

# IMAGE CODING BY BLOCK PREDICTION OF MULTIREOLUTION SUBIMAGES

*R. Rinaldo and G. Calvagno \**

## ABSTRACT

The redundancy of the multiresolution representation has been clearly demonstrated in the case of fractal images, but it has not been fully recognized and exploited for general images. Recently, fractal block coders have exploited the self-similarity among blocks in images. In this work we devise an image coder in which the causal similarity among blocks of different subbands in a multiresolution decomposition of the image is exploited. In a pyramid subband decomposition, the image is decomposed into a set of subbands which are localized in scale, orientation and space. The proposed coding scheme consists of predicting blocks in one subimage from blocks in lower resolution subbands with the same orientation. Although our prediction maps are of the same kind of those used in fractal block coders, which are based on an iterative mapping scheme, our coding technique does not impose any contractivity constraint on the block maps. This makes the decoding procedure very simple and allows a direct evaluation of the mean squared error (mse) between the original and the reconstructed image at coding time. More importantly, we show that the subband pyramid acts as an automatic block classifier, thus making the block search simpler and the block matching more effective. These advantages are confirmed by the experimental results, which show that the performance of our scheme is superior for both visual quality and mse to that obtainable with standard fractal block coders and also to that of other popular image coders like JPEG.

(Suggested EDICS Category: IP 1.1 Image Processing - Coding.)

---

\*Authors are with the Dipartimento di Elettronica e Informatica, Università di Padova, 35131 Padova, Italy. Reply Address: Roberto Rinaldo, Dipartimento di Elettronica e Informatica, Via Gradenigo 6/a, 35131 Padova, Italy. Ph. +39-49-828 7735. FAX +39-49-828 7699. E-mail: rinaldo@paola.dei.unipd.it

Fractal models for images have proven to be effective for image compression and rendering [1]–[7]. Fractal images share the common property that they exhibit self-similarity across scales. The redundancy of the multiresolution representation for such images has been fully recognized [8, 9].

The wavelet transform has been proposed [10] to provide a multiresolution representation of signals: its connections to subband coding have been recently noted and analyzed [11, 12]. A good example of the redundancy of the wavelet representation for fractal images is offered by figure 1 which shows a typical fractal image  $f(x_1, x_2)$  and the log – log plot of the magnitude of its wavelet transform  $T(a, b_1, b_2)$  versus scale  $a$  [13]. In this example, the analyzing wavelet is the “Mexican hat”

$$\psi(x, y) = (2 - x^2 - y^2) \exp\left(-\frac{x^2 + y^2}{2}\right)$$

and the wavelet transform is defined as

$$T(a, b_1, b_2) = \int \psi\left(\frac{x_1 - b_1}{a}, \frac{x_2 - b_2}{a}\right) f(x_1, x_2) dx_1 dx_2.$$

The plot shows the log-magnitude of the wavelet transform centered at  $(b_1, b_2) = (\bar{b}_1, \bar{b}_2)$ . The simple mapping between scale and wavelet transform suggests one could devise coding methods based on a multiresolution representation of images. Indeed, the methods proposed by Pentland [14] and Shapiro [15] seem to be inspired by this basic idea.

Fractal based coders have recently received attention in the literature [16]–[20]. In these coders, the image is partitioned into a set of non-overlapping range blocks, and a set of possibly overlapping domain blocks is chosen *from the same image*. Each of the range blocks is coded by mapping a domain block to the range block: the mapping consists of a spatial contraction, pixel reshuffling, contrast scaling and addition of an offset. By composing the action of the individual block maps, the result of the coding procedure is a transformation  $W$  on the image  $f$  that satisfies  $W(f) \simeq f$ . The essence of the method is that the description of the map  $W$  can be used as a code for  $f$ . If this description is simpler than that of the original image, some compression can be obtained. Fractal block coders give performances that are comparable to standard techniques [21]. From a subjective point of view, and as in most other block-based coding techniques, artifact edges between range block boundaries may appear at low bit rates [19].

In this paper, we propose a coding procedure that aims to exploit similarities among detail signals in a multiresolution decomposition of the image  $f$ . In a subband pyramid [22], an image is recursively decomposed into a low resolution approximation and three high pass detail images. The result of the decomposition is a set of subbands which are localized in scale (spatial frequency), orientation and space [11]. Figure 2 shows the original image “Lenna” and its pyramid subband decomposition.

Our coding scheme consists of choosing range blocks of different dimensions in each of the subimages corresponding to the same scale and orientation. The domain blocks are chosen from the subimage at the next lower resolution. The maps from a domain block to a range block are similar to those used in fractal block coders. Rather than recursively coding range blocks from other blocks in the image, as in standard fractal coders, our scheme *predicts* the range blocks of a subimage from the blocks of the smaller subimage at the next lower resolution. This simplifies considerably the decoding procedure, since no iteration of the map  $W$  is necessary, and

allows a more accurate control of the reconstruction error between the original and the coded image. Furthermore, the subband decomposition acts as an automatic classifier for blocks, since the blocks in different subimages corresponding to the same orientation have similar spectral content. This can be used to reduce the block searching time and to have smaller mean squared error in the block matching procedure. The proposed coding procedure gives better results than state-of-the-art fractal coders, and other popular coding algorithms such as JPEG, in terms of signal to reconstruction error ratios. From a subjective point of view, no blocking effect is visible in the reconstructed images, due to the nature of the coding procedure that operates in the subband coefficient domain.

In section II of the paper we briefly review subband coding principles and describe the pyramid subband decomposition scheme adopted as the first stage of our coder. Section III is a review of the main ideas used in fractal block coders: a description of the domain-range block mappings is given. Section IV describes our predictive technique. Section V is a detailed description of the coder. Section VI shows the results obtained with our scheme, and in section VII conclusions are drawn.

## II SUBBAND CODING PRINCIPLES

In this section we briefly review subband coding. For a more complete treatment, reference [23] is appropriate. Subband decomposition of signals was first introduced by Crochiere et al. [24] for speech coding, and later extended to image coding by Woods and O'Neil [25]. Figure 3 shows the schematic diagram relative to a two-channel subband coder for one-dimensional signals, where the transmitter and the receiver are connected back-to-back (therefore assuming no coding error).

In this scheme, the subband signal  $y^l(n)$  is obtained by filtering the input signal  $y(n)$  with the low-pass filter  $H_0$  and then subsampling the result by a factor two. Similarly, the other subband signal  $y^h(n)$  is obtained by subsampling the output of the high-pass filter  $H_1$  applied to  $y(n)$ . The pair of filters  $H_0$  and  $H_1$  is usually referred to as the analysis filter bank. At the receiver, the reconstruction is performed by first upsampling by a factor two the subband components  $y^l(n)$  and  $y^h(n)$ , then filtering the obtained signals with the synthesis filter pair  $G_0$  and  $G_1$ , and finally adding the results to get the output signal  $v(n)$ .

From figure 3, the  $z$ -transform of the output signal  $v(n)$  can be expressed as

$$V(z) = T(z)Y(z) + S(z)Y(-z)$$

where

$$T(z) = \frac{1}{2} [G_0(z)H_0(z) + G_1(z)H_1(z)]$$

corresponds to a linear shift-invariant transfer function between the input and the output, while

$$S(z) = \frac{1}{2} [G_0(z)H_0(-z) + G_1(z)H_1(-z)]$$

contributes to the aliasing components [26]. Ideally, one would like to reconstruct at the output the input signal or a delayed version of it, i.e.,  $V(z) = z^{-L}Y(z)$  with  $L$  the number of delay samples. Therefore, a perfect reconstruction analysis/synthesis subband system is one for which

$$S(z) = 0, \quad T(z) = z^{-L}.$$

This frequency domain analysis of the subband coding system can be alternatively replaced by a time (space in 2-D) domain analysis using the matrix notation of linear algebra [12, 27]. In this case, the signals  $y(n)$  and  $v(n)$  are represented by the infinite length vectors  $\mathbf{y}$  and  $\mathbf{v}$  respectively. The input/output relation is then expressed as

$$\mathbf{v} = GH^t\mathbf{y}$$

where  $G$  and  $H$  are infinite size matrices which represent linear transformations on infinite length vectors [27]. The columns of matrix  $H$  consist of even-shifted versions of the time-reversed kernels of the analysis filters  $H_0$  and  $H_1$  (zero-padded in the case of FIR filters), while the columns of  $G$  are even-shifted copies of the synthesis kernel filters  $G_0$  and  $G_1$ . Based on this analysis, a perfect reconstruction subband system with zero delay satisfies the constraint that the matrix product  $GH^t$  is the infinite size identity matrix  $I$ .

In addition to the perfect reconstruction condition we usually require the orthogonality of the analysis transformation, and we obtain for matrices  $G$  and  $H$  the constraints  $GG^t = I$ ,  $G = H$ . One possible way to satisfy these conditions is to choose the analysis/synthesis filter coefficients such that

$$\begin{aligned} g_0(n) &= h_0(-n) \\ g_1(n) &= h_1(-n) \\ h_1(n) &= (-1)^{1-n}h_0(1-n), \end{aligned} \tag{1}$$

where  $h_0(n)$  is a filter whose  $z$ -transform  $H_0(z)$  satisfies

$$H_0(z)H_0(z^{-1}) + H_0(-z)H_0(-z^{-1}) = 2. \tag{2}$$

In [11] it is shown that orthonormal bases of wavelets correspond to a subband coding scheme with orthogonal filters satisfying the equations (1) and (2). Considering the case of FIR filters, it is possible to show [12] that conditions (1) and (2) can be met exactly only for even-length filters. Furthermore, only trivial solutions with linear phase are possible. Approximate solutions with odd length linear phase FIR filters can be found in [11, 27].

A scheme with linear phase FIR analysis and synthesis filters has been proposed in [28], and a design method with several examples can be found in [29]. The scheme uses a symmetric linear phase low pass filter  $H_0$ , while the filter  $H_1$  is given by  $H_1(z) = H_0(-z)$ . The two filters are called Quadrature Mirror Filters (QMF). A consequence of the linear phase constraint is that relation (2) can be satisfied only approximately.

More general solutions for exact perfect reconstruction even-length, odd-length FIR linear phase filters are given by biorthogonal systems [12, 30], where the orthogonality conditions (1) and (2) are relaxed.

An interesting consequence of the orthogonality of the transformation matrix  $G$  is that the synthesis performed in the subband coding scheme is equivalent to the projection of the vector  $\mathbf{y}$  onto two orthogonal subspaces spanned by the even translations of the synthesis filter kernels  $g_0(n)$  and  $g_1(n)$ . Thus the Parseval theorem assures that the energy of  $\mathbf{y}$  is equal to the sum of the energies of  $\mathbf{y}^l$  and  $\mathbf{y}^h$ . Furthermore, whenever the synthesis stage is fed with any pair of sequences  $\tilde{\mathbf{y}}^l$  and  $\tilde{\mathbf{y}}^h$ , the energy of the output sequence  $\mathbf{v}$  equals the sum of the energies of  $\tilde{\mathbf{y}}^l$  and  $\tilde{\mathbf{y}}^h$ . This property can be used to relate in a simple way the coding error energy in the subbands to the error energy in the reconstructed signal. The same relation holds with a good approximation for nearly orthogonal systems.

The subband decomposition scheme of figure 3 is easily extended to two-dimensional signals (images) following a separable approach [25]. This is done by decomposing the input image  $y$

into four subimages (or subbands)  $y^{lh}$ ,  $y^{hl}$ ,  $y^{hh}$ ,  $y^{ll}$ , where the pair of superscript letters denotes the row-column filtering operations performed to obtain the subimage. For instance, subimage  $y^{lh}$  is obtained by low-pass filtering the rows and high-pass filtering the columns of  $y$ , followed by a factor two subsampling in each direction.

This procedure can be iterated to obtain a multilevel *pyramidal* decomposition of the image  $y$  [11]. Denoting with  $y_0^{ll}$  the input image  $y$ , at each decomposition level  $i$  the image  $y_{i-1}^{ll}$  is decomposed into the four subimages  $y_i^{lh}$ ,  $y_i^{hl}$ ,  $y_i^{hh}$ ,  $y_i^{ll}$ , for  $i = 1, \dots, i_M$ . The result of such a decomposition is a set of subimages which are localized in scale, orientation and space.

Figure 4 shows the organization of the subimages in a 5 level decomposition.

### III FRACTAL CODING

Fractal block coders are essentially based on the work done by A. Jacquin in [16, 17], with several improvements and variations presented in [18]–[20]. In standard fractal block coders, the image  $f$  is partitioned into a set of non-overlapping *range* blocks  $\{r_i, i = 1, \dots, N_R\}$  such that

$$\bigcup_{i=1}^{N_R} r_i = f, \quad r_i \cap r_j = \phi \text{ for } i \neq j.$$

Each of the range blocks has size  $B_i \times B_i$  pixels. As proposed by Jacquin, similarities between range blocks and *domain* blocks of another size  $D_j \times D_j$ , taken from other parts of the image, are exploited. The set of domain blocks  $\{d_j, j = 1, \dots, N_D\}$  plays the role of a codebook for the range blocks  $\{r_i, i = 1, \dots, N_R\}$  as in vector quantization [31], with the important difference that the domain blocks  $\{d_j\}$  are taken from the image itself.

When exploiting similarities between a range block  $r_i$  and a domain block  $d_j$ , the domain block is spatially scaled from size  $D_j \times D_j$  to size  $B_i \times B_i$ , it is isometrically transformed (i.e., reshuffling of the domain block pixels is carried out), it is contrast scaled by a factor  $\alpha_i$  and added to an offset value  $o_i$ . The mapping from the domain block  $d_j$  to the range block  $r_i$  is given by

$$\hat{r}_i = \tau_i(d_j) = \alpha_i \mathcal{I}_i(S_i(d_j)) + o_i,$$

where  $S_i$  denotes spatial contraction (e.g. pixel averaging),  $\mathcal{I}_i$  represents the isometry,  $\alpha_i$  is the contrast scaling and  $o_i$  is the offset. The map  $\tau_i$  is chosen in order to minimize the distance between the range block  $r_i$  and its approximation  $\hat{r}_i$ . Typically, we want to minimize the mse distance

$$D(r_i, \hat{r}_i) \triangleq \frac{1}{B_i^2} \left[ \sum_{l=1}^{B_i} \sum_{m=1}^{B_i} (r_i(l, m) - \hat{r}_i(l, m))^2 \right], \quad (3)$$

where  $l, m$  denote the pixel position inside the range block.

The framework in which fractal coders operate is the theory of iterated functions in complete metric spaces [17]. Once a metric is defined in the space  $F$  of images (for example, the mse metric), we define a map  $W : F \rightarrow F$  by composing the action of the individual block maps. Specifically, we define the transformation  $w_i : F \rightarrow F$  as

$$w_i(f) = \tau_i(d_j), \quad i = 1, \dots, N_R,$$

and build the map  $W$  as

$$\hat{f} = W(f) = \bigcup_{i=1}^{N_R} w_i(f).$$

Hence,  $W(f)$  is obtained as the union of approximations  $\hat{r}_i$  of the range blocks  $r_i$ , where each approximation  $\hat{r}_i$  is obtained by transforming a domain block  $d_j$  taken from the same image.

We now give some definitions and state the main theorems. A map  $W$  is contractive with respect to the metric  $D$  if it satisfies

$$D(W(f_1), W(f_2)) \leq s D(f_1, f_2), \quad \forall f_1, f_2 \in F, \quad 0 \leq s < 1,$$

where the number  $s$  is called a *contractivity factor* for  $f$ . Thus,  $W$  is contractive if the application of  $W$  to any two images  $f_1, f_2 \in F$  reduces the distance between them. Furthermore, an image  $\bar{f}$  such that  $W(\bar{f}) = \bar{f}$  is called a *fixed point* of the transformation  $W$ .

It can be shown [32] that if  $W$  is contractive, then  $W$  possesses exactly one fixed point image  $\bar{f} \in F$ , and moreover, for any image  $f_0 \in F$ , the sequence of iterated maps  $\{W(f_0), W(W(f_0)), \dots\}$  converges to  $\bar{f}$ . Thus

$$\bar{f} = \lim_{n \rightarrow \infty} W^{on}(f_0), \quad \forall f_0 \in F,$$

where “ $on$ ” denotes  $n$  iterations of the map  $W$ , that is  $W^{o0}(f_0) = f_0$  and  $W^{o(n+1)}(f_0) = W(W^{on}(f_0))$ . This result is known as the Fixed Point Theorem or Contraction Mapping Theorem. We note that this result permits to conclude that a description of  $W$  can be used as an alternative and equivalent definition of the fixed point  $\bar{f}$ .

The following theorem, which we restate from [3], is central to the design of fractal coders. **Theorem.** [The Collage Theorem] *Let  $F$  be a complete metric space with metric  $D$ . Let  $f \in F$  and  $W : F \rightarrow F$  a contractive transformation with contractivity factor  $0 \leq s < 1$  such that*

$$D(f, W(f)) \leq \epsilon.$$

*Then the distance between  $f$  and the fixed point  $\bar{f} = \lim_{n \rightarrow \infty} W^{on}(f_0)$  satisfies*

$$D(f, \bar{f}) \leq \frac{\epsilon}{1 - s}. \tag{4}$$

In words, if  $f$  and  $W(f)$  are *close*, then  $f$  is *close* to the fixed point  $\bar{f} = W(\bar{f})$  if the map  $W$  is sufficiently contractive (i.e.,  $s$  is not too close to 1 in (4)).

In summary, fractal block coding proceeds as follows. Block maps  $\tau_i$  are searched in order to minimize (3) for each individual range block. A map  $W$  on the entire image is then built by composing the action of the block maps. If  $W$  is contractive with respect to the mse metric and verifies  $D(W(f), f) \leq \epsilon$ , then the iteration of the map  $W$  on any starting image  $f_0$  will result in an image that is close to the original one  $f$  because of the bound in (4).

The map  $W$  can be described by specifying, for each range block  $r_i$ , the transformation  $w_i$ , i.e., the address of the domain block  $d_j$ , the isometry  $\mathcal{I}_i$ , the scaling parameter  $\alpha_i$  and the offset  $o_i$ . This description of  $W$  is a code for  $f$ . At the decoder, it is possible to reconstruct an approximation of image  $f$  by iterating  $W$  on any starting image (e.g., a uniform gray image).

It is important to understand the conditions under which the transformation  $W$  is contractive. It is noted in [33] that  $W$  can be *eventually* contractive and still have a unique fixed point. If  $|\alpha_i| < 1$  for each transformation  $\tau_i$ ,  $W$  results to be eventually contractive with respect to the mse metric. However, this condition is excessively restrictive. In general, the conditions under which  $W$  is eventually contractive can be difficult to determine: moreover, the exact determination of the contractivity factor for  $W$  could be important to bound the reconstruction error [19].

In our work we abandon the idea of a recursive map  $W$ . More precisely, we try to exploit a *causal* interdependence of the subband images in a multiresolution decomposition of  $f$ . The image is first subband decomposed, using a pyramid subband decomposition, as described in section II.

In our scheme, we *predict* blocks in the subimage  $y_i^{lh}$  (or  $y_i^{hl}, y_i^{hh}$ ) from blocks of the same dimension in the subimage  $y_{i+1}^{lh}$  (or  $y_{i+1}^{hl}, y_{i+1}^{hh}$ ). The subimages at the lowest resolution  $i_M$  are coded independently using PCM [34], as described in section V. After all the subband images are predicted from the blocks in lower resolution subimages, an approximation to the original image is reconstructed from the subband coefficients.

The connections between our scheme and standard fractal block coders are as follows. Each subimage  $y_i^{lh}$  (or  $y_i^{hl}, y_i^{hh}$ ) is divided into a set of non-overlapping range blocks  $\{r_i\}$ , whose size may vary depending on the different frequency band: the pool of domain blocks for image  $y_i^{lh}$  (or  $y_i^{hl}, y_i^{hh}$ ) consists of the blocks of the same dimension found in the image  $y_{i+1}^{lh}$  (or  $y_{i+1}^{hl}, y_{i+1}^{hh}$ ). For each range block  $r_i$  in the subimage  $y_i^{lh}$ , we find a domain block  $d_j$  in the subimage  $y_{i+1}^{lh}$  such that the map  $\tau_i$  defined as

$$\hat{r}_i = \tau_i(d_j) = \alpha_i (\mathcal{I}_i(d_j))$$

minimizes  $D(r_i, \hat{r}_i)$  with respect to all the possible choices of the isometries and the contrast scaling  $\alpha_i$ . In our scheme we neither use an offset parameter  $o_i$  nor a spatial contraction  $S_i$ .

We now show that it is reasonable to predict range blocks from domain blocks of the same size. Consider a 1-D system with input  $y(n)$  and subband signals  $y_i(n), i = 1, \dots, i_M$ , where  $y_i(n), i = 1, \dots, i_M - 1$  is obtained by filtering  $y(n)$  with  $H_0$  and subsampling by a factor of two ( $i - 1$ ) times, followed by high-pass filtering with  $H_1$  and subsampling. Signal  $y_{i_M}(n)$  is obtained by filtering with  $H_0$  and subsampling  $i_M$  times. We note that all the subband signals except  $y_{i_M}(n)$  are obtained by subsampling the output of the high pass filter  $H_1$ . Therefore, their energy will be localized in correspondence of the irregularities of the input signal. As an example, figure 5 shows the signals  $y_1(n)$  and  $y_2(n)$  when the input sequence  $y(n)$  is a step function and  $H_0$  and  $H_1$  are the 9-tap symmetric quasi-perfect reconstruction filters described in [27] that we use in our coder. As seen, the practical duration of the high energy events in the two signals is the same. So, if we want to predict blocks of coefficients in  $y_1(n)$  from blocks in  $y_2(n)$ , they should be of the same duration. The same reasoning can be applied to the following stages of the pyramid decomposition. These considerations extend to the 2-D case at least for one direction of filtering (row or column).

Our approach leads to some advantages with respect to standard fractal block coders. Since we are simply predicting subband subimages from lower resolution ones, there is no need to force contractivity of the block maps. Furthermore, this allows a direct evaluation at coding time of the mse between the original and reconstructed image, unlike the case of standard fractal block coders, where knowledge of the contractivity factor of  $W$  is required to evaluate the error bound [19]. The decoding procedure becomes very simple, because a one-step mapping is necessary to predict the subband images.

The most important advantage of our scheme, however, is that the pyramid subband decomposition acts as an automatic block classifier, with the result of simplifying the block search and block matching procedures. To make this evident, we consider the case of a 1-D signal  $y(n)$  with low-pass characteristics and consider the spectrum of the signals obtained from  $y(n)$  with

ideal subband filters. The results immediately extend to the case of a 2-D image with a low-pass power spectral density [34] and a separable 2-D subband analysis scheme.

Let  $y(n)$  be generated by a first order Markov model:

$$y(n) = \rho y(n-1) + z(n), \quad 0 < \rho < 1, \quad (5)$$

where  $z(n)$  is white noise with zero mean and variance  $\sigma_z^2$ . The power spectral density of  $y(n)$  is given by

$$S_y(e^{j\omega}) = \frac{\sigma_z^2}{|1 - \rho e^{j\omega}|^2}$$

and has strong low-pass characteristics when  $\rho$  is close to 1. We consider a pyramid subband decomposition of the signal  $y(n)$  ( $\rho = 0.9$  in (5)) with an ideal low-pass filter

$$H_0(e^{j\omega}) = \begin{cases} 1 & \text{for } |\omega| < \frac{\pi}{2} \\ 0 & \text{otherwise} \end{cases}$$

and a high-pass filter  $H_1(e^{j\omega}) = 1 - H_0(e^{j\omega})$ . Figure 6 shows the power spectral densities of the signals  $y_i(n), i = 1, \dots, 5$ . The power spectral densities in figure 6 are multiplied by an appropriate constant: it is easily seen that, apart from a multiplicative factor, the spectral contents of the subband signals are very similar to each other. An actual example is offered by figure 7 which shows the magnitude of the 2-D FFT of subimages  $y_1^{hl}$  and  $y_2^{hl}$  corresponding to a subband decomposition of ‘‘Lenna’’. Specifically, figure 7.a shows the magnitude of the  $256 \times 256$  2-D FFT of  $y_1^{hl}$ , while figure 7.b shows the magnitude of the  $256 \times 256$  2-D FFT of the zero-padded  $y_2^{hl}$ .

The similarities in the spectral content of the subband subimages show that the blocks in different subimages with the same orientation are of the same class and have similar band-pass characteristics. This is also suggested by the visual closeness of the subimages. The analysis indicates that better block matching is possible with a scheme that operates on a pyramid subband decomposition of the image. Therefore, we expect a lower mse between matched blocks than in standard fractal block coders, and a lower reconstruction error for the entire image at the same bit rate. This is confirmed by the experimental results of section VI.

## V DESCRIPTION OF THE CODER

In this section we will describe in some detail our predictive pyramid coder (PPC). In the following, the coder organization is described for  $512 \times 512$  images, but it can be easily adapted to images with different dimensions. We consider a pyramid subband decomposition with five levels, organized as in figure 4. The filters used to compute the subband decomposition are 9-tap symmetric nearly orthogonal and quasi-perfect reconstruction FIR quadrature mirror filters [27]. To avoid the introduction of artifacts at the boundary of the image, row and column filtering was performed using the symmetric extension method [35] rather than the circular convolution technique [25].

The coding procedure can be conceptually divided into two steps: a *block prediction* (BP) step and a *residual block coding* (RBC) step. In the first step we try to predict blocks from low-resolution subimages recursively, as described below. If the prediction for a range block  $r_i$  is not satisfactory at this stage, the actual coding of  $r_i$  is deferred to the RBC step.



### A. Block Prediction

To provide initial conditions to the BP step, subband  $y_5^{ll}$  is coded using an 8 bit uniform quantizer. Subbands  $y_5^{lh}$ ,  $y_5^{hh}$ ,  $y_5^{hl}$  are coded with a 7 bit laplacian quantizer. The prediction procedure begins at level 4 of the pyramid decomposition. Subband  $y_4^{lh}$  ( $y_4^{hh}$ ,  $y_4^{hl}$ ) is divided into  $4 \times 4$  blocks  $\{r_i\}$ . For each block  $r_i$ , a  $4 \times 4$  block  $d_j$  is searched in the  $16 \times 16$  region  $y_5^{lh}$  ( $y_5^{hh}$ ,  $y_5^{hl}$ ) at level 5 of the pyramid decomposition, in order to minimize  $D(r_i, \alpha_i \mathcal{I}_i(d_j))$ . We only consider 4 rotations [17] for the isometry  $\mathcal{I}_i$ : this has been shown to be sufficient, since nearly no improvement is obtained by considering reflections as well [20]. For each of the four rotations, the optimal parameter  $\alpha_i^o$  is readily obtained by setting to zero the derivative of  $D(r_i, \tau_i(d_j))$  with respect to  $\alpha_i$ . Note that 4 bits in one direction and 4 bits in the other direction are sufficient to specify the position of the  $4 \times 4$  domain block in subregion  $y_5^{lh}$  ( $y_5^{hh}$ ,  $y_5^{hl}$ ).

Once the domain block  $d_j$  and the optimal transformation are found, the mean squared error  $D(r_i, \tau_i(d_j))$  is compared to a target value  $P$  of the reconstruction error for the entire image. If  $D(r_i, \tau_i(d_j))$  exceeds  $P$ , the range block will be coded during the RBC step using a pixel-based coding scheme.

When  $D(r_i, \tau_i(d_j)) \leq P$ , the range block  $r_i$  is coded with 8 bits for the address of the domain block, 2 bits to specify the isometry, and 6 bits to code the scaling parameter  $\alpha_i$ . An additional bit per block is required to distinguish between the two coding schemes, corresponding to the BP or RBC alternatives.

We note that, apart from the blocks that are actually coded in the RBC step, we are in a position to predict subbands at level 4 from those at level 5. In order not to propagate the block matching error, subsequent subbands are actually predicted from the approximated subimages at the lower level, and not from the original subband coefficients. Thus, subbands at level 3 are predicted from the approximation at level 4 obtained with the procedure described above. We note that the decoder has in fact to predict subbands in the same way, i.e., it starts from *coded* lower resolution subimages to predict the new ones.

Subbands  $y_3^{lh}$ ,  $y_3^{hh}$ ,  $y_3^{hl}$  are divided into  $8 \times 8$  range blocks. Domain blocks with the same dimension are searched in subimages  $y_4^{lh}$ ,  $y_4^{hh}$ ,  $y_4^{hl}$  respectively. As before, we compute the mean squared error between matched blocks. If it exceeds the target parameter  $P$ , the range block is divided into four  $4 \times 4$  blocks, like in the quadtree scheme of [19], and domain blocks of dimension  $4 \times 4$  are considered. The block splitting procedure is due to the fact that the smaller block dimension should make it possible to satisfy the mse constraint between matched blocks. In case  $D(r_i, \tau_i(d_j)) > P$  for all  $j$ , the 16 coefficients in the range block will be coded during the RBC step.

The  $8 \times 8$  and  $4 \times 4$  blocks are coded with 10 bits to specify the location of the domain block in the  $32 \times 32$  subimages  $y_4^{lh}$ ,  $y_4^{hh}$ ,  $y_4^{hl}$ , 2 bits for the isometry, and 6 bits for the scaling parameter. Two additional bits are required for each block to indicate its dimension ( $8 \times 8$  or  $4 \times 4$ ) and to distinguish between the BP or RBC alternatives for the  $4 \times 4$  blocks.

Subimages  $y_2^{lh}$ ,  $y_2^{hh}$ ,  $y_2^{hl}$  are coded similarly: this time, blocks of dimension  $16 \times 16$ ,  $8 \times 8$ ,  $4 \times 4$  are considered, and domain blocks are searched in subimages  $y_3^{lh}$ ,  $y_3^{hh}$ ,  $y_3^{hl}$  respectively. If the  $P$  parameter constraint is not met for one block, it is split into four subblocks: the smallest allowed dimension for the blocks is  $4 \times 4$ . If the match is still not satisfactory, the  $4 \times 4$  block will be coded during the RBC step. The domain blocks have the same dimension as the range blocks, but their position is not arbitrary in the  $64 \times 64$  subimages  $y_3^{lh}$ ,  $y_3^{hh}$ ,  $y_3^{hl}$ . Only  $4 \times 4$  domain blocks located at even pixel positions are considered, so that 10 bits are still sufficient to specify their location in the subimages. We use 4 bits to specify the location of the  $8 \times 8$

blocks and 2 bits to specify the location of the  $16 \times 16$  blocks. Therefore, the  $8 \times 8$  domain blocks are searched with steps of 4 pixels in each direction, and a step 16 search is adopted for the  $16 \times 16$  blocks. Two additional bits for each block indicate the block dimension and the BP or RBC alternatives for the  $4 \times 4$  blocks.

In subimages  $y_1^{lh}, y_1^{hh}, y_1^{hl}$  we consider domain blocks of dimension  $32 \times 32, 16 \times 16, 8 \times 8$  and  $4 \times 4$ . Each block is split into 4 subblocks of smaller dimension every time the  $P$  parameter constraint is not met: however, no RBC step is considered for the  $4 \times 4$  blocks in these subimages. The  $4 \times 4$  domain blocks in subimages  $y_2^{lh}, y_2^{hh}, y_2^{hl}$  are located at positions that are multiples of 4 in each direction, so that 10 bits are sufficient to specify their location. We use a step 16 search for the  $32 \times 32, 16 \times 16, 8 \times 8$  blocks, with 6 bits necessary to specify their location.

A particular care has been taken of the quantization of the scaling parameters  $\alpha_i$ . As a matter of fact, an error  $\Delta\alpha_i$  in the quantization of  $\alpha_i$  can contribute considerably to the overall mse because it affects the error in the pixels of an entire block.

Let us respectively denote with  $r_i$  and  $d_j$  the range block and the (isometrically transformed) domain block. The optimal value of  $\alpha_i$  that minimizes the mse error for range block  $i$

$$P_i(\alpha_i) = \sum_{l,m} [r_i(l, m) - \alpha_i d_j(l, m)]^2 \quad (6)$$

is given by

$$\alpha_i^o = \frac{\sum_{l,m} r_i(l, m) d_j(l, m)}{\sum_{l,m} d_j^2(l, m)}. \quad (7)$$

If  $\alpha_i^o$  is quantized with an error  $\Delta\alpha_i \triangleq q_\alpha(\alpha_i) - \alpha_i$ , the mse (6) increases to

$$P_i(\alpha_i^o + \Delta\alpha_i) = P_i(\alpha_i^o) + (\Delta\alpha_i)^2 \sum_{l,m} d_j^2(l, m),$$

as it is immediately seen by evaluating (6) in  $\alpha_i^o + \Delta\alpha_i$  and taking (7) into account.

In order to minimize the expected mse we have to design the quantizer  $q_\alpha(\alpha_i)$  such that

$$\sum_i (\Delta\alpha_i)^2 \sum_{l,m} d_j^2(l, m) \triangleq \sum_i (\Delta\alpha_i)^2 S(\alpha_i) \quad (8)$$

is minimum.

If we model  $\alpha_i$  as the outcome of a random variable  $\alpha$ , equation (8) is proportional (by a factor  $\sum_i S(\alpha_i)$ ) to the sample mean of a function  $(\Delta\alpha)^2$  of  $\alpha$ , where each occurrence  $\alpha_i$  is weighted by  $S(\alpha_i)$ . A histogram of  $\alpha_i$ , where each occurrence is counted  $S(\alpha_i)$  times, shows that the best fitting distribution is laplacian, with a variance given by

$$\sigma_\alpha^2 = \frac{\sum_i \alpha_i^2 S(\alpha_i)}{\sum_i S(\alpha_i)}. \quad (9)$$

In our coder we therefore compute the variance from the scaling parameters and the domain block energies by using (9) and we employ a 6 bit laplacian quantizer for  $\alpha_i$ .

The scaling parameter  $\alpha_i$  is set to zero whenever the variance of the range block is less than or equal to  $0.5P$ . The blocks corresponding to the same spatial location at finer scales with the same orientation are also tested to determine if their variance is negligible. In such a case, these blocks are not coded at all. The first range block with  $\alpha_i = 0$  is encoded with a special symbol indicating the insignificance of all the corresponding blocks in finer scales. The technique is similar to that described in [36, 15]. Similarly, we do not code and set to zero the subbands with a total variance less than or equal to  $0.5P$ .

## B. Residual Block Coding

The block prediction procedure described in the previous section can give unsatisfactory results for certain  $4 \times 4$  blocks. In that case, one possible strategy could be to propagate the block splitting and consider  $2 \times 2$  range blocks. If we used the same coding scheme as described above, we would devote around 20 bits to code 4 coefficients, and 80 bits to code a  $4 \times 4$  block. However, well-known results from the subband coding literature [25] suggest that it is possible to obtain good quality reconstructed images while using less bits per coefficient on average. We therefore decided to code the individual coefficients in those  $4 \times 4$  blocks by using a laplacian quantizer and an optimal bit allocation strategy.

There are  $M = 9$  subbands, from level 4 to level 2 of the pyramid decomposition (see figure 4) that possibly have  $4 \times 4$  range blocks that were not coded during the BP step. As mentioned in the previous section, the RBC step is not considered for the blocks in subbands at level 1.

Let  $\sigma_k^2, k = 1, \dots, M$ , be the energy of the  $4 \times 4$  residual blocks in subband  $k$  and  $f_k$  the corresponding total number of subband coefficients.

Optimum bit allocation is obtained by coding subband  $k$  with  $R_k$  bits [34], where

$$R_k = R + \frac{1}{2} \log_2 \frac{\sigma_k^2 / f_k}{\prod_{h=1}^M \left( \frac{\sigma_h^2}{f_h} \right)^{\frac{f_h}{N}}}, \quad k = 1, \dots, M, \quad (10)$$

$R$  is the average number of bits per coefficient, and  $N = \sum_k f_k$ . The minimum reconstruction error energy results to be

$$\sigma_{r_{min}}^2 = \epsilon^2 2^{-2R} \left[ \prod_{k=1}^M \left( \frac{N \sigma_k^2}{f_k} \right)^{\frac{f_k}{N}} \right]. \quad (11)$$

The RBC step in our coder therefore consists of computing energies  $\sigma_k^2$  of the residual blocks in each subband  $k$ , together with the total number of coefficients  $f_k$ . We fix the average rate  $R = 3.5$  bits/coefficient, and allocate  $R_k$  bits to each coefficient in the blocks of subband  $k$ , as computed by (10).  $R_k$  is rounded for simplicity to the nearest integer and a laplacian quantizer with  $2^{R_k}$  levels is used for the coefficients [34]. Finally, the quantizer output levels are entropy coded by using a Huffman code [34].

## VI RESULTS

In this section, we present some experimental results to evaluate the performance of the proposed PPC. Original images are  $512 \times 512$  gray-level images, coded with 8 bits per pixel (bpp).

The peak-signal-to-noise ratio (PSNR) is used to determine image reconstruction fidelity. PSNR is defined as

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

where mse is given by

$$\text{mse} = \frac{1}{512^2} \sum_{l=0}^{511} \sum_{m=0}^{511} [\hat{f}(l, m) - f(l, m)]^2,$$

with  $f$  and  $\hat{f}$  denoting the original and reconstructed image respectively.

Figure 8 shows the PSNR versus the bit rate used to code “Lenna”. In the same plot, the results of our coder are compared with the JPEG coding system [37]. As it may be seen, the PPC performs better over the entire range of bit rates, with an improvement in PSNR that is almost independent of the bit rate.

To compare the visual quality of the two coding systems, we run our PPC and JPEG coder to reconstruct “Lenna” at 0.26 bpp. The reconstructed images are shown in figure 9.a and figure 9.b, respectively. As it may be seen, no blocking effect can be noticed in figure 9.a, even though some artifacts and smearing can be detected. Ringing effects, typical of subband coded images, are also negligible.

Table 1 reports a summary of the coding results for “Lenna” at 0.26 bpp. For each subband, we give the number of blocks of each size obtained during the coding procedure. For the  $4 \times 4$  blocks that were coded in the RBC step, we report the number of quantization levels, as computed with the bit allocation strategy described in section V.

The total coding time for “Lenna” was about 5 minutes on a Sun SPARC station IPC. Similar coding time figures were obtained for different bit rates and other images. The coding time is mainly devoted to the block search. The decoding procedure is instead very fast, and the system candidates itself very well for applications where a large variety of pre-encoded images has to be decoded quickly.

As reported in the last line of table 1, the value of the reconstruction error  $P = 150$  was used during the coding procedure. Notice that the actual mse is much lower (mse=34 dB). Therefore,  $P$  is used as a qualitative input parameter to the PPC. Smaller values of  $P$  were used to obtain a better quality (i.e., a higher PSNR) at the expense of a higher bit rate. However, the value of  $P$  is not directly related to the actual mse that will be obtained after coding. Nevertheless, the mse can be tracked during the coding steps, as explained in section V.

To make the artifacts introduced by our technique more evident, we show in figure 10 the image “Lenna” coded at 0.15 bpp using the PPC and JPEG. The PPC coded image appears to be smeared, and ringing artifacts are noticeable around edges. However, the image quality is acceptable.

Figures 11.a and 11.b show the original and PPC reconstructed image “Building” at 0.36 bpp. In this case, the performance of the coder is worse than for “Lenna”, because of the greater high frequency content of “Building”. Again, the visual quality of the reconstructed image is fairly good, even though some artifacts and ringing effects are noticeable near the edges. Table 2 reports a summary of the coding results.

Another coding example is offered by figures 12.a and 12.b which show the original and reconstructed image “Clown” at 0.25 bpp, while the corresponding coding results are reported in table 3.

## VII CONCLUSION

In this paper we have proposed an image coding technique which exploits the similarities among blocks in different subbands of a pyramid representation of the image. These similarities are used to predict blocks of a subband from blocks in subbands with the same orientation at lower resolution. The predicted blocks are therefore simply coded by specifying the location of the domain block and the parameters of the map. In our coder, we use block mappings similar to those used in fractal block coders. However, our technique is not fractal since no map iteration is implied.

The advantages of this scheme with respect to standard fractal block coder derive from

the fact that the multiresolution decomposition inherently classifies image blocks. We have shown that the spectral content of blocks in subbands with the same orientation has similar characteristics. The block search procedure is simplified because the search region is limited to the nearest lower resolution subband. Moreover, similar spectral characteristics allow a better matching of most blocks.

The block prediction scheme is conceptually simpler than the iterative scheme adopted in standard fractal block coders. As a consequence, in our coding algorithm, no map contractivity issue arises. Furthermore, a direct evaluation of the mse between the original and the reconstructed image is possible at coding time and the decoding procedure is not iterative but very simple.

Experimental results are satisfactory both in terms of the visual appearance of the reconstructed image and in terms of PSNR versus bit rate. No blocking effect is visible in the reconstructed images, as expected from a technique that operates in the subband domain.

A drawback of the proposed technique is the larger encoding time in comparison to other coding techniques like JPEG. The block matching procedure employed by the PPC is responsible for such an impairment, with the result of making the coder suitable for applications where non-symmetric coders and decoders can be tolerated. We are currently investigating the possibility of reducing the encoding time by constraining the search performed for each block.

As a final remark, we note that, when the BP step does not give satisfactory results, our scheme actually reduces to standard subband coding. We believe that better schemes for the RBC step could further improve the image quality at the same bit rate.

#### ACKNOWLEDGMENT

The authors would like to thank the people of the Image Laboratory of the Dipartimento di Elettronica e Informatica for helpful and fruitful discussions. Thanks are also due to Prof. A. Zakhor of the University of California at Berkeley for introducing the first author to the topic of fractal coding. We are also grateful to the anonymous Reviewers for their comments and suggestions.

#### REFERENCES

- [1] B. Mandelbrot, *The Fractal Geometry of Nature*, San Francisco, CA, Freeman, 1982.
- [2] A. Fournier, D. Fussel, L. Carpenter, "Computer rendering of stochastic models," *Comm. of the ACM*, vol. 25, pp. 371-384, 1982.
- [3] M. Barnsley, *Fractals everywhere*, Boston, Academic Press, 1988.
- [4] H-O. Peitgen, P.H. Richter, *The beauty of fractals*, Berlin: Springer, 1986.
- [5] H-O. Peitgen, D. Saupe, *The science of fractal images*, Springer-Verlag, 1988.
- [6] M.F. Barnsley, A. Jacquin, F. Malassenet, L. Reuter, "Harnessing chaos for image synthesis," *Proc. SIGGRAPH 1988*, pp. 131-140, 1988.
- [7] H-O. Peitgen, H. Jürgens, D. Saupe, *Chaos and fractals, new frontiers of science*, Springer-Verlag, 1992.

- [8] G.C. Freeland, T.S. Durrani, "IFS Fractals and the Wavelet Transform," in *Proc. ICASSP 1990*, pp. 2345–2348.
- [9] A. Arneodo, G. Grasseau, M. Holschneider, "Wavelet Transform of Invariant Measures of some Dynamical Systems," in *Wavelets*, Eds. J.M. Combes, A. Grossman, P. Tchamitchian, Springer, Berlin, 1989.
- [10] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," *IEEE Trans. on Information Theory*, pp. 961–1005, Sep. 1990.
- [11] S. Mallat, "Multifrequency channel decomposition of images and wavelet models," *IEEE Trans. on Acoustics, Speech and Signal Proc.*, pp. 2091–2110, Dec. 1989.
- [12] M. Vetterli, C. Herley, "Wavelets and Filter Banks - Theory and Design," *IEEE Trans. on Signal Processing*, vol. 40, no. 9, pp. 2207–2232, Sep. 1992.
- [13] R. Rinaldo, A. Zakhor, "Inverse and Approximation Problem for Two-Dimensional Fractal Sets," accepted for publication in *IEEE Trans. on Image Processing*.
- [14] A. Pentland, B. Horowitz, "A Practical Approach to Fractal-Based Image Compression," in *Proc. Data Compression Conference*, Snowbird, Utah, IEEE Computer Society Press, 1991.
- [15] J. M. Shapiro, "An Embedded Wavelet Hierarchical Image Coder," in *proc. ICASSP 1992*, pp. IV-657–660.
- [16] A. E. Jacquin, "A novel fractal block-coding technique for digital images," in *Proc. ICASSP 1990*, Albuquerque, NM, pp. 2225-2228, April 1990.
- [17] A. E. Jacquin, "Image Coding based on a Fractal theory of Iterated Contractive Image Transformations," *IEEE Trans. on Image Processing*, vol. 1, no. 1, pp. 18–31, Jan 1992.
- [18] A. E. Jacquin, "Fractal image coding: a review," *Proceedings of the IEEE*, vol. 81, no. 10, pp. 1451–1465, Oct. 1993.
- [19] Y. Fisher, E. W. Jacobs, R. D. Boss, "Fractal Image Compression Using Iterated Transforms," in *Image and text compression*, edited by Storer, J.A. Dordrecht, Netherlands, Kluwer Academic Publishers, 1992, pp. 35–61.
- [20] D.M. Monro, "Generalized Fractal Transforms: Complexity Issues," in *proc. Data Compression Conference*, Snowbird, Utah, March 1993, pp. 254–261.
- [21] R.J. Clarke, *Transform Coding of Images*, Academic Press, New York, 1985.
- [22] M.J. Smith, T.P. Barnwell, "Exact Reconstruction Techniques for Tree-Structured Subband Coders," *IEEE Trans. on Acoustics, Speech and Signal Proc.*, 1986, pp. 434–441.
- [23] P.P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice Hall, A.V. Oppenheim Ed., 1993.
- [24] R.E. Crochiere, S.A. Webber, and J.L. Flanagan, "Digital coding of speech subbands," *Bell. Syst. Tech. J.*, vol. 55, pp 1096–1085, Oct. 1976.

- [25] J.W. Woods and S.D. O'Neil, "Subband coding of images," *IEEE Trans. on Acoustics Speech and Signal Processing*, vol. 34, no. 5, Oct. 1986.
- [26] A. Oppenheim and R. Schaffer, *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1975.
- [27] E.P. Simoncelli, E.H. Adelson, "Subband Transforms," in *Subband Image Coding* edited by J. W. Woods, Kluwer Academic Publishers, 1991, pp. 143–192.
- [28] D. Esteban, G. Galand, "Application of Quadrature Mirror Filters to Split Band Voice Coding Schemes," in *proc. IEEE ICASSP*, pp. 191–195, May 1977.
- [29] J.D. Johnston, "A Filter Family Designed for Use in Quadrature Mirror Filter Banks," in *proc. IEEE ICASSP*, pp. 291–294, April 1980.
- [30] A. Cohen, I. Daubechies, J.C. Feauveau, "Biorthogonal Bases of Compactly Support Wavelets," *Communications on Pure and Applied Mathematics*, V45 N5:485-560, June 1992.
- [31] N.M. Nasrabadi, R.A. King, "Image Coding Using Vector Quantization: a Review," *IEEE Trans. on Communications*, vol. 36, no.8, pp. 957–971, Aug. 1988.
- [32] E. Kreyszig, *Introductory Functional Analysis with Applications*, John Wiley & Sons, New York, 1978.
- [33] E.W. Jacobs, Y. Fisher, R.D. Boss, "Image compression - a study of the Iterated Transform Method," *Signal Processing*, vol. 29, no. 3, pp. 251–263, Dec. 1992.
- [34] N.S. Jayant, P. Noll, *Digital Coding of Waveforms*, Englewood Cliffs, N.J., Prentice Hall, 1984.
- [35] M.J.T. Smith and S.L. Eddins, "Analysis/synthesis techniques for subband image coding", *IEEE Trans. on Acoustics Speech and Signal Processing*, vol. 38, no. 8, Aug. 1990.
- [36] A.S. Lewis, G. Knowles, "Image Compression Using the 2-D Wavelet Transform," *IEEE Trans. on Image Processing*, vol. 1, no. 2, pp. 244–250, Apr. 1992.
- [37] G.K. Wallace, "The JPEG Still Picture Compression Standard", *Communications of the ACM*, vol. 34, no. 4, pp. 30–44, Apr. 1991.

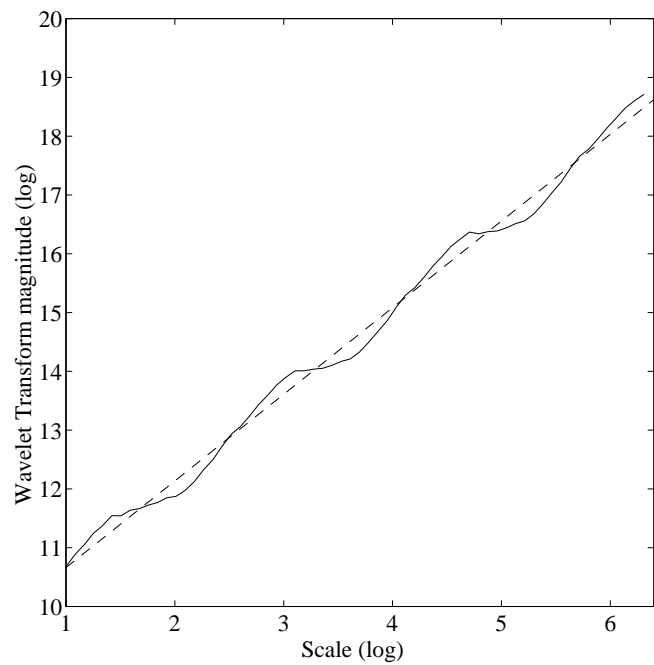
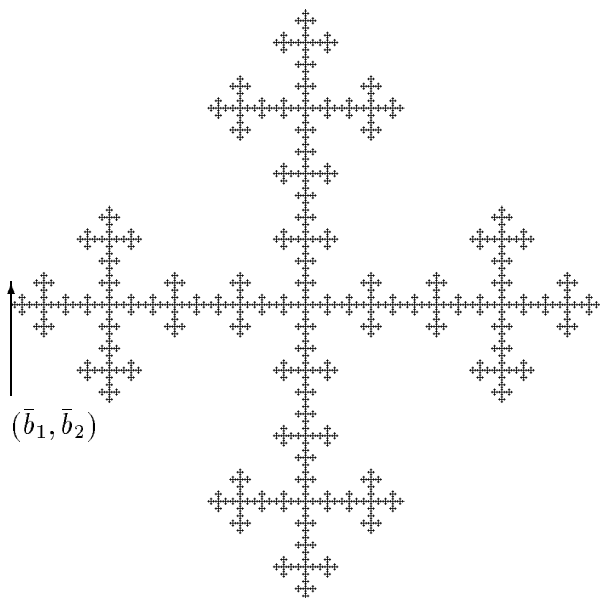


Figure 1: (a) The “Snowflake”; (b) Wavelet Transform versus Scale.

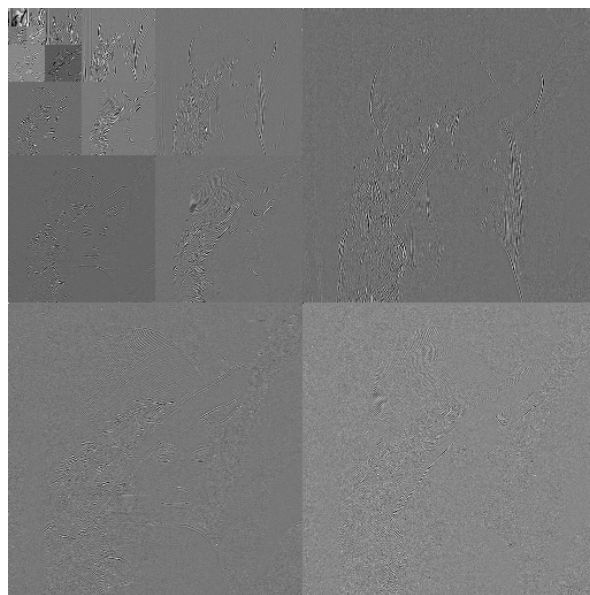


Figure 2: “Lenna”: (a) Original image; (b) Multiresolution decomposition.



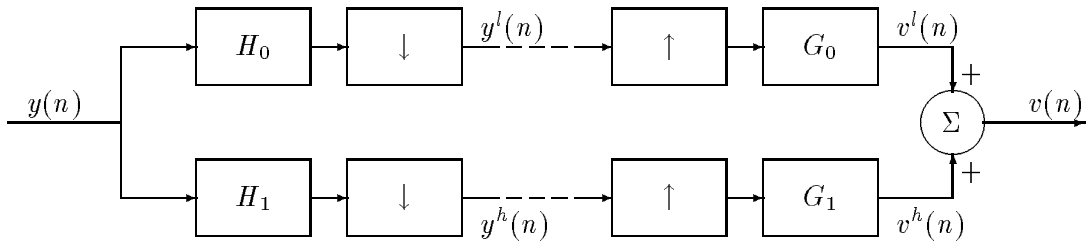


Figure 3: Two-channel subband coding scheme.

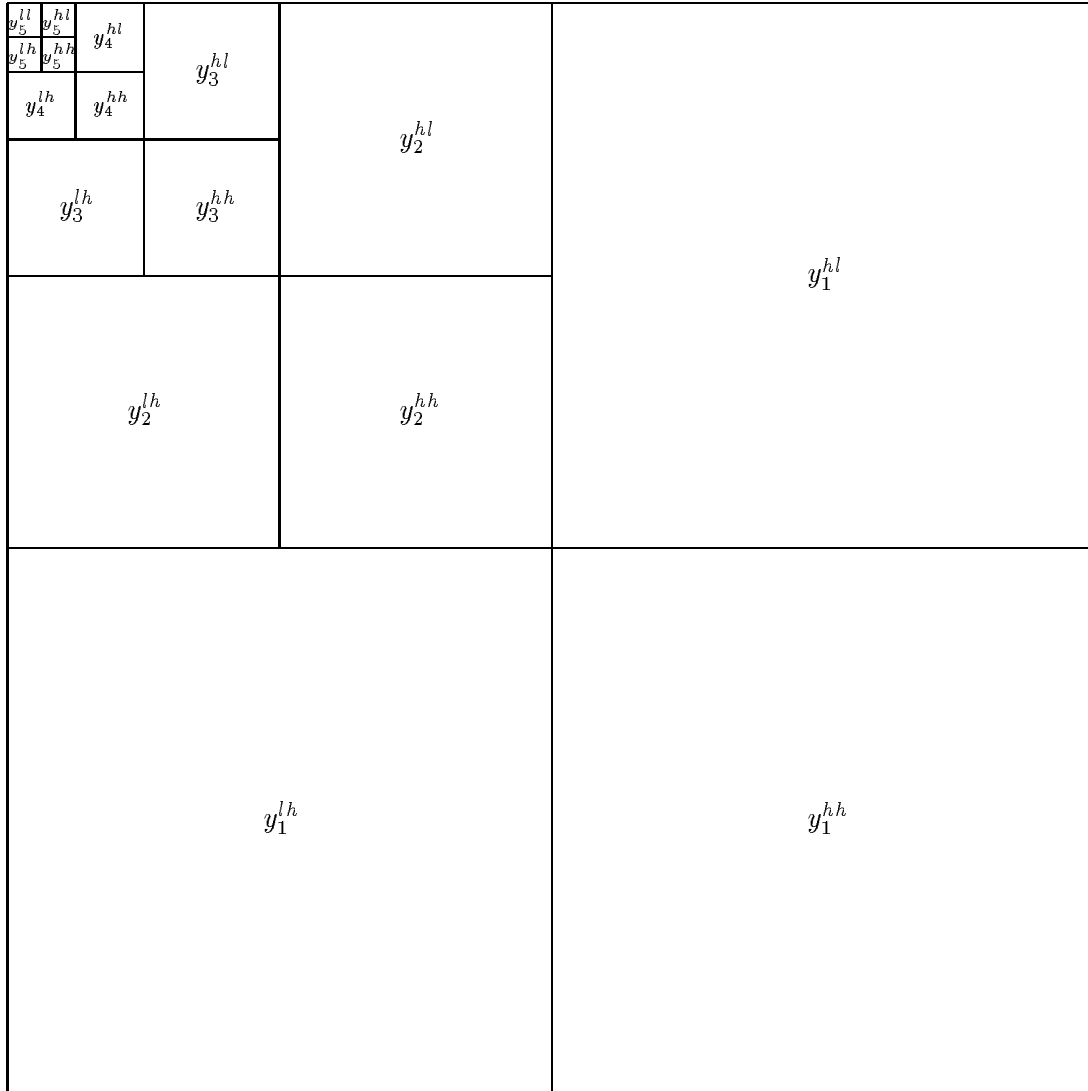


Figure 4: Organization of Subbands in the pyramid decomposition.

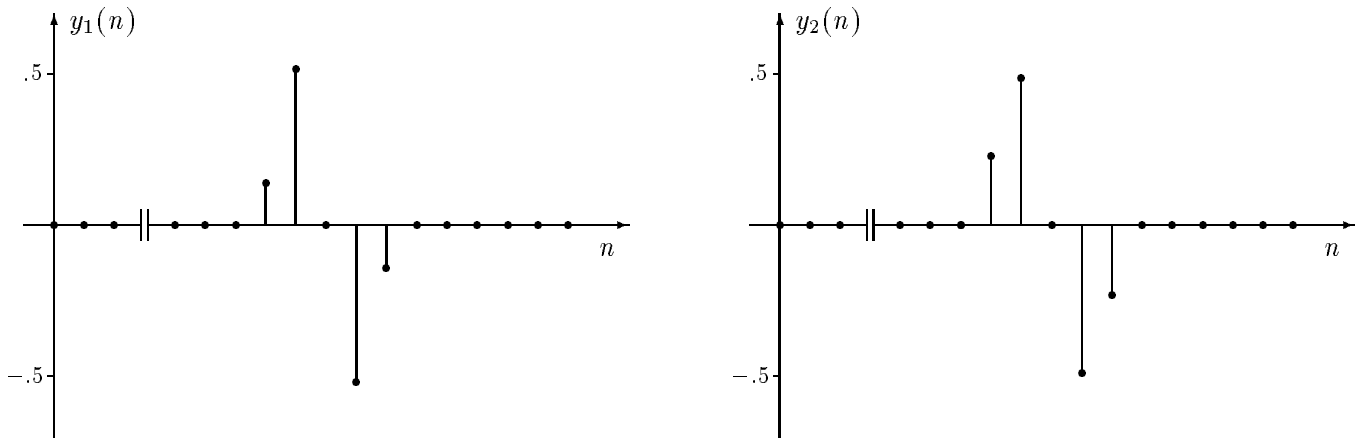


Figure 5: Subband components  $y_1(n)$  and  $y_2(n)$  of a unit step function.

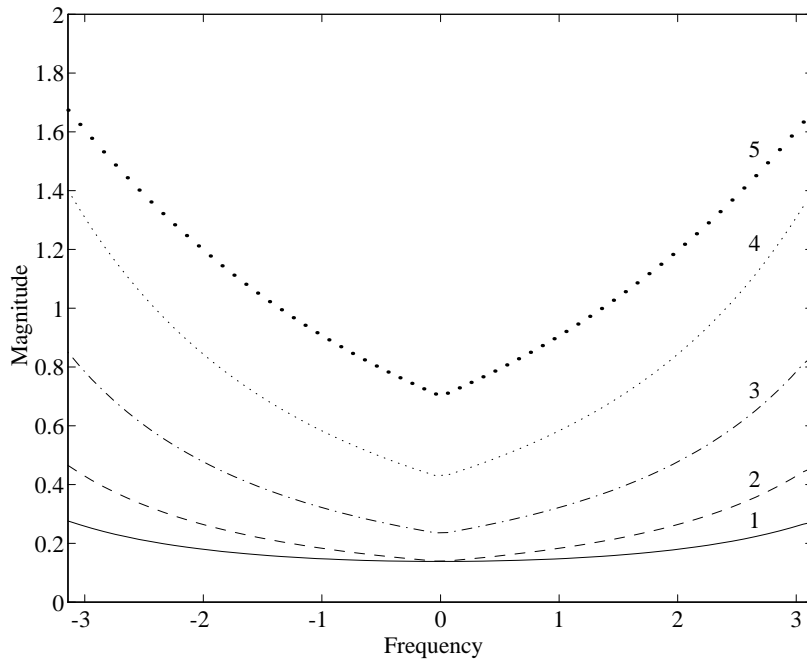
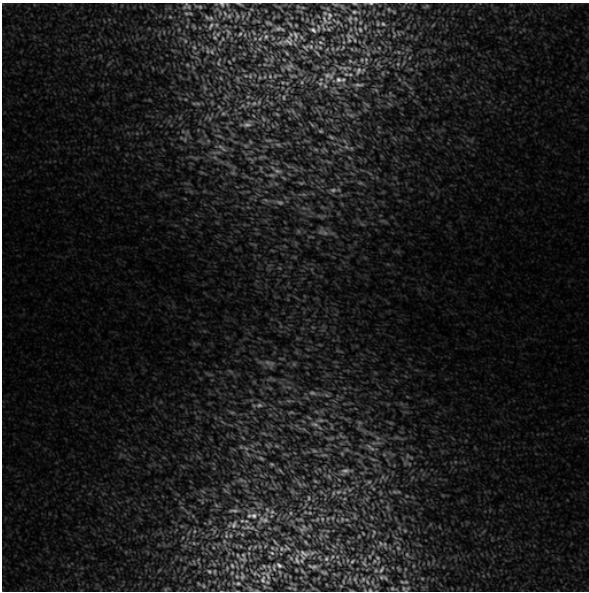
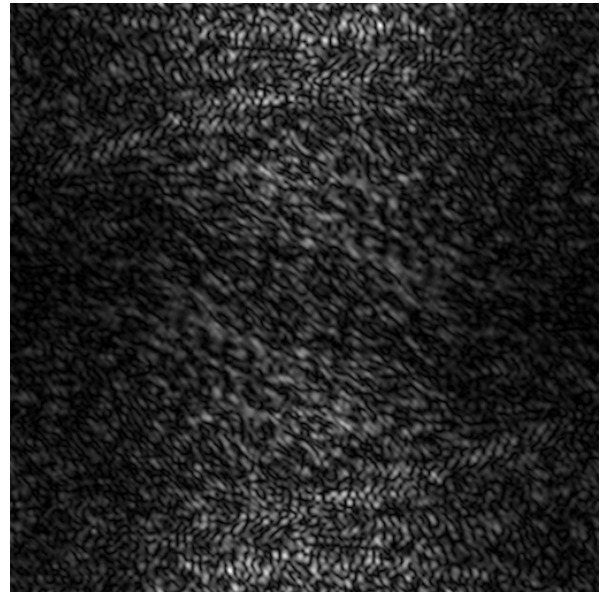


Figure 6: Power spectral densities of  $y_i(n)$ ,  $i = 1, \dots, 5$ .



(a)



(b)

Figure 7: 2-D FFT magnitude of  $y_1^{hl}$  and  $y_2^{hl}$  of "Lenna".

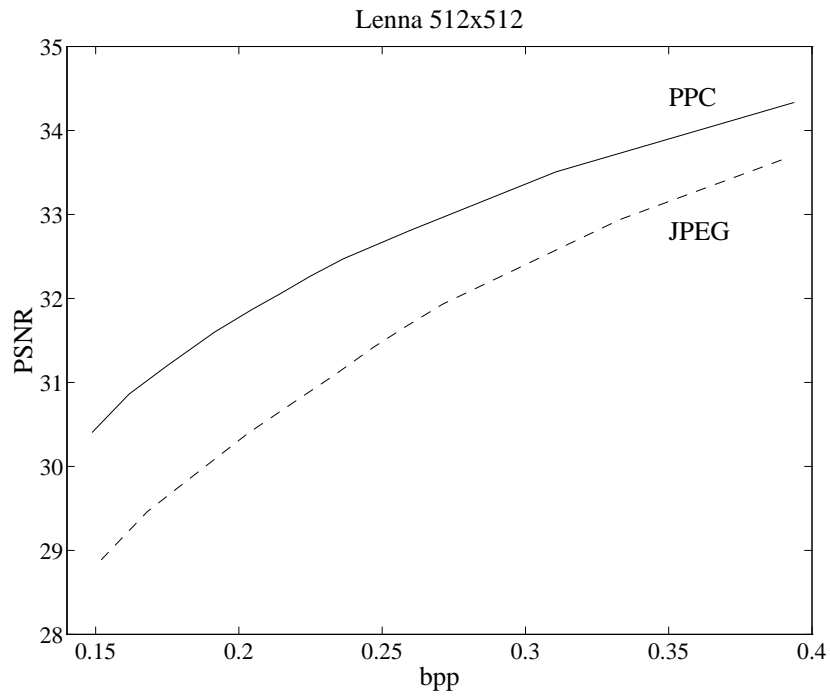


Figure 8: PSNR versus bpp for "Lenna": comparison of PPC and JPEG.



(a)



(b)

Figure 9: Reconstructed “Lenna” at 0.26 bpp. (a) PPC; (b) JPEG.



(a)



(b)

Figure 10: Reconstructed “Lenna” at 0.15 bpp. (a) PPC; (b) JPEG.



(a)



(b)

Figure 11: “Building”: (a) Original; (b) Reconstructed at 0.36 bpp.



(a)



(b)

Figure 12: “Clown”: (a) Original; (b) Reconstructed at 0.25 bpp.

Table 1: Coding results for “Lenna”.

Subband	#4 × 4 (RBC) / Lev.	#4 × 4 (BP)	#8 × 8	#16 × 16	#32 × 32
$y_4^{lh}$	53/16	11	–	–	–
$y_4^{hh}$	51/16	13	–	–	–
$y_4^{hl}$	59/32	5	–	–	–
$y_3^{lh}$	96/8	76	21	–	–
$y_3^{hh}$	91/8	57	27	–	–
$y_3^{hl}$	149/16	71	9	–	–
$y_2^{lh}$	100/8	148	42	38	–
$y_2^{hh}$	72/8	136	20	46	–
$y_2^{hl}$	202/8	162	49	29	–
$y_1^{lh}$	–	–	–	–	–
$y_1^{hh}$	–	–	–	–	–
$y_1^{hl}$	–	136	22	18	56
$P=150$	PSNR=32.78	bpp=0.26	Coding Time:		5 m

Table 2: Coding results for “Building”.

Subband	#4 × 4 (RBC) / Lev.	#4 × 4 (BP)	#8 × 8	#16 × 16	#32 × 32
$y_4^{lh}$	63/64	1	–	–	–
$y_4^{hh}$	61/16	3	–	–	–
$y_4^{hl}$	63/32	1	–	–	–
$y_3^{lh}$	209/16	39	2	–	–
$y_3^{hh}$	133/8	95	7	–	–
$y_3^{hl}$	168/16	68	5	–	–
$y_2^{lh}$	431/8	273	44	9	–
$y_2^{hh}$	16/4	60	17	55	–
$y_2^{hl}$	181/8	283	76	16	–
$y_1^{lh}$	–	120	26	10	58
$y_1^{hh}$	–	–	–	–	–
$y_1^{hl}$	–	–	–	–	–
$P=150$	PSNR=31.20	bpp=0.36	Coding Time:		6 m

Table 3: Coding results for “Clown”.

Subband	#4 × 4 (RBC) / Lev.	#4 × 4 (BP)	#8 × 8	#16 × 16	#32 × 32
$y_4^{lh}$	57/32	7	–	–	–
$y_4^{hh}$	48/16	16	–	–	–
$y_4^{hl}$	59/32	5	–	–	–
$y_3^{lh}$	129/16	79	12	–	–
$y_3^{hh}$	72/8	72	28	–	–
$y_3^{hl}$	165/16	63	7	–	–
$y_2^{lh}$	71/8	177	42	38	–
$y_2^{hh}$	3/8	17	11	60	–
$y_2^{hl}$	162/8	298	53	22	–
$y_1^{lh}$	–	–	–	–	–
$y_1^{hh}$	–	–	–	–	–
$y_1^{hl}$	–	–	–	–	–
$P=225$	PSNR=32.93	bpp=0.25	Coding Time:	4 m	