A Review of the Fractal Image Coding Literature

Brendt Wohlberg and Gerhard de Jager

***Abstract*— Fractal image compression is a relatively recent technique based on the representation of an image by a contractive transform, on the space of images, for which the fixed point is close to the original image. This broad principle encompasses a very wide variety of coding schemes, many of which have been explored in the rapidly growing body of published research. While certain theoretical aspects of this representation are well established, relatively little attention has been given to the construction of a coherent underlying image model which would justify its use. Most purely fractal-based schemes are not competitive with the current state of the art, but hybrid schemes incorporating fractal compression and alternative techniques have achieved considerably greater success. This review represents a survey of the most significant advances, both practical and theoretical, since the publication in 1990 of Jacquin's original fractal coding scheme.**

***Index Terms*— Image coding, fractals**

I. Introduction

The fundamental principle of fractal coding consists of the representation of an image by a contractive transform of which the fixed point is close to that image. Banach's fixed point theorem guarantees that, within a complete metric space, the fixed point of such a transform may be recovered by iterated application thereof to an arbitrary initial element of that space [1]. Images are represented within this framework by viewing them as vectors [2] [3, ch. 7] within a Hilbert space, the metric being derived from the inner product via the norm [1, pg. 129]. Encoding is not as simple, since there is no known algorithm for constructing the transform with the smallest possible distance, given the constraints on the transform, between the corresponding fixed point and the image to be encoded. The usual approach is based on the collage theorem (see Section V-A) which provides a bound on the distance between the image to be encoded and the fixed point of a transform, in terms of the distance between the transform of the image and the image itself. A suitable, although suboptimal, transform may therefore be constructed as a "collage" or union of mappings from the image to itself, a sufficiently small "collage error" (the distance between the collage and the image) guaranteeing that the fixed point of that transform is close to the original image.

In the original approach, devised by Barnsley, this transform was composed of the union of a number of affine mappings on the entire image - an Iterated Function System (IFS) [3, ch. 2] [4]. While a few impressive examples of image modelling were generated by this method (Barnsley's fern [4] [5, pg. 256], for example), no automated encoding algorithm was found. Fractal compression became a practical reality with the introduction by Jacquin[1] of the Partitioned IFS (PIFS) [3, ch. 2], which differs from an IFS in that each of the individual mappings operates on a subset of the image, rather than the entire image. Since the image support is tiled by "range blocks", each of which is mapped from one of the "domain[2] blocks" as depicted in Figure 1, the combined mappings constitute a transform on the image as a whole. The transform minimising the collage error within this framework is constructed by individually minimising the collage error for each range block, which requires locating the domain block which may be made closest to it under an admissible block mapping. This transform is then represented by specifying, for each range block, the identity of the matching domain block together with the block mapping parameters minimising the collage error for that block. Distances are usually measured by the MSE (Mean-Squared Error), equivalent to the distance derived from the l^2 inner product [1, pg. 133], since optimisation[3] of the standard block mappings is simple under this measure [3, pp. 20-21].

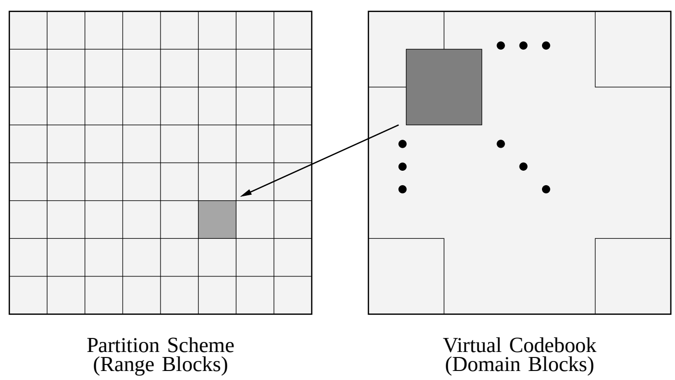


Fig. 1. One of the block mappings in a PIFS representation.

The fundamental principle of fractal coding clearly leaves considerable latitude in the design of a particular implementation. Within this broad framework, the differences between the majority of existing fractal coding schemes may be classified into the following categories:

- The partition imposed on the image support by the range blocks.
- The composition of the pool of domain blocks.
- The class of transforms applied to the domain blocks.
- The type of search used in locating suitable domain blocks.

B. Wohlberg was with the Digital Image Processing Laboratory, Electrical Engineering Department, University of Cape Town, South Africa. He is now with Los Alamos National Laboratory, Los Alamos, NM 87545, USA. G. de Jager is with the Digital Image Processing Laboratory, Electrical Engineering Department, University of Cape Town, Rondebosch 7701, South Africa.

[1] Note that there is an error with respect to a contractivity criterion [6, pp. 207-208] in Jacquin's early work.

[2] The names of these blocks are derived from their roles in the mappings. Note, though, that these labels are reversed by Barnsley [7, pg. 181].

[3] Optimisation with respect to the sup norm has also been considered [8].

- The representation and quantisation of the transform parameters.

There are unfortunately very few theoretical results on which design decisions in any of these aspects may be based, and choices are often made on a rather *ad hoc* basis. In addition, these categories are not independent, in the sense that any comparative analysis of coding performance between different options in one of these categories is usually contingent on the corresponding choices in the other categories; a meaningful comparison between the relative merits of particular choices in each category is consequently very difficult. This review is therefore intended primarily as an overview of the variety of schemes that have been investigated, although brief comparisons are made where possible. Details of the more theoretical aspects of fractal compression, such as the collage theorem and convergence conditions, are presented where appropriate, and the review is concluded with a wavelet based analysis of fractal compression, and a comparison of the performance of the most effective fractal coding based compression algorithms in the literature.

While fractal coding of colour images [9] [10] and video [11] [12] [13] have been investigated, space limitations necessitate the restriction of the scope of this review to the coding of greyscale images (all of which may be assumed to have 8 bits/pixel). Since publications responsible for introducing new concepts are usually cited in derived work, we have in some cases referenced the more recent or easily accessible work. In addition to the proceedings [14] [15] of the 1995 NATO conference on the subject, of which many of the papers are referenced in this review, there are currently three books devoted entirely to this subject. The book by Barnsley and Hurd [7], the first on the subject, reveals relatively little practical detail. The book edited by Fisher [3] contains two introductory chapters and a collection of significant work by a number of authors, while the recent book by Lu [16] combines introductory material with an in-depth discussion of many aspects of fractal coding.

II. Partition Schemes

The first decision to be made when designing a fractal coding scheme is in the choice of the type of image partition used for the range blocks. Since domain blocks must be transformed to cover range blocks, this decision, together with the choice of block transformation described later, restricts the possible sizes and shapes of the domain blocks. A wide variety of partitions have been investigated, the majority being composed of rectangular blocks.

A. Fixed size square blocks

The simplest possible range partition consists of the fixed size square blocks [17] [18] [19] depicted in Figure 2a. This type of block partition is successful in transform coding of individual image blocks[4] since an adaptive quantisation mechanism is able to compensate for the varying "activity" levels of different blocks, allocating few bits to blocks with little detail and many to detailed blocks.

Fractal coding based on the standard block transform, in contrast, is not capable of such adaptation, representing a significant disadvantage of this type of block partition for fractal coding. This deficiency may be addressed by introducing adaptivity to the available block transforms as described in Section III-B, but the usual solution is to introduce an adaptive partition with large blocks in low detail regions and small blocks where there is significant detail. There is, of course, a trade-off between the lower distortion expected by adapting the partition to the image content, and the additional bits required to specify the partition details.

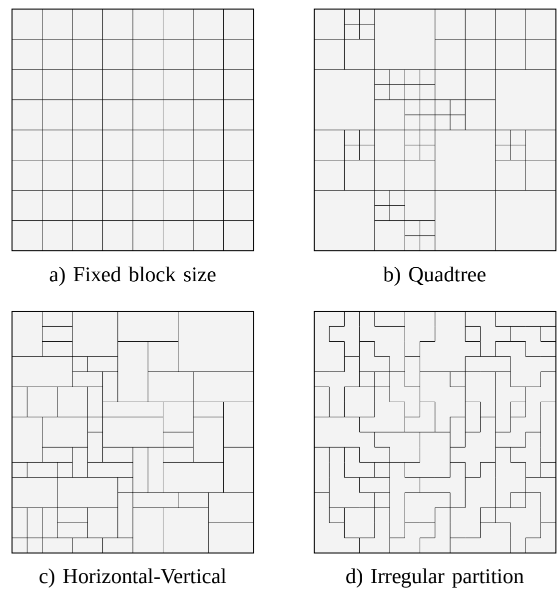


Fig. 2. Right-angled range partition schemes.

B. Quadtree

The quadtree partition (see Figure 2b) employs the well-known image processing technique based on a recursive splitting of selected image quadrants, enabling the resulting partition to be represented by a tree structure in which each non-terminal node has four descendents. The usual top-down construction starts by selecting an initial level in the tree, corresponding to some maximum range block size, and recursively partitioning any block for which a match better than some preselected threshold is not found [3, ch. 3] [16, pp. 93-105] [21] (or more efficiently, by deciding whether to split a block by examining the variance of its pixels [16, pp. 105-106] [22]). The alternative bottom-up construction begins with a uniform partition using the smallest block size, and then proceeds to merge those neighbouring blocks for which a more efficient representation is provided by the resulting larger block one level up the quadtree [16, pp. 93-105] [23]. Compact coding of partition details is possible by taking advantage of the tree structure of the partition.

Jacquin's original scheme [24] [25] [26] used a variant of the quadtree partition in which the block splitting was restricted to two levels. Instead of automatically discarding

[4] Such as implemented in the JPEG standard [20].

the larger block prior to splitting it into four subblocks if an error threshold was exceeded, it was retained if additional transforms on up to two subblocks were sufficient to reduce the error below the threshold.

C. Horizontal-vertical

The Horizontal-Vertical (HV) partition [3, ch. 6] [5, app. A] [27] [28] (see Figure 2c), like the quadtree, produces a tree-structured partition of the image. Instead of recursively splitting quadrants, however, each image block is split into two by a horizontal or vertical line. Splitting positions may be constructed so that boundaries tend to fall along prominent edges [3, pg. 120], or based on the accuracy of approximation by constant pixel values in each of the new blocks created by a particular split [28]. Compact coding of the partition details, similar to that utilised for the quadtree partition, is possible.

D. Irregular regions

A tiling of the image by right-angled irregular-shaped ranges may be constructed by a variety of merging strategies on an initial fixed square block [29] [30] [31] [32] (see Figure 2d) or quadtree [33] partition; chain codes allow the range shapes to be coded efficiently.

E. Polygonal blocks

A number of different constructions of triangular partitions (see Figures 3a-3c) have been investigated. Starting by splitting the image into two main triangles by the insertion of a suitable diagonal, progressively smaller triangles may be placed where necessary by a 3-side split [5, app. A] in which a new vertex is created on each of the sides of an existing triangle, or by a 1-side split [34] [35] in which an existing triangle is split into two by inserting a line from a vertex of the triangle to a point on the opposite side. An alternative triangular partition is based on a *Delaunay triangulation* [36] of the image, which is constructed on an initial set of "seed points", and is adapted to the image by adding extra seed points in regions of high image variance [37] [38] [39].

Polygonal partitions have been constructed by recursive subdivision of an initial coarse grid by the insertion of line segments at various angles [40] (see Figure 3d), as well as by merging triangles, in a Delaunay triangulation, to form quadrilaterals [41].

F. Overlapped blocks

Overlapping range blocks have been used to reduce blocking artifacts, without a corresponding improvement in MSE, within a quadtree partition [42], and with multiple domain transforms (such as those described in Section III-B.4) in a fixed block size partition [43]. A more complex form of block overlapping, but with a fixed block size range partition, provided improved MSE and subjective quality [44]. These techniques, while promising, have been overtaken to a large extent by developments in wavelet domain fractal coding, reviewed in Section IX.

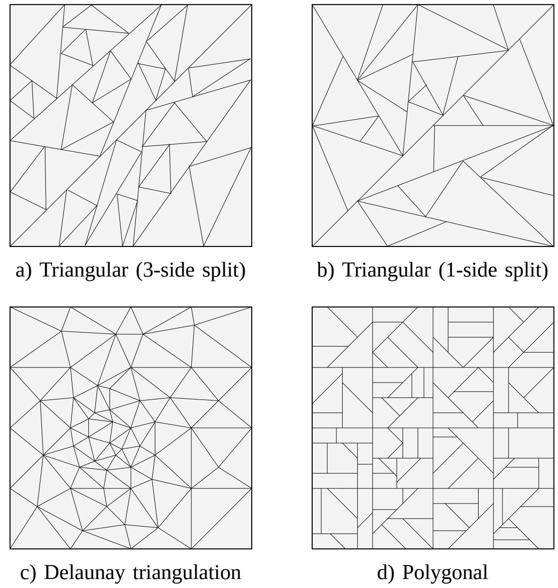


Fig. 3. Triangular and polygonal range partition schemes.

G. Comparison

The simplest partition (quadtree) was found to provide the best rate distortion results in a comparison of polygonal, HV, and quadtree partitions [40]. An independent comparison between the quadtree and HV partitions, in contrast, found the HV partition to be superior [45], while irregular partitions have been found to outperform a fixed square block partition [29] [31] as well as a quadtree partition [30] [33]. A disadvantage of partitions which are not right-angled is the interpolation required in performing the block transforms when there is no simple pixel-to-pixel correspondence between domain and range blocks.

III. Block Transforms

The type of block transform selected is a critical element of a fractal coding scheme since it determines the convergence properties on decoding, and its quantised parameters comprise the majority of the information in the compressed representation. A distinction is made here between transforms operating on the block support ("geometric" transforms in Jacquin's terminology[5] [26]) and those operating on the pixel values (termed "massic" transforms by Jacquin).

A. Block support

The permissible transforms on the block support are restricted by the block partition scheme, since domain block supports are required to be mapped onto range block supports.

1) Rectangular blocks: The block support transform for rectangular blocks may be separated into an initial spatial contraction followed, for square blocks, by one of the square isometry operations.

The spatial contraction of domains as introduced by Jacquin [25] is almost universally applied, despite being inessential for

[5]The block isometries are considered to be block support transforms here, in contrast to Jacquin's usage.

the contractivity of the image map as a whole [6] [16, pp. 127-129] [27]. While contraction by a factor of two in width and height is standard, smaller factors have also been considered [46], and increasing[6] this to a factor of three has been found to improve decoder convergence [48]. Contraction is usually achieved by the averaging of neighbouring pixels, which may be improved by the addition of an anti-aliasing filter [49]. The alternative of decimating by discarding pixels [3, pg. 141] is slightly faster, but results are inferior to those obtained by averaging [27].

The symmetry operations utilised by Jacquin are widely used as a means of enlarging the domain pool. While some authors have reported similar frequency of usage for all of the isometry operations [50] [51], others have presented evidence to the contrary [16, pp. 123-125] [52]. These conflicting results are possibly due to the sensitivity to design choices in each of the categories listed in the introduction. Despite their widespread usage, there is evidence that their application is counter-productive in a rate distortion sense [51] [53] [54] [55]. Affine transforms other than the isometries have also been considered [16, pp. 129-131], and generalised square isometries constructed by conformal mapping from a square to a disk are reported to be capable of improved performance over the true square isometries [56].

2) Non-rectangular blocks: An affine mapping on the image support is sufficiently general to transform domain triangles to range triangles in a triangular partition. These affine transforms are determined by requiring that the transformed vertices of the domain blocks match those of the range blocks. Depending on their structure, polygonal blocks may require transforms more general than affine in transforming domain to range blocks [41].

B. Block intensity

The simplest intensity transform in common use is that introduced by Jacquin

$$M\mathbf{u} = s\mathbf{u} + o\mathbf{1}, \tag{1}$$

where s and o are variable scaling and offset coefficients, $\mathbf{u}$ is a suitable vector representation [2] of the domain block after application of any block support operations such as spatial contraction, and $\mathbf{1}$ is a vector of unit components.

1) Orthogonal projection: The subtraction of the DC component of the domain block prior to scaling [3, ch. 8] [57]

$$M\mathbf{u} = s\left(\mathbf{u} - \frac{\langle \mathbf{u}, \mathbf{1}\rangle}{\|\mathbf{1}\|^2}\mathbf{1}\right) + o\mathbf{1}, \tag{2}$$

(where $\langle\cdot,\cdot\rangle$ and $\|\cdot\|$ are the inner product and derived norm of an appropriate inner product space - usually l^2) creates transformed domains which are orthogonal to the fixed block $\mathbf{1}$, with the desirable effect of decorrelating the s and o coefficients. In addition, convergence at the decoder within a fixed number of iterations may be guaranteed by imposing the additional restrictions of a quadtree range partition, a domain pool constructed so that every domain block contains an integer number of range blocks, and spatial contraction by pixel averaging [3, pg. 160].

2) Frequency domain: Selective manipulation of the block spectral contents is allowed by the transform [49] [58] [59]

$$M\mathbf{u} = C^{-1}\left[\begin{pmatrix} a_0 & 0 & 0 & \dots \\ 0 & a_1 & 0 & \dots \\ 0 & 0 & a_2 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} C\mathbf{u} + \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \end{pmatrix}\right], \tag{3}$$

where C is the Discrete Cosine Transform (DCT) matrix. Adaptivity to block activity levels may be achieved by varying the number of a_i and b_i that are individually specified, the remainder being set to zero. This hybrid scheme constitutes a transition between conventional fractal coding and transform coding, being equivalent to the former when all of the a_i are equal, and only b_0 is non-zero, and to the latter when a full set of b_i values is utilised, and all of the a_i are zero.

Alternative hybrids between fractal and transform coding have been constructed by DCT coding of the error image resulting from fractal coding [19] [60].

3) Multiple fixed blocks: Instead of the usual single fixed block $\mathbf{1}$, multiple fixed blocks $\mathbf{v}_i$ may be employed in the transform

$$M\mathbf{u} = a\mathbf{u} + \sum_i b_i \mathbf{v}_i. \tag{4}$$

Orthogonalisation of the domain block term with respect to the fixed block terms may be achieved by projecting the domain block perpendicular to the subspace spanned by the fixed domain blocks [3, ch. 8].

Transform (1) may be extended by including fixed blocks with constant gradient in the vertical and horizontal directions respectively [61] [62]. Further extensions [18] [51] [63] to "order 2 polynomials" by including blocks with quadratic form, and to "order 3" with the addition of cubic form blocks have also been considered. The "order 2" transform was found to be best, in a rate distortion sense, in experiments with limited domain searching [64].

If all of the a_i in the frequency domain transform (3) are equal, it becomes equivalent to transform (4) with DCT basis vectors as the fixed blocks (such a transform, with orthogonalisation with respect to the subspace spanned by the first few DCT basis vectors, has been examined [65] [66]). Although no explicit comparison has been made between the use of polynomial or DCT basis fixed blocks, in the absence of experimental evidence the DCT basis blocks are likely to be superior, since they are known to form an efficient basis for image blocks and, unlike the polynomial bases, are mutually orthogonal.

4) Multiple domains: A transform constructed by adding independently scaled *domain* blocks has also been considered [3, ch. 10]. Computational tractability was achieved by creating an orthogonal basis of the domain block set, representing each range by a scaling of as few basis vectors as possible. A variety of mappings using multiple fixed blocks as well as multiple domain blocks, including domain blocks with no spatial contractivity, have also been investigated [67] [68] [69],

[6] It is also possible, by an appropriate choice of spatial contractivity, to achieve decoding by a single iteration of the transform [3, pp. 171-172] [47, pp. 56-60].

a linear combination of these blocks being selected via a technique known as matching pursuit.

IV. Domain Pool Selection

The domain pool used in fractal compression is often referred to as a virtual codebook [52], in comparison with the codebook of Vector Quantisation (VQ) [70]. It should be clear from this comparison that a suitable domain pool is crucial to efficient representation since, although increased fidelity may be obtained by allowing searching over a larger set of domains, there is a corresponding increase in the number of bits required to specify the selected domain.

A bound $|s| \leq s_{\max}$ is usually placed on the block intensity transform scaling coefficients in order to guarantee contractivity (see Section V-B), in which case a scaling coefficient exceeding this bound is set equal to it prior to calculating the distance between the transformed domain and the range.

A. Global domain pool

The simplest domain pool design provides a fixed domain pool for all range blocks in the image, or for a particular class of range blocks distributed throughout the image (eg. range blocks of one size in a quadtree partition). This design choice is motivated by experiments indicating that the best domain for a particular range is not expected to be spatially close[7] to that range to any significant degree [3, pp. 69-71] [27, pp. 56-57] [50] (it is clear from the following section, however, that there is some disagreement over this issue).

In the fixed square block or quadtree partitions, domain blocks may be placed at intervals as small as one pixel. Since this results in an enormous domain pool which is slow to search, larger domain increments are usually selected, typically equal to the domain block width [3, ch. 3] [17] [27] or half the domain block width [3, ch. 3, 4] [24]. Improved convergence is also obtained with either of these increments [3, ch. 8], while the larger of the two corresponding domain pools was found to be superior in fidelity and compression ratio [27].

In adaptive partitions the domain pool usually consists of the larger blocks in the range pool [27], or larger blocks created by the same partitioning mechanism [29] [41].

B. Local domain pool

A number of researchers have noticed a tendency for a range block to be spatially close to the matching domain block [48] [49], based on the observed tendency for distributions of spatial distances between range and matching domain blocks to be highly peaked at zero [52] [64] [71]. Motivated by this observation, the domain pool for each range block may be restricted to a region about the range block [24], or a spiral search path may be followed outwards from the range block position [48] [49]. More complicated alternatives include using a domain position mask, centred at each range block, with positions in the mask dense near the centre and progressively less dense further away, and using widely spaced domain blocks together with a fine lattice in the vicinity of the best match in the coarse lattice [71].

The domain search may also be dispensed with entirely, either by selecting each domain in a fixed position relative to the range [61] [62], or by placing the domain so that it contains the range, and the dominant edge is in the same relative position in both blocks [65]. The search may also be restricted to a very small set about the range block [18]. There is some evidence that local codebooks outperform global ones [16, pg. 122], and that any domain searching is counter-productive in a rate distortion sense [64] for the "order 2" polynomial transform described in Section III-B.3.

C. Synthetic codebook

In a significant variation[8] from standard fractal coding, the domain pool may be extracted from a low resolution image approximation (which is coded independently), rather than from the image itself [72] [73] [74]. Decoding does not require iteration, and the collage error minimised at the encoder is also the true distortion.

D. Hybrid codebooks

A coding scheme allowing range blocks to be represented either as mappings from domain blocks or a fixed VQ codebook was found to perform significantly better[9] than when the VQ option was excluded, and slightly better than when the fractal option was excluded [75].

E. Comparison

The question of domain locality (the tendency for a range and matching domain to be spatially close) plays an important role in the design of an efficient domain pool. While the degree to which this effect is present may be dependent on the particular fractal coding scheme for which it is evaluated, this does not adequately explain the extent of the disagreement in the literature (see reference [76, pp. 46-47, 75-77] for further discussion of this issue).

V. Encoding

Fractal coding is achieved by representing a signal $\mathbf{x}$ by a quantised representation of a contractive transform T which is chosen such that the fixed point $\mathbf{x}_T$ of T is close to $\mathbf{x}$. Although $\mathbf{x}_T$ may be recovered from T by the iterative process described previously, there is usually no simple expression for $\mathbf{x}_T$ in terms of its quantised coefficients. As a result, and given the constraints on T imposed by its dependence on its constituent coefficients, it is not usually possible to optimise those coefficients to make the fixed point as close as possible to a given signal $\mathbf{x}$.

[7] Note that this is the distance (measured in pixels) in the image support between the range and domain block centres, and *not* the distortion resulting from representing the range block by that particular domain block (the collage error for that range block).

[8] This technique does not, strictly speaking, fall within the scope of fractal coding, since the representation is not in any sense a fractal.

[9] Encoder and decoder speeds were also improved.

A. The collage theorem

Since the the distortion $\|\mathbf{e}_T\|$ (where $\mathbf{e}_T = \mathbf{x} - \mathbf{x}_T$) introduced by the fractal approximation can usually not be directly optimised for these reasons, the standard approach is to optimise T to minimise the collage error $\|\mathbf{e}_C\|$ (where $\mathbf{e}_C = \mathbf{x} - T\mathbf{x}$), which is usually computationally tractable. The collage theorem guarantees that $\|\mathbf{e}_T\|$ may be made small by finding T such that $\|\mathbf{e}_C\|$ is sufficiently small[10].

The most common form of the collage theorem is

$$\|\mathbf{e}_T\| \leq (1-\alpha)^{-1}\|\mathbf{e}_C\|,$$

where T is a contractive transform with Lipschitz factor α (i.e. $\|T\mathbf{x} - T\mathbf{y}\| \leq \alpha\|\mathbf{x} - \mathbf{y}\|$). In image coding terms this implies that a transform T, for which the fixed point $\mathbf{x}_T$ is close to an original image $\mathbf{x}$, may be found by designing the transform T such that the "collage" $T\mathbf{x}$ is close to $\mathbf{x}$, achieved by minimising the collage error individually for each range block. A similar bound is possible for *eventual contractivity* [3, ch. 2], while a tighter collage bound is possible by imposing certain restrictions, consisting primarily of requiring DC subtraction in the block transform and setting the domain increment to be equal to the range block size [3, ch. 8] [57]. Despite the considerable improvement over the usual collage theorem bound, this bound is still rather loose [57].

The majority of existing fractal coding schemes restrict T to be an affine transform $T\mathbf{x} = A\mathbf{x} + \mathbf{b}$, where A is a linear transform (encapsulating the combined effects of the spatial contractions, isometry operations, and scalings of the individual domain to range mappings) and $\mathbf{b}$ is an offset vector (composed of the offsets in each of the individual domain to range mappings) [3, ch. 7]. In this case $\mathbf{e}_C = (I - A)\mathbf{e}_T$, and bounds[11]

$$(1 + \|A\|)^{-1}\|\mathbf{e}_C\| \leq \|\mathbf{e}_T\| \leq (1 - \|A\|)^{-1}\|\mathbf{e}_C\|$$

may be derived, in terms of an operator norm $\|A\|$ consistent with the vector norm [78], by noting that that $|\,\|\mathbf{u}\| - \|\mathbf{v}\|\,| \leq \|\mathbf{u} - \mathbf{v}\| \leq \|\mathbf{u}\| + \|\mathbf{v}\|$ for arbitrary vectors $\mathbf{u}$ and $\mathbf{v}$.

B. Convergence

It is clearly desirable that the encoding process produce a transform for which the decoding sequence is guaranteed to converge; while necessary and sufficient conditions for convergence are known, their computation during coding is generally not feasible, posing a significant problem for a practical encoder implementation. Contractivity under the $\sup$ norm may be guaranteed[12] by setting $s_{\max} < 1$ for each of the block transforms of which the image transform T is composed [3, ch. 2] [6, pp. 207-210]. This restriction is sufficient but not necessary for convergence, and empirical evidence indicates that convergence is often achieved for larger values of $s_{\max}$, for which reduced distortion is obtained in reconstruction [27], although smaller values of $s_{\max}$ provide more rapid convergence on decoding [3, pg. 62]. A disadvantage of increasing $s_{\max}$ is a corresponding increase in the cost or distortion in quantising the scaling coefficients.

1) Orthogonalisation: The introduction of an orthogonalisation operator to each domain block, making it orthogonal to the constant blocks, results in a transform which (given a few additional constraints) may be shown to converge exactly within a fixed number of iterations [3, ch. 8].

2) Mapping cycles: The interdependence between ranges at one iteration of decoding and domains at the next may be analysed in terms of "mapping cycles", each of which consists of an independent set of domain to range mappings [79] [80] [81]. The full image transform is convergent if each of its independent cycles is convergent.

3) Transform eigenvalues: When the transform is affine, a necessary and sufficient condition for convergence of the transform sequence on decoding is that the spectral radius[13] of the linear part be less than unity (equivalent to eventual contractivity) [78] [79] [82] [83]. It is possible, in simple cases, to determine the spectral radius in terms of the transform parameters, allowing analytic determination of convergence requirements on the transform coefficients. While computation of the spectral radius is difficult for the general case, the probability of contractivity may be estimated by considering a statistical distribution for the eigenvalues, based on probability distributions for the transform parameters [79].

C. Optimal encoding

Although the collage theorem currently forms the basis of virtually all fractal coders, it does not result in an optimal image representation given the constraints imposed on the transform. Suboptimality is, amongst others, a result of optimisation of individual block transforms with respect to the domains in the original image, whereas only the fixed point domains are available during decoding. It has been shown that optimal encoding is NP-hard [84], and that collage based coding may produce a solution of arbitrary distance from the optimal solution. Collage based encoding may, however, be shown to be optimal under certain restrictions [3, ch. 8] [47].

Updating the scaling and offset coefficients after coding, by re-optimising them with respect to domains extracted from the decoded image, was found to result in reduced distortion on reconstruction [49], as was a scheme involving multiple compression stages during each of which domains were extracted from the decoded image of the previous stage [16, pp. 81-82]. Improvements due to more computationally intensive optimisation techniques have also been reported [85] [86].

VI. Search Strategies

The significant computational requirements of the domain search resulted in lengthy coding times for early fractal

[10] It is important to note that the collage error $\|\mathbf{e}_C\|$ is usually smaller than the actual distortion $\|\mathbf{e}_T\|$ [76, pp. 87-88] [77], whereas the various forms of the collage theorem provide an upper bound *in terms of* the collage error.

[11] Note that the upper bound is only valid when $\|A\| < 1$.

[12] Note however, that, since it is not additive, the $\sup$ norm is not appropriate for independent blockwise collage minimisation, which is usually performed under the l^2 norm.

[13] The spectral radius $r(A)$ of linear transform A is the maximum absolute value of the eigenvalues of A.

compression algorithms. The design of efficient domain search techniques has consequently been one the most active areas of research in fractal coding, resulting in a wide variety of solutions. The survey presented here is rather brief due to space restrictions; the reader is referred to a comprehensive review [87] of these techniques for further details.

A. Invariant representation

The search for the best domain block for a particular range block is complicated by the requirement that the range matches a *transformed* version of a domain block; the problem is in fact to find for each range block, the domain block that can be *made* the closest by an admissible transform. Given a set of domain blocks $\mathbf{d}_i$ and the admissible transforms $M_\mathbf{p}$ parameterised by $\mathbf{p}$, the optimum domain block for range block $\mathbf{r}$ results in a collage error of $\min_{\mathbf{p},i} \|\mathbf{r} - M_\mathbf{p}\mathbf{d}_i\|$.

The problem may be simplified by constructing an appropriate invariant representation for each image block. Transforming range and contracted domain blocks to this representation allows direct distance comparisons between them to determine the best possible match [88].

The standard invariant representation for the block intensity transform[14] is constructed by applying the orthogonal projection onto the orthogonal complement of the space spanned by the fixed block terms, followed by normalisation. Alternative representations[15] for the single constant block transform utilise the DCT (or another orthogonal transform) of the vector followed by zeroing of the DC term and normalisation. This representation can decrease the time required for an efficient domain search [15, ch. 6] [48] [91], and allows the utilisation of a distance measure adapted to the properties of the human visual system [16, pp. 190-193] [48] [58].

B. Domain pool reduction

One of the simplest ways of decreasing coding time is to decrease the size of the domain pool in order to decrease the number of domains to be searched, which is often achieved by a spatial constraint on the domain pool for each range, as described in Section IV-B. Noting that a contractive mapping requires a domain with a higher variance than the range to which it is mapped, domains with low variance may be excluded from the domain pool [92]. Alternatively, the domain pool may be pruned in order to exclude domains which have similar invariant representations [93] to other domains in the pool.

C. Classification

Classification based search techniques often do not explicitly utilise an invariant representation as formalised above, but rely instead on features which are at least approximately invariant to the transforms applied. Domain and range blocks may either be classified into a fixed number of classes according to these features [3, ch. 3] [24] [25] [71], a matching domain for each range only being sought within the same class, or inspection of domains may be restricted to those with feature values close to those of the range [34] [50].

[14]An appropriate invariant representation with respect to the block isometries is not possible, although invariant *features* are [89] [90].

[15]These alternatives are equivalent to the standard representation in a different basis.

D. Distance bounds

Instead of locating likely matches, impossible matches may be excluded by utilising features in terms of which distance inequalities are available. Examples include inner products with a fixed set of vectors [6] which provide lower bounds on distances between domain and range blocks, allowing many of the domains to be excluded from the actual distance calculation, and features based on the distribution of energy within image blocks [94].

E. Multiresolution search

A tree search has been applied to a pyramid of progressively coarser resolution domains, the search at each level progressing in the region of the best match in the previous level [95] [96]. A similar technique, using collage errors at coarse resolutions as lower bounds for those at finer resolutions, has also been implemented [15, ch. 7] [97].

F. Clustering

Clustering of the domain blocks, under a distance measure invariant to the block transforms, allows a fast search by locating the optimum cluster centre and then the optimum domain within that cluster. The Generalised Lloyd Algorithm [3, ch. 9] [37] [47], the Pairwise Nearest Neighbour algorithm [98] and Self-Organising Maps [99] have been utilised in the construction of these clusters. The computational cost of clustering during encoding may be avoided by designing the clusters on an initial training set rather than determining them adaptively for each image [3, ch. 4] [99].

G. Efficient distance computation

Since a significant fraction of the computational cost of the domain search lies in the actual calculation of distances between domain and range blocks, the time required for the search may be reduced by improving the efficiency of these calculations.

A simple technique for decreasing search time is the *partial distance* [70, pp. 479-480] method used in VQ. The efficiency of this search is improved by constructing an invariant representation from Hadamard transform coefficients in zig-zag scan order [48], since the energy packing property of this transform shifts most of the variance to the initial elements of the vector. A similar approach based on the Haar transform has also been investigated [94].

Efficient computation of the inner products between domain and range blocks can result in a significant improvement, since these calculations dominate the computational cost of the distance computations. These calculations may be efficiently performed in the frequency domain by considering the calculation of the inner products between a particular range block and all domain blocks as a convolution of the image with that range block [100].

H. Nearest neighbour search

Efficient nearest neighbour search techniques utilise a pre-processing stage to arrange the set to be searched in an appropriate data structure, usually a tree representing a hyperplane induced partition of the search space, allowing the vector in the search set closest (the invariant representation of range and domain blocks is used) to the specified vector to be located without actually examining every point in the set. Existing techniques [101, ch. 2, 3] [102] have been applied to domain searching [15, ch. 6] [16, pp. 179-200] [88] [103] [104], as have algorithms specifically designed for this purpose [94] [105] [106].

VII. Transform Representation

Domain positions, and any additional partition information required in an adaptive partition, are represented by discrete values and are not subjected to quantisation. There are usually compact methods of representing the range partition details in adaptive partitions such as quadtree or HV [3, ch. 3, 6]. Efficient representation of the domain positions [16, pp. 114-121, 132-133] may be achieved by indexing in decreasing order of probability of a match, as in the spiral search [49] described in Section IV-B, a Finite State approach based on the corresponding VQ technique [70, ch. 14] also having been considered [107].

A. Quantisation

Although the distributions for the scaling and offset coefficients have been observed to be non-uniform, quantisation is usually uniform [3, ch. 3] [17], but with the possibility of compensation for inefficiency by subsequent entropy coding. Bit allocations[16] for the scaling and offset coefficients have been respectively 2 and 6 [52], 5 and 8 [108], and between 2 and 4 for the scaling and between 3 and 8 for the offset [9]. An allocation of 5 and 7 bits to the scaling and offset coefficients respectively provided the best performance in a comparison over a number of bit allocations [3, pp. 61-65].

Logarithmic [3, pg. 63] and pdf optimised [108] quantisation of the scaling coefficients have been investigated, the former not resulting in an improvement over uniform quantisation, with which the latter was not compared. Since the scaling coefficients are often rather coarsely quantised, there is a significant advantage in calculating collage errors for each domain block using quantised transform coefficients [27, pg. 45] [108], although this may be difficult to achieve for some of the fast domain search methods [15, ch. 6].

It has been observed that the standard block transform (without DC subtraction[17]) results in correlated scaling and offset coefficients [9] [109]. Alternative responses to this observation have been VQ of combined scaling and offset coefficients [109] [110], and linear prediction of the offset from the scaling [9]. Since there is usually also some correlation between the offset coefficients for neighbouring blocks, some form of predictive coding is indicated [16, pp. 140-144], but presents practical difficulties for some range partitions [108].

Quantisation optimisation has also been investigated for polynomial fixed block transforms [51] [63], and VQ of the transform coefficients has been considered [49] for the frequency domain transform.

[16]Constant scaling coefficients fixed at 0.50 [53] and 0.75 [16, pp. 156-159] have also been used, and the scaling coefficients have been restricted to the set $\{0.0, 0.5, 1.0\}$ in a hybrid scheme [19].

[17]The same transform with DC subtraction does not result in a significant correlation between these coefficients [108].

B. Rate-distortion optimisation

An adaptive block coding technique may provide a number of options (eg. either splitting the block into smaller blocks or adding additional fixed blocks into the block transform), each associated with a different cost in bits, for reducing the distortion in representing a particular block. In this case the appropriate choice is not the option providing the lowest distortion, but the option for which the ratio between the decrease in distortion and the associated bit cost is the greatest.

Such rate-distortion optimisation has been applied in the selection between adaptive block transforms [58] [66], in the construction of an optimum range partition [16, pp. 93-105] [28] [66], in the selection of local domain search regions [16, pp. 114-123], in the selection of an optimum linear combination of basis blocks [69], and in the decision whether a mapping from a domain block is beneficial in a hybrid coding scheme [65]. As a result of the encoding difficulties (described in Section V) necessitating the use of the collage theorem, complete rate-distortion optimisation over all components of the representation is, however, usually impractical.

VIII. Decoding

Reconstruction of the encoded image is achieved by computing the fixed point of the image transform T from its encoded coefficients. Since the encoded representation of a transform may be independent of the size of the encoded image, a form of interpolation is possible by reconstructing the fixed point at a higher resolution than the encoded image [3, pg. 59].

A. Standard decoding

Reconstruction of the fractal coded approximation of a signal is theoretically based on Banach's fixed point theorem which guarantees that the sequence constructed by the iterative application of a contractive transform T to an arbitrary initial element $\mathbf{x}_0$ of a complete metric space converges to the fixed point $\mathbf{x}_T = \lim_{n\to\infty} T^n \mathbf{x}_0$ of that transform.

When the transform T is affine, with $T\mathbf{x} = A\mathbf{x} + \mathbf{b}$, the fixed point may, in principle, be expressed as $\mathbf{x}_T = (I - A)^{-1}\mathbf{b}$ if $|I - A| \neq 0$ (equivalent to the condition that A has no eigenvalues equal to 1). If the spectral radius $r(A) < 1$, a Taylor series expansion of the term $(I - A)^{-1}$ provides an alternative derivation of the reconstruction series $\mathbf{x}_T = \mathbf{b} + A\mathbf{b} + A^2\mathbf{b} + \ldots$ resulting from iterated application of the transform T to an initial zero vector.

B. Successive correction decoding

Improved decoding speed has been achieved by a successive correction scheme (such as Gauss-Seidel [111]), updating each range block in place as soon as the corresponding domain is mapped to it, rather than mapping the domains into a temporary image on each iteration [53] [112] [113]. This technique was found to provide a further improvement when decoding of range blocks was ordered so that regions containing the most highly utilised domain blocks were decoded first on each iteration [112] [114].

C. Hierarchical decoding

If the domain increment is equal to the range block size, a PIFS may be iteratively decoded to a minimum-length vector in which each range block consists of a single pixel. Given a few additional restrictions [3, pg. 95], one may consider the domain to range mappings as providing a relationship between consecutive resolution approximations in the Haar wavelet basis. This relationship provides an algorithm in which the range block dimensions are doubled at each step, until the desired size is reached [3, ch. 5] [115] [116]; a considerable computational saving is obtained over applying the standard iterative method to full-sized blocks.

D. Pixel chaining

If spatial contraction is achieved by subsampling, each pixel (considered as part of a range block) has a single associated reference pixel (in the corresponding domain block) from which it is mapped by the image transform T. Since the reference pixel itself has an associated reference pixel, a chain of associated pixels may be constructed in this way. These chains may be utilised in decoding by either tracing back the path of influence of a pixel until a known pixel value is encountered, or by utilising a segment of the chain long enough to provide an acceptable approximation of the desired pixel value [3, pp. 305-307] [16, pp. 207-210].

E. Postprocessing

Postprocessing in the form of smoothing along block boundaries has been found to be beneficial in reducing blocking artifacts [3, pg. 59] [16, pp. 222-224].

F. Resolution independence

While “resolution independence” has been cited in the popular technical press as one of the main advantages of fractal compression [117], there is little evidence for the efficacy of this technique. Subsampling an image to a reduced size, fractal encoding it, and decoding at a larger size has been reported to produce results comparable to fractal coding of the original image [21], although there is no indication that replacing the fractal interpolation stage by another form of interpolation would not produce comparable results. Comparisons with classical interpolation techniques indicate that, while fractal techniques result in more visually acceptable straight edges than linear interpolators, they are inferior in terms of the MSE measure [118]. An alternative study [119] found slightly better results for the fractal technique in isolated cases, but a general superiority for the classical techniques.

IX. Wavelet Analysis

The most significant recent development in fractal coding theory is the independent discovery by a number of researchers of a multiresolution analysis description of certain classes of fractal coding [77] [120] [121] [122]. This discovery has not only resulted in improved fractal coders, but a better understanding of the mechanism underlying standard fractal coding.

A. Mappings between wavelet subtrees

If the domain increment is equal to the domain block size, and subject to a few additional restrictions [3, pg. 95], there is a direct correspondence between the domain and range blocks (without DC component) in a signal, and subtrees rooted at consecutive resolutions in the Haar wavelet transform of that signal (essentially in an extension of the analysis described in Section VIII-C). The domain to range mappings may be expressed as mappings between subtrees if the block transform (2) with DC subtraction is used, a domain subtree being mapped to a range subtree by scaling the detail coefficients, shifting the entire subtree one resolution higher, and discarding the highest resolution detail coefficients.

The same analysis may be extended to images by considering the non-standard [123, pp. 313-316] extension of the Haar basis to two-dimensions, in which subtrees in each of the directional subbands are combined to form a composite subtree (see Figure 4). The square isometries may also be applied within this framework [77].

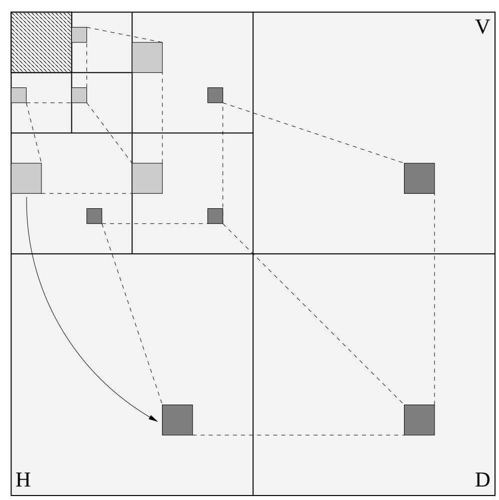


Fig. 4. Detail coefficient extrapolation by mappings between subtrees.

Encoding is achieved by locating the best matching domain subtree for each range subtree, in the sense that the MSE distance between the range subtree and appropriately scaled domain subtree is minimised. Decoding within this framework is achieved within a fixed number of iterations, since the corresponding linear operator is strictly lower triangular below a few initial rows - convergence problems for small domain increments may be seen as a result of dependency loops from high to low resolution detail coefficients [124] [125].

B. General wavelet bases

This interpretation of fractal coding naturally suggests the substitution of a smooth wavelet basis for the Haar basis. Note, of course, that strict correspondence with standard fractal coding breaks down under this extension, particularly for biorthogonal bases, where spatial and transform domain energies are not equal. Such an extension was found to reduce blocking artifacts and improve the reconstruction MSE [77] [120] [126].

A number of hybrid coders have been implemented, combining the subtree mapping of fractal coding with scalar quantisation techniques of varying complexity [122] [124] [127] [128].

C. Alternative schemes

In contrast to the generalisation of the usual *subtree* prediction described above, a *subband* prediction scheme in the non-standard image decomposition has also been proposed [129] [130]. Each image subband is covered by range blocks which are mapped from domain blocks of the same size from the next lower resolution subband (see Figure 5). Since each subband is predicted from the *coded* version of the previous subband, contractivity is not required[18], and the coding error may be evaluated at coding time. Low resolution subbands and residual errors after block prediction were coded by Laplacian scalar quantisers [130], or by a sophisticated Lattice Vector Quantisation technique [129].

Standard fractal coding of individual subbands (i.e. domain and range blocks are extracted from the same subband) has also been considered [131]. Block shapes within each subband were designed to reflect the correlation structure within that subband, the blocks in the horizontal directional subbands being horizontally elongated, for example.

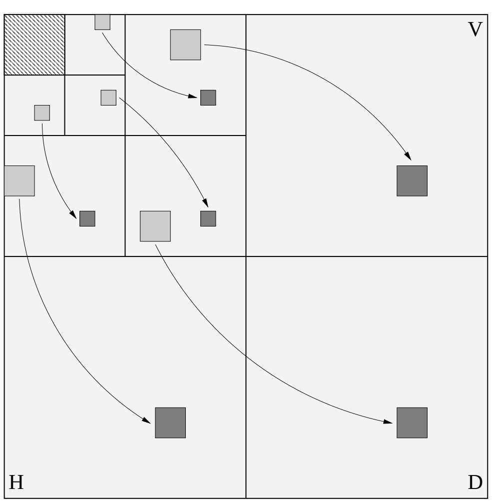


Fig. 5. Mappings between subbands (note that these are not constrained by the tree structure as in Figure 4) in the subband prediction algorithm [130].

[18]As a result, it is not, strictly speaking, fractal coding, despite the considerable similarities.

X. Performance Comparisons

The greatest difficulty in comparing results of different lossy coding algorithms is the absence of an objective distortion measure which accurately reflects perceived distortion. A further complication in the comparison of fractal coding algorithms is the scarcity of theoretical results to support design choices; as a result, most stages of coder design are based on empirical studies, and the lack of consensus on important issues is probably largely a result of the dependence between different aspects of fractal coder design referred to in the introduction. Coder design by a “greedy algorithm” which optimises each stage separately is therefore bound to fail.

Since the most widely used test image is the 8 bits/pixel 512×512 Lena image, PSNR (Peak Signal-to-Noise Ratio) results published for coding of this image may be used as a basis for comparison[19], as displayed in Figures 6 and 7, between a variety of coding schemes. The wide range in performance is striking, a number of the more effective algorithms offering performance comparable to that of Shapiro’s EZW algorithm [132], which is often used as a benchmark in the recent literature. While it is difficult to identify the primary factors responsible for the superior performance of the better algorithms, a few general tendencies may be observed:

Partition The best algorithms tend to utilise quadtree partitions or their wavelet domain equivalents, although one of the irregular partition algorithms also offers superior performance. None of the non-right-angled partitions offer competitive performance.

Transform The majority[20] of the best algorithms either operate in the wavelet transform domain, or utilise frequency domain block transforms.

Transform Representation As might be expected, attention to quantisation of transform parameters, and rate-distortion optimisation strategies appear to play a significant role in improving performance.

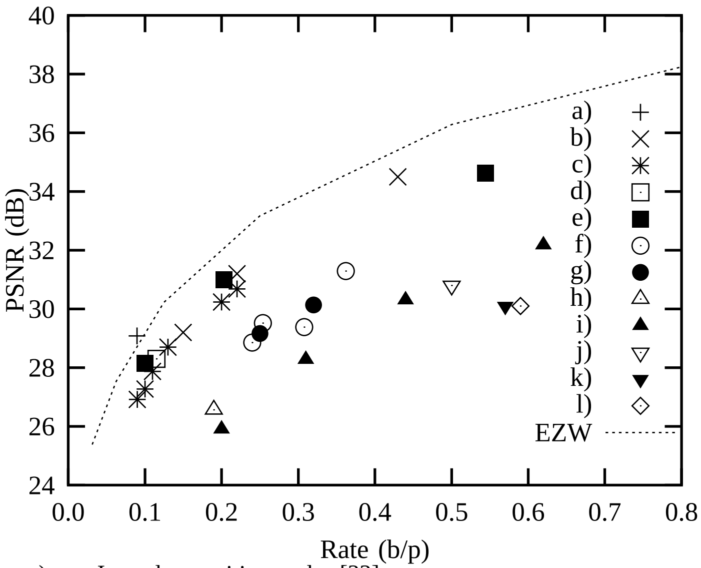
Hybrids Many of the best algorithms are constructed as hybrids of fractal coding and alternative techniques. In many of these cases, and in particular, for the coder in Figure 7a, the role of the fractal part in these hybrids is relatively small.

XI. Conclusions

Despite the considerable attention received by the technical aspects of constructing a fractal representation of an image, it is certainly not clear why a contractive transform should be expected to provide an efficient representation for natural images [133]; most authors assume, without direct evidence,

[19]Caution should be exercised in evaluating this comparison. First, there are, unfortunately, different versions of the same image in common use, one of which is significantly easier to code than the other. Second, PSNR is an unreliable measure of perceived image quality, and while its definition involves the dynamic range of an image, this value is usually taken as 255, despite images such as Lena not utilising the full 8 bits available. Finally, the exclusion of algorithms for which published results for this image were not available makes a fair comparison across all schemes impossible.

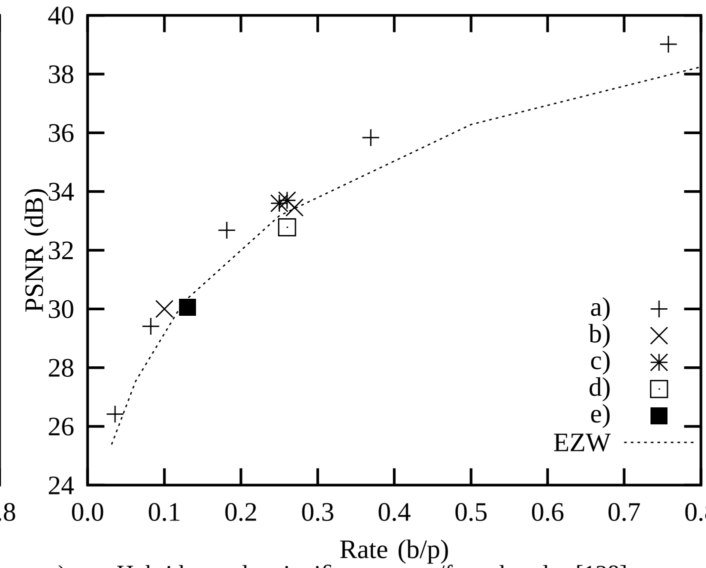
[20]The notable exception of Figure 6a was probably tested on the more easily coded Lena image.



a) Irregular partition coder [33].
b) Multiple domain transform [69].
c) Quadtree partition coder with VQ of transform parameters [110].
d) Irregular partition coder [32].
e) HV partition coder [3, ch. 6].
f) Quadtree partition coder [3, ch. 3].
g) Triangular partition coder [41].
h) Irregular partition coder [29]
i) Quadtree partition coder [21].
j) Fixed square block partition coder [17].
k) Original Jacquin[a] 2-level coder [24].
l) Triangular partition coder [35].
EZW EZW coder results [132].

[a] In a later publication [52] Jacquin reports on a similar coder for which a PSNR of 31.4dB is achieved at a rate of 0.06 b/p; since the coder described is very similar to the earlier one [24], it is likely that a typographical error transformed 0.6 b/p into 0.06 b/p.

Fig. 6. Comparison of the performance of fractal coding (pure fractal coders) and EZW for the 8 b/p 512×512 Lena image.



a) Hybrid wavelet significance map/fractal coder [128].
b) DCT domain block transform coder [49].
c) Subband block prediction with PVQ [129].
d) Subband block prediction with scalar quantisation [130].
e) Hybrid wavelet scalar quantisation/fractal coder [124].
EZW EZW coder results [132].

Fig. 7. Comparison of the performance of fractal coding (hybrid coders) and EZW for the 8 b/p 512×512 Lena image.

that natural images exhibit significant "self-affinity". Nevertheless, an understanding of the statistical image model[21] underlying fractal compression, together with associated flaws, has recently begun to emerge.

Motivation for the representation has been proposed in terms of comparisons with alternative techniques such as predictive coding [134], classified transform coding [135], and VQ [47, ch. 5]. More direct statistical examination [76] [127, ch. 4] has revealed the role played by the second order statistics of the image model, a decaying power spectrum and the statistical self-similarity of fractional Brownian motion models being most significant [125]. There is evidence, however, that the underlying model does not represent a particularly accurate characterisation of natural images [136]. An optimised VQ codebook generally outperforms the domain pool of a fractal representation, and domain pools extracted from *different* images are generally no less effective than those extracted from the *same* image as the range blocks [47, ch. 5] [76] [137].

[21] Representing a coherent description of the image statistics required for fractal coding to be effective.

Furthermore, fractal coding is less effective than transform coding for the underlying models of transform coding, even when these models are statistically self-similar [76] [138], and it appears as if the simpler zerotree recently introduced in wavelet scalar quantisation [132] is able to account for similar higher-order dependencies to those represented by the underlying fractal coding model [125] [127, ch. 4].

While the performance comparisons presented here imply that the better fractal coders offer rate distortion performance at least comparable with the current state of the art, it should be noted that the majority of these algorithms are not classical fractal coders relying purely on image self-affinity, but incorporate the ability to exploit alternative forms of redundancy for which there is better evidence. It remains to be seen whether fractal compression captures any statistical property of natural images which can not be exploited as effectively by alternative techniques.

Acknowledgement

The authors wish to thank the anonymous reviewers for valuable comments on the manuscript.

References

[1] E. Kreyszig, *Introductory Functional Analysis with Applications.* New York, NY, USA: John Wiley & Sons, 1978.
[2] W. K. Pratt, "Vector space formulation of two-dimensional signal processing operations," *Computer Graphics and Image Processing*, vol. 4, pp. 1–24, Sept. 1975.
[3] Y. Fisher, ed., *Fractal Image Compression: Theory and Application.* New York, NY, USA: Springer-Verlag, 1995.
[4] M. Barnsley, *Fractals Everywhere.* San Diego, CA, USA: Academic Press, 1988.

[5] H.-O. Peitgen, H. Jürgens, and D. Saupe, *Chaos and Fractals: New Frontiers of Science*. New York, NY, USA: Springer-Verlag, 1992.

[6] T. Bedford, F. M. Dekking, and M. S. Keane, "Fractal image coding techniques and contraction operators," *Nieuw Archief Voor Wiskunde*, vol. 10, pp. 185–217, Nov. 1992.

[7] M. F. Barnsley and L. P. Hurd, *Fractal Image Compression*. Wellesley, MA, USA: AK Peters, 1993.

[8] K. Belloulata, A. M. Baskurt, H. Benoit-Cattin, and R. Prost, "Fractal coding of medical images," in *Medical Imaging X: Image Display* (Y. Kim, ed.), vol. 2707 of *SPIE Proceedings*, (Newport Beach, CA, USA), pp. 598–609, Feb. 1996.

[9] B. Hürtgen, P. Mols, and S. F. Simon, "Fractal transform coding of color images," in *Visual Communications and Image Processing '94* (A. K. Katsaggelos, ed.), vol. 2308 of *SPIE Proceedings*, (Chicago, IL, USA), pp. 1683–1691, July 1994.

[10] Y. Zhang and L.-M. Po, "Fractal color image compression using vector distortion measure," in *Proceedings ICIP-95 (IEEE International Conference on Image Processing)*, vol. III, (Washington, D.C., USA), pp. 284–287, Oct. 1995.

[11] M. S. Lazar and L. T. Bruton, "Fractal block coding of digital video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 4, pp. 297–308, June 1994.

[12] Y. Fisher, D. N. Rogovin, and T.-P. J. Shen, "Fractal (self-VQ) encoding of video sequences," in *Visual Communications and Image Processing '94* (A. K. Katsaggelos, ed.), vol. 2308 of *SPIE Proceedings*, (Chicago, IL, USA), pp. 1359–1370, Sept. 1994.

[13] D. L. Wilson, J. A. Nicholls, and D. M. Monro, "Rate buffered fractal video," in *Proceedings ICASSP-94 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 5, (Adelaide, Australia), pp. 505–508, Apr. 1994.

[14] Y. Fisher and F. Dudbridge, eds., *Fractals*, vol. 5 (Supplementary Issue), Apr. 1997. Proceedings of the 1995 NATO ASI on Fractal Image Encoding and Analysis.

[15] Y. Fisher, ed., *Fractal Image Encoding and Analysis*, vol. 159 of *NATO ASI Series. Series F: Computer and Systems Sciences*. New York, NY, USA: Springer-Verlag, 1997.

[16] N. Lu, *Fractal Imaging*. San Diego, CA, USA: Academic Press, 1997.

[17] G. E. Øien, S. Lepsøy, and T. A. Ramstad, "An inner product space approach to image coding by contractive transformations," in *Proceedings ICASSP-91 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 4, (Toronto, Canada), pp. 2773–2776, May 1991.

[18] D. M. Monro, "Class of fractal transforms," *Electronics Letters*, vol. 29, pp. 362–363, Feb. 1993.

[19] N. T. Thao, "A hybrid fractal-DCT coding scheme for image compression," in *Proceedings ICIP-96 (IEEE International Conference on Image Processing)*, vol. I, (Lausanne, Switzerland), pp. 169–172, Sept. 1996.

[20] W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standard*. New York, NY, USA: Van Nostrand Reinhold, 1993.

[21] G. Lu and T.-L. Yew, "Image compression using partitioned iterated function systems," in *Image and Video Compression* (M. Rabbani and R. J. Safranek, eds.), vol. 2186 of *SPIE Proceedings*, (San Jose, CA, USA), pp. 122–133, Feb. 1994.

[22] D. Saupe and S. Jacob, "Variance-based quadtrees in fractal image compression," *Electronics Letters*, vol. 33, pp. 46–48, Jan. 1997.

[23] D. J. Jackson, W. Mahmoud, W. A. Stapleton, and P. T. Gaughan, "Faster fractal image compression using quadtree recomposition," *Image and Vision Computing*, vol. 15, pp. 759–767, October 1997.

[24] A. E. Jacquin, "Fractal image coding based on a theory of iterated contractive image transformations," in *Visual Communications and Image Processing '90* (M. Kunt, ed.), vol. 1360 of *SPIE Proceedings*, (Lausanne, Switzerland), pp. 227–239, Oct. 1990.

[25] A. E. Jacquin, "A novel fractal block-coding technique for digital images," in *Proceedings ICASSP-90 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 4, (Albuquerque, NM, USA), pp. 2225–2228, Apr. 1990.

[26] A. E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Transactions on Image Processing*, vol. 1, pp. 18–30, Jan. 1992.

[27] Y. Fisher, E. W. Jacobs, and R. D. Boss, "Fractal image compression using iterated transforms," in *Image and Text Compression* (J. A. Storer, ed.), ch. 2, pp. 35–61, Norwell, MA, USA: Kluwer Academic Publishers, 1992.

[28] D. Saupe, M. Ruhl, R. Hamzaoui, L. Grandi, and D. Marini, "Optimal hierarchical partitions for fractal image compression," in *Proceedings ICIP-98 (IEEE International Conference on Image Processing)*, vol. 1, (Chicago, IL, USA), pp. 737–741, October 1998.

[29] L. Thomas and F. Deravi, "Region-based fractal image compression using heuristic search," *IEEE Transactions on Image Processing*, vol. 4, pp. 832–838, June 1995.

[30] D. Saupe and M. Ruhl, "Evolutionary fractal image compression," in *Proceedings ICIP-96 (IEEE International Conference on Image Processing)*, vol. I, (Lausanne, Switzerland), pp. 129–132, Sept. 1996.

[31] M. Tanimoto, H. Ohyama, and T. Kimoto, "A new fractal image coding scheme employing blocks of variable shapes," in *Proceedings ICIP-96 (IEEE International Conference on Image Processing)*, vol. 1, (Lausanne, Switzerland), pp. 137–140, Sept. 1996.

[32] M. Ruhl, H. Hartenstein, and D. Saupe, "Adaptive partitionings for fractal image compression," in *Proceedings ICIP-97 (IEEE International Conference on Image Processing)*, vol. II, (Santa Barbara, CA, USA), pp. 310–313, Oct. 1997.

[33] Y.-C. Chang, B.-K. Shyu, and J.-S. Wang, "Region-based fractal image compression with quadtree segmentation," in *Proceedings ICASSP-97 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 4, (Munich, Germany), pp. 3125–3128, Apr. 1997.

[34] M. Novak, "Attractor coding of images," in *Proceedings PCS'93 (International Picture Coding Symposium)*, (Lausanne, Switzerland), p. 15.6, Mar. 1993.

[35] M. Novak, "Attractor coding of images." Licentiate dissertation, Linköping University, Linköping, Sweden, May 1993.

[36] F. P. Preparata and M. I. Shamos, *Computational Geometry*. Texts and Monographs in Computer Science, New York, NY, USA: Springer-Verlag, 1985.

[37] F. Davoine, M. Antonini, J.-M. Chassery, and M. Barlaud, "Fractal image compression based on Delaunay triangulation and vector quantization," *IEEE Transactions on Image Processing*, vol. 5, pp. 338–346, Feb. 1996.

[38] F. Davoine, E. Bertin, and J.-M. Chassery, "From rigidity to adaptive tessellations for fractal image compression: Comparative studies," in *Proceedings of the 8th IEEE Workshop on Image and Multidimensional Signal Processing*, (Cannes, France), pp. 56–57, Sept. 1993.

[39] F. Davoine and J.-M. Chassery, "Adaptive Delaunay triangulation for attractor image coding," in *Proceedings of the 12th International Conference on Pattern Recognition*, (Jerusalem, Israel), pp. 801–803, Oct. 1994.

[40] E. Reusens, "Partitioning complexity issue for iterated functions systems based image coding," in *Proceedings EUSIPCO'94 (European Signal Processing Conference)*, vol. 1, (Edinburgh, Scotland), pp. 171–174, Sept. 1994.

[41] F. Davoine, J. Svensson, and J.-M. Chassery, "A mixed triangular and quadrilateral partition for fractal image coding," in *Proceedings ICIP-95 (IEEE International Conference on Image Processing)*, vol. III, (Washington, D.C., USA), pp. 284–287, Oct. 1995.

[42] E. Reusens, "Overlapped adaptive partitioning for image coding based on the theory of iterated functions systems," in *Proceedings ICASSP-94 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 5, (Adelaide, Australia), pp. 569–572, Apr. 1994.

[43] D. J. Bone, "Orthonormal fractal image encoding using overlapping blocks," *Fractals*, vol. 5 (Supplementary Issue), pp. 187–199, Apr. 1997.

[44] H.-L. Ho and W.-K. Cham, "Attractor image coding using lapped partitioned iterated function systems," in *Proceedings ICASSP-97 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 4, (Munich, Germany), pp. 2917–2920, Apr. 1997.

[45] Y. Fisher, T. P. Shen, and D. Rogovin, "Comparison of fractal methods with discrete cosine transform (DCT) and wavelets," in *Neural and Stochastic Methods in Image and Signal Processing III* (S.-S. Chen, ed.), vol. 2304 of *SPIE Proceedings*, (San Diego, CA, USA), pp. 132–143, July 1994.

[46] H. Kuroda, D. C. Popescu, and H. Yan, "Fast block matching method for image data compression based on fractal models," in *Visual Communications and Image Processing '95* (L. T. Wu, ed.), vol. 2501 of *SPIE Proceedings*, (Taipei, Taiwan), pp. 1257–1266, May 1995.

[47] S. Lepsøy, *Attractor Image Compression - Fast Algorithms and Comparisons to Related Techniques*. PhD thesis, The Norwegian Institute of Technology, Trondheim, Norway, June 1993.

[48] J. M. Beaumont, "Advances in block based fractal coding of still pictures," in *Proceedings IEE Colloquium: The Application of Fractal Techniques in Image Processing*, pp. 3.1–3.6, Dec. 1990.

[49] K. U. Barthel and T. Voyé, "Adaptive fractal image coding in the frequency domain," in *Proceedings of the International Workshop on*

Image Processing, vol. XLV of *Theory, Methodology, Systems and Applications*, (Budapest, Hungary), pp. 33–38, June 1994.

[50] C. Frigaard, J. Gade, T. Hemmingsen, and T. Sand, “Image compression based on a fractal theory,” Internal Report S701, Institute for Electronic Systems, Aalborg University, Aalborg, Denmark, 1994.

[51] D. M. Monro and S. J. Woolley, “Rate/distortion in fractal compression: Order of transform and block symmetries,” in *Proceedings ISSIPNN’94 (International Symposium on Speech, Image Processing and Neural Networks)*, vol. 1, (Hong Kong), pp. 168–171, Apr. 1994.

[52] A. E. Jacquin, “Fractal image coding: A review,” *Proceedings of the IEEE*, vol. 81, pp. 1451–1465, Oct. 1993.

[53] H. A. Kaouri, “Fractal coding of still images,” in *IEE 6th International Conference on Digital Processing of Signals in Communications*, pp. 235–239, 1991.

[54] D. Saupe, “The futility of square isometries in fractal image compression,” in *Proceedings ICIP-96 (IEEE International Conference on Image Processing)*, vol. I, (Lausanne, Switzerland), pp. 161–164, Sept. 1996.

[55] S. J. Woolley and D. M. Monro, “Rate-distortion performance of fractal transforms for image compression,” *Fractals*, vol. 2, no. 3, pp. 395–398, 1994.

[56] D. C. Popescu, A. Dimca, and H. Yan, “A nonlinear model for fractal image coding,” *IEEE Transactions on Image Processing*, vol. 6, pp. 373–382, Mar. 1997.

[57] G. E. Øien, Z. Baharav, S. Lepsøy, E. Karnin, and D. Malah, “A new improved collage theorem with applications to multiresolution fractal image coding,” in *Proceedings ICASSP-94 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 5, (Adelaide, Australia), pp. 565–568, Apr. 1994.

[58] K. U. Barthel, J. Schüttemeyer, T. Voyé, and P. Noll, “A new image coding technique unifying fractal and transform coding,” in *Proceedings ICIP-94 (IEEE International Conference on Image Processing)*, vol. III, (Austin, TX, USA), pp. 112–116, Nov. 1994.

[59] K. U. Barthel, “Entropy constrained fractal image coding,” *Fractals*, vol. 5 (Supplementary Issue), pp. 17–26, Apr. 1997.

[60] Y. Zhao and B. Yuan, “Image compression using fractals and discrete cosine transform,” *Electronics Letters*, vol. 30, pp. 474–475, Mar. 1994.

[61] D. M. Monro and F. Dudbridge, “Fractal block coding of images,” *Electronics Letters*, vol. 28, pp. 1053–1055, May 1992.

[62] D. M. Monro and F. Dudbridge, “Fractal approximation of image blocks,” in *Proceedings ICASSP-92 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 3, (San Francisco, CA, USA), pp. 485–488, Mar. 1992.

[63] D. M. Monro and S. J. Woolley, “Fractal image compression without searching,” in *Proceedings ICASSP-94 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 5, (Adelaide, Australia), pp. 557–560, Apr. 1994.

[64] S. J. Woolley and D. M. Monro, “Optimum parameters for hybrid fractal image coding,” in *Proceedings ICASSP-95 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 4, (Detroit, MI, USA), pp. 2571–2574, May 1995.

[65] P. D. Wakefield, D. M. Bethel, and D. M. Monro, “Hybrid image compression with implicit fractal terms,” in *Proceedings ICASSP-97 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 4, (Munich, Germany), pp. 2933–2936, Apr. 1997.

[66] G. Melnikov and A. K. Katsaggelos, “Non uniform segmentation optimal hybrid fractal/DCT image compression algorithm,” in *Proceedings ICASSP-98 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 5, (Seattle, WA, USA), pp. 2573–2576, May 1998.

[67] M. Gharavi-Alkhansari and T. S. Huang, “Generalized image coding using fractal-based methods,” in *Proceedings PCS’94 (International Picture Coding Symposium)*, (Sacramento, CA, USA), pp. 440–443, Sept. 1994.

[68] M. Gharavi-Alkhansari and T. S. Huang, “Fractal-based techniques for a generalized image coding method,” in *Proceedings ICIP-94 (IEEE International Conference on Image Processing)*, vol. III, (Austin, TX, USA), pp. 122–126, Nov. 1994.

[69] M. Gharavi-Alkhansari and T. S. Huang, “Fractal image coding using rate-distortion optimized matching pursuit,” in *Visual Communications and Image Processing ’96* (R. Ansari and M. J. Smith, eds.), vol. 2727 of *SPIE Proceedings*, (Orlando, FL, USA), pp. 1386–1393, Mar. 1996.

[70] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Communications and Information Theory, Norwell, MA, USA: Kluwer Academic Publishers, 1992.

[71] B. Hürtgen and C. Stiller, “Fast hierarchical codebook search for fractal coding of still images,” in *Video Communications and PACS for Medical Applications* (R. A. Mattheus, A. J. Duerinckx, and P. J. van Otterloo, eds.), vol. 1977 of *SPIE Proceedings*, (Berlin, Germany), pp. 397–408, Apr. 1993.

[72] C.-S. Kim, R.-C. Kim, and S.-U. Lee, “Fractal vector quantizer for image coding,” *IEEE Transactions on Image Processing*, vol. 7, pp. 1598–1602, November 1998.

[73] K. Kim and R.-H. Park, “Image coding based on fractal approximation and vector quantization,” in *Proceedings ICIP-94 (IEEE International Conference on Image Processing)*, vol. III, (Austin, TX, USA), pp. 132–136, Nov. 1994.

[74] K. Kim and R.-H. Park, “Still image coding based on vector quantization and fractal approximation,” *IEEE Transactions on Image Processing*, vol. 5, pp. 587–597, Apr. 1996.

[75] R. Hamzaoui, M. Müller, and D. Saupe, “VQ-enhanced fractal image compression,” in *Proceedings ICIP-96 (IEEE International Conference on Image Processing)*, vol. I, (Lausanne, Switzerland), pp. 153–156, Sept. 1996.

[76] B. Wohlberg, *Fractal Image Compression and the Self-Affinity Assumption: A Stochastic Signal Modelling Perspective*. PhD thesis, University of Cape Town, Cape Town, South Africa, Aug. 1996.

[77] G. Davis, “Self-quantized wavelet subtrees: A wavelet-based theory of fractal image compression,” in *Wavelet Applications II* (H. H. Szu, ed.), vol. 2491 of *SPIE Proceedings*, (Orlanda, FL, USA), pp. 141–152, Apr. 1995.

[78] B. Hürtgen and T. Hain, “On the convergence of fractal transforms,” in *Proceedings ICASSP-94 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 5, (Adelaide, Australia), pp. 561–564, Apr. 1994.

[79] B. Hürtgen, “Statistical evaluation of fractal coding schemes,” in *Proceedings ICIP-95 (IEEE International Conference on Image Processing)*, vol. III, (Washington, D.C., USA), pp. 280–283, Oct. 1995.

[80] J. Kominek, “Convergence of fractal encoded images,” in *Proceedings DCC’95 (IEEE Data Compression Conference)* (J. A. Storer and M. Cohn, eds.), (Snowbird, UT, USA), pp. 242–251, Mar. 1995.

[81] M. Gharavi-Alkhansari and T. S. Huang, “A system/graph theoretical analysis of attractor coders,” in *Proceedings ICASSP-97 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 4, (Munich, Germany), pp. 2705–2708, Apr. 1997.

[82] B. Hürtgen, “Contractivity of fractal transforms for image coding,” *Electronics Letters*, vol. 29, pp. 1749–1750, Sept. 1993.

[83] B. Hürtgen and S. F. Simon, “On the problem of convergence in fractal coding schemes,” in *Proceedings ICIP-94 (IEEE International Conference on Image Processing)*, vol. III, (Austin, TX, USA), pp. 103–106, Nov. 1994.

[84] M. Ruhl and H. Hartenstein, “Optimal fractal coding is NP-hard,” in *Proceedings DCC’97 (IEEE Data Compression Conference)*, (Snowbird, UT, USA), pp. 261–270, Mar. 1997.

[85] J. Domaszewicz and V. A. Vaishampayan, “Iterative collage coding for fractal compression,” in *Proceedings ICIP-94 (IEEE International Conference on Image Processing)*, vol. III, (Austin, TX, USA), pp. 127–131, Nov. 1994.

[86] B. Hürtgen, “Performance bounds for fractal coding,” in *Proceedings ICASSP-95 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 4, (Detroit, MI, USA), pp. 2563–2566, May 1995.

[87] D. Saupe and R. Hamzaoui, “Complexity reduction methods for fractal image compression,” in *Proceedings of the IMA Conference on Image Processing: Mathematical Methods and Applications* (J. M. Blackledge, ed.), (Oxford, England), pp. 211–229, Sept. 1994.

[88] D. Saupe, “Accelerating fractal image compression by multi-dimensional nearest neighbor search,” in *Proceedings DCC’95 (IEEE Data Compression Conference)* (J. A. Storer and M. Cohn, eds.), (Snowbird, UT, USA), pp. 222–231, Mar. 1995.

[89] D. C. Popescu and H. Yan, “MR image compression using iterated function systems,” *Magnetic Resonance Imaging*, vol. 11, no. 5, pp. 727–732, 1993.

[90] D. Götting, A. Ibenthal, and R.-R. Grigat, “Fractal image coding and magnification using invariant features,” *Fractals*, vol. 5 (Supplementary Issue), pp. 65–74, Apr. 1997.

[91] B. Wohlberg and G. de Jager, “Fast image domain fractal compression by DCT domain block matching,” *Electronics Letters*, vol. 31, pp. 869–870, May 1995.

[92] D. Saupe, “Lean domain pools for fractal image compression,” in *Still-Image Compression II* (R. L. Stevenson, A. I. Drukarev, and T. R. Gardos, eds.), vol. 2669 of *SPIE Proceedings*, (San Jose, CA, USA), pp. 150–157, Jan. 1996.

[93] J. Signes, "Geometrical interpretation of IFS based image coding," *Fractals*, vol. 5 (Supplementary Issue), pp. 133–143, July 1997.

[94] G. Caso, P. Obrador, and C.-C. J. Kuo, "Fast methods for fractal image encoding," in *Visual Communications and Image Processing '95* (L. T. Wu, ed.), vol. 2501 of *SPIE Proceedings*, (Taipei, Taiwan), pp. 583–594, May 1996.

[95] M. Kawamata, M. Nagahisa, and T. Higuchi, "Multi-resolution tree search for iterated transformation theory-based coding," in *Proceedings ICIP-94 (IEEE International Conference on Image Processing)*, vol. III, (Austin, TX, USA), pp. 137–141, Nov. 1994.

[96] H. Lin and A. Venetsanopoulos, "A pyramid algorithm for fast fractal image compression," in *Proceedings ICIP-95 (IEEE International Conference on Image Processing)*, vol. III, (Washington, D.C., USA), pp. 596–599, Oct. 1995.

[97] F. M. Dekking, "Fractal image coding: some mathematical remarks on its limits and its prospects," Tech. Rep. 95-95, Faculty of Technical Mathematics and Informatics, Delft University of Technology, Delft, The Netherlands, 1995.

[98] C. J. Wein and I. F. Blake, "On the performance of fractal compression with clustering," *IEEE Transactions on Image Processing*, vol. 5, pp. 522–526, Mar. 1996.

[99] R. Hamzaoui, "Codebook clustering by self-organizing maps for fractal image compression," *Fractals*, vol. 5 (Supplementary Issue), pp. 27–38, Apr. 1997.

[100] D. Saupe and H. Hartenstein, "Lossless acceleration of fractal image compression by fast convolution," in *Proceedings ICIP-96 (IEEE International Conference on Image Processing)*, vol. I, (Lausanne, Switzerland), pp. 185–188, Sept. 1996.

[101] H. Samet, *The Design and Analysis of Spatial Data Structures*. Addison-Wesley Series in Computer Science, Reading, MA, USA: Addison-Wesley, 1989.

[102] S. Arya and D. M. Mount, "Algorithms for fast vector quantization," in *Proceedings DCC'93 (IEEE Data Compression Conference)* (J. Storer and M. Cohn, eds.), (Snowbird, UT, USA), pp. 381–390, Mar. 1993.

[103] D. Saupe, "Breaking the time complexity of fractal image compression," Tech. Rep. 53, Institut für Informatik, University of Freiburg, Freiburg, Germany, 1994.

[104] J. Kominek, "Algorithm for fast fractal image compression," in *Digital Video Compression: Algorithms and Technologies 1995* (A. A. Rodriguez, R. J. Safranek, and E. J. Delp, eds.), vol. 2419 of *SPIE Proceedings*, (San Jose, CA, USA), pp. 296–305, Feb. 1995.

[105] B. Bani-Eqbal, "Speeding up fractal image compression," in *Still-Image Compression* (M. Rabbani, E. J. Delp, and S. A. Rajala, eds.), vol. 2418 of *SPIE Proceedings*, (San Jose, CA, USA), pp. 67–74, Feb. 1995.

[106] B. Bani-Eqbal, "Enhancing the speed of fractal image compression," *Optical Engineering*, vol. 34, pp. 1705–1710, June 1995.

[107] H. T. Chang and C. J. Kuo, "Finite-state fractal block coding of images," in *Proceedings ICIP-96 (IEEE International Conference on Image Processing)*, vol. I, (Lausanne, Switzerland), pp. 133–136, Sept. 1996.

[108] G. E. Øien, "Parameter quantization in fractal image coding," in *Proceedings ICIP-94 (IEEE International Conference on Image Processing)*, vol. III, (Austin, TX, USA), pp. 142–146, Nov. 1994.

[109] K. U. Barthel, T. Voyé, and P. Noll, "Improved fractal image coding," in *Proceedings PCS'93 (International Picture Coding Symposium)*, (Lausanne, Switzerland), p. 1.5, Mar. 1993.

[110] H. Hartenstein, D. Saupe, and K.-U. Barthel, "VQ-encoding of luminance parameters in fractal coding schemes," in *Proceedings ICASSP-97 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, vol. 4, (Munich, Germany), pp. 2701–2704, Apr. 1997.

[111] G. Strang, *Linear Algebra and its Applications*. Orlando, FL, USA: Harcourt Brace Jovanovich College Publishers, 3rd ed., 1988.

[112] H.-S. Kang and S.-D. Kim, "Fractal decoding algorithm for fast convergence," *Optical Engineering*, vol. 35, pp. 3191–3198, Nov. 1996.

[113] R. Hamzaoui, "Decoding algorithm for fractal image compression," *Electronics Letters*, vol. 32, no. 14, pp. 1273–1274, 1996.

[114] R. Hamzaoui, "Ordered decoding algorithm for fractal image compression," in *Proceedings PCS'97 (International Picture Coding Symposium)*, (Berlin, Germany), pp. 91–95, Sept. 1997.

[115] Z. Barahav, D. Malah, and E. Karnin, "Hierarchical interpretation of fractal image coding and its application to fast decoding," in *Proceedings of the IEEE International Conference on Digital Signal Processing*, (Nicosia, Cyprus), pp. 190–195, July 1993.

[116] D. M. Monro and F. Dudbridge, "Rendering algorithms for deterministic fractals," *IEEE Computer Graphics and Applications*, vol. 15, pp. 32–41, Jan. 1995.

[117] R. Dettmer, "Form and function: Fractal-based image compression," *IEE Review*, vol. 38, no. 9, pp. 323–327, 1992.

[118] E. Polidori and J.-L. Dugelay, "Zooming using iterated function systems," *Fractals*, vol. 5 (Supplementary Issue), pp. 111–123, Apr. 1997.

[119] M. Gharavi-Alkhansari, R. DeNardo, Y. Tenda, and T. S. Huang, "Resolution enhancement of images using fractal coding," in *Visual Communications and Image Processing '97* (J. Biemond and E. J. Delp, eds.), vol. 3024 of *SPIE Proceedings*, (San Jose, CA, USA), pp. 1089–1100, Feb. 1997.

[120] H. Krupnik, D. Malah, and E. Karnin, "Fractal representation of images via the discrete wavelet transform," in *Proceedings of the 18th IEEE Convention of Electrical and Electronics Engineers in Israel*, (Tel-Aviv, Israel), pp. 2.2.2/1–5, Mar. 1995.

[121] B. Simon, "Explicit link between local fractal transform and multiresolution transform," in *Proceedings ICIP-95 (IEEE International Conference on Image Processing)*, vol. I, (Washington, D.C., USA), pp. 278–281, Oct. 1995.

[122] A. van de Walle, "Merging fractal image compression and wavelet transform methods," *Fractals*, vol. 5 (Supplementary Issue), pp. 3–15, Apr. 1997.

[123] I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1992.

[124] G. Davis, "Adaptive self-quantization of wavelet subtrees: A wavelet-based theory of fractal image compression," in *Wavelet Applications in Signal and Image Processing III* (A. F. Laine, M. A. Unser, and M. V. Wickerhauser, eds.), vol. 2569 of *SPIE Proceedings*, (San Diego, CA, USA), pp. 294–307, July 1995.

[125] G. M. Davis, "A wavelet-based analysis of fractal image compression," *IEEE Transactions on Image Processing*, vol. 7, pp. 141–154, Feb. 1998.

[126] G. Caso and C.-C. J. Kuo, "New results for fractal/wavelet image compression," in *Visual Communications and Image Processing '96* (R. Ansari and M. J. Smith, eds.), vol. 2727 of *SPIE Proceedings*, pp. 536–547, 1996.

[127] A. van de Walle, "Relating fractal image compression to transform methods," Master's thesis, University of Waterloo, Waterloo, Canada, 1995.

[128] J. Li and C.-C. J. Kuo, "Fractal wavelet coding using a rate-distortion constraint," in *Proceedings ICIP-96 (IEEE International Conference on Image Processing)*, vol. II, (Lausanne, Switzerland), pp. 81–84, Sept. 1996.

[129] R. Rinaldo and G. Calvagno, "An image coding scheme using block prediction of the pyramid subband decomposition," in *Proceedings ICIP-94 (IEEE International Conference on Image Processing)*, vol. II, (Austin, TX, USA), pp. 878–882, Nov. 1994.

[130] R. Rinaldo and G. Calvagno, "Image coding by block prediction of multiresolution subimages," *IEEE Transactions on Image Processing*, vol. 4, pp. 909–920, July 1995.

[131] K. Belloulata, A. M. Baskurt, H. Benoit-Cattin, and R. Prost, "Fractal coding of subbands with an oriented partition," *Signal Processing: Image Communication*, vol. 12, no. 3, pp. 243–252, 1998.

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

[133] R. J. Clarke and L. M. Linnett, "Fractals and image representation," *Electronics & Communication Engineering Journal*, vol. 5, pp. 233–239, Aug. 1993.

[134] D. W. Lin, "Fractal image coding as generalized predictive coding," in *Proceedings ICIP-94 (IEEE International Conference on Image Processing)*, vol. III, (Austin, TX, USA), pp. 117–121, Nov. 1994.

[135] J. Domaszewicz, S. Kuklinski, and V. A. Vaishampayan, "Fractal coding versus classified transform coding," in *Proceedings ICIP-96 (IEEE International Conference on Image Processing)*, vol. I, (Lausanne, Switzerland), pp. 149–152, Sept. 1996.

[136] B. Wohlberg and G. de Jager, "A class of multiresolution stochastic models generating self-affine images," *IEEE Transactions on Signal Processing*, vol. 47, pp. 1739–1742, June 1999.

[137] S. Lepsøy, P. Carlini, and G. E. Øien, "On fractal compression and vector quantization," in *Fractal Image Encoding and Analysis* (Y. Fisher, ed.), vol. 159 of *NATO ASI Series. Series F: Computer and Systems Sciences*, New York, NY, USA: Springer-Verlag, 1997.

[138] B. Wohlberg and G. de Jager, "Fractal coding performance for first-order Gauss-Markov models," *Electronics Letters*, vol. 32, pp. 441–442, Feb. 1996.

PLACE
PHOTO
HERE

Brendt Wohlberg received the BSc(Hons) degree in applied mathematics, and the MSc(Applied Science) and PhD degrees in electrical engineering from the University of Cape Town, South Africa, in 1990, 1993 and 1996 respectively. He was a postdoctoral research fellow in the Electrical Engineering Department at the University of Cape Town from 1996 to 1998, and is currently a postdoctoral research associate at Los Alamos National Laboratory, Los Alamos, NM. His research interests include image compression, sparse representations, and pattern recognition.

PLACE
PHOTO
HERE

Gerhard de Jager (M'86) received the BSc degree in physics, mathematics and applied mathematics in 1957, BSc (Hons) and MSc degrees in physics from Rhodes University, Grahamstown, South Africa in 1958 and 1963 respectively, PhD in radioastronomy from Manchester University, Manchester, UK in 1967 and MBL from the University of South Africa in 1979. He was lecturer in radioastronomy at Manchester University from 1967 to 1969, lecturer in physics at Rhodes University from 1970 to 1980, professor in electronics at Rhodes University from 1981 to 1987 and professor of electronic engineering at the University of Cape Town from 1988 to the present.

Dr de Jager was an Alexander von Humboldt fellow at the Max Planck Institut for Radioastronomy in Bonn in 1975, a visiting scientist at the Australian National University in Canberra in 1981, a visiting scientist at the Research Laboratory of Electronics at the Massachusetts Institute of Technology in 1986 and a visiting professor at the Katolieke Universiteit Leuven in 1999. He was a recipient of the a Junior Leverhulme fellowship and the Ernest Oppenheimer travelling fellowships. Dr de Jager's current interests are in image and video compression and digital television, medical imaging and computer vision and he has consulted for various firms in these areas.