

# Fractal Image Denoising

Mohsen Ghazel, George H. Freeman, and Edward R. Vrscay

**Abstract**—Over the past decade, there has been significant interest in fractal coding for the purpose of image compression. However, applications of fractal-based coding to other aspects of image processing have received little attention. In this paper we propose a fractal-based method to enhance and restore a noisy image. If the noisy image is simply fractally coded, a significant amount of the noise is suppressed. However, one can go a step further and estimate the fractal code of the original noise-free image from that of the noisy image, based upon a knowledge (or estimate) of the variance of the noise, assumed to be zero-mean, stationary and Gaussian. The resulting fractal code yields a significantly enhanced and restored representation of the original noisy image. The enhancement is consistent with the human visual system where extra smoothing is performed in flat and low activity regions and a lower degree of smoothing is performed near high frequency components, e.g., edges, of the image. We find that, for significant noise variance ( $\sigma > 20$ ), the fractal-based scheme yields results that are generally better than those obtained by the Lee filter which uses a localized first order filtering process similar to fractal schemes. We also show that the Lee filter and the fractal method are closely related.

**Index Terms**—Fractal image coding, fractals, image denoising, image restoration.

## I. INTRODUCTION

THE need for image restoration is encountered in many practical applications. Distortion due to additive white Gaussian noise (AWGN) can be caused by poor quality image acquisition, images observed in a noisy environment or noise inherent in communication channels. Linear filtering and smoothing operations have been widely used for image restoration because of their relative simplicity. However, since these methods are based upon the assumption that the image signal is stationary and formed through a linear system, their effectiveness, although generally acceptable, is limited. In reality, real-world images have typically nonstationary statistical characteristics, and are formed through a nonlinear system process where the intensity distribution arriving at the imaging system is the product of the reflectance of the object or the scene of interest and the illumination distribution falling on the scene. There exist various adaptive and nonlinear

image restoration methods that account for the variations in the local statistical characteristics. These methods achieve better enhancement and restoration of the image while preserving high frequency features of the original images such as edges [13], [22], [23].

In this paper, we explore an adaptive, locally linear yet globally nonlinear method of image restoration and enhancement based on fractal image coding. As discussed in Section II, fractal image coding [5], [8], [9], [19] has received much interest over the past decade, mostly in the context of image compression. Very briefly, it seeks to approximate an image function  $u(x, y)$  as a union of modified copies of itself. The net result is that the target  $u$  is approximated by the attractive fixed point  $\bar{u}$  of a contractive *fractal transform* operator  $T$  that performs the shrinking and gray-level modifying operations on image functions. It is the transform  $T$ , in terms of the *fractal code* parameters that define its geometric and gray-level transformations, that is stored in computer memory, often requiring much less storage than the original image  $u$ . The approximation  $\bar{u}$  may be generated by the iteration procedure  $u_{n+1} = Tu_n$ , where  $u_0$  is a suitable starting “seed”, for example, a blank screen.

Over the years, a variety of highly competitive fractal and fractal-wavelet image compression methods have been developed. Most share a common feature, performing some kind of block *collage coding* as originally introduced by Jacquin [14]. Their variety lies in the different strategies employed to obtain the best possible matching of blocks within an image, subject to constraints. (The books [5], [8], [9], [19] represent excellent surveys of the field. Perhaps the most complete listing of papers on fractal image coding is to be found at the Leipzig fractal compression ftp site <ftp://axes.informatik.uni-leipzig.de/pub/Fractal/index.html>.) These schemes yield results that are comparable, in terms of rate-distortion performance, to some of the most efficient still image compression methods.

However, to the best of our knowledge, little or no attention has been given to the use of such fractal-based methods for the purpose of image enhancement and restoration. Indeed, one of the original motivations for this study was the observation that a noisy image is somewhat denoised when it is fractally coded. This led to the question of whether such a simple fractal encoding of the *noisy* image could be used as a starting point to estimate the fractal code of the *noiseless* image, perhaps with some knowledge of the noise, e.g., its variance. We examine these questions in this paper and show that the answers are in the affirmative.

First we show that straightforward fractal-based coding performs rather well as a denoiser. In retrospect, this is not surprising since the (white Gaussian) noise process is not represented well by the (local) linear transform that maps parent

Manuscript received November 30, 2001; revised February 21, 2003. This work was supported in part by the Natural Sciences and Engineering Council of Canada (individual research grants for G. H. Freeman and E. R. Vrscay), an Ontario Graduate Scholarship (for M. Ghazel), and the Electrical and Computer Engineering Department of the University of Waterloo (for M. Ghazel). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Nasser Kehtarnavaz.

M. Ghazel and G. H. Freeman are with the Department of Electrical and Computer Engineering, Faculty of Engineering, University of Waterloo, Waterloo, ON, Canada N2L 3G1 (e-mail: mghazel@engmail.uwaterloo.ca; freeman@uwaterloo.ca).

E. R. Vrscay is with the Department of Applied Mathematics, Faculty of Mathematics, University of Waterloo, Waterloo, ON, Canada N2L 3G1 (e-mail: ervrscay@uwaterloo.ca).

Digital Object Identifier 10.1109/TIP.2003.818038

blocks to child blocks. As such, we expect a reduction in noise. Indeed, this is essentially the basis of the local linear minimum mean squared error Lee filter [16] so that we may consider fractal coding to be closely related to Lee filtering.

Given a noisy image, we then show that its fractal code parameters—in particular, the gray-level map coefficients—can be used to estimate those of its *noiseless* counterpart, assuming that we know (or can estimate) the variance of the (white Gaussian) noise. This leads to an improvement in the fractal approximation to a target image  $u$ . We show theoretically how this method performs in a manner similar to that of the human visual system [22], producing extra smoothing in flat, low activity regions and a lower degree of smoothing near high activity regions, including edges. We also show that, for high noise, the results of the fractal-based scheme are better both qualitatively and quantitatively (in terms of SNR) than those obtained by the Lee filter.

The layout of the paper is as follows. We briefly outline the basics of fractal image coding in Section II. Only two methods of partitioning the image—uniform and quadtree—are considered. We then show that the fractal coding of noisy images results in some denoising. In Section III, we derive a theoretical relationship between the fractal code of a noisy image and the fractal code of its noiseless counterpart, keeping the parent–child assignments the same and assuming additive white Gaussian noise. In Section IV, we outline a scheme to predict the true fractal code of the noiseless image and thereby enhance the denoising. Some experimental results are presented. In Section V, we compare the results of this fractal-based method with those obtained by the Lee filter. We also compare the Lee filter method to the fractal transform method. Some additional experimental results and comparisons with the Lee filter are then presented.

## II. ESSENTIALS OF FRACTAL IMAGE CODING

Fractal image coding is based on the original theory of Iterated Function Systems (IFS) [2], [3]. In what follows, we provide some bare essentials of fractal image coding. For those readers interested in seeing the larger mathematical picture, we summarize the main ideas of contraction maps and associated inverse problems in Appendix A. Let  $\mathcal{I}$  denote an image of interest as defined by an *image function*  $u(x, y)$  supported over a region  $X \in \mathbf{R}^2$ . Here  $x, y \in X$  denote spatial coordinates of a point or pixel of  $\mathcal{I}$ . Now suppose that there exists a suitable partition  $\mathcal{R}$  of  $X$  into range sub-blocks  $R_i$  so that  $X = \cup_i R_i$ . For simplicity, the  $R_i$  are assumed to be “nonoverlapping,” intersecting only over their boundary curves. (In the discrete case, i.e., pixels, there is no overlapping.) Assume that associated with each sub-block  $R_i$  is a larger domain sub-block  $D_i \subseteq X$  so that  $R_i = w_i(D_i)$ , where  $w_i$  is a one-to-one contraction map in the continuous (nondigitized) plane. As illustrated in Fig. 1, typically, the  $R_i$  and  $D_i$  blocks are square pixel blocks and the  $w_i$  are affine contractions that may also rotate or invert the  $R_i$ . In this case there are eight such possible maps.

Now suppose that the image function  $u(R_i)$  supported on  $R_i$  is well approximated by a modified copy of the image function  $u(D_i)$  supported on  $D_i$  as follows:

$$u(R_i) \cong \phi_i(u(D_i)) = \phi_i(u(w_i^{-1}(R_i))) \quad (1)$$

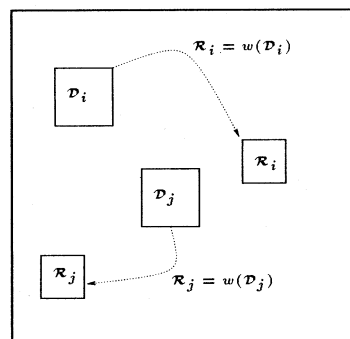


Fig. 1. Fractal transformation of a domain block  $D_i$  into a range block  $R_i$ .

where  $\phi_i : \mathbf{R} \rightarrow \mathbf{R}$  is a *gray-level* map that operates on the pixel intensities. Because of the nonoverlapping nature of the partition, we may write

$$u(x, y) \cong (Tu)(x, y) = \sum_i \phi_i(u(w_i^{-1}(x, y))). \quad (2)$$

In other words, the image function  $u$  is approximated by a union of spatially-contracted ( $w_i$ ) and gray-level-distorted ( $\phi_i$ ) copies of itself. We may consider this union of modified copies as defining a special kind of *fractal transform* operator  $T$  on image functions.

Under suitable conditions on the spatial maps  $w_i$  and the gray-level maps  $\phi_i$ , the operator  $T$  is *contractive*. From Banach’s fixed point theorem for contractive maps, there then exists a unique fixed point function,  $\bar{u}$ , such that  $\bar{u} = T\bar{u}$ . Furthermore  $\bar{u}$  is *attractive*: If we start with any image function  $u_0$  supported on  $X$  and construct the iteration sequence of functions  $u_{n+1} = Tu_n$ , then the  $u_n$  converge to  $\bar{u}$  as  $n \rightarrow \infty$ . (In the discrete alphabet case, e.g.,  $N \times N$  pixel images, the sequence converges after a finite number of iterations.)

Moreover, a simple triangle inequality (the so-called *Collage Theorem*—see Appendix A) establishes that if  $u$  is “close” to  $Tu$ , then  $u$  is also “close” to  $\bar{u}$ , implying that  $\bar{u}$  is a good approximation to  $u$ . This is the essence of fractal image coding.

### A. Practicalities of Fractal Image Coding

We now address some practical issues left unaddressed in the above discussion. For example, what kind of partitioning for the  $R_i$  should be used? Given a *child* or *range block*  $R_i$ , how does one determine an “optimal” *parent* or *domain block*  $D_i$ , with corresponding gray-level mapping  $\phi_i$ , that produces the best approximation in (2)?

For the remainder of this paper, unless otherwise indicated, we consider images as square arrays of pixels, for example  $512 \times 512$  arrays. Associated with each pixel  $p_{ij}$ , is a gray-level value  $u_{ij} \geq 0$ .

1) *Range Block Partitioning*: A variety of partitioning strategies have been employed in the literature. Since it is not our purpose to review these methods we refer the reader to [8], [9], [19] for discussions on this topic. In this study we consider two simple, yet fundamentally different, schemes.

- 1) **Uniform partitioning**: The image is simply partitioned into  $N \times N$  square sub-blocks, where  $N \geq 2$ . Typically we choose  $N = 2^l$ , for some integer  $l \geq 1$ .

2) **Quadtree partitioning:** This is perhaps the most common method of adaptive partitioning. Beginning with the original image, square pixel blocks are broken down into quadrants in a recursive tree structure. The partitioning, which will vary throughout the image, is terminated when a particular condition is satisfied. Typically, regions of higher image activity, for example edges, will produce partitions of finer resolution, i.e., small block sizes. As such, edge information is rather well represented in quadtree-based coding schemes, including fractal coding.

2) *Domain “Pools”:* We assume that for each range block  $R_i$  there exists a set or “pool”  $\mathcal{D}_i$  of domain blocks  $D_i \subset X$ . The domain blocks are usually assumed to be twice the length and width (hence four times the area) of the range blocks. Naturally, the larger the domain pool, the better the approximation that can be achieved in (2). However, a larger domain pool requires more searching time as well as more memory for the storage of indices that specify the locations of optimal domain blocks. From a compression viewpoint, some compromise between domain pool size and fidelity must be established.

In the “regular” partitioning scheme, where  $N^2$  nonoverlapping child blocks of identical size are used, a very simple, yet limited, strategy is to partition the image region  $X$  into  $M^2$  nonoverlapping parent blocks, where  $M = N/2$ . The resulting domain pool is then common to all range blocks  $R_i$ . This will be one of the parent-child schemes used in this paper. In this scheme, each domain block contains four range blocks. (A number of other strategies can be found in the literature. For example, in the various “Bath Fractal Transforms” [20], [21], [25], no searching is performed in an effort to reduce the coding time. One strategy is to use the larger domain block that is co-centric with the range block. Some of the bits that are saved by not having to code parent block indices are used to code additional gray-level map parameters that improve the collage fitting. Other strategies consider shifts of the co-centric domain block.)

In the adaptive quadtree scheme, if one assumes that the partitioning has produced a range block  $R_i$  of size  $n \times n$  pixels, then one may wish to consider a domain pool of nonoverlapping  $2n \times 2n$  pixel blocks of the image region  $D$ . This is the strategy that we have used in this paper.

3) *Geometric Mappings  $w$ :* There are eight contractive affine mappings that can transform a square parent block into a smaller square child block—four rotations, a horizontal flipping, a vertical flipping and two diagonal flippings. We consider such mappings as the composition of a nonrotating affine transformation followed by an isometry. (In some applications in the literature, only a subset of the eight possible isometries are employed. In some cases, only the zero rotation mapping is used.) We shall denote these geometric mappings of domain block  $D_i$  to range block  $R_j$  as  $w_{ij}^{(k)}$ , where  $1 \leq k \leq 8$ . In most applications, including those in this paper, the domain blocks are contracted by a factor of two in each of the  $x$  and  $y$  directions.

The action of these geometric contractions is straightforward when the  $x$  and  $y$  spatial coordinates are continuous, i.e., real-valued. However, in discrete pixel space, the shrinking of an  $2n \times 2n$  parent pixel block to an  $n \times n$  child pixel block must be achieved by some kind of reduction or *decimation* procedure.

Typically, a sub-block of  $2 \times 2$  neighboring pixels in the parent block is replaced by a single pixel and the four gray-level values are replaced by their average value. In the discrete case, we assume that decimation is included in the action of the maps  $w_{ij}^{(k)}$ . The gray-level maps (Section II-A4) are then applied to these average values.

4) *Gray-Level Mappings  $\phi$ :* Once the parent blocks  $D_i$  are reduced to the same pixel size as the child blocks  $R_j$ , we perform a first order linear prediction to estimate the latter from the former using affine transformations of the form

$$\phi(t) = \alpha t + \beta. \quad (3)$$

The gray-level map that best maps the gray-level values supported on the (decimated) block  $D_i$  to their counterparts on  $R_j$  is the map  $\phi(t)$  that minimizes the so-called *collage distance* over  $R_j$

$$\Delta_{ij}^{(k)} = \left\| \phi \left( u \left( w_{ij}^{(k)}(D_i) \right) \right) - u(R_j) \right\|_2. \quad (4)$$

In practice, the  $\mathcal{L}^2$  norm is used so that the optimal gray-level map performs a least-squares fit of the parent-child gray-level data. If we let  $x_m$  and  $y_m$ ,  $1 \leq m \leq n$ , denote the gray-level values on, respectively, the geometrically transformed (plus decimated) parent block  $D_i$  and the child block  $R_j$ , then minimization of the squared  $\mathcal{L}^2$  distance

$$\left( \Delta_{ij}^{(k)} \right)^2 = \sum_{m=1}^n [y_m - (\alpha x_m + \beta)]^2 \quad (5)$$

yields a set of linear equations in  $\alpha$  and  $\beta$  with solution (assuming a nonzero determinant)

$$\alpha^* = \frac{n \sum_{m=1}^n x_m y_m - \left[ \sum_{m=1}^n x_m \right] \left[ \sum_{m=1}^n y_m \right]}{n \sum_{m=1}^n x_m^2 - \left[ \sum_{m=1}^n x_m \right]^2} \quad (6)$$

$$\beta^* = \frac{1}{n} \sum_{m=1}^n y_m - \alpha^* \frac{1}{n} \sum_{m=1}^n x_m. \quad (7)$$

There is one complication, however, in that the contractivity of the fractal transform operator  $T$  is dependent upon the  $\alpha$  scaling coefficients. There is no simple relationship between the  $\mathcal{L}^2$  contractivity factor of  $T$  and the  $\alpha$  coefficients because of the local nature of the parent-child mappings. However, in the  $\mathcal{L}^\infty$  norm, contractivity is guaranteed if all  $\alpha$  satisfy the condition  $|\alpha| < 1$ . For this reason, most fractal coding algorithms “clamp” the  $\alpha$  coefficients, i.e.,  $\alpha = \text{sgn}(\alpha) \min(|\alpha|, 1)$ . The resulting fractal transform operators  $T$  are almost always contractive in  $\mathcal{L}^\infty$ , hence in  $\mathcal{L}^2$ , due to the equivalence of the norms in finite pixel space. (Some authors, e.g., [8], have performed the truncation at  $\sqrt{2}$  with no noticeable effects on the contractivity.)

### B. Fractal Encoding Algorithm

We now have all of the ingredients necessary to perform a fractal encoding of an image  $\mathcal{I}$ . For simplicity, we assume that a particular partitioning  $\mathcal{R}$  of range blocks  $R_k$ ,  $k = 1, \dots, N_R$  has been constructed. Associated with each block  $R_k$  is a pool of domain blocks  $\mathcal{D}_k$ , as discussed earlier.

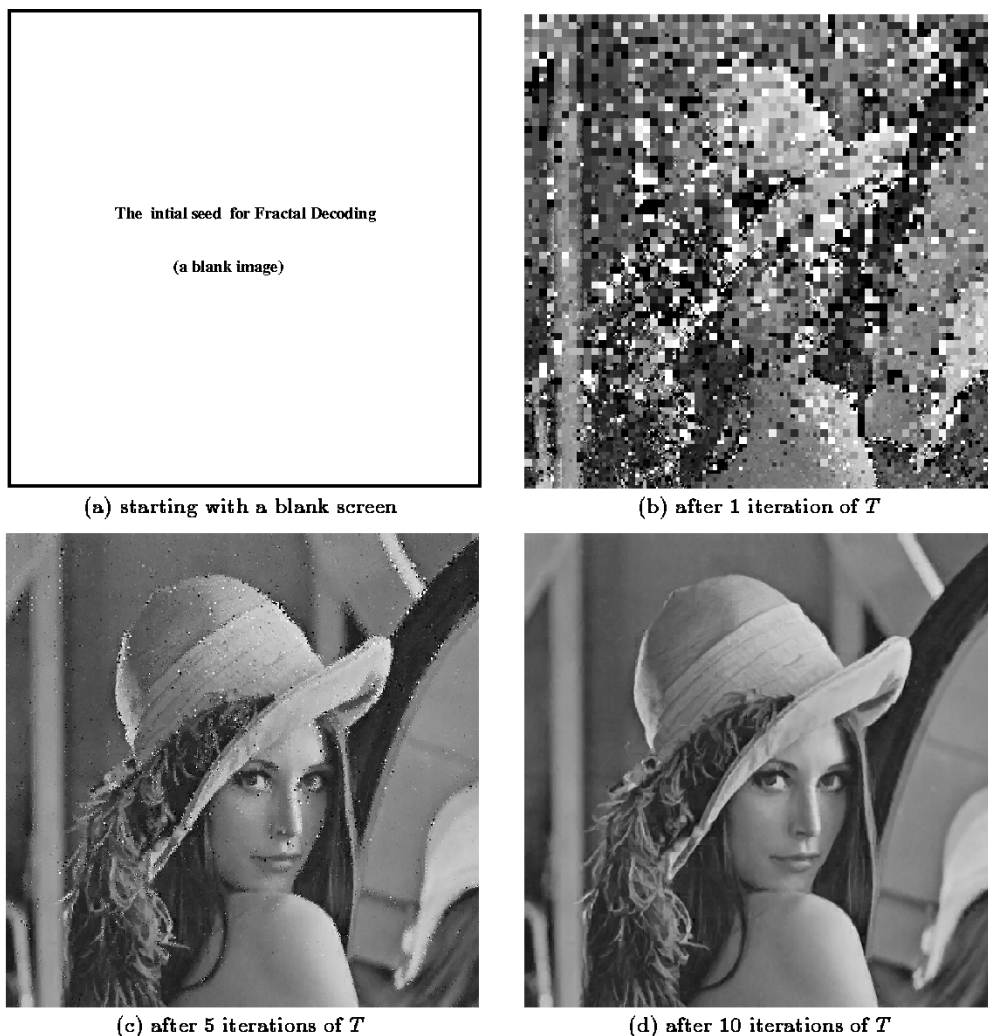


Fig. 2. Generating a fractal approximation of the target image “Lena” by iterating the fractal transform  $T$ , starting with a blank initial image. Standard uniform partitioning scheme, using  $(M, N) = (32, 64)$ .

For each range block,  $R_k$ ,  $1 \leq k \leq N_R$ , we seek the domain block  $D_{i(k)} \in \mathcal{D}_k$  such that the subimage  $u(D_{i(k)})$  best approximates the subimage  $u(R_k)$  after a geometric transformation/decimation,  $w_{ik}^{(m)}$ . In other words, we find the indices

$$(i(k), m(k)) = \arg \left[ \min_{D_i \in \mathcal{D}_k} \min_{1 \leq m \leq 8} \left\| \alpha_{ik}^m u(w_{ik}^{(m)}(D_i)) + \beta_{ik}^m - u(R_k) \right\|_2^2 \right] \quad (8)$$

where  $\alpha_{ik}^m$  and  $\beta_{ik}^m$  denote the least-squares gray-level coefficients associated with the  $m^{\text{th}}$  geometric mapping  $w_{ik}^{(m)}$ . (We omit the starred superscripts for the moment.)

The *fractal code* associated with the partition  $\mathcal{R}$  and the associated domain pools  $\mathcal{D}$  will then be, for each range block,  $R_k \in \mathcal{R}$ ,  $1 \leq k \leq N_R$ .

- 1) The index  $i(k)$  of its optimal parent block  $D_{i(k)} \in \mathcal{D}_k$ .
- 2) The optimal isometry  $m(k)$ . (Note that we may choose *not* to consider all eight possible isometries.)
- 3) The optimal (least-squares) gray-level coefficient pair  $(\alpha_{i(k),k}^{(m(k))}, \beta_{i(k),k}^{(m(k))})$ .

In the case of the uniform partitioning scheme, one will also have to provide the partition parameters  $M$  and  $N$ . In the case of

quadtree partitioning, the adaptive partitioning will have to be coded. This can be done in a straightforward recursive manner, first adopting a scanning order over a set of four quadrants (e.g., clockwise) and then coding a “1” if a quadrant is partitioned or a “0” if it is not. In the case of a “1” we then scan the four subquadrants of the quadrant, etc. [8].

The fractal code defines a fractal transform operator  $T$  that acts on an image function  $u$  according to (2). To obtain the image values of  $Tu$  in (range) block  $R_k$ , we take the (domain) block  $D_{i(k)}$  and apply the following operations:

- 1) decimate the domain block to produce a block of the same size as  $R_k$ ;
- 2) apply the appropriate isometry,  $1 \leq m(k) \leq 8$ , to the block;
- 3) modify the gray-level values of this block according to the appropriate gray-level map  $\phi(t) = \alpha_{i(k),k}^{(m(k))}t + \beta_{i(k),k}^{(m(k))}$ ;
- 4) replace the image values in (range) block  $R_k$  by those of the above block.

Under suitable conditions (involving the  $\alpha$  coefficients, as discussed earlier)  $T$  is contractive, implying the existence of a unique fixed point image  $\bar{u} = T\bar{u}$ . Beginning with any starting image (for example, a blank screen where  $u_0 = 0$ ), the iteration sequence  $u_{n+1} = Tu_n$  converges to  $\bar{u}$ . This process is illustrated in Fig. 2.



Fig. 3. Fractal representations of the test image of “Lena” for various resolutions values  $(M, N)$ . Note that the  $(64, 128)$  fractal representation is visually identical to the original image.

Now let  $u$  be a target image that we wish to fractally encode. From the Collage Theorem (Appendix A), if we find an operator  $T$  such that the *collage distance*,  $\|u - Tu\|_2$ , is small, then  $\|u - \bar{u}\|_2$  is small, i.e.,  $\bar{u}$  is a good approximation to  $u$ . It is the fractal code defining  $T$  that is stored in the computer and used to generate the approximation  $\bar{u}$  to  $u$ .

Historically, it was found that much less computer memory was needed to store the fractal code than the image  $u$ , resulting in *data compression*. As is the case for any compression algorithm, the goal of *fractal image compression* is to produce approximations  $\bar{u}$  of maximal fidelity, i.e., lowest error  $\|u - \bar{u}\|_2$ , subject to constraints on the memory required to store the fractal code. This also involves the question of optimal storage of fractal coefficients in terms of quantization and entropy coding. This paper, however, is not concerned with fractal coding for compression purposes. As such, compression ratios, quantization and entropy coding will not play a role here.

### C. Example Using Uniform Image Partitioning

In Fig. 3, the fractal representation of the standard test image of “Lena,”  $(512 \times 512)$  pixels, 8 bpp—which we consider to be the *noiseless* image—is shown at various domain-range block

resolutions  $(M, N)$ . RMSE denotes the usual  $\mathcal{L}^2$  or root mean squared error: The distance between two image functions  $u$  and  $v$  with pixel values  $0 \leq u_{ij}, v_{ij} \leq 255$ ,  $1 \leq i, j \leq 512$ , is given by

$$\text{RMSE} = \frac{1}{512} \left[ \sum_{i,j} (u_{ij} - v_{ij})^2 \right]^{1/2}.$$

If we consider  $v$  to be an approximation to  $u$  then the PSNR is given by

$$\text{PSNR (dB)} = 20 \log_{10} \left[ \frac{255}{\text{RMSE}} \right].$$

As expected, the quality of the representation increases with the fractal resolution  $(M, N)$ .

### III. IMAGE RESTORATION USING SIMPLE FRACTAL CODING

In this section, we investigate the use of fractal coding for image denoising and restoration. The noisy image used here is the Lena test image that has been degraded by an additive white Gaussian noise (AWGN) with variance  $\sigma_N^2 = 25^2$ . Both the uniform as well as quadtree partitioning schemes will be used.



Fig. 4. (a) Original noisy image with noise standard deviation  $\sigma = 25$  and (b)-(d) the standard fractal representations of the noisy image of “Lena” for various resolutions  $(M, N)$ .

In Fig. 4 the fractal representation of the noisy image at various resolutions  $(M, N)$  is illustrated. We make the following observations:

- The fractal representation at the resolution  $(M, N) = (32, 64)$  is the best, quantitatively, as reflected by the RMSE and PSNR fidelity measures. However, although most of the noise has been smoothed and suppressed, the image appears blurry since some of the edges and other high-frequency features have been smoothed.
- For the lower resolution,  $(M, N) = (16, 32)$ , significant blockiness and blurriness artifacts, including blurry edges, are noticeable in the fractal approximation.
- For the higher resolution,  $(M, N) = (64, 128)$ , the fractal representation appears rather close to the noisy image, as expected. Since the noise is being more accurately represented, the fractal approximation appears rather noisy.

Clearly, there is a tradeoff between the resolution and the quality of the image. A low resolution results in smoothing most of the noise at the expense of blurriness of edges and blockiness artifacts. On the other hand, a higher resolution results in a greater reconstruction of the noise, while preserving the high frequency content (edges) of the image.

#### A. Quadtree Image Partitioning

In principle, the human visual system is less sensitive to noise near edges and more sensitive to noise in smoother regions of the image. This motivates us to investigate the quadtree partitioning scheme which permits the use of different resolutions or block sizes for different parts of the image. In particular, we expect that finer resolutions (i.e., smaller sub-blocks) will be used near edges and other high activity areas. From the Section II, we expect these areas to yield noisy fractal representations while preserving the edges well.

The presence of an acceptable amount of noise in these high frequency regions of the fractal representation may not be a problem due to the fact that the human visual system is less sensitive to noise near edges. On the other hand, we expect that a more coarse partitioning will be suitable for flat and low activity regions, resulting in a higher degree of smoothing and denoising.

The above is achieved if we use a *signal-to-noise ratio* (SNR) quadtree decomposition criterion. The SNR,  $\gamma$ , of a child sub-block  $X$  is computed as

$$\gamma = \frac{\sigma_X^2}{\sigma_N^2}. \quad (9)$$

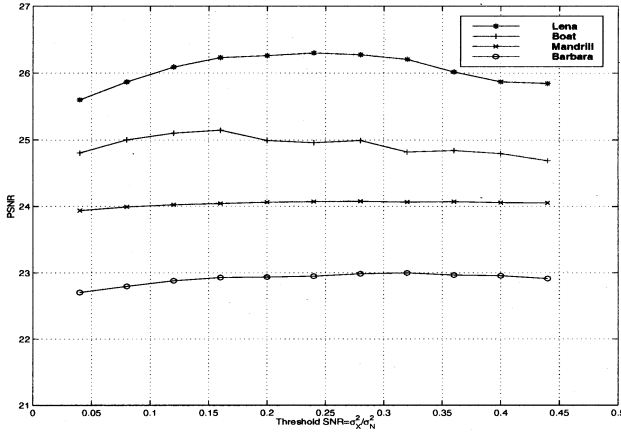


Fig. 5. Distortion curve illustrating the quality of the fractal representations as a function of the SNR threshold, for the noisy images of Lena, Barbara, Boat, and Mandrill, with  $\sigma_N^2 = 625$ . The optimal threshold for Lena is seen to be  $\gamma_c \approx 0.25$ .

As we shall show in (18), the SNR  $\gamma$  can be computed from the noisy child sub-block  $\hat{X}$  and the noise variance  $\sigma_N^2$  as follows:

$$\gamma = \frac{\sigma_{\hat{X}}^2}{\sigma_N^2} - 1. \quad (10)$$

A sub-block is partitioned into four quadtree if its SNR,  $\gamma$ , exceeds a prescribed threshold  $\gamma_c$ . Otherwise, the sub-block is simply fractally encoded.

Higher SNR values reflect the existence of edges or other high pixel variability within the sub-block. Fig. 5 illustrates the relationship between the quality of the quadtree-based fractal representations and the signal to noise threshold,  $\gamma_c$ , for the the noisy images of Lena, Barbara, Boat, and Mandrill, with noise variance  $\sigma_N^2 = 625$ . Note that there is a trade-off between the quadtree-resolution and the reconstruction of the noise. On the one hand, a higher SNR threshold of  $\gamma_c$  results in coarse quadtree partitioning which, in turn, produces smoother yet coarser fractal representations of the image. On the other hand, a lower SNR threshold yields a finer quadtree partitioning, resulting in finer but noisier fractal representations. For instance, Fig. 5 shows that there is an optimal critical SNR threshold,  $\gamma_c \approx 0.25$ , for Lena. This represents an optimal trade-off between the quadtree-based fractal resolution and the noise reconstruction.

Whenever using the quadtree partitioning algorithm for the purpose of fractal image coding, one has to choose an optimal decomposition criterion. In our case, we had to determine the “optimal” value,  $\gamma_c$ , for the SNR threshold decomposition criterion. In Fig. 5, we note that quality of the various fractal representations is rather stable and is not highly sensitive to various values of  $\gamma_c$ , especially for the Barbara and Mandrill images. For the image of interest, we note that,  $25.6 \leq \text{PSNR} \leq 26.3$ , whenever  $0 \leq \gamma_c \leq 0.5$ , with an optimal value for the SNR threshold is  $\gamma_c \approx 0.25$ . From (10), this seems to indicate that we partition a noisy image sub-block  $\hat{X}$  if

$$\sigma_{\hat{X}}^2 \geq 1.25\sigma_N^2. \quad (11)$$

From Fig. 5, this decomposition criterion seems reasonable for the other test images as well. In general, we expect the op-

timal value for  $\gamma_c$  depends on the image and the noise level, so one can, experimentally, generate lookup tables for classes of images and various noise levels. Thus,  $\gamma_c$ , can be viewed as a denoising *fine-tuning* parameter that measures the tradeoff between suppressing the noise and reconstructing the high frequency content of the image, such as edges. This is somewhat similar to the idea of using fine-tuning parameters adopted in JPEG, which are obtained from lookup tables. These parameters are used to visually enhance the quality of the image representation.

Fig. 6 illustrates the quadtree segmentation of the image as well as the fractal representation of the image using this SNR threshold. Note that the fractal representation has a relatively high RMSE (or low PSNR) due to the presence of noise near edges—a careful observation of the fractal representation will reveal such noise. However, the edges remain sharp, in contrast to the uniform partitioning case. Flat regions of the image, such as the shoulder and the background, are relatively smooth—most of the noise in these regions has been suppressed in the fractal representation. It is important to mention that the quadtree-based fractal representation is indeed visually better than the qualitative measures (i.e., RMSE and PSNR) seem to indicate. This is because, as mentioned previously, most of the remaining noise is localized in the vicinity of edges and other high-frequency content of the image where the human visual system is less sensitive to noise. Hence the noise is less noticeable.

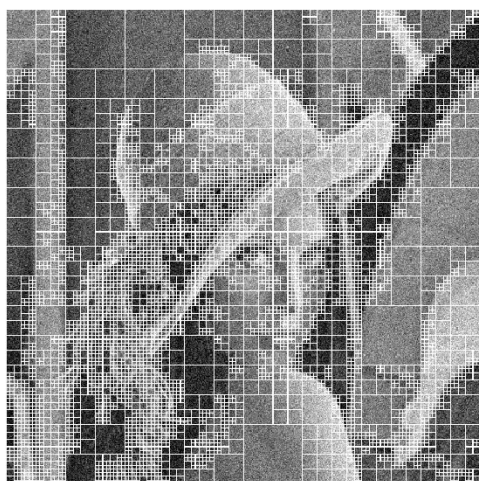
In summary, the quadtree partitioning scheme permits more smoothing away from edges and lesser smoothing near edges, thus achieving the smoothing and denoising of an image without degrading its edges.

#### IV. PREDICTING THE FRACTAL CODE OF THE NOISE-FREE IMAGE

In this section, we examine the relation between a noisy image  $\hat{\mathcal{I}}$  and its noiseless counterpart  $\mathcal{I}$ , specifically in terms of their respective fractal gray-level map coefficients. This relationship provides a method of estimating the fractal parameters of the noiseless image from those of the noisy image. From the former, a fractal representation of the noiseless image can then be reconstructed. In the discussion that follows, variables and coefficients that correspond to a noisy image will have hats, e.g.,  $\hat{X}$ ,  $\hat{Y}$  for the noisy image as opposed to  $X$ ,  $Y$  for the noiseless image.

Before proceeding further, it will be useful to rewrite the least-squares gray-level coefficients in (6) in terms of standard statistical quantities. We regard an image as a random signal so that the gray-level values  $\{x_m, m = 1, 2, \dots, n\}$  and  $\{y_m, m = 1, 2, \dots, n\}$  in (6) can be considered as random samples of the random variables  $X$  and  $Y$  representing the gray-level distribution of the (decimated) parent block  $D$  and child block  $R$ . The least-squares coefficients can then be written as

$$\begin{aligned} \alpha^* &= \frac{\text{Cov}(X, Y)}{\sigma_X^2} \\ \beta^* &= E[Y] - \alpha^* E[X] \end{aligned} \quad (12)$$



(a) Quadtree partitioning of the image



(b) Fractal denoising: RMSE=12.35, PSNR=26.30.

Fig. 6. Standard fractal image restoration using quadtree partitioning using the optimal SNR threshold,  $\gamma_c = 0.25$ , for the noisy Lena with  $\sigma_{\mathcal{N}}^2 = 625$ .

where

$$\begin{aligned} Cov(X, Y) &= \frac{\sum_{i=1}^n x_i y_i}{n} - \frac{\sum_{i=1}^n x_i}{n} \frac{\sum_{i=1}^n y_i}{n}, \\ \sigma_X^2 &= \frac{\sum_{i=1}^n x_i^2}{n} - \left[ \frac{\sum_{i=1}^n x_i}{n} \right]^2, \\ E[X] &= \frac{\sum_{i=1}^n x_i}{n} \text{ and } E[Y] = \frac{\sum_{i=1}^n y_i}{n}. \end{aligned}$$

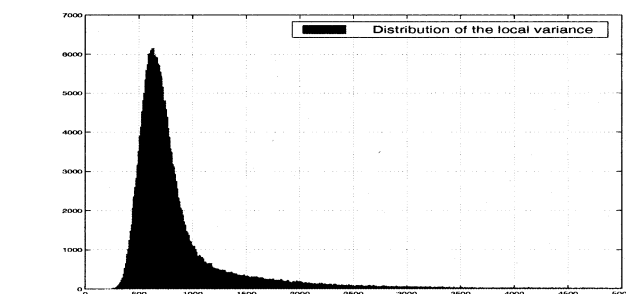
Strictly speaking, the above expressions are approximations to the statistical quantities of the random variables  $X$  and  $Y$  since they represent (finite) sample statistics. For large  $n$ , the sample statistics provide good estimates of the population statistics. The fact that  $n$  will not be large in our applications will contribute to errors in estimating the local image statistics and, subsequently, optimal fractal codes for the noiseless images.

We now address the problem of estimating the noise variance in an image.

#### A. Estimation of Noise Variance $\sigma_{\mathcal{N}}^2$

Throughout most of this paper, for illustrative purposes, we are using a noisy version of the  $512 \times 512$  pixel Lena test image that has been degraded by AWGN of variance  $\sigma_{\mathcal{N}}^2 = 25^2$ . Since in practice the noise variance is generally not known, we first outline how to estimate the noise variance from the noisy image.

One method of estimating the noise variance is based on the assumption that an image has many regions of almost uniform intensity and that any changes in these regions of insignificant variation is due to the noise. This assumption is generally valid for many real-world images. The background of a scene is an example of such a region of insignificant variation. We also assume that the noise  $\mathcal{N}$  has a constant variance  $\sigma_{\mathcal{N}}^2$  throughout the image because of the assumed stationarity of the white Gaussian process. Calculating local variance estimates in small windows ( $5 \times 5$  or  $7 \times 7$  pixel blocks) and taking the

Fig. 7. The histogram of the local noise variance computed from  $7 \times 7$  masks of the noisy “Lena” image. Note that the distribution of the local variance follows a  $\chi^2$  distribution.

minimum gives one noise estimate. Generally, the minimum local variance estimate provides an underestimate of the noise variance. A better estimate might be obtained by examining the distribution of the local variance across the entire image. If most local regions have no signal variations, the *mode* (i.e., the most frequent value) of the local variance histogram is a better estimate of the noise variance.

Fig. 7 illustrates the distribution of the of the local variances using  $7 \times 7$  windows. We observe that the local noise variance  $\sigma_{\mathcal{N}}^2$  follows approximately a  $\chi^2$  distribution. It was found that the minimum variance  $\hat{\sigma}_{\min}^2 = 225$  underestimate the true noise variance, but the mode variance  $\hat{\sigma}_{\text{mode}}^2 = 636$  yields a better estimate. In what follows, we shall use the mode estimate of the noise variance.

#### B. Prediction of the Gray-Level Coefficients

As above, let  $X$  and  $Y$  denote the random variables representing the gray-level values in a transformed parent block  $D$  and its corresponding child block  $R$ , respectively, for the noise-free original image. Also, let  $\hat{X}$  and  $\hat{Y}$  denote the corresponding gray-level random variables for the noisy image. Recall that the random variable  $\hat{X}$  represents the gray-level values of the pixels in the transformed parent block.

Various decimation methods have been used in the literature to produce from the parent block a transformed block of the same size as the child block, generally four times smaller. These



include 1) downsampling by taking every fourth pixel and 2) averaging over  $2 \times 2$  pixel blocks. The averaging operation is preferred and will be used in this study. However, averaging over  $2 \times 2$  pixel blocks affects the noise variance since

$$\hat{x}_i = x_i + \bar{n}_{i,4}$$

where

$$\bar{n}_{i,4} = \frac{n_i + n_{i+1} + n_{i+2} + n_{i+3}}{4}.$$

Since the noise  $\mathcal{N}$  is stationary,  $\bar{n}_{i,4}$  is independent of the location index  $i$  and it is a sample from the averaged random noise

$$\bar{\mathcal{N}}_4 = \frac{\mathcal{N}_1 + \mathcal{N}_2 + \mathcal{N}_3 + \mathcal{N}_4}{4}, \text{ where } \mathcal{N}_i \sim N(0, \sigma_{\mathcal{N}}^2).$$

Clearly,  $\bar{\mathcal{N}}_4$  is also Gaussian with

$$E[\bar{\mathcal{N}}_4] = 0, \text{ Var}(\bar{\mathcal{N}}_4) = \frac{\sigma_{\mathcal{N}}^2}{4}.$$

Thus, the averaged noise  $\bar{\mathcal{N}}_4$  is an AWGN process

$$\bar{\mathcal{N}}_4 \sim N\left(0, \frac{\sigma_{\mathcal{N}}^2}{4}\right).$$

The relationship between the random variables  $\hat{X}$  and  $X$  corresponding to a domain block of the noisy and the noiseless image, respectively, can be written as follows:

$$\hat{X} = X + \bar{\mathcal{N}}_4 \Rightarrow E[\hat{X}] = E[X], \text{ since } E[\bar{\mathcal{N}}_4] = 0.$$

On the other hand, for the child blocks, no averaging or downsampling is required, so the relationship between  $\hat{Y}$  and  $Y$  is given by

$$\hat{Y} = Y + \mathcal{N} \Rightarrow E[\hat{Y}] = E[Y].$$

Assuming that the image signal and the noise signal are independent, the variance of the noisy vector  $\hat{X}$  is

$$\sigma_{\hat{X}}^2 = \text{Var}(X + \bar{\mathcal{N}}_4) = \sigma_X^2 + \frac{\sigma_{\mathcal{N}}^2}{4} \quad (13)$$

where  $\sigma_X^2$  is the variance of the noise-free vector  $X$ . Also, under the independence assumption between the noise and the image signal as well as the independence between  $\bar{\mathcal{N}}_4$  and  $\mathcal{N}$ , we have

$$\text{Cov}(\hat{X}, \hat{Y}) = \text{Cov}(X + \bar{\mathcal{N}}_4, Y + \mathcal{N}) = \text{Cov}(X, Y). \quad (14)$$

The independence between  $\bar{\mathcal{N}}_4$  and  $\mathcal{N}$  can be achieved by insuring that parent block  $\hat{X}$  does not overlap with child block  $\hat{Y}$ . From (12)–(14), we can express the *gain*  $\hat{\alpha}^*$  in terms of the statistics of the noiseless image  $\mathcal{I}$  and the noise variance

$$\hat{\alpha}^* = \frac{\text{Cov}(\hat{X}, \hat{Y})}{\sigma_{\hat{X}}^2} = \frac{\text{Cov}(X, Y)}{\sigma_X^2 + \frac{\sigma_{\mathcal{N}}^2}{4}} = \frac{\frac{\text{Cov}(X, Y)}{\sigma_X^2}}{1 + \frac{\sigma_{\mathcal{N}}^2}{4\sigma_X^2}}. \quad (15)$$

The *offset*  $\hat{\beta}^*$  is given by

$$\hat{\beta}^* = E[\hat{Y}] - \hat{\alpha}^* E[\hat{X}] = E[Y] - \hat{\alpha}^* E[X]. \quad (16)$$

From this result and (15) we have

$$\hat{\alpha}^* = \frac{\alpha^*}{1 + \frac{1}{4\gamma}} \quad (17)$$

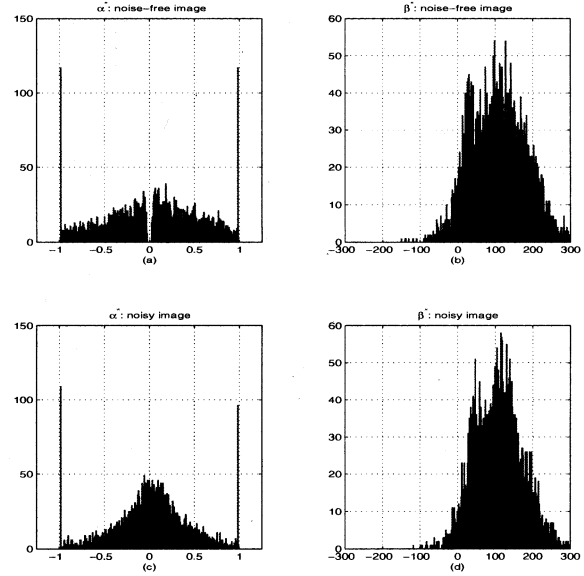


Fig. 8. Histogram plots of fractal gray-level coefficients  $\alpha^*$  and  $\beta^*$ , respectively. (a) and (b): Fractal coding of the noise-free image. (c) and (d): Fractal coding of the noisy image when the parent–child assignments of the noise-free image are used.

where the *signal-to-noise ratio* is defined as

$$\gamma = \frac{\sigma_X^2}{\sigma_{\mathcal{N}}^2} = \frac{\sigma_{\hat{X}}^2 - \sigma_{\mathcal{N}}^2}{\sigma_{\mathcal{N}}^2} = \frac{\sigma_{\hat{X}}^2}{\sigma_{\mathcal{N}}^2} - 1. \quad (18)$$

We now rearrange the above to express the noise-free image fractal gray-level coefficients ( $\alpha^*, \beta^*$ ) in terms of the noisy image coefficients ( $\hat{\alpha}^*, \hat{\beta}^*$ );

$$\begin{aligned} \alpha^* &= \left(1 + \frac{1}{4\gamma}\right) \hat{\alpha}^* \\ \beta^* &= E[\hat{Y}] - \hat{\alpha}^* E[\hat{X}]. \end{aligned} \quad (19)$$

These relations now allow us to predict the noise-free image coefficients from the noisy image. Note the following.

- In the absence of noise, that is, as  $\sigma_{\mathcal{N}}^2 \rightarrow 0$ , then  $\gamma \rightarrow \infty$  so that

$$\alpha^* \rightarrow \hat{\alpha}^* \text{ and } \beta^* \rightarrow \hat{\beta}^*$$

as expected.

- Near edges or other high activity regions of the image, where  $\gamma$  is relatively large, we have

$$\hat{\alpha}^* \approx \alpha^* \text{ and } \hat{\beta}^* \approx \beta^*.$$

It is worthwhile to recall that the relationships between fractal codes of noiseless and noisy images, as expressed in (19), assume that the parent–child matching is the same in both images. In order to illustrate these relationships, we present in Fig. 8 the histograms of the least squares gray-level coefficients for the noise-free and the noisy Lena image using the parent–child matching of the noise-free image.

The fractal code of the noise-free image of “Lena” consisting of the  $\alpha^*, \beta^*$  and the optimal parent–child pairs and the corresponding geometric maps are first obtained. The histograms of the gray-level parameters,  $\alpha^*, \beta^*$ , for the noise-free image are presented in Fig. 8(a) and (b), respectively. The histograms of

the gray-level parameters,  $\hat{\alpha}^*$ ,  $\hat{\beta}^*$ , for the noisy image are presented in Fig. 8(c) and (d). Note that the distribution of the  $\hat{\alpha}^*$  in Fig. 8(c) is more packed toward the origin than the distribution of the  $\alpha^*$  in Fig. 8(a). This indicates that, in general, we have  $|\alpha^*| \geq |\hat{\alpha}^*|$ , which is consistent with (19).

### C. Prediction of the Optimal Parent–Child Matching

The results of the Section IV-B suggest a simple algorithm to fractally denoise an image. First, estimate the variance  $\sigma_{\mathcal{N}}^2$  of the noise in the image. Then, fractally encode the noisy image to obtain the noisy gray-level coefficients  $(\hat{\alpha}^*, \hat{\beta}^*)$ . Use (19) to estimate the noise-free gray-level coefficients  $(\alpha^*, \beta^*)$ . There is one problem, however: It is not guaranteed that the parent–child assignments of the noisy image  $\hat{\mathcal{I}}$  are optimal for the noiseless image  $\mathcal{I}$  in the  $\mathcal{L}^2$  sense, i.e., that the collage distance is minimized. We now propose a method that estimates an optimal collage coding procedure over the noiseless image. As before, we use the  $X, Y$  random variables to denote the parent and (transformed) child blocks. ( $\hat{X}, \hat{Y}$  for the noisy image.)

The method is as follows. For each child block  $R_k \in \mathcal{R}$ ,  $1 \leq k \leq N_R$ , of the noisy image, we perform the following steps.

- 1) For each parent block  $D_i \in \mathcal{D}_k$ , the domain pool of  $R_k$ , consider all possible geometric maps  $w_{ik}^{(m)}$  and compute the least-squares noisy gray-level gain coefficient  $\hat{\alpha}_{ik}^*$ . Then use this coefficient, the estimated noise variance and the estimated statistics from the noisy image to compute the noise-free gray-level coefficients  $(\alpha_{ik}^*, \beta_{ik}^*)$  using (19) as follows:

$$\begin{aligned} \alpha^* &= \left(1 + \frac{1}{4\gamma}\right) \hat{\alpha}^* \\ \beta^* &= E[\hat{Y}] - \alpha^* E[\hat{X}]. \end{aligned} \quad (20)$$

- 2) For the fractal of the original noise-free image, we wish to minimize the collage error, measured in terms of the *mean squared error* of the noiseless image as

$$\Delta_{ik}^2 = E[(Y_k - (\alpha_{ik}^* X_i + \beta_{ik}^*))^2 | (\alpha_{ik}^*, \beta_{ik}^*)].$$

Expanding the above quantity yields

$$\begin{aligned} \Delta_{ik}^2 &= E[Y_k^2] + \alpha_{ik}^{*2} E[X_i^2] - 2\alpha_{ik}^* E[X_i Y_k] \\ &\quad - 2\beta_{ik}^* E[Y_k] + 2\alpha_{ik}^* \beta_{ik}^* E[X_i] + \beta_{ik}^{*2} \end{aligned}$$

which can be expressed in terms of the statistics of the noisy image as

$$\begin{aligned} \Delta_{ik}^2 &= \left(E[\hat{Y}_k^2] - \sigma_{\mathcal{N}}^2\right) + \alpha_{ik}^{*2} \left(E[\hat{X}_i^2] - \frac{\sigma_{\mathcal{N}}^2}{4}\right) \\ &\quad - 2\alpha_{ik}^* E[\hat{X}_i \hat{Y}_k] - 2\beta_{ik}^* E[\hat{Y}_k] + 2\alpha_{ik}^* \beta_{ik}^* E[\hat{X}_i] + \beta_{ik}^{*2}. \end{aligned}$$

As a result, the collage error for the noiseless image is estimated from the noisy image statistics.

- 3) Select the optimal parent block  $i(k)$  and associated geometric map  $w_{i(k),k}^{m(k)}$  such that

$$\Delta_{i(k),k}^2 \leq \Delta_{j,k}^2 \text{ for all } j \neq i(k).$$

The result is an estimated collage-based matching criterion for the original noise-free image.

We have now estimated the fractal code of the noise-free image—parent–child matchings with associated gray-level parameters—from that of the noisy image. This fractal code is expected to be closer to the code of the noiseless image as the variance of the noise decreases, converging to the latter in the limit  $\sigma_{\mathcal{N}} \rightarrow 0$ .

### D. Quantization of the Gray-Level Coefficients

In order to guarantee the contractivity of the fractal transform with respect of the  $\mathcal{L}_\infty$  for various classes of images, one has to insure that the absolute value of the quantized scaling parameter  $\alpha$  is smaller than unity. In the quantization methods used in this work, the scaling coefficients are restricted to the interval  $[-0.99, 0.99]$ . Also, adding noise to images often results in many pixels of the noisy image having gray-level values outside the appropriate  $[0, 255]$  range for 8-bit gray-scale images. In order to insure that the gray-level values of the fractal approximations to the denoised image lie within this proper range, we devise the following quantization strategy for the offset coefficient  $\beta$ . For a quantized scaling coefficient  $\alpha_q$  and offset coefficient  $\beta_q$ , we require that

$$0 \leq \alpha_q x + \beta_q \leq 255 \quad (21)$$

for each pixel value  $x$  in the parent block. There are two cases.

- 1) If  $\alpha_q \geq 0$

$$\begin{aligned} 0 \leq x \leq 255 &\Rightarrow \beta_q \leq \alpha_q x + \beta_q \leq 255\alpha_q + \beta_q \\ &\Rightarrow 0 \leq \beta_q \leq 255(1 - \alpha_q). \end{aligned}$$

- 2) If  $\alpha_q < 0$

$$\begin{aligned} 0 \leq x \leq 255 &\Rightarrow \beta_q \geq \alpha_q x + \beta_q \geq 255\alpha_q + \beta_q \\ &\Rightarrow -255\alpha_q \leq \beta_q \leq 255. \end{aligned}$$

This guarantees that the fractal representation of a noisy image will lie within the  $[0, 255]$  range provided that the seed image also satisfies this condition. Since we typically use a blank image, i.e.,  $u_{ij} = 0$ , as the seed, the condition is satisfied.

In view of the above derivations, we quantize the offset coefficient  $\beta$  to be within the interval  $[\beta_{\min}, \beta_{\max}]$ , depending on the sign of the scaling coefficient  $\alpha$ , as follows:

$$[\beta_{\min}, \beta_{\max}] = [0, 255(1 - \alpha_q)], \text{ if } \alpha_q \geq 0 \quad (22)$$

$$[\beta_{\min}, \beta_{\max}] = [-255\alpha_q, 255], \text{ if } \alpha_q < 0. \quad (23)$$

This insures that the various fractal representations have pixel values within the suitable  $[0, 255]$  range. Consequently, one can use the standard RMSE and PSNR qualitative measures to assess the quality of these representations and display these representations on any standard 8-bit graphic device.

### E. Experimental Results

We now present some results of the procedure outlined above, as applied to the noisy image of ‘‘Lena.’’

Presented in Fig. 9(a) and (b) are the histograms of the gray-level coefficients obtained by fractally encoding the noisy Lena image. Shown in Fig. 9(c) and (d) are the histograms of the predicted gray-level coefficients for the noise-free image, according to the predictive method described in Section IV-D. We

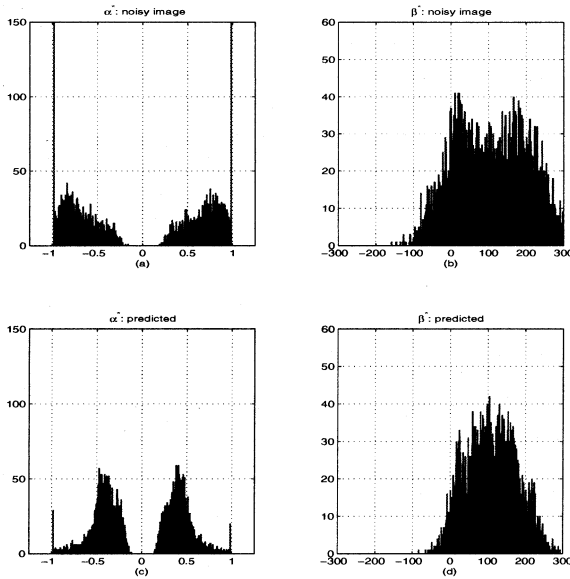


Fig. 9. Histogram plots of the fractal gray-level coefficients,  $\alpha^*$  and  $\beta^*$ , respectively. (a) and (b): Fractal coding of the noisy image. (c) and (d): Fractal coding using the predictive scheme of Section IV-B.

note that the latter set of distributions are closer to the distributions in Fig. 8(a) and (b) of the “true” noiseless gray-level coefficients, especially in the case of the  $\beta$  offset coefficients. We also find that this estimation method correctly predicts a significant fraction of optimal parent–child matchings and their corresponding geometric maps.

In Fig. 10(a) and (b) are shown the results of the fractal restoration scheme using both uniform as well as quadtree partitioning schemes. In these fractal representations, most of the noise appears to have been suppressed without blurring the edges or other high frequency components of the image.

It is interesting to note that, as illustrated in Figs. 4 and 6, the standard uniform-based fractal coding of the noisy image results in further noise reduction than quadtree-based fractal coding of the noisy image. This is because the quadtree-based fractal coding reconstructs most of the noise in the vicinity of edges and other high-frequency content of the image. However, after applying the proposed fractal code prediction method, the quadtree-based fractal denoising scheme results in further noise reduction than the use of the uniform partition fractal scheme. This can be explained by the fact that the quadtree-based fractal denoising scheme performs a content-based denoising where a high degree of denoising is performed in uniform sub-regions of the image and a lower degree of denoising is performed in the vicinity of edges without compromising their sharpness. However, the uniform-based fractal denoising scheme performs a uniform degree of smoothing throughout the image, regardless of its content.

## V. COMPARISON WITH THE LEE FILTER

In this section, we compare the performance of the proposed fractal-based image restoration scheme to a commonly used adaptive image restoration filtering technique known as the Lee filter [16]. We also explore the relationship between the Lee filter and the fractal transform method.

### A. Lee Filter

One of the limitations of conventional linear filtering methods is that they are based on the assumption that the formation image signal is stationary and formed through a linear process. However, such an assumption is generally not valid for most real-world images. It is expected that local characteristics of an image would be more suitable for effective image restoration and enhancement. The Lee filter is such an adaptive local method. It is a local statistics filter that employs local masks whose coefficients are functions of the local signal and noise characteristics [16]. In this approach, an optimal filter estimator for a signal in additive noise is defined as

$$\hat{s}(m, n) = \alpha_{m, n} f(m, n) + \beta_{m, n} \quad (24)$$

for the observation  $f$  given by

$$f(m, n) = s(m, n) + \mathcal{N}(m, n) \quad (25)$$

where  $s$  is the original noise-free image and  $\mathcal{N}$  is an additive white Gaussian noise. Also, the noise is assumed to be a zero mean white Gaussian noise process with variance  $\sigma_{\mathcal{N}}^2$ , independent of the original signal  $s$ .

The parameters  $\alpha_{m, n}$  and  $\beta_{m, n}$  are chosen to minimize the mean squared estimation error criterion

$$\Delta_{m, n}^2 = E [(\alpha_{m, n} f + \beta_{m, n} - s)^2]. \quad (26)$$

These parameters are computed adaptively and locally from the local statistics of a  $7 \times 7$  mask, centered at the pixel  $(m, n)$ . Setting the partial derivatives of  $\Delta_{m, n}^2$  with respect to  $\alpha_{m, n}, \beta_{m, n}$  to zero and solving for  $\alpha_{m, n}, \beta_{m, n}$ , yields the following form of the Lee filter:

$$\begin{aligned} \hat{s}(m, n) &= \frac{\sigma_s^2(m, n)}{\sigma_f^2(m, n)} f(m, n) + \frac{\sigma_{\mathcal{N}}^2}{\sigma_f^2(m, n)} \bar{f}(m, n) \\ &= \frac{\sigma_f^2(m, n) - \sigma_{\mathcal{N}}^2}{\sigma_f^2(m, n)} f(m, n) \\ &\quad + \frac{\sigma_{\mathcal{N}}^2}{\sigma_f^2(m, n)} \bar{f}(m, n). \end{aligned} \quad (27)$$

The quantities  $\bar{f}(m, n)$  and  $\sigma_f^2(m, n)$  denote, respectively, the local sample mean and variance of  $f$ . They are computed locally and are dependent on the location of the center pixel  $(m, n)$ .

### B. Relationship Between the Lee Filter and the Fractal Transform

From (24), the Lee filter is an affine transformation that operates on single pixel values  $f(m, n)$ . On the one hand, it may be viewed as an extreme case of a fractal transform for which the parent of a single-pixel child block is the child block itself. From (27), the gray-level coefficients of this transform are determined from the statistics of a  $7 \times 7$  pixel block that is centered at the child block:

$$\begin{aligned} \alpha_{m, n} &= \frac{\sigma_f^2(m, n) - \sigma_{\mathcal{N}}^2}{\sigma_f^2(m, n)}, \\ \beta_{m, n} &= \frac{\sigma_{\mathcal{N}}^2}{\sigma_f^2(m, n)} \bar{f}(m, n). \end{aligned}$$

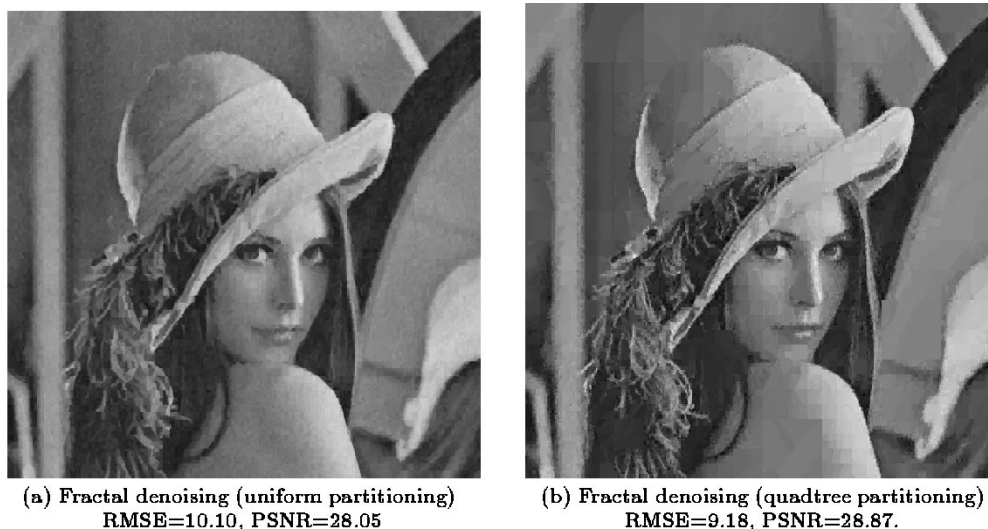


Fig. 10. Fractal restoration of the noisy image of “Lena,” with  $\sigma_{\mathcal{N}}^2 = 625$ , using uniform and quadtree partitioning of the image.

On the other hand, the Lee filter may be viewed as a local affine transformation of a  $7 \times 7$  parent block onto the single-pixel child block that lies at its center. This latter action is somewhat reminiscent of the Bath Fractal Transform discussed in Section II, in which parent blocks are chosen to be co-centric with child blocks.

Clearly, the fractal denoising method proposed above allows for more flexibility in choosing the optimal parent sub-block, however at the cost of having to search for such blocks. These insights into the relationship between the fractal and the Lee filter methods have motivated us to further investigate the theoretical and practical connections between these two techniques.

### C. Experimental Results

In Fig. 11 are presented the results of the fractal-based and Lee filter methods as applied to the noisy Lena image with noise variance  $\sigma_{\mathcal{N}}^2 = 625$ . In the fractal-based scheme, uniform partitioning was used with  $8 \times 8$  pixel child blocks in an effort to provide a fair comparison with the optimal  $7 \times 7$  pixel mask used by the Lee filter.

The fractal coding method is seen to yield better results both quantitatively, in terms of PSNR, as well as qualitatively, in terms of quality of the denoised image. In Fig. 11(c), where the difference between the fractally denoised image and the noiseless image is plotted, we see that the fractal method suppresses noise throughout the entire image without blurring edges.

Another advantage of the fractal method is that it yields a representation of the noisy image with pixel values that lie in the range  $[0, 255]$  because of the quantization imposed upon the gray-level coefficients,  $\alpha^*$  and  $\beta^*$ , as discussed in Section IV-D. The Lee filter representation of the noisy image will not always satisfy this  $[0, 255]$  condition.

In Fig. 11(d), where the error associated with the Lee filter denoised image is plotted, we see that the Lee filter allows much more visible noise, especially in close proximity of the edges. Nevertheless, the edges remain sharp, unlike most other linear filtering and smoothing operations. In smoother regions of the image, e.g., the background, where the local signal variance is mostly due to noise, the Lee filter performs a high degree of smoothing through local

averaging. In summary, the Lee filter is seen to exploit the characteristic features of the human visual system.

### D. Additional Experimental Results

In Figs. 12–15, are presented additional comparisons of the fractal and Lee filter methods using four standard  $512 \times 512$ , 8 bpp test images—Lena, Barbara, Mandrill, and Boat—for the following values of noise variance:  $\sigma^2 = 10^2, 20^2, 30^2, 40^2$ . For larger values of the noise variance, e.g.,  $\sigma = 30, 40$ , the noisy images look distinctly unnatural due to speckle in the smooth regions. However, the level of detail and sharpness of edges remain visually pleasing. We observe that, compared to the fractal method, the Lee filter performs well at low noise variance but its performance decreases rapidly as the noise variance increases. Under heavy noise, the Lee filter leaves visible speckle around edges, becomes blotchy in smooth regions, and looks somewhat blurry or ragged in spots. On the other hand, the fractal method appears to become stable as the quality of the image degrades with increasing noise variance since no sharp differences are noticed as we increase  $\sigma$ . Also, the fractal representations do not seem to become blotchy, perhaps because there is a good deal of mixing among different areas of the image in the fractal reconstruction. Eventually, however, the fractal representations also become blurry and ragged along edges.

We now identify some characteristic regions of each image to see the above effects.

- 1) *Lena*: As the noise variance is increased, the Lee filter shows some speckle near the shadow line on the nose and blotchiness in the background and on the skin, especially the shoulder.
- 2) *Barbara*: The image is busier so the blotchiness of the Lee filter is not as visible. However, for large noise variance, the hand and face look more natural in the fractal reconstructions than in the Lee filter reconstructions.
- 3) *Mandrill*: This image is extremely busy. The most visible artifact in the Lee case is some blurriness in the middle to lower left and right, as well as the nose when the noise variance is large. The fractal reconstruction becomes blurry in the same regions but on a much smaller scale.

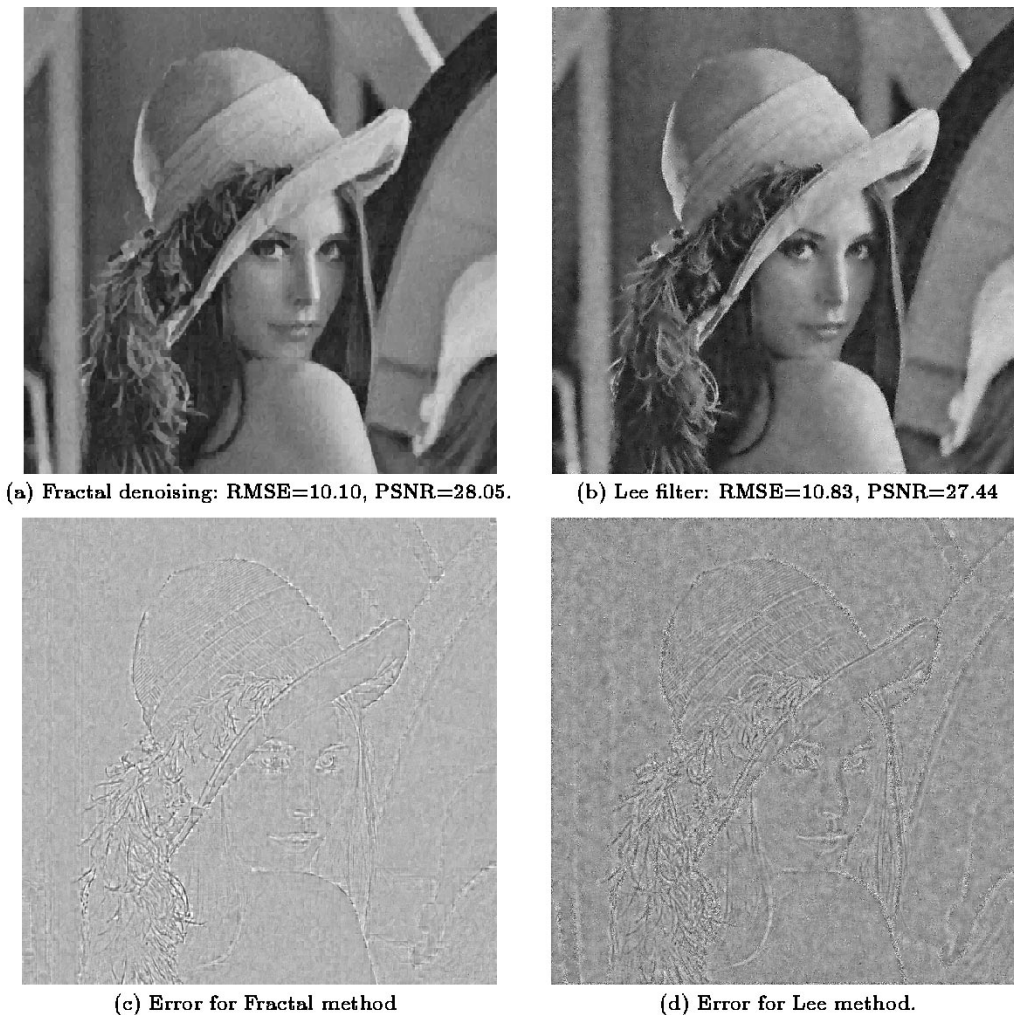


Fig. 11. Comparison between the fractal-based image method and the Lee filter: (a) the fractal-based image denoising approach using uniform partitioning with  $8 \times 8$  mask and (b) the Lee filter, using a  $7 \times 7$  mask. (c) and (d) Illustrate the difference between the fractal and the Lee filter representation and the original noise-free image, respectively.

- 4) *Boat*: This image has a few fine details, e.g., the cables and poles, and a significant background subregion. The remarks made for the other test images apply to this image as well. For light noise the Lee filter performs well and then quickly degrades as the variance becomes large. The fractal method is more stable and consistent even when the noise is rather heavy.

In summary, for most of the test images the Lee filter method performs better for low noise variance but sharply degrades as the noise variance becomes large. For heavy noise ( $\sigma \geq 20$ ), the fractal method performs consistently better than the Lee filter method. A significant factor in this difference is that the fractal representations, unlike those of the Lee filter, are constrained to satisfy the requirement that pixel values lie in the range  $[0, 255]$ . As the noise variance increases, it becomes more difficult for the Lee filter method to satisfy this requirement.

## VI. SUMMARY AND CONCLUDING REMARKS

In this work, we have developed some fractal-based methods of image restoration and assessed their performance. Much interest has been given to the use of fractal block coding schemes

for the purpose of image compression. However, to the best of our knowledge, little or no attention has been given to their possible use in image restoration and enhancement. This work may, in fact, represent the first attempt of its kind.

We began by simply applying the uniform fractal coding scheme to a noisy image. The decimation associated with the contractive spatial maps used in the fractal transform is probably responsible for most of the denoising. It was observed that the standard fractal scheme performs too much smoothing for lower resolutions  $(M, N)$ , producing a blurring of the edges as well as blockiness artifacts. At higher resolutions  $(M, N)$ , an insufficient amount of smoothing is performed, resulting in a noisy fractal representation. This competition between quality and resolution can be resolved by using an adaptive partitioning scheme such as quadtrees, which allows different resolutions/block sizes to be used for different parts of an image. It permits a greater degree of smoothing away from edges (larger blocks) and a lesser degree of smoothing near edges (smaller blocks), hence minimizing their degradation.

A second major result of this paper lies in the estimation of the fractal code of the noise-free image from the fractal code of the noisy image. However, it is not guaranteed that optimal

Noisy image	Fractal restoration	Lee filter restoration
 $\sigma = 10$	 RMSE=8.87, PSNR=29.17	 RMSE=7.59, PSNR=30.52
 $\sigma = 20$	 RMSE=9.58, PSNR=28.50	 RMSE=9.78, PSNR=28.32
 $\sigma = 30$	 RMSE=11.31, PSNR=27.06	 RMSE=11.16, PSNR=27.06
 $\sigma = 40$	 RMSE=12.81, PSNR=25.98	 RMSE=13.31, PSNR=25.64

Fig. 12. Comparison between the fractal (using uniform partitioning) and the lee filter restoration methods for the test image of “Lena.”

parent–child pairings of the noisy image will coincide with optimal parent–child pairings of the noiseless image. In an effort to

predict the latter, thereby producing a better fractal reconstruction of the noiseless image, a collage-based method employing

Noisy image	Fractal restoration	Lee filter restoration
 $\sigma = 10$	 RMSE=12.44, PSNR=26.23	 RMSE=7.82, PSNR=30.26
 $\sigma = 20$	 RMSE=13.31, PSNR=25.64	 RMSE=12.69, PSNR=26.06
 $\sigma = 30$	 RMSE=14.95, PSNR=24.64	 RMSE=16.12, PSNR=23.98
 $\sigma = 40$	 RMSE=15.73, PSNR=24.20	 RMSE=18.82, PSNR=22.64

Fig. 13. Comparison between the fractal (using uniform partitioning) and the lee filter restoration methods for the test image of "Barbara."

the noise statistics was proposed. Experiments indicate that the method works very well, improving upon the other fractal-based

methods and yielding results that, especially in the case of heavy noise, are superior to those obtained by the Lee filter method.

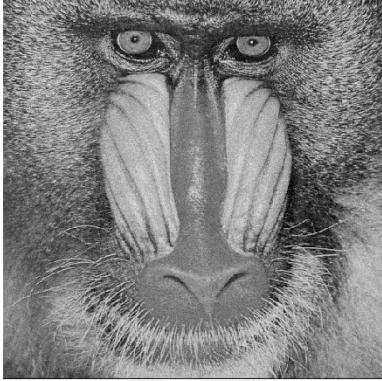
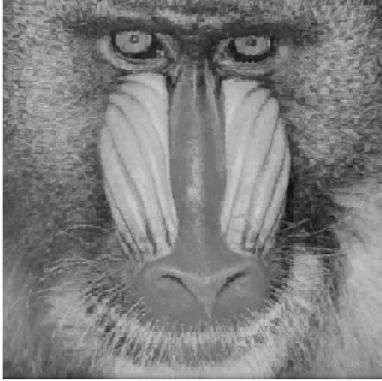
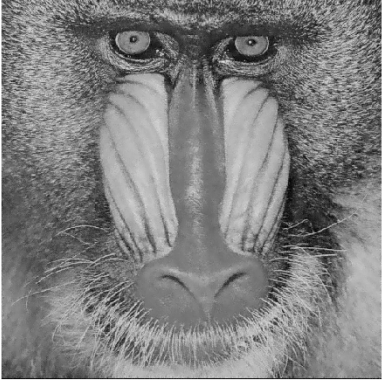
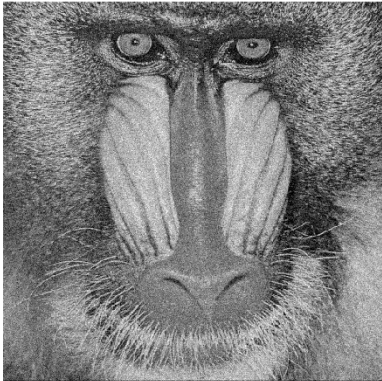
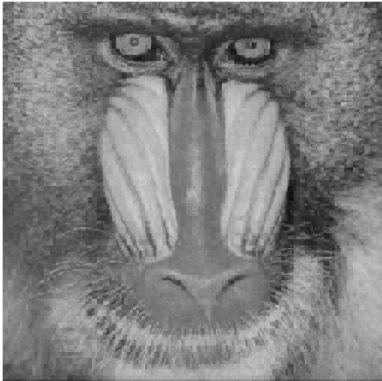
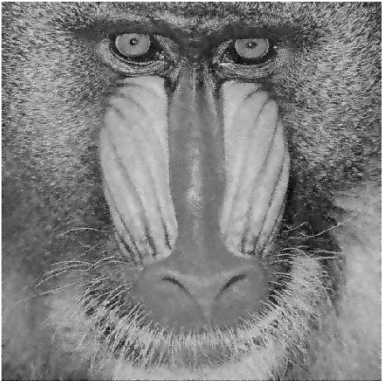
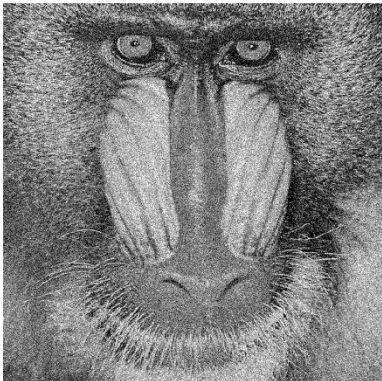
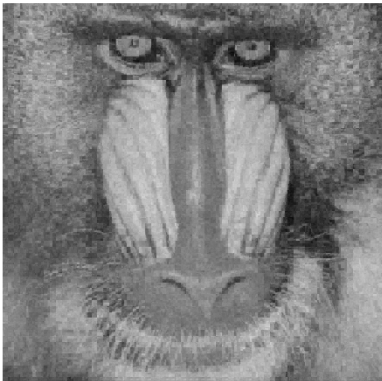
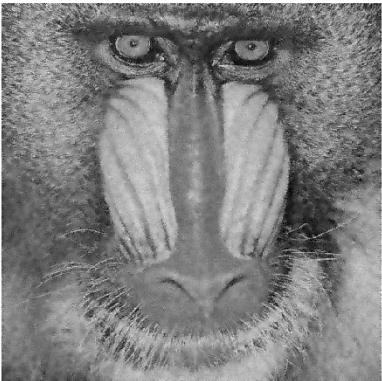
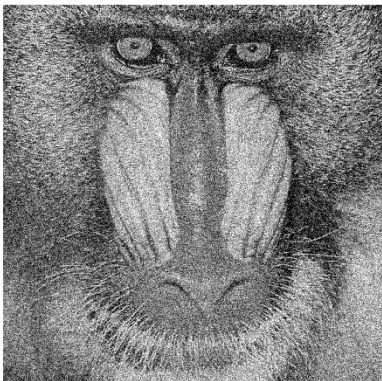
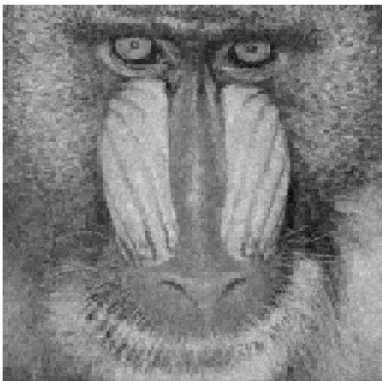
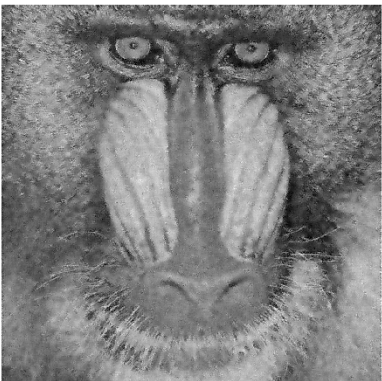
Noisy image	Fractal restoration	Lee filter restoration
 <p><math>\sigma = 10</math></p>	 <p>RMSE=8.32, PSNR=29.72</p>	 <p>RMSE=8.02, PSNR=30.05</p>
 <p><math>\sigma = 20</math></p>	 <p>RMSE=9.55, PSNR=28.53</p>	 <p>RMSE=12.84, PSNR=25.96</p>
 <p><math>\sigma = 30</math></p>	 <p>RMSE=10.35, PSNR=27.83</p>	 <p>RMSE=16.21, PSNR=23.93</p>
 <p><math>\sigma = 40</math></p>	 <p>RMSE=12.36, PSNR=26.36</p>	 <p>RMSE=18.89, PSNR=22.60</p>

Fig. 14. Comparison between the fractal (using uniform partitioning) and the lee filter restoration methods for the test image of “Mandrill.”

Finally, another possible advantage of the proposed fractal denoising scheme over the Lee filter and related methods is that the

former can, if desired, perform compression of the noisy image, unlike the latter. We are currently examining the possibility of













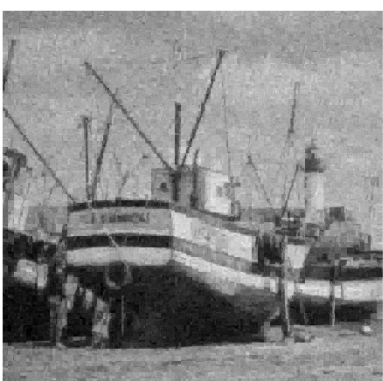

Noisy image	Fractal restoration	Lee filter restoration
 $\sigma = 10$	 RMSE=11.39, PSNR=26.99	 RMSE=7.62, PSNR=30.48
 $\sigma = 20$	 RMSE=12.28, PSNR=26.34	 RMSE=12.06, PSNR=26.50
 $\sigma = 30$	 RMSE=13.66, PSNR=25.41	 RMSE=15.11, PSNR=24.54
 $\sigma = 40$	 RMSE=15.32, PSNR=24.26	 RMSE=17.63, PSNR=23.20

Fig. 15. Comparison between the fractal (using uniform partitioning) and the lee filter restoration methods for the test image of "Boat."

fractal-based schemes that are optimized to achieve both image compression as well as denoising.

We should qualify, however, that other fractal and "multi-fractal" methods [7] have been shown to be successful in image

processing applications, including denoising as well as segmentation, texture analysis, approximation and compression. Over the past decade, much of this work has been done by Lévy Véhel and coworkers at INRIA, Rocquencourt, France [12], [17]. The denoising methods developed by this group rely on multifractal methods and more sophisticated methods of analysis (e.g., “2-microlocalization”) that are deeply rooted in wavelet theory.

There are also connections between fractal coding and wavelet theory. One may define a fractal-type operation of scaling and copying of wavelet coefficient subtrees onto lower subtrees—the resulting *fractal-wavelet transform* induces fractal transform type operations on the image function ([6], [10], [15], to name but a few references). The fractal-based method devised in this paper has a fractal-wavelet analogue. The extraction of noiseless gray-level coefficients from their noisy counterparts can be performed in fractal-wavelet coding and the resulting smoothing can be analyzed in terms of wavelet theory. Some preliminary results have already been presented [11]. A more detailed discussion is planned for a future paper.

APPENDIX

CONTRACTION MAPPINGS AND INVERSE PROBLEMS

Fractal-based methods of image approximation and compression are centered around Banach’s celebrated fixed point theorem [1] which we review below. We then introduce a class of inverse problems involving contraction mappings and finally present a simple consequence of Banach’s theorem that yields a reformulation of the inverse problem suitable for practical applications.

In what follows, we let  $(Y, d_Y)$  denote a complete metric space: In our applications,  $Y$  will typically be an appropriate space of functions that are used to represent images, for example,  $\mathcal{L}^2(X)$  where  $X = [0, 1]^2$ , the support of the images (the “computer screen”).

A mapping  $T : Y \rightarrow Y$  is said to be contractive if there exists a  $c \in [0, 1)$  such that

$$d(T(y_1), T(y_2)) \leq cd_Y(y_1, y_2), \quad \forall y_1, y_2 \in Y.$$

The smallest such  $c \geq 0$  is called the *contractivity factor* of  $T$ . If  $T$  is contractive then, by Banach’s Theorem.

- 1) There exists a unique  $\bar{y} \in Y$  such that  $T\bar{y} = \bar{y}$ , i.e., a unique fixed point.
- 2) Moreover, the fixed point  $\bar{y}$  is globally attractive: For any  $y \in Y$ , the sequence  $\{T^n(y)\}$  converges to  $\bar{y}$  in the  $d_Y$  metric.

Fractal transforms—see Section I—are operators that take subsets of an image, modify their gray-scale values and then map these modified subimages onto smaller regions. Under suitable conditions on the fractal parameters that define them, the fractal transforms are contractive. The goal of fractal image compression is then to approximate an image  $y$  with the fixed point  $\bar{y}$  of a contractive fractal transform  $T$ . It is  $T$ , as opposed to its fixed point  $\bar{y}$ , that is stored since the fractal parameters that define it often require much less memory than the original image  $y$  as well as the approximation  $\bar{y}$ . The latter can be generated by the iteration process  $y_{n+1} = Ty_n$ , for any “seed” image  $y_0$ .

Fractal image compression can be viewed as a special case of a more general class of problems concerned with the approximation of elements of a set by fixed points of contractive operators:

Given a “target”  $y \in Y$ , find a contraction map  $T : Y \rightarrow Y$  with fixed point  $\bar{y}$  such that  $d_Y(y, \bar{y})$  is small.

The question now remains, “How do we find such suitable fixed point approximations?” A naive method would be to select a map  $T$ , compute its fixed point  $\bar{y}$  by iteration and finally compute the distance  $d_Y(y, \bar{y})$ . By some kind of systematic variation of the fractal parameters defining  $T$  (e.g., simplex algorithm), one might be able to zero in on optimal fractal parameters that minimized the approximation error  $d_Y(y, \bar{y})$ . Needless to say, this is a computationally expensive procedure.

A great simplification is achieved thanks to a quite simple yet important consequence of Banach’s theorem. Given a “target”  $y \in Y$  and a contraction map  $T$  with fixed point  $\bar{y}$ , note that the approximation error  $d_Y(y, \bar{y})$  is bounded by the triangle inequality as follows:

$$d_Y(y, \bar{y}) \leq d_Y(y, Ty) + d_Y(Ty, \bar{y}).$$

But

$$\begin{aligned} d_Y(Ty, \bar{y}) &= d_Y(Ty, T\bar{y}) \\ &\leq cd_Y(y, \bar{y}) \end{aligned}$$

due to the contractivity of  $T$ . Substitution and rearrangement into the earlier inequality yields the important result

$$d_Y(y, \bar{y}) \leq \frac{1}{1-c} d_Y(y, Ty).$$

In the modern fractal compression literature, this result has been referred to as the *Collage Theorem* [4]. In other words, if one can find a contraction map  $T$  that maps the target  $\bar{y}$  close to itself, then the fixed point  $\bar{y}$  of  $T$  will closely approximate  $y$ . In this reformulated inverse problem, the object is to find a map  $T$  that minimizes the so-called *collage distance*  $d_Y(y, Ty)$ . This procedure is often referred to as *collage coding*.

It is not guaranteed that collage coding will yield the optimal fixed point approximation  $\bar{y}$  to a target  $y$ . Nevertheless, collage coding is almost universally employed because of its relative simplicity. In fact, Hartenstein and Ruhl [24] have shown that optimal fractal coding—trying to minimize the error  $d_Y(y, \bar{y})$  with fractal transforms—is an NP-hard problem.

REFERENCES

- [1] S. Banach, “Sur les opérations dans les ensembles abstraits et leur applications aux équations intégrales,” *Fund. Math.*, vol. 3, pp. 133–181, 1922.
- [2] M. F. Barnsley, *Fractals Everywhere*. New York: Academic, 1988.
- [3] M. F. Barnsley and S. Demko, “Iterated function systems and the global construction of fractals,” *Proc. R. Soc. Lond.*, vol. A399, pp. 243–275, 1985.
- [4] M. F. Barnsley, V. Ervin, D. Hardin, and J. Lancaster, “Solution of an inverse problem for fractals and other sets,” *Proc. Nat. Acad. Sci. USA*, vol. 83, pp. 1975–1977, 1985.
- [5] M. F. Barnsley and L. P. Hurd, *Fractal Image Compression*. Wellesley, MA: A. K. Peters, 1993.
- [6] G. Davis, “A wavelet-based analysis fractal image compression,” *IEEE Trans. Image Processing*, vol. 7, pp. 141–154, 1998.
- [7] C. J. G. Evertsz and B. B. Mandelbrot, “Multifractal measures,” in *Chaos and Fractals: New Frontiers of Science*, H.-O. Peitgen, H. J. Jürgen, and D. Saupe, Eds. New York: Springer-Verlag, 1994.

- [8] Y. Fisher, Ed., *Fractal Image Compression, Theory and Application*. New York: Springer-Verlag, 1995.
- [9] Y. Fisher, Ed., *Fractal Image Encoding and Analysis*. ser. NATO ASI Series F 159. New York: Springer-Verlag, 1998.
- [10] B. Forte and E. R. Vrscay, "Theory of generalized fractal transforms," in *Fractal Image Encoding and Analysis*. ser. NATO ASI Series F 159, Y. Fisher, Ed. New York: Springer-Verlag, 1998.
- [11] M. Ghazel, G. Freeman, and E. R. Vrscay, "Fractal-wavelet image denoising," in *Proc. ICIP 2002*, Rochester, NY, Sept. 2002.
- [12] B. Guiheneuf and J. Lévy Véhel, *2-Microlocal Analysis and Applications in Signal Processing*: INRIA Rocquencourt, 1997.
- [13] *IEEE Trans. Image Processing*, vol. 5, no. 6, June 1996.
- [14] A. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Trans. Image Processing*, vol. 1, pp. 18–30, 1992.
- [15] H. Krupnik, D. Malah, and E. Karnin, "Fractal representation of images via the discrete wavelet transform," in *Proc. IEEE Conv. EE*, Tel-Aviv, Israel, Mar. 7–9, 1995.
- [16] J. S. Lee, "Digital image enhancement and noise filtering by use of local statistics," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-2, no. 2, pp. 165–168, 1980.
- [17] J. Lévy Véhel, "Introduction to the multifractal analysis of images," in *Fractal Image Encoding and Analysis*. ser. NATO ASI Series F 159, Y. Fisher, Ed. New York: Springer-Verlag, 1998.
- [18] J. Lévy Véhel and B. Guiheneuf, *Multifractal Image Denoising*: INRIA Rocquencourt, 1997.
- [19] N. Lu, *Fractal Imaging*. New York: Academic, 1997.
- [20] D. M. Monro, "A hybrid fractal transform," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, vol. 5, 1993, pp. 162–172.
- [21] D. M. Monro and F. Dudbridge, "Fractal block coding of images," *Electron Lett.*, vol. 28, pp. 1053–1054, 1992.
- [22] R. Pinter, Ed., *Nonlinear Vision*. Boca Raton, FL: CRC, 1992.
- [23] I. Pitas and A. N. Venetsanopoulos, *Nonlinear Digital Filters*. Norwell, MA: Kluwer, 1990.
- [24] M. Ruhl and H. Hartenstein, "Optimal fractal coding is np-hard," in *Proc. IEEE Data Compression Conference*, J. Storer and M. Cohn, Eds., Snowbird, UT, 1997.
- [25] S. J. Woolley and D. M. Monro, "Rate distortion performance of fractal transforms for image compression," *Fractals*, vol. 2, pp. 395–398, 1994.

**Mohsen Ghazel** received the B.Sc. degree in engineering mathematics (1992) from the University of Arizona, Little Rock, and the M.Math. (1994) and M.A.Sc. (1999) degrees from the University of Waterloo, Waterloo, ON, Canada. He is currently pursuing the Ph.D. degree in the Department of Electrical and Computer Engineering at the University of Waterloo.

His main research topics are fractal and wavelet image compression and denoising.

**George H. Freeman** received the B.A.Sc. and Ph.D. degrees in electrical engineering from the University of Waterloo, Waterloo, ON, Canada, in 1979 and 1984, respectively.

He then joined the Continuous Speech Recognition Group at IBM T. J. Watson Research Center, Yorktown Heights, NY, where he worked on statistical language modeling and array signal-processing algorithms for speech recognition. In 1985, he returned to the University of Waterloo where he is currently an Associate Professor and Associate Chair for Undergraduate Studies in the Department of Electrical and Computer Engineering. A one-year sabbatical in 1993-1994 was spent working with NCR Canada Ltd., Waterloo, exploring fractal and wavelet methods for document image compression. Past research interests have also included lossless data compression, trellis quantization, and resource management for multimedia wideband CDMA systems. He is now interested in wavelet analysis for biomedical image processing, particularly transrectal ultrasound images of the prostate gland, and articulatory models for speech recognition.

**Edward R. Vrscay** received a B.Sc. degree in chemistry (1975) and the M. Math. (1977) and Ph.D. (1984) degrees in applied mathematics from the University of Waterloo, Waterloo, ON, Canada.

From 1984 to 1986, he was an NSERC Postdoctoral Fellow and Visiting Assistant Professor at the School of Mathematics, Georgia Institute of Technology, Atlanta. In 1986, he returned to the Department of Applied Mathematics, University of Waterloo, as an NSERC University Fellow. His original research interests were in mathematical physics, in particular, quantum mechanics. At the Georgia Institute of Technology, he became involved with dynamical systems, fractal geometry and fractal image coding. He has worked on the mathematical formulation and solution of inverse problems for fractal and fractal-wavelet approximation and is now interested in other problems of mathematical imaging. He is co-editor (with M. Barnsley and D. Saupe) of the book *Fractals in Multimedia*, a collection of invited papers at a workshop sponsored by the Institute of Mathematics and its Applications (IMA), University of Minnesota, Minneapolis, in January 2001 as part of the IMA's Mathematics in Multimedia Program, 2000-2001.