both a regular implementation, and the ladder structure. The processing was performed on a standard Pentium, with no particular optimization. The theoretical speed up factor is reached for images of size 1024×1024. As the image size decreases, the speed up factor becomes 3.67. No other existing algorithm would permit to achieve comparable speed up for short filters.

TABLE I
AVERAGE PROCESSOR TIME NEEDED FOR COMPUTING THE CONVOLUTION-DECIMATION ON AN ENTIRE IMAGE

Image size	Convolution-decimation	Ladder	speed up
256 × 256	110 ms	30 ms	3.67
512 × 512	477 ms	130 ms	3.67
1024 × 1024	2146 ms	536 ms	4.0

V. QUANTIZATION AND ENTROPY CODING

A. Laplacian Based Scalar Quantization

Within each subband the distribution of the wavelet packet coefficients is approximated to a Laplacian distribution [18]. We use a particularly efficient scalar quantizer, with a performance that is very close to the performance of the optimal entropy constrained scalar quantizer for Laplacian distribution [8]. It is a uniform scalar quantizer, with a symmetric dead-zone, $[-\Delta+\delta, \Delta-\delta]$, and a step size Δ. The optimal (for mean square error) reconstruction offset is $\delta = 1 - \Delta((e^{-\Delta})/(1-e^{-\Delta}))$. Finally, we apply a dichotomic search to find the optimal value of Δ that exhausts the budget.

B. Ordering of the Coefficients and Entropy Coding

After quantization, the positions of the nonzero output levels are recorded in a significance map. The lossless compression of the significance map takes advantage of the fact that large output levels often appear in clusters. If one uses a wavelet basis, then one can also take advantage of the self similar structure of subbands [19], using zero-trees [2], [3], [20]. Unfortunately, the wavelet packet coefficients fail to exhibit a self similar structure. This is due to the fact that, in principle, if the basis is well "adapted" to a given image $\boldsymbol{f}$, then it is unlikely that the coefficients $\langle \psi^n_{j,l}, \boldsymbol{f} \rangle$ at a given position l, are significant at all the scales j. In other words, the coefficients of a salient feature in the image will only appear at a single scale j that characterizes the feature. Another technical difficulty [21] comes from the fact that we usually cannot define the "parent" subband in a general wavelet packet tree. Because of all these issues, we limit the context to be a spatial context. Subbands are scanned by increasing frequency. We then scan all the pixels inside any given subband using a Hilbert space filling curve. This self similar curve guarantees that 1) all pixels in an area are visited before moving on to another area, and 2) the sequentially visited pixels are always adjacent in the 2-D image plane. The spatial context is then defined as the n_C pixels that appear before the current pixel in the Hilbert scan. In the experiments we use $n_C = 3$. This efficient context modeling permits to condition the probability of significance of a given pixel on the probability of significance of its neighbors using a n_C order arithmetic coder.

VI. EXPERIMENTS

We implemented the fast wavelet packet (FWP) coder and decoder, and an actual bit stream is generated by the coder. In accordance with Clærbout's "reproducible research" philosophy [22], we provide the software for reproducing all experiments in this paper. The FWP code is available online from http://ece-www.colorado.edu/~meyer/profile.html. We present the results of the wavelet packet compression algorithm on four 512×512 images: "Barbara," "fingerprints," "houses," and "lighthouse." These images are all difficult to compress because they contain a mixture of large smooth regions, and long oscillatory patterns. We compared our coder to one of the best wavelet coder that was available to us: the SPIHT wavelet coder of Said and Pearlman [2]. A comparison with other wavelet coders (e.g.[3], [4]) would result in different but comparable results. The performance of the algorithm is summarized in Table II. It took 1564 ms to calculate all the coefficients of a six-level full wavelet packet tree of the image Barbara, to calculate the cost of ach node, and to prune the tree. The quantization then took another 6899 ms (compression ratio: 32). Most of the time during the quantization is spent on the dichotomic search for the optimal value of Δ. All computations were performed on a regular Pentium, running Linux.

A. Artifacts Created by Wavelet Packets

In general the quantization of a wavelet coefficient $\langle \boldsymbol{x}, \psi_k \rangle$ has the following effect: it will add on the original image the

TABLE II
RATE DISTORTION PERFORMANCES

Barbara

Rate (bpp)	1	0.5	0.4	0.25	0.2	0.125
SPIHT	36.41	31.39	30.10	21.58	26.65	24.86
FWP	37.24	32.82	31.53	29.12	28.11	26.22

Fingerprints

Rate (bpp)	1	0.5	0.4	0.25	0.2	0.125
SPIHT	36.01	31.27	29.91	27.12	26.00	23.97
FWP	36.85	32.79	30.43	27.59	26.80	24.71

Houses

Rate (bpp)	1	0.5	0.4	0.25	0.2	0.125
SPIHT	30.84	26.15	25.06	23.17	22.33	20.98
FWP	30.64	26.48	25.39	23.41	22.59	21.11

Lighthouse

Rate (bpp)	1	0.5	0.4	0.25	0.2	0.125
SPIHT	34.03	30.25	29.29	27.43	26.58	24.98
FWP	34.06	30.44	29.55	27.98	27.28	25.86

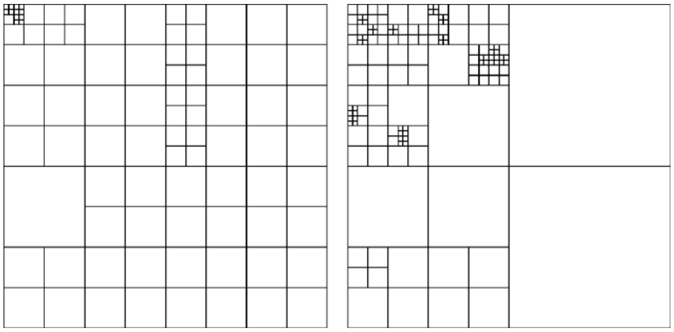


Fig. 3. Best basis geometry: left: Barbara and right: fingerprints.

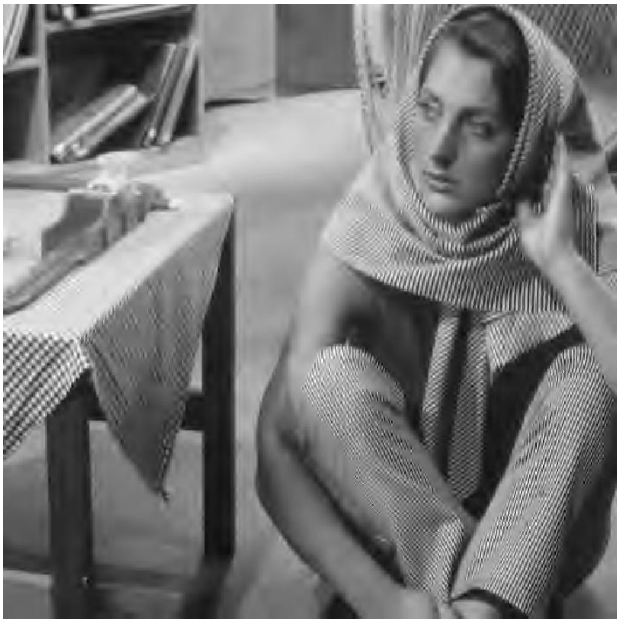


Fig. 4. SPIHT: decoded Barbara, 0.25 bpp, PSNR = 27.58 dB.

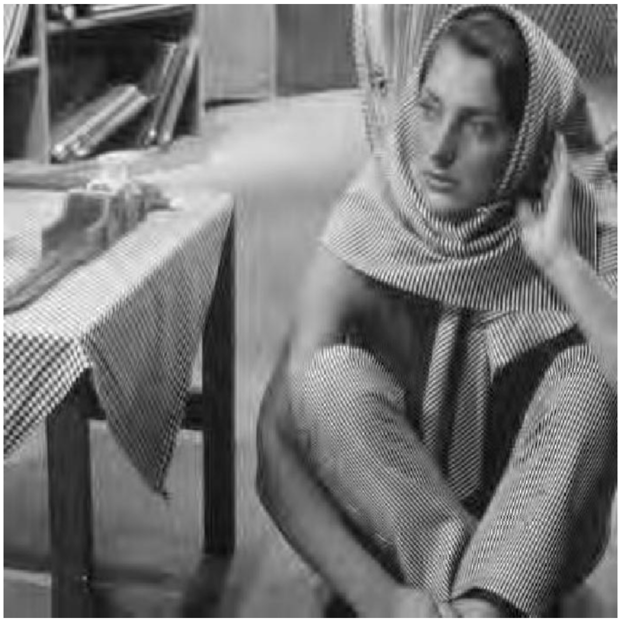


Fig. 5. Decoded Barbara using FWP, 0.25 bpp, PSNR = 29.12 dB.

vector $(Q(\langle \boldsymbol{x}, \psi_k\rangle) - \langle \boldsymbol{x}, \psi_k\rangle)\psi_k$. The size of the artifacts thus depends on the size of the support of the function ψ_k. Large wavelet coefficients occur around edges. If the quantization of these coefficients only affects fine scale wavelets (large j), then the ringing artifacts will be localized. The case of wavelet packets is more complicated. A wavelet packet basis usually contains many functions similar to sine or cosine functions (very well localized in frequency), and as result the fine scale wavelets may no longer be part of the basis. Edges need then to be reconstructed from oscillatory wavelet packets. The quantization of these coefficients will consequently introduce Gibbs phenomenon, that will affect large regions around edges. Our implementation addresses this problem by preventing further splits of the higher frequency bands below a certain scale.

B. Barbara

The smaller boxes in the central column of the best basis, shown in Fig. 3 left, correspond to large patterns oscillating in the horizontal direction. These basis functions obviously match the texture on the scarf, and on the legs of Barbara, as well as the checker pattern on the tablecloth. Because the basis is well fitted for the image, the FWP coder has no difficulty preserving the oscillatory texture everywhere in the mage (see Fig. 6), whereas SPIHT entirely fails to represent the texture in these regions. As mentioned in Section VI-A, ringing artifacts are visible at sharp edges (around the legs of the table, and the elbow of Barbara; see Figs. 4 and 5), for both SPIHT and FWP. While artifacts created by wavelets and wavelet packets have similar intensity, artifacts created by wavelet packets affect more pixels around the edge. Nevertheless, because of its ability to reproduce the texture so well, FWP still significantly out performs SPIHT, by 1.14 dB on the average.

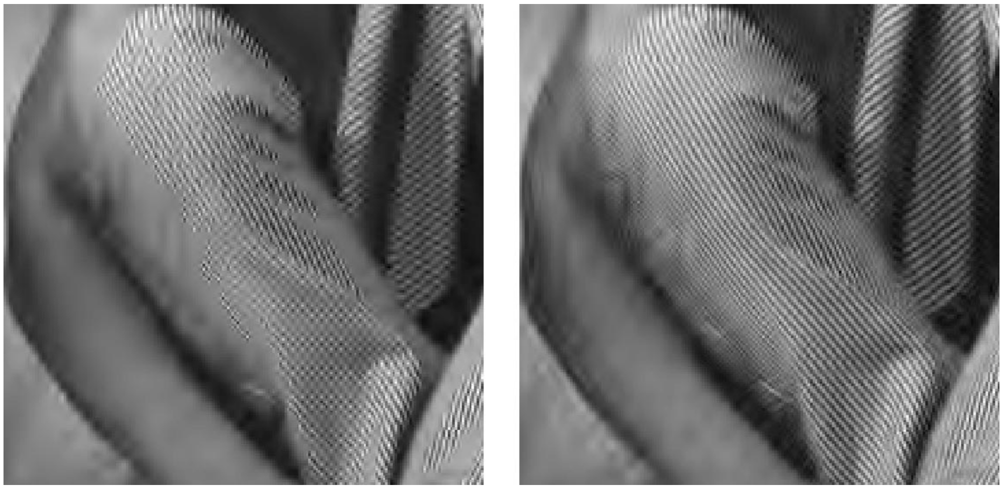


Fig. 6. Magnified detail: SPIHT: left, FWP: right, 0.25 bpp.

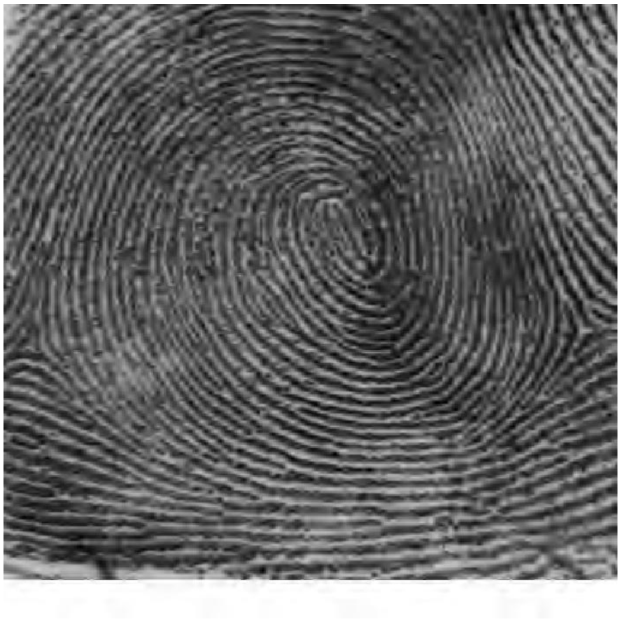


Fig. 7. SPIHT: decoded fingerprints, 0.20 bpp, PSNR = 26.00 dB.

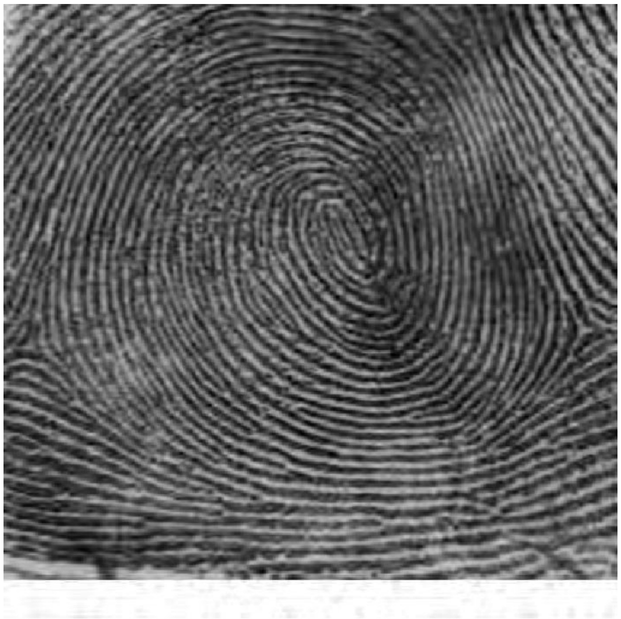


Fig. 8. FWP: decoded fingerprints, 0.20 bpp, PSNR = 26.80 dB.

1) Fingerprints: The best basis, shown in Fig. 3 right, contains large patterns oscillating in the vertical, horizontal, and diagonal directions, with a frequency that match the frequency of the concentric circular patterns. Unfortunately, wavelet packets can only provide “criss-cross” patterns: a better basis should contain steerable filters [23]. Fig. 7 shows the result of a compression of 32 using SPIHT, and Fig. 8 shows the result of FWP at the same compression rate. We notice in Fig. 9 that the FWP image is much crisper than the SPIHT image. FWP also out performs SPIHT in terms of PSNR.

2) Houses: Small boxes on the first row and first column of the best basis (shown in Fig. 10 left) correspond to the many horizontal or vertical oscillating patterns that are present in the image. Fig. 11 shows the result of a compression of 25, using SPIHT, and Fig. 12 is the result of FWP at the same compression rate. We notice in Fig. 14 that FWP has kept all the details on the shutters, as well as the texture on the roof. All these details have been erased by SPIHT (see Fig. 13). Ringing artifacts are visible on the left border of the central house, where the intensity abruptly changes. While similar artifacts are visible for wavelets, the artifacts have a larger extent for the wavelet packets.

3) Lighthouse: Fig. 15 shows the result of a compression of 40 using SPIHT, and Fig. 16 shows the result of FWP at the same compression rate. Again, the best basis (shown in Fig. 10) is selecting many basis functions that correspond to horizontal or vertical oscillating patterns. A detailed view, shown in Fig. 17, demonstrates that the wavelet packet coder has better preserved the texture on the lighthouse, and has not smeared the fence. Again, artifacts on the limb of the lighthouse are clearly noticeable both for SPIHT (see Fig. 17), and FWP (see Fig. 18). While FWP out performs SPIHT in terms of PSNR at low bit rates, SPIHT performs as poorly as FWP in terms of ringing artifacts.

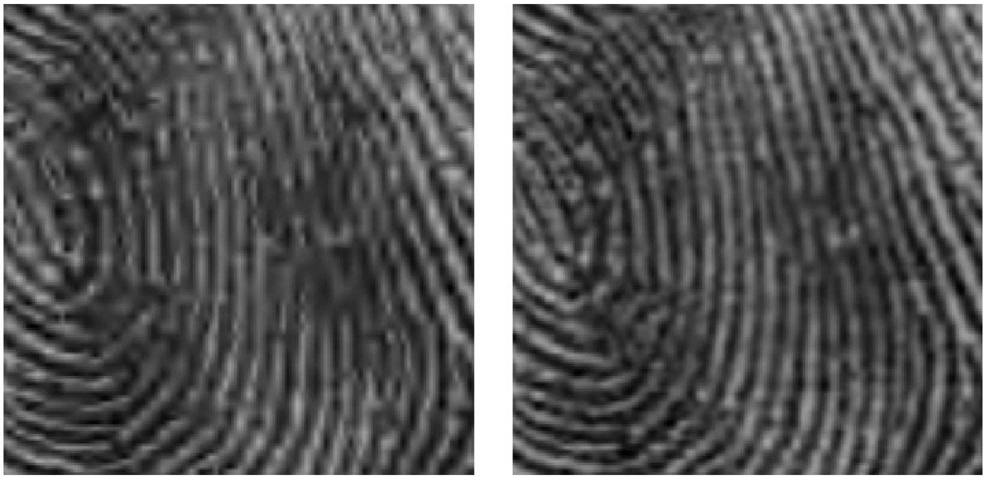


Fig. 9. Magnified detail: SPIHT: left, FWP: right, 0.20 bpp.

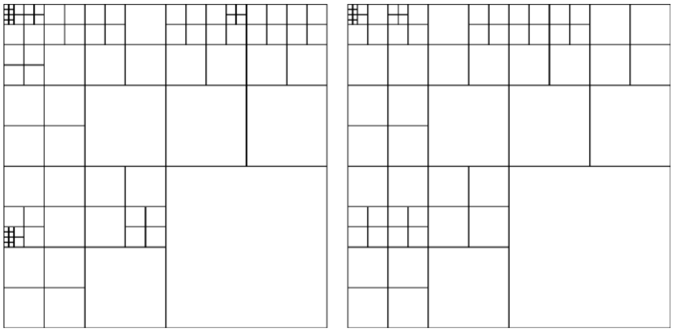


Fig. 10. Best basis: left: houses, right: lighthouse.

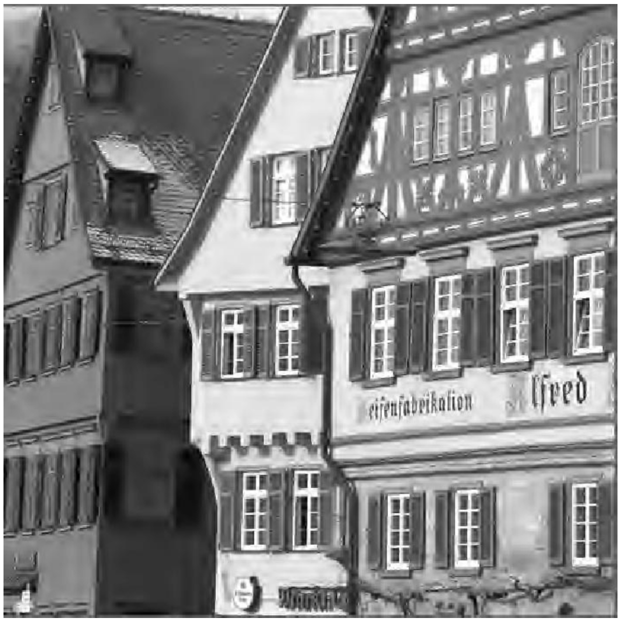


Fig. 11. SPIHT: decoded houses, 0.32 bpp, PSNR = 24.17 dB.

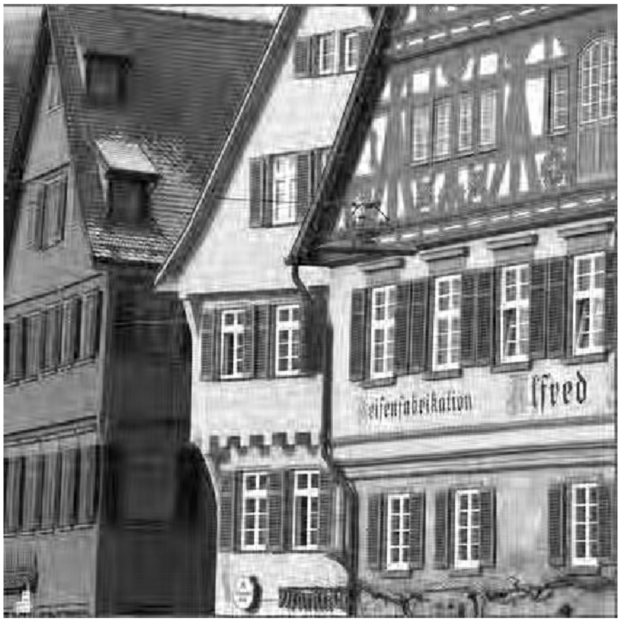


Fig. 12. FWP: decoded houses, 0.32 bpp, PSNR = 24.40 dB.

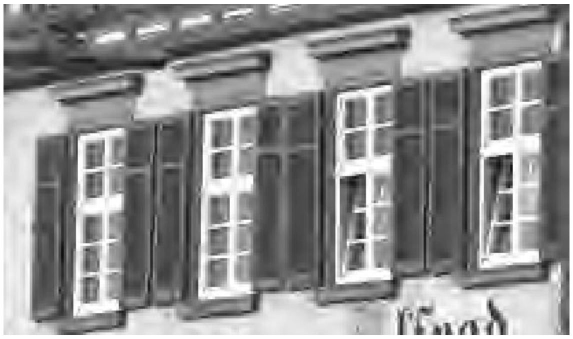


Fig. 13. SPIHT: magnified detail of decoded houses, 0.32 bpp.

VII. Discussion and Conclusion

This work provides a fast numerical implementation of the best wavelet packet algorithm, and demonstrates that an advantage can be gained by constructing a basis adapted to a target image without requiring an absurdly large amount of time. We designed a fast wavelet packet coder that, combined with a simple quantization scheme, could significantly outperform a sophisticated wavelet coder, with a negligible increase in computational load. Our evaluation, performed on a large class of textured images, included not only quantitative figures (PSNR), but also subjective visual appearance. On one hand,

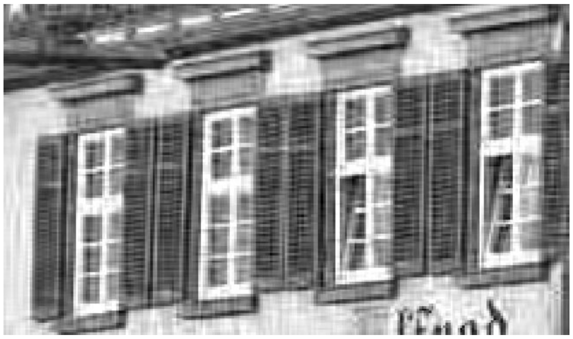


Fig. 14. FWP: magnified detail of decoded houses, 0.32 bpp.

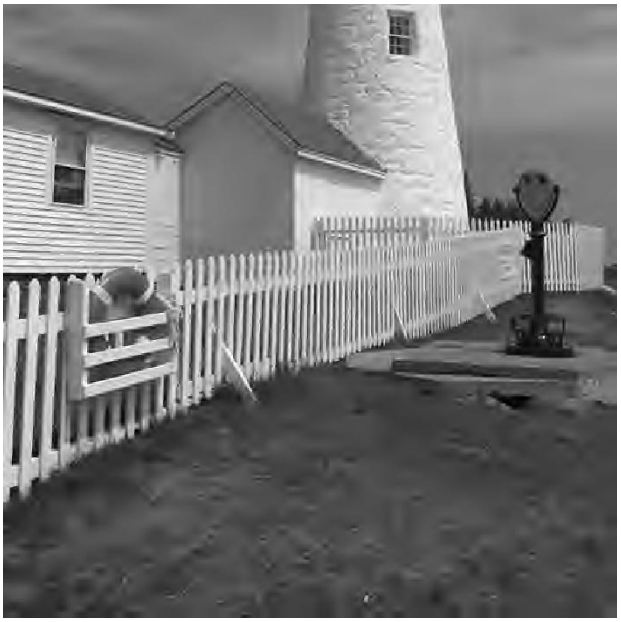


Fig. 15. SPIHT: decoded lighthouse, 0.20 bpp, PSNR = 26.58 dB.

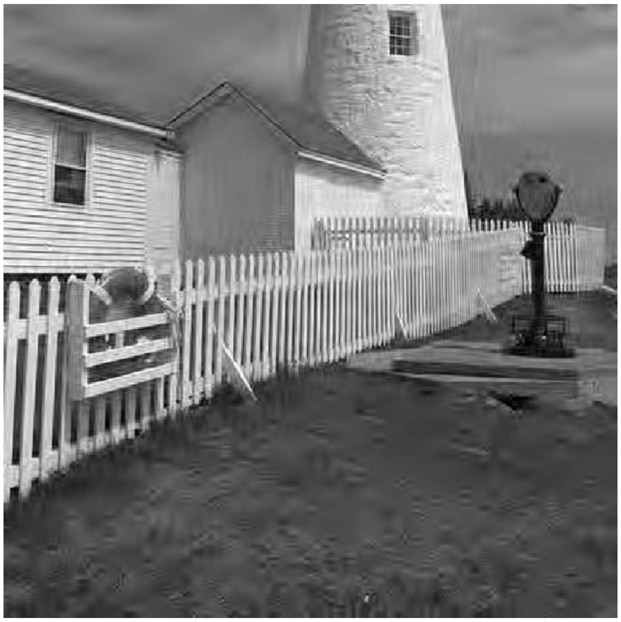


Fig. 16. FWP: decoded lighthouse, 0.20 bpp, PSNR = 27.28 dB.

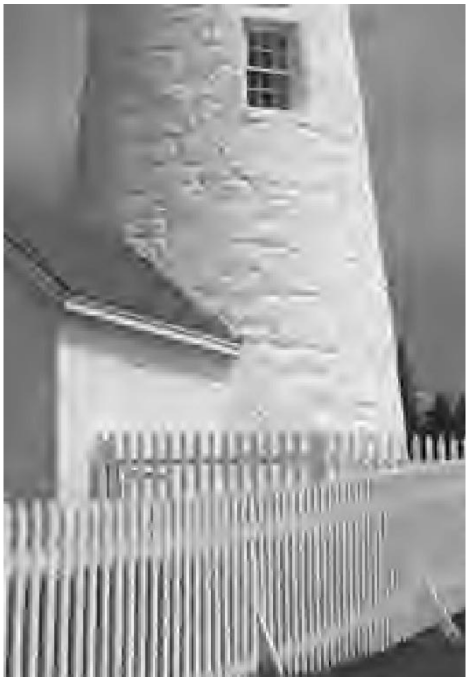


Fig. 17. SPIHT: magnified detail of decoded lighthouse, 0.20 bpp.

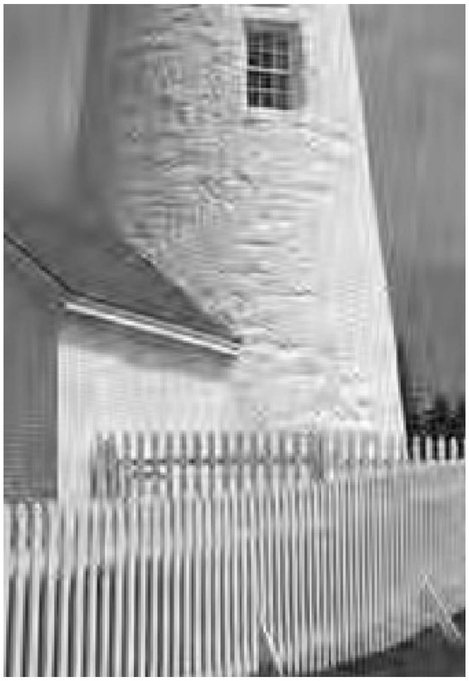


Fig. 18. FWP: magnified detail of decoded lighthouse, 0.20 bpp.

our results indicate that our wavelet packet coder tends to create artifacts at the same locations (i.e., at strong edges) as a wavelet coder does, with a similar intensity. The main difference is that artifacts created by wavelet packets affect more pixels, than those created by wavelets. On the other hand, the basis selected by the algorithm is usually well adapted to the target image, and the wavelet packet coder has no difficulty preserving the oscillatory textures. Because of its ability to reproduce textures so well, the FWP coder significantly out performs SPIHT on images such as Barbara, and fingerprints, both visually and in term of PSNR. A number of open interesting problems have been raised by this work. We realized that when coding images that contain a mixture of smooth and textured features, the best basis algorithm is always trying to find a compromise between two conflicting goals: —describe the large scale smooth regions and edges, and describe the oscillatory patterns. As a result the best basis may not always yield "visually pleasant" images; and we noticed ringing artifacts on the border of smooth regions when the basis is mostly composed of oscillatory patterns. Addressing this problem may require replacing an orthonormal basis with a collection of functions selected from a large dictionary, that can include wavelets and oscillatory waveforms [24], [25]. In our very recent work [26] we explore yet another approach. We propose to encode an image with a multilayered representation technique, based on a cascade of compressions, using at each time a different basis.

REFERENCES

[1] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Trans. Inform. Theory*, vol. 38, pp. 713–718, Mar. 1992.

[2] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on set partioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243–250, 1996.

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

[4] P. Sriram and M. W. Marcellin, "Image coding using wavelet transforms and entropy-constrained trellis quantization," *IEEE Trans. Image Processing*, vol. 4, pp. 725–733, 1995.

[5] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense," *IEEE Trans. Image Processing*, vol. 2, pp. 160–175, Apr. 1993.

[6] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 1445–1453, Sept. 1988.

[7] Z. Xiong, K. Ramchandran, and M. T. Orchard, "Wavelet packets coding using space-frequency quantization," *IEEE Trans. Image Processing*, vol. 7, pp. 892–898, June 1998.

[8] G. J. Sullivan, "Efficient scalar quantization of exponential and Laplacian random variables," *IEEE Trans. Inform. Theory*, vol. 42, no. 5, pp. 1365–1374, Sept. 1996.

[9] M. Nelson and J.-L. Gailly, *The Data Compression Book*. New York: M&T, 1996.

[10] L. Cooper and M. W. Cooper, *Introduction to Dynamic Programming*. New York: Pergamon, 1988.

[11] T. Cover and J. Thomas, *Elements of Information Theory*, New York: Wiley, 1991.

[12] O. Rioul and P. Duhamel, "Fast algorithms for discrete and continuous wavelet transforms," *IEEE Trans. Inform. Theory*, vol. 38, pp. 569–586, Mar. 1992.

[13] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Applicat.*, to be published.

[14] R. E. Van Dyck and T. G. Marshall, "Ladder realizations of fast subband/VQ coders with diamond support for color images," *IEEE Int. Symp. Circuits Syst.*, pp. I-670–I-677, 1993.

[15] A. A. C. Kalker and I. A. Shah, "Ladder structures for multidimensional linear phase perfect reconstruction filter banks and wavelets," in *Proc. Visual Communication Image Processing '92*, 1992, pp. 12–20.

[16] T. G. Marshall, "U-L block-triangular matrix and ladder realizations of subband coders," in *Proc. ICASSP 93*, 1993, pp. 177–180.

[17] E. Fossgaard, "Fast computational algorithms for the discrete wavelet transform," M.S. thesis, Dept. Math. Statist., Univ. Tromsø, Norway, Nov. 1997.

[18] K. A. Birney and T. R. Fischer, "On the modeling of DCT and subband image data for compression," *IEEE Trans. Image Processing*, vol. 4, pp. 186–193, Feb. 1995.

[19] G. M. Davis, "A wavelet-based analysis of fractal image compression," *IEEE Trans. Image Processing*, vol. 7, pp. 141–1545, Feb. 1998.

[20] A. S. Lewis and G. Knowles, "Image compression using the 2-D wavelet transform," *IEEE Trans. Image Processing*, vol. 1, pp. 244–250, Feb. 1992.

[21] N. M. Rajpoot, F. G. Meyer, R. G. Wilson, and R. R. Coifman, "On zerotree quantization for embbeded wavelet packet image coding," in *Proc. Int. Conf. Image Processing*, 1999.

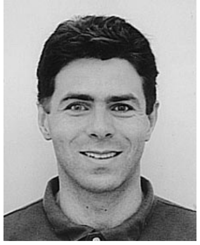
[22] J. B. Buckheit and D. L. Donoho, "Wavelab and reproducible research," in *Wavelets and Statistics*, A. Antoniadis and G. Oppenheim, Eds. Berlin, Germany: Springer-Verlag, 1995, pp. 55–82.

[23] F. G. Meyer and R. R. Coifman, "Brushlets: A tool for directional image analysis and image compression," *Appl. Comput. Harmon. Anal.*, pp. 147–187, 1997.

[24] S. S. Chen, "Basis pursuit," Ph.D. dissertation, Dept. Statist., Stanford Univ., Stanford, CA, Nov. 1995.

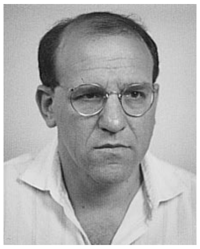
[25] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Processing*, vol. 41, pp. 3397–3415, Dec. 1993.

[26] F. G. Meyer, A. Z. Averbuch, J.-O. Strömberg, and R. R. Coifman, "Multi-layered image representation: Application to image compression," in *Proc. IEEE Int. Conf. Image Processing*, 1998.



François G. Meyer (M'94) received the M.S. (Hon.) degree in computer science and applied mathematics from Ecole Nationale Superieure d'Informatique et de Mathématiques Appliquées, Grenoble, France, in 1987, and the Ph.D. degree in electrical engineering from INRIA, France, in 1993.

From 1988 to 1989, he was with the Department of Radiotherapy, Institut Gustave Roussy, Villejuif, France. From 1989 to 1990, he was with Alcatel, Paris, France. From 1990 to 1993, he was a Research and Teaching Assistant with INRIA. From 1993 to 1995, he was a Postdoctoral Associate with the Departments of Diagnostic Radiology and Mathematics, Yale University. In 1996 he joined the French National Science Foundation (CNRS), France. In 1996, he was also an Associate Research Scientist with the Departments of Diagnostic Radiology and Mathematics, Yale University. From 1997 to 1999, he was an Assistant Professor with the Departments of Radiology and Computer Science, at Yale University, New Haven, CT. He is currently an Assistant Professor with the Department of Electrical Engineering, University of Colorado, Boulder. He is also an Assistant Professor with the Department of Radiology, School of Medicine, University of Colorado Health Sciences Center. His research interests include image processing, biomedical signal and image analysis, image and video compression, and diverse applications of wavelets to signal and image processing.



Amir Z. Averbuch was born in Tel Aviv, Israel. He received the B.Sc. and M.Sc. degrees in mathematics from the Hebrew University, Jerusalem, Israel, in 1971 and 1975, respectively. He received the Ph.D degree in computer science from Columbia University, New York, in 1983.

From 1976 to 1986 he was a Research Staff Member at IBM T. J. Watson Research Center, Yorktown Heights, NY. In 1987, he joined the Department of Computer Science, School of Mathematical Sciences, Tel Aviv University, where, since 1995, he has been an Associate Professor. His research interests include wavelet applications for signal/image processing and numerical computation, parallel processing and supercomputing (software and algorithms), and fast algorithms.

Jan-Olov Strömberg was born in Örebro, Sweden, on July 19, 1947. He received the Fil.Kand degree from Stockholm University, Stockholm, Sweden, in 1969, and the Ph.D degree in mathematics at Upsala College, East Orange, NJ, in 1977.

He was an Instructor and Assistant Professor at Princeton University, Princeton, NJ, from 1977 to 1980, Forskarassistent at Stockholm University from 1980 to 1982, and a Professor at University of Tromsø, Sweden, from 1982 to 1998. Since 1998, he has been a Professor of computational harmonic analysis at the Royal Institute of Technology, Stockholm. He has written about 30 publications and conference proceedings, most in harmonic analysis. His construction of an unconditional basis of Hardy spaces is the first known continuous orthonormal wavelet basis. Recently, his mathematical interest has been in applied mathematics.