7

FRACTAL-BASED IMAGE AND VIDEO CODING

Mohammad Gharavi-Alkhansari and Thomas S. Huang

Beckman Institute for Advanced Science and Technology,
University of Illinois at Urbana-Champaign,
Urbana, Illinois 61801,
USA

ABSTRACT

This chapter reviews the theoretical foundations and implementation issues of fractal-based image coding methods. The concepts of fractals, iterated function systems, and local iterated function systems are discussed and different implementations of compression of both still images and image sequences are reviewed.

1 INTRODUCTION

Fractal-based image coding, which is sometimes called *fractal image coding* or *attractor image coding*, is a new method of image compression. In this method, similarities between different scales of the same image are used for compression. The method is rooted in the work of Mandelbrot, who introduced the concept of fractals and the fractal dimension.

This chapter is organized as follows. Section 1 gives a brief review of the concept of fractals and its applications especially in image compression. In Section 2 the principles of iterated function systems (IFS) will be studied. IFS makes the basis of most fractal-based image compression methods. In Section 3, we will see how the this theory has been used for compression of images and video sequences. Finally in Section 4, conclusions will be drawn.

1.1 Fractals and Self-Similarity at Different Scales

In late 70s and early 80s, Mandelbrot showed that many natural and man-made phenomena have the very fundamental characteristic of invariance under change of scale [86, 88, 87]. Mandelbrot coined the name *fractal* for the geometry of these phenomena.

The mathematical definition of fractals suggested by Mandelbrot is that they are sets for which the Hausdorff-Besicovitch dimension D is strictly larger than their topological dimension D_T [87]. However, computing Hausdorff-Besicovitch dimension is often difficult, and in many cases the *fractal dimension* [12] is used instead. Fractal dimension is defined as

$$D = \lim_{d \to 0} \frac{\ln N(d)}{\ln(\frac{1}{d})} \tag{7.1}$$

where $N(d)$ is the minimum number of balls of diameter d which are needed to cover the set[1].

This definition implies that if d is small enough, we can write the approximate power law

$$N(d) = K(1/d)^D \tag{7.2}$$

where K is a constant. This means that as d decreases, $N(d)$ grows with the Dth power of $1/d$, no matter how small d is. D can be considered as a measure of the roughness of a set, where rougher sets have larger Ds [105]. A classical example of a natural fractal set is the coastline of an island, and an example of an artificial fractal set is the Koch curve [87]. In practice, a natural set is considered fractal if its D is stable over a wide range of scales.

Figure 1 shows different steps in the construction of the Koch curve. In this construction we begin with a line segment of length 1 (Figure 1a). Then divide the line into three equal parts and replace the middle part with two line segments of length 1/3, obtaining the graph of Figure 1b. If we apply the operation that generated Figure 1b from Figure 1a on every line segment in Figure 1b, we get Figure 1c. Repeating this process once more results in Figure 1d, and continuing to apply this process infinitely many times, results in the set shown (approximately) in Figure 1e, which is known as the Koch curve.

[1] For more detailed information on the definition of fractals and different dimensions see [42] or [44].

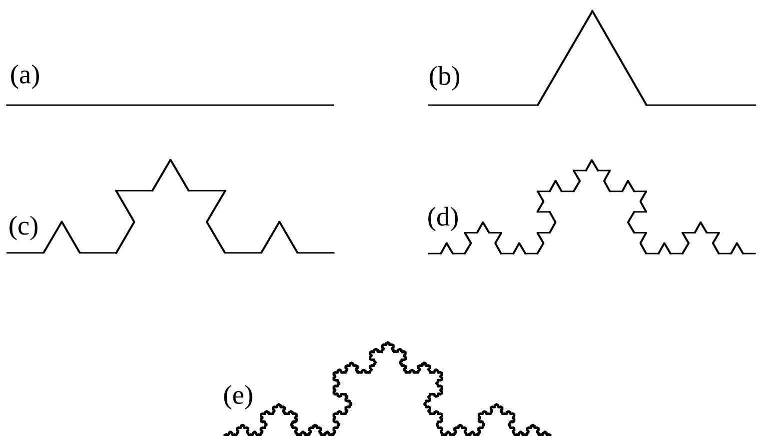

Figure 1 Different stages of construction of the Koch Curve.

It is easy to show that the set shown in Figure 1e has a fractal dimension of

$$D = \frac{\ln 4^n}{\ln 3^n} = \frac{\ln 4}{\ln 3} \approx 1.26 > 1 = D_T.$$

The power law in Equation 7.2 states that dividing d by any factor a, always increases $N(d)$ by a factor a^D, for any small value of d. In the case of integer Ds, this result seems trivial, but when D is not an integer this means that, for example, for a fractal curve, magnifying any part of the curve does not result in a curve that is smoother than the original curve.

This power law means that a fractal set has the same roughness, independent of scale. Therefore one can say that a fractal set is self-similar at different scales in the above general sense. Of course many simple non-fractal geometrical sets, like lines and planes, are also self similar in this sense, but are not fractals as they are not 'rough' sets and always become smooth when magnified enough. Fractals always reveal more and more details under magnification and these details do not diminish by magnification.

Some fractals possess self-similarities in a more restricted sense. The self-similarity of this class of fractals can be either deterministic or statistical. The deterministic self-similarity is when the shape of the set is similar to itself in an deterministic way as the scale changes. A particular kind of fractals of this class are fractals with exact self-similarity which do not change at all under change of scale, e.g. the Koch curve.

The statistical self-similarity is the property of fractals which retain all of their statistical parameters at different scales while a deterministic relationship does not necessarily exist between different scales of the set.

1.2 Applications of Fractals

Fractal geometry in nature is more a rule than an exception. Since the introduction of the concept of fractals by Mandelbrot, the concept has been used in many different branches of science including mathematics, physics, chemistry, geophysics, botany, biology, computer graphics, computer vision, and image processing.

Fractals with statistical self-similarity have been of great interest in the area of computer graphics, where the concept was used to generate complex and strikingly natural-looking graphics of natural scenes using simple rules [87, 99].

In the area of computer vision, the fractal dimension has been used for image modeling, segmentation and shape extraction for natural scenes by Pentland [105, 106] and others [4, 5, 6, 74, 100, 102, 101, 119, 128]. The fractal dimension of different natural objects can be different from each other or from those of man-made objects. On the other hand, under certain conditions, 2-D images taken from some 3-D fractal geometries are also fractals. Pentland [105] used the fractal dimension as a parameter for segmenting images. This method can be used to discriminate between different natural objects in a scene or between man-made objects and natural objects [102, 101].

1.3 Fractals in Image Compression

Fractal geometry has been used for image compression in a few basically different ways.

Fractal curves, specially the Peano curve, were used for scanning images instead of the standard raster scanning [120, 137, 54, 126].

A 'yardstick' method was used for image compression [133, 138, 122] and for shape classification [36].

Fractal dimension has been used as a tool in different aspects of image compression algorithms.

- Fractal dimension was used in a fractal image coder for adjusting error thresholding [60]. Also in [72, 73], fractal dimension was used for image segmentation. After segmentation, fractal dimension was also used as a measure of complexity of the segment to determine how the segments should be coded.
- In the context of image coding, fractal dimension was also used for selecting the optimal scale parameter in an edge detector [38].

Wavelet decomposition has also been used to exploit self-similarities of images at different scales. In this approach, a wavelet decomposition is applied to an image, and the similarity of same-size blocks in different subbands is used to reduce the size of the code needed for representing them [104, 108, 109, 41]. Pentland [104] reports typical compression ratios of 38:1 with 33 dB PSNR for 256×256 pixel images and Rinaldo and Calvagno [108] report compression ratios of about 54:1 (PSNR of 31.4 dB) to 20:1 (PSNR of 35.5 dB) for 512×512 test image Lena.

However, the method that has attracted the most attention, and that will be discussed in more detail in this chapter, is based on the work by Barnsley [17, 9, 10, 21, 26, 12, 23, 27, 24, 19, 11, 18, 28, 13, 20, 14, 15, 29, 22, 25, 16], summarized in [25][2]. His work was based on that of Hutchinson [64], who set up a theory for deterministically self-similar sets, and studied transformations that can generate this kind of sets. These transformations, which Barnsley later named *Iterated Function Systems (IFS)*, were originally used for generating fractals, but because many non-fractal sets can also be deterministically self-similar, these sets can also be generated by IFS. Iterated function systems will be discussed in detail in Section 2.

Barnsley's early work was based on the following assumptions,

- The images of many natural objects can be approximated by members of a class of deterministically self-similar sets.
- These sets can be generated by IFS transformations which have a relatively small number of parameters.

[2]The methods based on Barnsley's work have strong relationship with some of the wavelet methods mentioned earlier. For studies on this, see for example [78, 41].

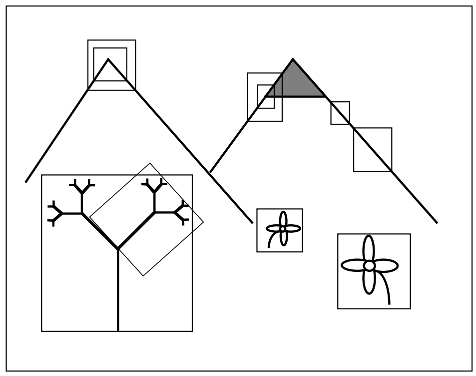

Figure 2 The essence of fractal coding methods is to try to approximate each segment of the image by applying some (contractive) transformation on some bigger segment(s) in the image.

Barnsley observed that even very simple IFSs with very short codes can generate complex sets with infinite details that resemble natural objects. IFS transformations describe the relationship between the whole image and its parts and exploit the similarities that exist between an image and its smaller parts.

Given an IFS, generating the image corresponding to it is quite straightforward and easy. However, the *inverse problem* of finding the IFS which can generate (or closely approximate) a given image has yet to be solved. In other words, the problem was that of how the similarities between the whole image and its parts could be found automatically. Another problem was: what could be done for images for which the smaller parts do not resemble the whole image? To solve the second problem, in 1988, Barnsley generalized the theory of IFS to the theory of *Local Iterated Function Systems* which exploited similarities between parts of the image which were of different sizes. Using this theory, image parts were not required to resemble the whole image; they only needed to be similar to some other bigger parts in the image, as shown in Figure 2. But there was still the first problem: how could these similarities be found automatically? In the early implementations of this theory, these similarities were found by human interaction and hence the images were encoded by interactive computer programs. This resulted in codes for images which were extremely compact in size, but their decoded images had very low quality [12]. This was until the work by Arnaud E. Jacquin (a student of Barnsley) who automated this method for the first time [67, 68, 69, 70]. The code generated by Jacquin's program for an image was not as compact as before, but the compression ratio and the quality of the decoded images looked promising. The work by Jacquin provided a platform for others to continue this line of research. Since then several extensions and generalizations of this method have been found, and many of its properties are better understood, which has resulted in more efficient algorithms. Some of these methods will be discussed in Section 3.

Barnsley and Sloan founded the company 'Iterated Systems, Inc.' in 1987 for the development of products based on the fractal theory, and patented some of the basic algorithms in fractal coding [15, 29]. This company has made different hardware and software products for image and video compression/decompression especially on personal computers. Although many articles have been published on the basics of Barnsley's theory, many of the details of the algorithms used in these products have not been revealed.

1.4 Fractal Techniques in Second Generation Image Coding

Second generation image coding methods take special advantage of the properties of the human visual system and many of them are segmentation-based. In this section we will briefly see how fractals are related into these two features of second generation coding methods.

- *Fractals and the Human Visual System*
  Many researchers have studied the relation between fractals and the human visual system [105, 106, 77].
  - To human eye, many fractal curves and surfaces look very similar to natural curves and surfaces and for this reason they have been extensively used in computer graphics. In model-based coding, this similarity has the potential of being used for coding of natural images by modeling the underlying processes that generates parts of these images.
  - Experiments have shown that the fractal dimension of a curve or set is closely related to human's perception of its roughness [105]. Although fractal dimension alone is not enough for generating a visually good approximation of a set [4, 7], it may be used as one of the parameters for its representation.
  - It is known that human visual system's sensitivity to details in any part of an image is dependent on the amount of activity in the back-

ground of that part of the image. Fractal dimension of image regions has been used as an objective measure of this activity [60].

- *Segmentation Using Fractal Dimension*
  Many researchers have used fractal dimension for image segmentation [105, 72, 113, 6, 121, 79, 82]. Fractal dimension is usually computed locally [123, 91] and is used as the texture feature for segmentation.

- *Edge Detection Using Fractal Dimension*
  In the context of image coding, fractal dimension was also used for selecting the optimal scale parameter in a multiscale edge detector [38]. In this method, edge points were detected by wavelet transform and the dilation parameter is controlled by the fractal dimension.

- *Fractal Coding of Contours*
  One of the first applications of the theory of iterated function systems proposed by Barnsley and Jacquin was in contour coding [23, 67]. A similar method was later used for this purpose by Jacobs et al. [65].

- *Jacquin's Method as a Second Generation Method*
  Fractal coding methods based on Jacquin's method basically use redundancies in an image at different scales, i.e., they use the fact that different parts of the image at different scales are similar. Due to computational complexity limitations, most fractal coding methods find similarities between image 'blocks', after applying limited transformations, even though the most natural choice is finding similarities between 'objects' or 'segments' with more free deformations. The use of blocks instead of segments is more a matter of speed than anything else, especially because fractal coders are usually computationally intensive. In the basic theory, the shape of the domain segments is not restricted in any sense. Simple block splitting methods have been used by many researchers (including Jacquin) for adjusting the size of the blocks to the feature sizes of the image.

  Thomas and Deravi [124, 125] devised a method for merging of blocks using a region growing procedure based on fractal coding. This method results in range 'regions' with rather free shapes that are adapted to the content of the image. Using this method, a region in the image is approximated with another larger region (but with a similar shape) in the same image.

  Franich et al. [52] proposed a method for merging quadtree block splitting method for shape description with the quadtree block splitting method used for fractal coding and used it in an object-based video coding system.

  From this view point, the fractal coding techniques originated by the work of Jacquin, can be well adapted to both first and second generation image coding techniques, although during the recent years most of the advancements of these techniques have been in the direction of combining them with waveform-based coding methods.

2 BASIC THEORY

The essence of most fractal-based image coding methods is to approximate each segment of the image by applying a (contractive) transformation on some bigger segment(s) in the image. One can then reconstruct the image (with some error) by using only the parameters of the transformations [12, 25]. In these methods, most of the information in the image is basically encoded by coding relations among different segments (of different sizes) of the image. The mathematical framework of this theory is presented in the following sections.

2.1 Iterated Function Systems

We begin with a complete metric space (X, d), where $d(.,.)$ denotes the metric[3]. Now, consider a transformation $w\colon X \mapsto X$, for which there is a constant s such that for all $x, y \in X$,

$$d(w(x), w(y)) \leq s\ d(x, y).$$

If $0 \leq s < 1$, then w is said to be *contractive* (or a *contraction*) with *contractivity factor* s. If w is contractive, then according to the *Contraction Mapping Theorem*,

1. w possess a unique fixed point $x^* \in X$, i.e., $w(x^*) = x^*$.
2. For any $x \in X$, $\lim_{n\to\infty} w^{(n)}(x) = x^*$.

The transformation w defined on X also induces a transformation on subsets of X. This can be done by defining

$$w(B) = \{w(x), \forall x \in B\} \quad \forall B \subseteq X.$$

Let $(H(X), h)$ denote the metric space whose points are non-empty compact subsets of X, and h is the *Hausdorff Distance* [25]. An *Iterated Function System*

[3]For more details on the basic theory brought in this section see [12], [25], or [22].

(IFS) consists of a complete metric space (X, d) and a number of contractive mappings w_i defined on X, i.e. $\{X; w_i, i = 1, \ldots, N\}$. The *fractal transformation* associated with an IFS is the transformation $W : H(X) \mapsto H(X)$ defined by

$$W(B) = \bigcup_{i=1}^{N} w_i(B) \tag{7.3}$$

for all $B \in H(X)$. If the mappings w_i are contractive with contractivity factors $s_i,\ i = 1, 2, \cdots, N$, then W is also contractive with contractivity factor $s = \max_i s_i$, and W has a unique fixed point $A \in H(X)$ for which

$$A = W(A) = \bigcup_{i=1}^{N} w_i(A),$$

and for all $B \in H(X)$ we have $\lim_{n\to\infty} W^{(n)}(B) = A$. A is called the *attractor* of IFS. w_i's are usually chosen to be affine transformations. For the two-dimensional case, this becomes

$$w_i\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} a_i & b_i \\ c_i & d_i \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \end{bmatrix} \tag{7.4}$$

defined on points in R^2. For the three-dimensional case (gray-scale images) this becomes

$$w_i\left(\begin{bmatrix} x \\ y \\ I(x,y) \end{bmatrix}\right) = \begin{bmatrix} a_{1,1,i} & a_{1,2,i} & a_{1,3,i} \\ a_{2,1,i} & a_{2,2,i} & a_{2,3,i} \\ a_{3,1,i} & a_{3,2,i} & a_{3,3,i} \end{bmatrix}\begin{bmatrix} x \\ y \\ I(x,y) \end{bmatrix} + \begin{bmatrix} b_{1,i} \\ b_{2,i} \\ b_{3,i} \end{bmatrix}, \tag{7.5}$$

where $I(x, y)$ denotes the gray-scale value at location (x, y). For an image, the *fractal code* is made up of the parameters of the fractal transformation W, which consists of the number N and the parameters of w_i's. The mappings w_i defined by 7.4 and 7.5 are contractions under suitable constraints on the parameters and therefore the resulting Ws are also contractions.

As an example of an IFS and its attractor in R^2, let us consider an IFS of the form

$$\{R^2; w_1, w_2, w_3\},$$

where

$$w_1\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} \tag{7.6}$$

$$w_2\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.5 \\ 0 \end{bmatrix} \tag{7.7}$$

$$w_3\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}. \tag{7.8}$$

The attractor of this IFS can be found by iteratively applying the induced W on any non-empty compact subset of X. Figure 3 shows how a sequence of sets generated by iterative application of W on an arbitrary initial set B converges to the attractor of W and how the attractor is dependent only on W and not the initial set.

In general, A is completely described by W and is independent of B. Therefore, W gives a complete representation of A and the set of parameters that represent W can be considered as a code for A. In the above example, it can be seen that A has a visually complex shape, but W has a very simple mathematical form which can be specified by three affine transformations. Considering the plot of A as a black and white image, the parameters of W make the code for this image.

2.2 The Collage Theorem

Although the theory of generating the attractor of an IFS is well developed, the inverse problem of finding the IFS code for approximating an arbitrary given set, like many other inverse problems in mathematics, has proven to be a rather difficult problem.

Several studies have been made on finding the exact mathematical solution to this inverse problem using tools such as Fourier transform [43], wavelet transform [53, 8], moment method [3, 1, 2, 32, 34, 49, 61, 89, 127, 131, 130, 132, 50, 51], chaotic optimization [90, 89], genetic algorithms [118], combination of the wavelet transform and the moments method [110, 111, 112], fuzzy sets [35] and other methods [95, 37]. However, this problem, in the general case, is not yet solved.

As discussed before, given a W, the decoding process is based on the Contraction Mapping Theorem. The transformation W is applied iteratively on an (arbitrary) initial image until the transformed image does not change significantly. As W is contractive, the convergence of this sequence of images is guaranteed by the Contraction Mapping Theorem.

However, for a given set C, the encoding problem of finding a contractive transformation W such that its attractor A is close to C, is a rather difficult

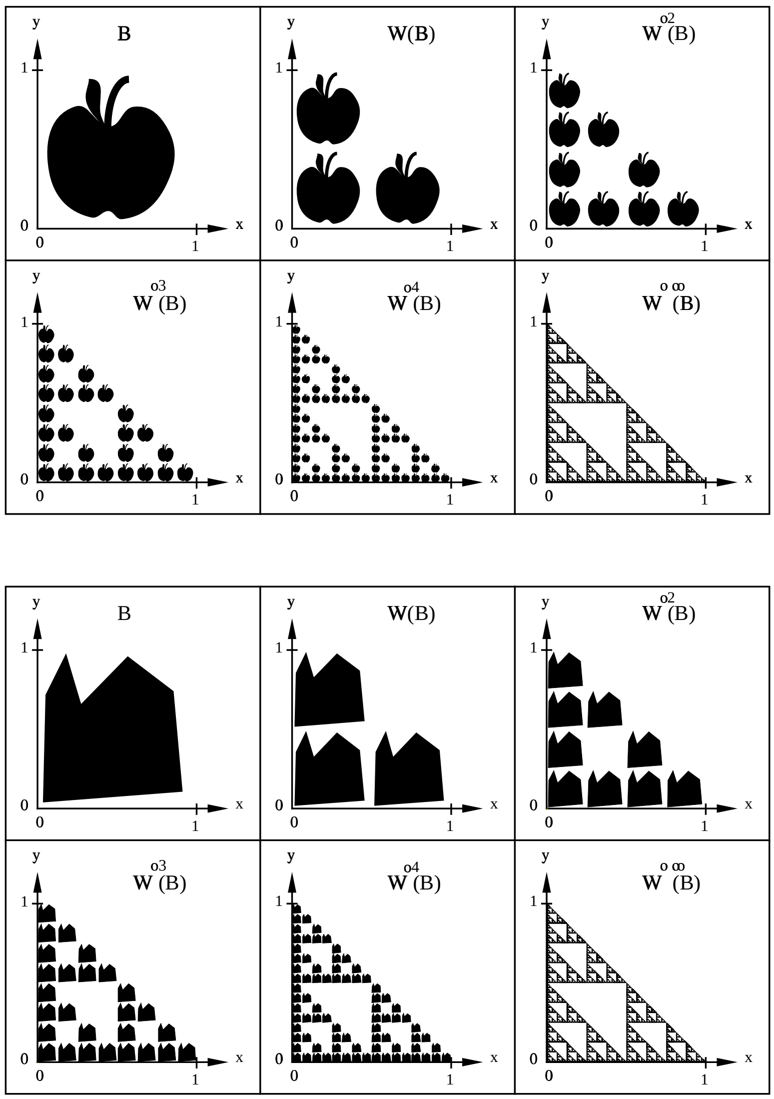

Figure 3 Sequences of sets generated by iterative application of the IFS transformation W defined by 7.3, 7.6, 7.7, and 7.8 on two different arbitrary initial sets (B), converging to the attractor of IFS.

problem. The *Collage Theorem* [21, 12] provides a guideline for solving this problem. It says that for a set C and a contraction W with attractor A,

$$h(C, A) \leq \frac{h(C, W(C))}{1 - s}. \tag{7.9}$$

This means that in order for C and A to be close, it is sufficient that C and $W(C)$ be close, i.e., W may be found in such a way that $W(C)$ be as close to C as possible. $W(C)$ is sometimes called the *collage* of C.

In terms of w_i, we have

$$\left.\begin{array}{l} W(C) \approx C \\ W(C) = \bigcup_{i=1}^{N} w_i(C) \end{array}\right\} \Rightarrow \bigcup_{i=1}^{N} w_j(C) \approx C$$

This can be done by partitioning C into parts C_i,

$$C = \bigcup_{i=1}^{N} C_i$$

such that each part C_i can be closely approximated by applying a contractive affine transformation w_i on the whole C, i.e.,

$$C_i = w_i(C).$$

If we denote $h(C, W(C))$ by ε_E and call it the *encoding error* (or *collage error*), and denote $h(C, A)$ by ε_D and call it the *decoding error*, then according to 7.9,

$$\varepsilon_D \leq \frac{1}{1 - s} \varepsilon_E$$

which gives an upper bound for ε_D in terms of ε_E.

2.3 Local Iterated Function Systems

For most natural images, it is not possible to closely approximate all parts of the image by a small number of transformations applied on the *whole* image. To solve this problem, the theory of Iterated Function Systems was extended to *Local Iterated Function Systems* [22], and its associated fractal transform. In contrast to an Iterated Function System which approximates each part of the

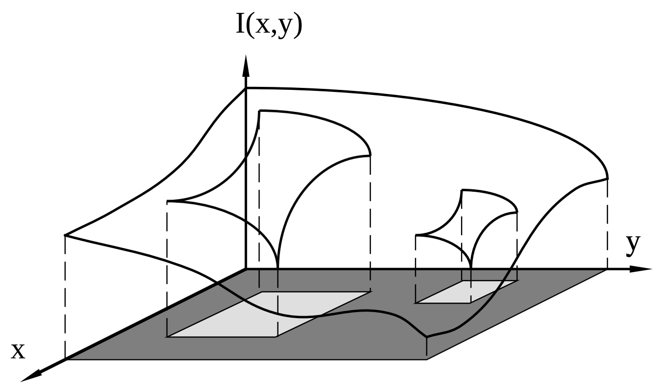

Figure 4 Approximation of a range block by a transformed domain block in a local IFS

image by a transformed version of the whole set, in the Local Iterated Function System each part of the image is approximated by applying a contractive affine transformation on another *part* of the image. In this case, the image C is partitioned into range segments C_i, where $C = \bigcup_{i=1}^{n} C_i$. Then each range segment C_i is approximated by a transformed version of a bigger domain segment D_i, i.e., $C_i \approx w_i(D_i) \Rightarrow C \approx W(C) = \bigcup_{i=1}^{N} w_i(D_i)$ as shown in Figure 4. The decoding process for Local IFS is very similar to that of IFS.

2.4 Resolution Independence

When the above theory is used for image compression, it is implemented in a discrete setting. However, the fractal code generated by encoding a digital image describes relationships (in the form of affine functions) between various segments of the image and is independent of the resolution of the original image. In other words, the fractal code is a *resolution independent* representation of the image and theoretically represents a continuous image approximating the original image. A decoder may decode this code to generate a digital image at any resolution. The resolution of the decoded image may as well be higher than the resolution of the original image. This increase of resolution is sometimes referred to as *fractal zoom.*

The higher resolution obtained is not created by a simplistic technique such as repeating the pixels of the image, but more detail is actually generated in the decoded images. In fact the additional higher resolution information are generated using information from the image at a lower resolution. When an image is reconstructed at the same resolution as the original encoded image, in the decoding process domain blocks of the image are shrunk (lowpass filtering followed by subsampling), which eliminates some of the details of the domain blocks. However, if the image is reconstructed at a higher resolution, in the shrinking of the domain block, the details of the domain block are only shrunk to generate the extra resolution in the range block. In fact, details of the domain blocks are used for missing details of the range block. The details in the domain block are also generated to some extent from details of other domain blocks, used for encoding each part of it. In other words, it is implicitly assumed that if the range block is similar to its corresponding domain block, then the details of the range block (which are *beyond* the resolution of the originally encoded image) are also similar to the details of the domain block (which are *within* the resolution of the encoded image). This assumption is a typical property of self-similarity of fractal sets at different scales, and the resolution independence is a property of the code generated by fractal-based methods.

3 IMPLEMENTATIONS

In view of the theory discussed in the previous section, some of the basic questions to be answered are:

- how to segment the image,
- what transformations to use,
- how to find the parameters of the transformations,
- where to find the matching segments.

These issues will be discussed in this section along with compression results reported for both still images and video sequences.

3.1 Still Images

In 1989 and 1990, Jacquin [67, 68, 69, 70] developed an automatic implementation of the Local IFS method by restricting B_is to squares of two fixed sizes,

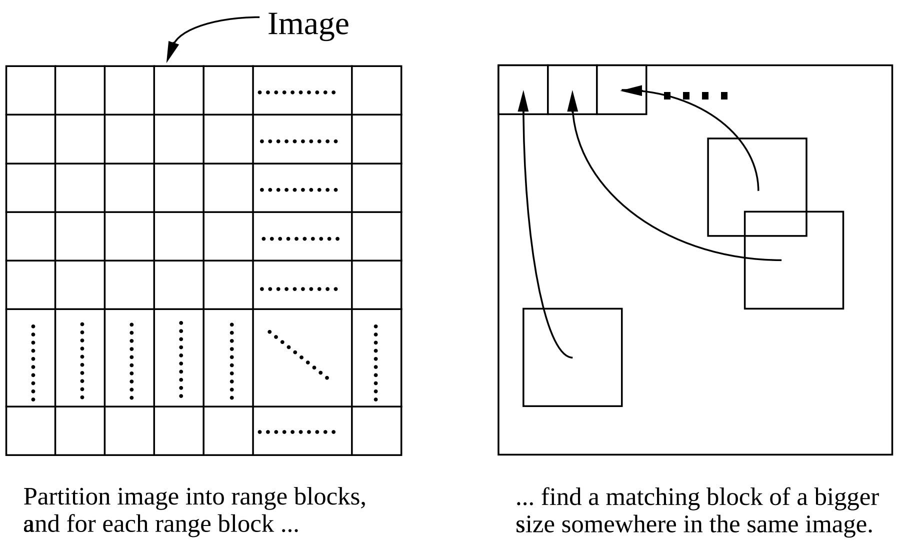

Figure 5 A demonstration of Jacquin's algorithm

and restricting the affine transformation to the following special case,

$$w_i\left(\begin{bmatrix} x \\ y \\ I(x,y) \end{bmatrix}\right) = \begin{bmatrix} a_{1,1,i} & a_{1,2,i} & 0 \\ a_{2,1,i} & a_{2,2,i} & 0 \\ 0 & 0 & p_{i,0} \end{bmatrix} \begin{bmatrix} x \\ y \\ I(x,y) \end{bmatrix} + \begin{bmatrix} b_i \\ c_i \\ p_{i,1} \end{bmatrix}.$$

where

$$\begin{bmatrix} a_{1,1,i} & a_{1,2,i} \\ a_{2,1,i} & a_{2,2,i} \end{bmatrix} = \begin{bmatrix} \pm a & 0 \\ 0 & \pm a \end{bmatrix} or \begin{bmatrix} 0 & \pm a \\ \pm a & 0 \end{bmatrix}$$

and the origin of the x and y axes is the center of the domain block, $a = 0.5$ and $p_{i,0} < 1$. For each i, the $p_{i,0}$, b_i, c_i are found by search, and $p_{i,1}$ is computed. The essence of Jacquin's method can be summarized as partitioning the image into square range blocks and searching the image for matching domain blocks of twice the size of the range block, as shown in Figure 5. In finding a matching block, we are allowed to apply simple transformations on the domain block, which include shrinking, adding a single value to the gray-scale of the pixels in the block, and scaling by a number less than one. Some shuffling of the pixel locations (*isometric* transformations) are also allowed, which include rotation by multiples of 90 degrees, and/or reflection against vertical or horizontal axes. The encoding process is also enhanced by a two-level hierarchical block splitting method and a range and domain block classification scheme for a faster search.

For the 512×512 standard Lena image, PSNRs of 30.1 dB and 31.4 dB were reported at bit rates of 0.57 and 0.6 bits per pixel (bpp) [68, 71].

In 1991, Øien et al. [96] extended this method to

$$w_i\left(\begin{bmatrix} x \\ y \\ I(x,y) \end{bmatrix}\right) = \begin{bmatrix} a_{1,1,i} & a_{1,2,i} & 0 \\ a_{2,1,i} & a_{2,2,i} & 0 \\ d_i & e_i & p_{i,0} \end{bmatrix} \begin{bmatrix} x \\ y \\ I(x,y) \end{bmatrix} + \begin{bmatrix} b_i \\ c_i \\ p_{i,1} \end{bmatrix}.$$

In this case, for each i, values of d_i, e_i, $p_{i,0}$, $p_{i,1}$ are found by least squares methods, and b_i, c_i are again found by search. Using this method, the 512×512 Lena image was encoded at a bit rate of 0.5 bpp with a 30.8 dB PSNR.

Monro and Dudbridge [92, 93], in 1992, suggested partitioning an image into small square images, and for each small image, an IFS (and not a Local IFS) was to be found. This is equivalent to a Local IFS with the domain block for each range block being a predetermined block which contains the range block.

Fisher, Jacobs and Boss, studied the effect of using blocks of different shapes, including squares, rectangles, and/or triangles combined with a multi-level hierarchical block splitting method [45, 46, 66]. They also compared the trade offs between compression ratio and signal to noise ratio (SNR) for their method [66]. In two of their experiments, the 512×512 Lena image was coded at 0.22 bpp with PSNR of 30.71 dB and at 0.45 bpp with PSNR of 33.40.

In 1993, Gharavi-Alkhansari and Huang [55, 56] extended Jacquin's method and showed that one can use a linear combination of a series of transformed domain blocks instead of only a single domain block.

Thomas and Deravi [124] used blocks of relatively free shapes in Jacquin's algorithm and showed that this could improve the performance of Jacquin's method for simple images. For the 512×512 image Lena, they obtained a PSNR of 27.7 dB at 0.30 bpp.

Lepsøy et al. [81] introduced a non-iterative decoding algorithm for fractal-based image compression.

Also Vines and Hayes [129] suggested limiting the search on b_i and c_i by looking for matching domain blocks only in the neighborhood of the corresponding range blocks. Using this method and a multi-level block-splitting scheme, the 512×512 Lena image could be compressed at a bit rate of 0.47 bpp with PSNR of 31.5 dB.

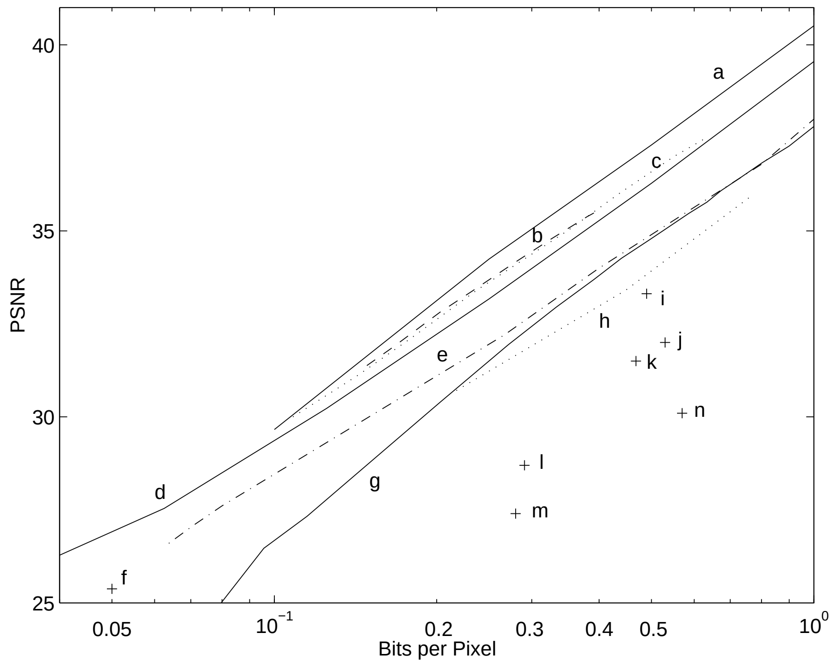

Figure 6 PSNR vs. bit rate for compression of 512 × 512 Lena image with some fractal-based and some non-fractal-based methods. See Table 1 for references.

Up to 1993 most of the attention of the papers published on fractal coding was concentrated on the fractal transform. Since then, more attention has been paid to the entropy coding stages following the fractal transform and on the problem of how the fractal transform parameters could be best modeled for entropy coding. This has resulted in more efficient algorithms.

Barthel et al., in 1994, published results on a fractal-based coding method with a performance of 35 dB PSNR at 0.35 bpp for the 512 × 512 Lena image [30]. In their method, after approximating a range block with a a domain block, any spectrum coefficient of the range block (in DCT domain) that is not well approximated by the domain block, is individually coded using transform coding. These spectral coefficients are then excluded from being approximated by the domain block. Rate-distortion optimality is also used as the criteria for selecting the best possible choice in places where there are several possible alternatives in the encoding process.

Table 1 References for Figure 6

	Year	Researchers	Reference	Method
a	1994	Xiong et al.	[136]	(Non-fractal) Wavelets
b	1994	Rinaldo and Calvagno	[108]	Fractal-Wavelet
c	1994	Barthel et al.	[30]	Fractal-DCT
d	1993	Shapiro	[117]	(Non-fractal) Wavelets
e	1995	Fisher and Menlove	[47]	Fractal
f	1995	Culik and Kari	[40]	Fractal
g	1991	JPEG	[134, 103]	(Non-fractal) JPEG
h	1992	Fisher et al.	[45, 46, 66]	Fractal
i	1994	Kim and Park	[75]	Fractal
j	1993	Lepsøy et al.	[81]	Fractal
k	1993	Vines and Hayes	[129]	Fractal
l	1994	Lu and Yew	[85]	Fractal
m	1993	Thomas and Deravi	[124, 125]	Fractal
n	1990	Jacquin	[68]	Fractal

Also in 1994, Gharavi-Alkhansari and Huang proposed a generalized image block coding method for unifying the three methods of block transform coding, vector quantization and fractal-based coding methods [57, 58]. In this method every block in the image is approximated by a linear combination of one or more blocks selected from a possibly large dictionary of (not necessarily orthogonal) library blocks. In the case of video coding, block prediction methods like DPCM and adaptive block prediction methods like block motion compensation methods are also special cases of this algorithm. They also proposed that the iterative nature of the fractal image decoders is related to the noncausality of the encoder and using a causal encoder results in a non-iterative decoder that converges in one iteration.

Lin [84] also studied fractal image coding as a generalized predictive coding method and showed how noncausal prediction in fractal coders necessitates an iterative decoding.

In 1994 and 1995, Rinaldo and Calvagno [108, 109] used similarities between blocks in different subbands of image for image coding and reported a performance of 32.78 PSNR at 0.26 bpp [108].

Figure 6 shows the reported performance of some fractal compression methods along with some non-fractal compression methods in terms of PSNR and bit rate for the 512×512 Lena image[4]. Different curves in this plot are assigned letters which are described in Table 1. Curves "g", "d", and "a" are for JPEG, wavelet-based zerotree, and improved wavelet-based zerotree methods which are non-fractal methods and are introduced here only for comparison. The JPEG results brought here are based on the "Independent JPEG Group's free JPEG software" implementation of JPEG. For an implementation based on JPEG standard with a significantly better performance the reader may see [39].

Figure 7 shows the iterative decoding of the 512×512 test image Lena. This image was compressed to 0.43 bits/pixel using a noncausal version of [57, 58, 59]. Beginning with an initial 512×512 black (all zero) image, the decoder generates a sequence of images that converge to the decoded Lena image. Figure 7 shows images resulting from the first five iterations, having PSNRs of (b) 24.22 dB, (c) 27.74 dB, (d) 30.42 dB, (e) 32.21 dB and (f) 33.24 dB. After about 10 iterations, at a PSNR of 34.50 dB the change in the image becomes negligible. The original Lena image is shown in Figure 8 and the final decoded image is shown in Figure 9. Figures 10, and 11 show Lena image compressed at 0.22 bpp (31.2 dB PSNR), and 0.15 bpp (29.2 dB PSNR) using the same method.

3.2 Image Sequences

Fractal-based techniques have also been explored for coding image sequences.

In 1991, Beamount [31] used fractal-based techniques for video compression. He tried two different approaches for this purpose. In one method, he extended Jacquin's method and used three dimensional blocks (or rectangular cubes) of video sequences instead of the 2-D blocks in still images. He reported that although the high compression could be achieved using this method, the quality of the decoded images were not good. Using another method, Beamount applied the 2-D Jacquin's method on individual frames but for each frame (except for the first frame) took the domain blocks from previous frame instead of the same frame. It was reported that 10 frame-per-second 352×288 gray scale video sequences could be coded at a data rate of 80 Kbits/s with "reasonable quality".

[4]The data shown in this plot are brought here only for a rough comparison. PSNR is not always a good measure of image quality. Also in regards to the 512×512, 8 bits per pixel grayscale test image Lena, authors are aware of at least two versions of this image that may have been used by researchers for obtaining these results. Some of the mentioned methods also did not use optimal entropy coders for coding of the fractal transform parameters.

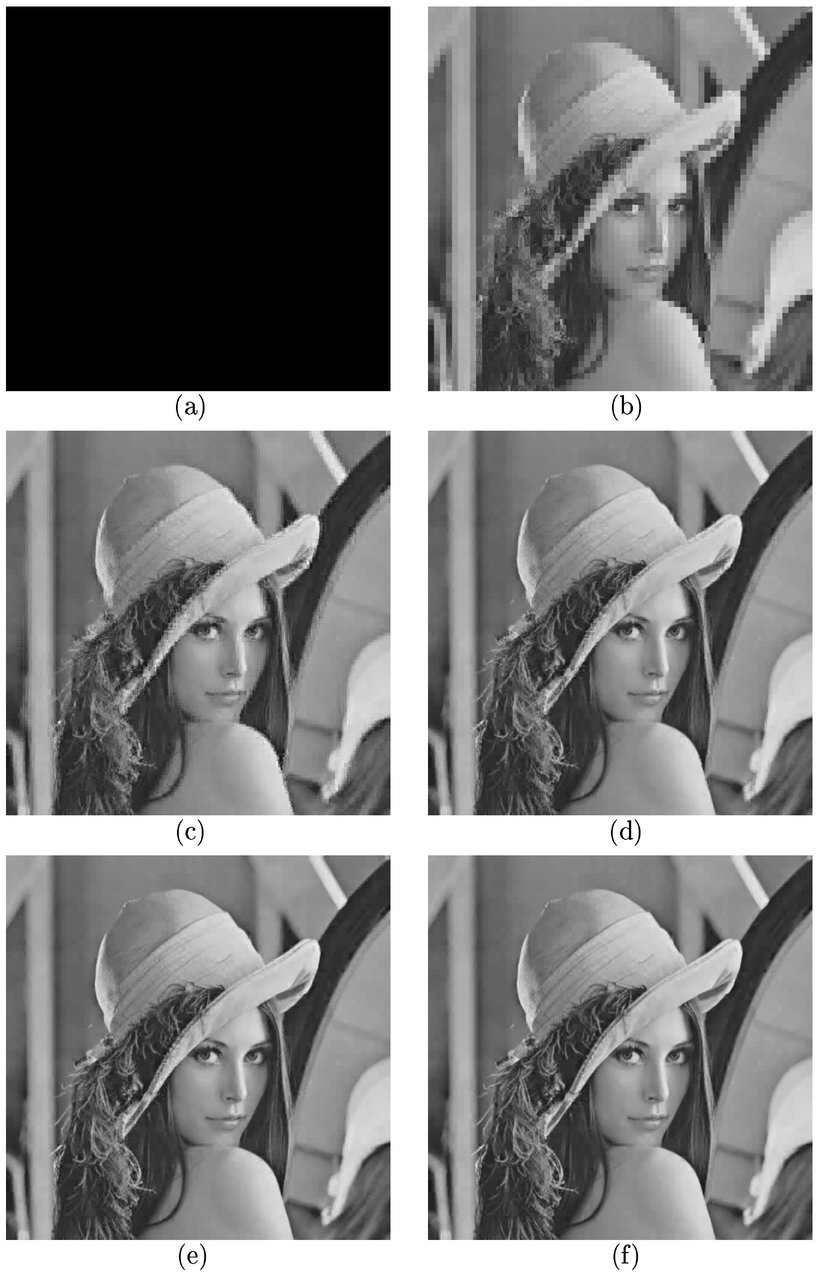

Figure 7 Five iterations of the decoding process

Figure 8 Original 512×512, 8 bits/pixel Lena image

Figure 9 Decoded Lena image: 0.43 bits/pixel at 34.5 dB PSNR

Figure 10 Decoded Lena image: 0.22 bits/pixel at 31.2 dB PSNR

Figure 11 Decoded Lena image: 0.15 bits/pixel at 29.2 dB PSNR

In 1992, Hurd et al. [62], from Iterated Systems, published results on fractal-based video compression claiming compression ratios from 21:1 (average PSNR of 39.2 dB) to 79:1 (average PSNR of 30.8 dB) for a 160×120, 8-bit grayscale video sequence. In their method, they encode the first frame using regular fractal coder. For the following frames, they always use the previous frame as the source of domain blocks. To approximate each range block in one frame they either (1) apply motion compensation and find a matching same-size domain block from previous decoded frame (no contraction) or (2) a single matching larger size domain block (with a contractive transformation applied on it) from the previous decoded frame is found. No residual error is sent for the frames. As the coding of this method is causal, the decoding process is non-iterative. In fact due to the low complexity of the decoding algorithm, this method has a very fast decompression.

In 1993, Hürtgen and Büttgen [63] applied fractal techniques for low bit rate video coding. They applied prediction by frame differencing with no motion compensation. Then for each frame, they applied the fractal transform only to those regions of the frame where prediction failed. For these regions they used a still fractal coding scheme. For range blocks located in these regions, domain blocks from the whole same frame were searched. In contrast to previous methods, for these regions they did not use previous frames. They also used a 3-level block splitting method in their algorithm. The 352×288, 8 1/3 Hz (25 Hz subsampled by 3) Miss America video sequence was reported to be coded at 128 Kbits/sec with an average PSNR of 36–37 dB, and at 64 Kbits/sec with an average PSNR of 34–35 dB, and at 32 Kbits/sec with an average PSNR of 30–32 dB. As the domain blocks for each range block were selected from the same frame, the decoder is iterative in this method.

Also in 1993, Li et al. [83] tried an extension of the still image compression method developed by Monro and Dudbridge [93] to video compression, and showed how compression ratios from 25:1 (average PSNR of 36.2 dB) to 51:1 (average PSNR of 27.2 dB) can be achieved for the 256×256, 15Hz 'Miss America' sequence. In this method the video sequence is partitioned into 3-D blocks. Each block is then partioned into 8 3-D sub-blocks each of which is approximated by a contractive transformation applied on the block that contains it.

In 1994, Lazar and Bruton [80] also extended Jacquin's 2-D algorithm to 3-D, and used 3-D range and domain blocks for image compression. They also used a 3-D block splitting method and the search for selecting domain blocks is done only in the neighborhood of the range block. They reported an average compression ratio of 74.39 at an average PSNR of 32–33 dB , for the 360×280, 8 bit/pixel 30 Hz 'Miss America' video sequence.

Some other researchers have also contribute to the implementations of fractal video coding [107, 76, 135, 94, 48, 52, 33, 97, 98].

As an example of results for fractal video coding, Figure 12 shows three original (a), (c), (e) and decoded (b), (d), (f) frames of the 352×288, 8bpp, 12.5 Hz (25 Hz subsampled by 2) Miss America video sequence. The sequence was coded using a method based on the generalized block coding algorithm described in [57, 58]. In this method, each range block in each frame is approximated by a linear combination of same-size blocks and larger-size blocks taken from the previous frame, and fixed blocks which in this example are DCT basis blocks. The number and type of selected blocks may vary from one block to another and are determined for each range block by an algorithm described in [57, 58]. The test sequence was coded at 80 Kbits/sec with an average PSNR of 36–37 dB.

3.3 Complexity

In terms of complexity, fractal-based image coding is asymmetric, i.e, the complexity of the encoder is typically much higher than that of the decoder. Complexities of these encoders are typically much higher than that of transform coders and vector quantizers. The most time consuming part of the encoding procedure is usually the search for finding the best matching domain blocks. Different techniques have been studied for limiting, structuring or approximating the search procedure [114, 116, 115].

In many implementations of fractal image coders the search is limited to the neighborhood of the range block where finding a good match is more likely. In the extreme case, the search may be totally avoided by using a predetermined domain block at the location of the range block. The search in Jacquin's original method included searching domain blocks that were generated by applying some isometric transformations on the image blocks (e.g. rotation by multiples of 90 degrees or reflection against horizontal or vertical axis). It has been found that it is more likely that best match being the domain block taken from the image rather than among the isometrically transformed versions and therefore in many fractal image and video coders these transformations are not used.

Figure 12 Three original (a,c,e) and decoded (b,d,f) frames of the Miss America sequence.

In some other implementations, the domain and range blocks are classified based on some criteria of structure of the blocks. Then for each range block the block matching search is done only among the domain blocks that are in the same class as of the range block.

Another approach to reduce the search is by doing a coarse to fine search. The search is done first using a coarse measure of similarity of blocks, and then another search with a finer measure of similarity is done among the blocks that have had high similarity using the coarser measure.

On the other hand, complexity of the fractal image decoders is usually much lower than their corresponding encoders and in some cases even less than some transform coding methods. This makes this method more suitable for publishing or broadcasting where the image must be compressed once by a central processor and decompressed many times by smaller receiving processors.

4 CONCLUSIONS

Although the field of fractal-based image coding is relatively young and its methods are different from other image coding methods, the performance of these methods has been comparable to those of the state of the art image compression methods in terms of combination of image quality and compression ratio. However, the complexity of the encoder for fractal-based image coders is typically high. These methods are typically based on the approximation of segments of the image with larger segments in the same image.

Acknowledgements

This work is supported in part by Joint Services Electronics Program Grant N00014-90-5-1270 and in part by a grant from Mitsubishi Electric.

REFERENCES

[1] S. Abenda. Inverse problem for one-dimensional fractal measures via iterated function systems and the moment method. *Inverse Problems*,

6(6):885–896, Dec. 1990.

[2] S. Abenda, S. Demko, and G. Turchetti. Local moments and inverse problem for fractal measures. *Inverse Problems*, 8(5):739–750, Oct. 1992.

[3] S. Abenda and G. Turchetti. Inverse problem for fractal sets on the real line via the moment method. *Il Nuovo Cimento*, 104B(2):213–227, Aug. 1989.

[4] A. Ait-Kheddache and S. A. Rajala. Texture classification based on higher-order fractals. In *Proceedings of IEEE ICASSP-88*, pages 1112–1115, Apr. 11–14, 1988.

[5] F. Arduini, C. Dambra, S. Dellepiane, S. B. Serpico, G. Vernazza, and R. Viviani. Fractal dimension estimation by adaptive mask selection. In *Proceedings of IEEE ICASSP-88*, pages 1116–1119. IEEE, 1988.

[6] F. Arduini, S. Fioravanti, and D. D. Giusto. A multifractal-based approach to natural scene analysis. In *Proceedings of IEEE ICASSP-91*, pages 2681–2684. IEEE, 1991.

[7] F. Arduini, S. Fioravanti, and D. D. Giusto. On computing multifractality for texture discrimination. In *Signal Processing VI: Theories and Applications. Proceedings of the Sixth European Signal Processing Conference (EUSIPCO-92)*, pages 1457–1460, Aug. 24–27, 1992.

[8] A. Arneodo, E. Bacry, and J. F. Muzy. Solving the inverse fractal problem from wavelet analysis. *Europhysics Letters*, 25(7):479–484, Mar. 1, 1994.

[9] M. F. Barnsley. Fractal functions and interpolation. *Constructive Approximation*, 2:303–329, 1986.

[10] M. F. Barnsley. Making chaotic dynamical systems to order. In M. F. Barnsley and S. G. Demko, editors, *Chaotic Dynamics and Fractals*, pages 53–68. Academic Press, Inc., New York, 1986. Proceedings of Conference on Chaotic Dynamics, Georgia Tech, March 25-29, 1985.

[11] M. F. Barnsley. Fractal modeling of real world images. In H. O. Peitgen and D. Saupe, editors, *The Science of Fractal Images*, chapter 5. Springer-Verlag, New York, 1988. Based on Lecture Notes for Fractals: Introductions, Basics and Perspectives, in SIGGRAPH'87 (Anaheim, California).

[12] M. F. Barnsley. *Fractals Everywhere*. Academic Press, Inc., New York, 1988.

[13] M. F. Barnsley, editor. *Constructive Approximation*, volume 5. Springer-Verlag, New York, 1989. Special issue on fractal approximation.

[14] M. F. Barnsley. Iterated function systems. In R. L. Devaney, L. Keen, K. T. Alligood, J. A. Yorke, M. F. Barnsley, B. Branner, J. Harrison, and P. J. Holmes, editors, *Chaos and Fractals: The Mathematics Behind the Computer Graphics*. American Mathematical Society, 1989.

[15] M. F. Barnsley. Methods and apparatus for image compression by iterated function systems. United States Patent Number 4,941,193, 1990.

[16] M. F. Barnsley and L. Anson. *The Fractal Transform*. Jones and Bartlett, Apr. 1993.

[17] M. F. Barnsley and S. Demko. Iterated function systems and the global construction of fractals. *Proceedings of the Royal Society of London*, A399:243–275, 1985.

[18] M. F. Barnsley, S. Demko, J. Elton, and J. Geronimo. Invariant measures for Markov processes arising from function iteration with place-dependent probabilities. *Annales de l'Institut Henry Poincare: Probabilites et statiques*, 24(3):367–394, 1988.

[19] M. F. Barnsley and J. Elton. A new class of Markov processes for image encoding. *Advances in Applied Probability*, 20:14–32, 1988.

[20] M. F. Barnsley, J. H. Elton, and D. P. Hardin. Recurrent iterated function systems. *Constructive Approximation*, 5(1):3–31, 1989.

[21] M. F. Barnsley, V. Ervin, D. Hardin, and J. Lancester. Solution of an inverse problem for fractals and other sets. *Proceedings of the National Academy of Sciences USA*, 83:1975–1977, Apr. 1986.

[22] M. F. Barnsley and L. P. Hurd. *Fractal Image Compression*. AK Peters, Ltd., Wellesley, Massachusetts, 1993.

[23] M. F. Barnsley and A. E. Jacquin. Application of recurrent iterated function systems to images. In *Proceedings of the SPIE, Visual Communications and Image Processing*, volume 1001, pages 122–131, 1988.

[24] M. F. Barnsley, A. E. Jacquin, F. Malassenet, L. Reuter, and A. Sloan. Harnessing chaos for image synthesis. In *Computer Graphics Conference Proceedings*, volume 22, pages 131–140. SIGGRAPH, Aug. 1988.

[25] M. F. Barnsley and H. Rising III. *Fractals Everywhere*. Academic Press Professional, Boston, second edition, 1993.

[26] M. F. Barnsley and A. D. Sloan. Chaotic compression. *Computer Graphics World*, pages 107–108, Nov. 1987.

[27] M. F. Barnsley and A. D. Sloan. A better way to compress images. *BYTE*, pages 215–223, Jan. 1988.

[28] M. F. Barnsley and A. D. Sloan. Fractal image compression. In H. K. Ramapriyan, editor, *Proceedings of the Scientific Data Compression Workshop*, pages 351–365, Snowbird, Utah, May 3–5, 1988. NASA Godard Space Flight Center. NASA conference publication 3025.

[29] M. F. Barnsley and A. D. Sloan. Method and apparatus for processing digital data. United States Patent Number 5,065,447, 1991.

[30] K. U. Barthel, J. Schüttemeyer, T. Voyé, and P. Noll. A new image coding technique unifying fractal and transform coding. In *Proceedings of IEEE International Conference on Image Processing*, volume 3, pages 112–116, Austin, Texas, Nov. 13–16, 1994.

[31] J. M. Beaumont. Image data compression using fractal techniques. *British Telecommunications Technical Journal*, 9(4):93–109, Oct. 1991.

[32] D. Bessis and S. Demko. Stable recovery of fractal measures by polynomial sampling. *Physica D*, 47:427–438, 1991.

[33] A. Bogdan. Multiscale (inter/intra-frame) fractal video coding. In *Proceedings of IEEE International Conference on Image Processing*, volume 1, pages 760–764, Austin, Texas, Nov. 13–16, 1994.

[34] C. Cabrelli, U. Molter, and E. R. Vrscay. Recurrent iterated function systems: Invariant measures, a collage theorem and moment relations. In H. . Peitgen, J. M. Henriques, and L. F. Penedo, editors, *Fractals in the Fundamental and Applied Sciences*, pages 71–80. Elsevier Science Publishers B. V. (North-Holland), 1991.

[35] C. A. Cabrelli, B. Forte, U. M. Molter, and E. R. Vrscay. Iterated fuzzy set systems: A new approach to the inverse problem for fractals and other sets. *Journal of Mathematical Analysis and Applications*, 171(1):79–100, Nov. 15, 1992.

[36] C. Chang and S. Chatterjee. Fractal based approach to shape description, reconstruction and classification. In *Proceedings of Twenty-Third Asilomar Conference on Signals, Systems and Computers*, pages 172–176, Oct. 30–Nov. 1, 1989.

[37] J. H. Chen and J. D. Kalbfleisch. Inverse problems in fractal construction: Hellinger distance method. *Journal of the Royal Statistical Society, Series B Methodological*, 56(4):687–700, 1994.

[38] C. K. Cheong, K. Aizawa, T. Saito, and M. Hatori. Structural edge detection based on fractal analysis for image compression. In *IEEE International Symposium on Circuits and Systems*, pages 2461–2464, San Diego, CA, May 10–13, 1992.

[39] M. Crouse and K. Ramchandran. Joint thresholding and quantizer selection for decoder-compatible baseline JPEG. In *Proceedings of IEEE ICASSP-95*, 1995.

[40] K. Culik II and J. Kari. Inference algorithms for WFA and image compression. In Y. Fisher, editor, *Fractal Image Compression: Theory and Application*, pages 243–258. Springer-Verlag, New York, 1995.

[41] G. Davis. Self-quantized wavelet subtrees: A wavelet-based theory for fractal image compression. In *DCC'95: Data Compression Conference*, Snowbird, Utah, Mar. 28–30, 1995.

[42] G. A. Edgar. *Measure, Topology, and Fractal Geometry.* Undergraduate Texts in Mathematics. Springer-Verlag, New York, 1990.

[43] J. H. Elton and Z. Yan. Approximation of measure by Markov processes and homogeneous affine iterated function systems. *Constructive Approximation*, 5:69–87, 1989.

[44] K. J. Falconer. *Fractal Geometry: Mathematical Foundations and Applications.* John Wiley and Sons, Chichester, 1990.

[45] Y. Fisher. A discussion of fractal image compression. In H.-O. Peitgen, H. Jürgens, and D. Saupe, editors, *Chaos and Fractals: New Frontiers of Science*, appendix A, pages 903–919. Springer-Verlag, 1992.

[46] Y. Fisher. Fractal image compression. In P. Prusinkiewicz, editor, *SIGGRAPH '92 Course Notes: From Folk Art to Hyperreality*, pages 1–21, 1992.

[47] Y. Fisher and S. Menlove. Fractal encoding with HV partitions. In Y. Fisher, editor, *Fractal Image Compression: Theory and Application*, pages 119–136. Springer-Verlag, New York, 1995.

[48] Y. Fisher, D. Rogovin, and T. P. Shen. Fractal (self-VQ) encoding of video sequences. In *Proceedings of the SPIE, Visual Communications and Image Processing*, Chicago, Illinois, Sept. 25–28, 1994.

[49] B. Forte and E. R. Vrscay. Solving the inverse problem for measures using iterated function systems. Submitted to Advances in Applied Probability, 1993.

[50] B. Forte and E. R. Vrscay. Solving the inverse problem for function/image approximation using iterated function systems, I. Theoretical basis. In *Proceedings of Fractals in Engineering'94*, pages 143–152, Montreal, Quebec, Canada, June 1–4, 1994. Published in volume 2, number 3 issues of the journal *Fractals* (World Scientific Publishing Corp.).

[51] B. Forte and E. R. Vrscay. Solving the inverse problem for function/image approximation using iterated function systems, II. Algorithm and computations. In *Proceedings of Fractals in Engineering'94*, pages 153–164, Montreal, Quebec, Canada, June 1–4, 1994. Published in volume 2, number 3 issue of the journal *Fractals* (World Scientific Publishing Corp.).

[52] R. E. H. Franich, R. L. Lagendijk, and J. Biemind. Fractal coding in an object-based system. In *Proceedings of IEEE International Conference on Image Processing*, volume 2, pages 405–408, Austin, Texas, Nov. 13–16, 1994.

[53] G. C. Freeland and T. S. Durrani. IFS fractals and the wavelet transform. In *Proceedings of IEEE ICASSP-90*, pages 2345–2348, 1990.

[54] I. Gerner and Y. Y. Zeevi. Generalized scanning and multiresolution image compression. In J. A. Storer and J. H. Reif, editors, *DCC'91: Data Compression Conference*, page 434, Snowbird, Utah, Apr. 8–11, 1991. IEEE Computer Society Press.

[55] M. Gharavi-Alkhansari and T. S. Huang. A fractal-based image block-coding algorithm. In *Proceedings of IEEE ICASSP-93*, volume V, pages 345–348, Minneapolis, Minnesota, Apr. 27–30, 1993.

[56] M. Gharavi-Alkhansari and T. S. Huang. A fractal-based image block-coding algorithm. In *Proceedings of Picture Coding Symposium*, page 1.7, Lausanne, Switzerland, Mar. 17–19, 1993.

[57] M. Gharavi-Alkhansari and T. S. Huang. Fractal-based techniques for a generalized image coding method. In *Proceedings of IEEE International Conference on Image Processing*, volume 3, pages 122–126, Austin, Texas, Nov. 13–16, 1994.

[58] M. Gharavi-Alkhansari and T. S. Huang. Generalized image coding using fractal-based methods. In *Proceedings of the International Picture Coding Symposium*, pages 440–443, Sacramento, California, Sept. 21–23, 1994.

[59] M. Gharavi-Alkhansari and T. S. Huang. Fractal image coding using rate-distortion optimized matching pursuit. In *Proceedings of the SPIE, Visual Communications and Image Processing*, volume 2727, Orlando, Florida, Mar. 17–20, 1996.

[60] B. D. Goel and S. C. Kwatra. A data compression algorithm for color images based on run-length coding and fractal geometry. In *IEEE International Conference on Communications'88*, pages 1253–1256, June 12–15, 1988.

[61] C. R. Handy and G. Mantica. Inverse problems in fractal construction: Moment method solution. *Physica D*, 43:17–36, May 1990.

[62] L. P. Hurd, M. A. Gustavus, and M. F. Barnsley. Fractal video compression. In *Digest of Papers. Thirty-Seventh IEEE Computer Society International Conference (COMPCON)*, pages 41–42, Feb. 24–28, 1992.

[63] B. Hürtgen and P. Büttgen. Fractal approach to low rate video coding. In *Proceedings of the SPIE, Visual Communications and Image Processing*, volume 2094, pages 120–131, Cambridge, Massachusetts, Nov. 8–11, 1993.

[64] J. E. Hutchinson. Fractals and self-similarity. *Indiana University Mathematics Journal*, 30(5):713–747, Sept.–Oct. 1981.

[65] E. W. Jacobs, R. D. Boss, and Y. Fisher. Fractal-based image compression, II. Technical Report 1362, Naval Ocean Systems Center, San Diego, CA, June 1990.

[66] E. W. Jacobs, Y. Fisher, and R. D. Boss. Image compression: A study of the iterated transform method. *Signal Processing*, 29(3):251–263, Dec. 1992.

[67] A. E. Jacquin. *A Fractal Theory of Iterated Markov Operators with Applications to Digital Image Coding*. PhD thesis, Georgia Institute of Technology, Aug. 1989.

[68] A. E. Jacquin. Fractal image coding based on a theory of iterated contractive image transformations. In *Proceedings of the SPIE, Visual Communications and Image Processing*, volume 1360, pages 227–239, Oct. 1–4, 1990.

[69] A. E. Jacquin. A novel fractal block-coding technique for digital images. In *Proceedings of IEEE ICASSP-90*, pages 2225–2228, Apr. 3–6, 1990.

[70] A. E. Jacquin. Image coding based on a fractal theory of iterated contractive image transformations. *IEEE Transactions on Image Processing*, 1(1):18–30, Jan. 1992.

[71] A. E. Jacquin. Fractal image coding: A review. *Proceedings of the IEEE*, 81(10):1451–1465, Oct. 1993.

[72] J. Jang and S. Rajala. Segmentation based image coding using fractals and the human visual system. In *Proceedings of IEEE ICASSP-90*, pages 1957–1960, 1990.

[73] J. Jang and S. A. Rajala. Texture segmentation-based image coder incorporating properties of the human visual system. In *Proceedings of IEEE ICASSP-91*, pages 2753–2756, 1991.

[74] J. M. Keller, R. M. Crownover, and R. Y. Chen. Characteristics of natural scenes related to the fractal dimension. *IEEE Transactions on Patthern Analysis and Machine Intelligence*, PAMI-9(5):621–627, Sept. 1987.

[75] I. K. Kim and R.-H. Park. Image coding based on fractal approximation and vector quantization. In *Proceedings of IEEE International Conference on Image Processing*, volume 3, pages 132–136, Austin, Texas, Nov. 13–16, 1994.

[76] O. Kiselyov and P. Fisher. Self-similarity of the multiresolutional image/video decomposition: Smart expansion as compression of still and moving pictures. In *DCC'94: Data Compression Conference*, page 514. IEEE Computer Society Press, Mar. 29–31, 1994.

[77] D. C. Knill, D. Field, and D. Kersten. Human discrimination of fractal images. *Journal of Optical Society of America. A, Optics and Image Science*, 7(6):1113–1123, June 1990.

[78] H. Krupnik, D. Malah, and E. Karnin. Fractal representation of images via the discrete wavelet transform. In *IEEE 18th Conv. of EE in Israel*, Tel-Aviv, Israel, Mar. 7–8, 1995.

[79] W. S. Kuklinski. Utilization of fractal image models in medical image processing. In *Proceedings of Fractals in Engineering'94*, pages 180–186, Montreal, Quebec, Canada, June 1–4, 1994. Published in volume 2, number 2 and 3 issues of the journal *Fractals* (World Scientific Publishing Corp.).

[80] M. S. Lazar and L. T. Bruton. Fractal coding of digital video. *IEEE Transactions on Circuits and Systems for Video Technology*, 4(3), June 1994.

[81] S. Lepsøy, G. Øien, and T. A. Ramstad. Attractor image compression with a fast non-iterative decoding algorithm. In *Proceedings of IEEE ICASSP-93*, volume V, pages 337–340, Apr. 27–30 1993.

[82] J. Lévy Véhel and P. Mignot. Multifractal segmentation of images. In *Proceedings of Fractals in Engineering'94*, pages 187–193, Montreal, Quebec, Canada, June 1–4, 1994. Published in volume 2, number 2 and 3 issues of the journal *Fractals* (World Scientific Publishing Corp.).

[83] H. Li, M. Novak, and R. Forchheimer. Fractal-based image sequence compression scheme. *Optical Engineering*, 32(7):1588–1595, July 1993.

[84] D. W. Lin. Fractal image coding as generalized predictive coding. In *Proceedings of IEEE International Conference on Image Processing*, volume 3, pages 117–121, Austin, Texas, Nov. 13–16, 1994.

[85] G. Lu and T. L. Yew. Image compression using quadtree partitioned iterated function systems. *Electronics Letters*, 30(1):23–24, Jan. 6, 1994.

[86] B. B. Mandelbrot. *Les Objets Fractals: Forme, Hasard et Dimension.* Flammarion, Paris, 1975. (In French).

[87] B. B. Mandelbrot. *The Fractal Geometry of Nature.* W. H. Freeman and Company, New York, 1982.

[88] B. B. Mandelbrot and R. F. Voss. *Fractals: Form, Chance and Dimension.* Freeman, San Francisco, 1977.

[89] G. Mantica. Techniques for solving inverse fractal problems. In H. . Peitgen, J. M. Henriques, and L. F. Penedo, editors, *Fractals in the Fundamental and Applied Sciences*, pages 255–268. Elsevier Science Publishers B. V. (North-Holland), 1991.

[90] G. Mantica and A. Sloan. Chaotic optimization and the construction of fractals: Solution of an inverse problem. *Complex Systems*, 3:37–62, Feb. 1989.

[91] B. Moghaddam, K. J. Hintz, and C. V. Stewart. A comparison of local fractal dimension estimation methods. *Pattern Recognition and Image Analysis*, 2(1), Mar. 1992.

[92] D. M. Monro and F. Dudbridge. Fractal approximation of image blocks. In *Proceedings of IEEE ICASSP-92*, volume III, pages 485–488, San Francisco, California, Mar. 1992.

[93] D. M. Monro and F. Dudbridge. Fractal block coding of images. *Electronics Letters*, 28(11):1053–1055, May 21, 1992.

[94] D. M. Monro and J. A. Nicholls. Real time fractal video for personal communications. In *Proceedings of Fractals in Engineering'94*, pages 206–209, Montreal, Quebec, Canada, June 1–4, 1994. Published in volume 2, number 2 and 3 issues of the journal *Fractals* (World Scientific Publishing Corp.).

[95] D. J. Nettleton and R. Garigliano. Evolutionary algorithms and a fractal inverse problem. *Biosystems*, 33(3):221–231, 1994.

[96] G. E. Øien, S. Lepsøy, and T. A. Ramstad. An inner product space approach to image coding by contractive transformations. In *Proceedings of IEEE ICASSP-91*, pages 2773–2776, May 14–17, 1991.

[97] B.-B. Paul and M. H. Hayes. Fractal-based compression of motion video sequences. In *Proceedings of IEEE International Conference on Image Processing*, volume 1, pages 755–759, Austin, Texas, Nov. 13–16, 1994.

[98] B.-B. Paul and M. H. Hayes III. Video coding based on iterated function systems. In *Proceedings of IEEE ICASSP-95*, volume 4, pages 2269–2272, Detroit, Michigan, May 9–12, 1995.

[99] H.-O. Peitgen and D. Saupe, editors. *The Science of Fractal Images*. Springer-Verlag, New York, 1988.

[100] S. Peleg, J. Naor, R. Hartley, and D. Avnir. Multiple resolution texture analysis and classification. *IEEE Transactions on Patthern Analysis and Machine Intelligence*, PAMI-6(4):518–523, July 1984.

[101] T. Peli. Multiscale fractal theory and object characterization. *Journal of Optical Society of America A*, 7:1101–1112, 1990.

[102] T. Peli, V. Tom, and B. Lee. Multi-scale fractal and correlation signatures for image screening and natural clutter suppression. In *Proceedings of the SPIE, Visual Communications and Image Processing IV*, volume 1199, pages 402–415, 1989.

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

[104] A. Pentland and B. Horowitz. A practical approach to fractal-based image compression. In J. A. Storer and J. H. Reif, editors, *DCC'91: Data Compression Conference*, pages 176–185. IEEE Computer Society Press, Los Alamitos, Apr. 8–11, 1991.

[105] A. P. Pentland. Fractal-based description of natural scenes. *IEEE Transactions on Patthern Analysis and Machine Intelligence*, PAMI-6(6):661–674, Nov. 1984.

[106] A. P. Pentland. Fractal surface models for communications about terrain. In *Proceedings of the SPIE, Visual Communications and Image Processing II*, volume 845, pages 301–306, Oct. 1987.

[107] E. Reusens. Sequence coding based on the fractal theory of iterated transformations systems. In *Proceedings of the SPIE, Visual Communications and Image Processing*, volume 2094, pages 132–140, Cambridge, Massachusetts, Nov. 8–11, 1993.

[108] R. Rinaldo and G. Calvagno. An image coding scheme using block prediction of the pyramid subband decomposition. In *Proceedings of IEEE International Conference on Image Processing*, Austin, Texas, Nov. 13–16, 1994.

[109] R. Rinaldo and G. Calvagno. Image coding by block prediction of multiresolution subimages. *IEEE Transactions on Image Processing*, 4(7):909–920, July 1995.

[110] R. Rinaldo and A. Zakhor. Inverse problem for two-dimensional fractal sets using the wavelet transform and the moment method. In *Proceedings of IEEE ICASSP-92*, volume IV, pages 665–668, San Francisco, California, Mar. 20–23, 1992.

[111] R. Rinaldo and A. Zakhor. Fractal approximation of images. In *DCC'93: Data Compression Conference*, page 451, Mar.–Apr. 1993.

[112] R. Rinaldo and A. Zakhor. Inverse and approximation problem for two-dimensional fractal sets. *IEEE Transactions on Image Processing*, 3(6):802–820, Nov. 1994.

[113] S. K. Rogers et al. Synthetic aperture radar segmentation using wavelets and fractals. In *Proceedings of the IEEE International Conference on Systems Enginering*, pages 21–24, 1991.

[114] D. Saupe. Breaking the time complexity of fractal image compression. Technical Report 53, Institut für Informatik, Universität Freiburg, 1994.

[115] D. Saupe. Accelerating fractal image compression by multi-dimensional nearest neighbor search. In J. A. Storer and M. Cohn, editors, *DCC'95: Data Compression Conference*, Snowbird, Utah, Mar. 28–30, 1995.

[116] D. Saupe and R. Hamzaoui. Complexity reduction methods for fractal image compression. In J. M. Blackledge, editor, *I.M.A. Conf. Proc. on Image Processing; Mathematical Methods and Applications*. Oxford University Press, Sept. 1994.

[117] J. M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on Signal Processing*, 41(12):3445–3462, Dec. 1993.

[118] R. Shonkwiler, F. Mendivil, and A. Deliu. Genetic algorithms for the 1-D fractal inverse problem. In *4th International Conference on Genetic Algorithms (ICGA 91)*, pages 495–501, San Diego, CA, July 13–16, 1991.

[119] M. Stein. Fractal image models and object detection. In *Proceedings of the SPIE, Visual Communications and Image Processing II*, volume 845, pages 293–306, Oct. 1987.

[120] R. J. Stevens, A. F. Lehar, and F. H. Preston. Manipulation and presentation of multidimensional image data using Peano scan. *IEEE Transactions on Patthern Analysis and Machine Intelligence*, PAMI-5(5):520–526, Sept. 1983.

[121] C. V. Stewart, B. Moghaddam, K. J. Hintz, and L. M. Novak. Fractaional Brownian motion models for synthetic aperture radar imagery. *Proceedings of the IEEE*, 81(10):1511–1522, Oct. 1993.

[122] M. Temerinac, A. Kozarev, Z. Trpovski, and B. Simsic. An efficient image compression algorithm based on filter bank analysis and fractal geometry. In J. Vandewalle, R. Boite, M. Moonen, and A. Oosterlinck, editors, *Proceedings of Signal Processing VI: Theories and Applications*, page 1373. Elsevier Science Publishers B. V., 1992.

[123] J. Theiler. Estimating fractal dimension. *Journal of Optical Society of America. A, Optics and Image Science*, 7(6):1055–1073, June 1990.

[124] L. Thomas and F. Deravi. Pruning of the transform space in block-based fractal image compression. In *Proceedings of IEEE ICASSP-93*, volume V, pages 341–344, Minneapolis, Minnesota, Apr. 27–30, 1993.

[125] L. Thomas and F. Deravi. Region-based fractal image compression using heuristic search. *IEEE Transactions on Image Processing*, 4(6):832–838, June 1995.

[126] K. S. Thyagarajan and S. Chatterjee. Fractal scanning for images compression. In *Proceedings of Twenty-Fifth Asilomar Conference on Signals, Systems and Computers*, Nov. 4–6, 1991.

[127] D. van Schooneveld. The moment method for invariant measure approximation. Internal report, Department of Mathematics and Computer Science, Delft University of Technology, 1990.

[128] A. M. Vepsalainen and J. Ma. Estimating of fractal and correlation dimension from 2D and 3D images. In *Proceedings of the SPIE, Visual Communications and Image Processing IV*, volume 1199, pages 431–438, 1989.

[129] G. Vines and M. H. Hayes, III. Adaptive IFS image coding with proximity maps. In *Proceedings of IEEE ICASSP-93*, volume V, pages 349–352, Minneapolis, Minnesota, Apr. 27–30, 1993.

[130] E. R. Vrscay. Moment and collage methods for the inverse problem of fractal construction with iterated function systems. In H. . Peitgen, J. M. Henriques, and L. F. Penedo, editors, *Fractals in the Fundamental and Applied Sciences*, pages 443–461. Elsevier Science Publishers B. V. (North-Holland), 1991. June 6–8 1990.

[131] E. R. Vrscay and C. J. Roehrig. Iterated function systems and the inverse problem of fractal construction using moments. In E. Kaltofen and S. M. Watt, editors, *Computers and Mathematics*, pages 250–259. Springer-Verlag, Berlin, 1989.

[132] E. R. Vrscay and D. Weil. Missing moment and perturbative methods for polynomial iterated function systems. *Physica D*, 50:478–492, July 1991.

[133] E. Walach and E. Karnin. A fractal-based approach to image compression. In *Proceedings of IEEE ICASSP-86*, pages 529–532, Tokyo, Japan, Apr. 7–11, 1986.

[134] G. K. Wallace. The JPEG still picture compression standard. *Communications of the ACM*, 34(4):30–44, Apr. 1991.

[135] D. L. Wilson, J. A. Nicholls, and D. M. Monro. Rate buffered fractal video. In *Proceedings of IEEE ICASSP-94*, volume 5, pages 505–508, Adelaide, Australia, Apr. 19–22, 1994.

[136] Z. Xiong, K. Ramchandran, M. T. Orchard, and K. Asai. Wavelet packets-based image coding using joint space-frequency quantization. In *Proceedings of IEEE International Conference on Image Processing*, volume 3, pages 324–328, Austin, Texas, Nov. 13–16, 1994.

[137] K.-M. Yang, L. Wu, and M. Mills. Fractal based image coding scheme using Peano scan. In *Proceedings of IEEE International Symposium on Circuits and Systems*, pages 2301–2304, Espoo, Finland, June 7–9, 1988.

[138] N. Zhang and H. Yan. Hybrid image compression method based on fractal geometry. *Electronics Letters*, 27(5):406–408, Feb. 28, 1991.