



Fractal Image Compression using Hurst Edge Operator

S.Sakthivel¹, Dr.N.Malmurugan²

Assistant professor, Dept. of Computer Applications, Annai College of Arts & Science, Kumbakonam, Thanjavur,
Tamil Nadu, India¹

Director, Mahendra Group of Institutions, Thiruchencode, Namakkal, Tamil Nadu, India²

ABSTRACT: The demand for images, video sequences and computer animations has increased drastically over the years. This has resulted in image and video compression becoming an important issue in reducing the cost of data storage and transmission. JPEG is currently the accepted industry standard for still image compression, but alternative methods are also being explored. Fractal Image Compression is one of them. This scheme works by partitioning an image into blocks and using Contractive Mapping to map range blocks to domains.

Keywords: Hurst Edge, Compression, Segmentation

I. INTRODUCTION

Fractal encoding is a mathematical process used to encode bitmaps containing a real-world image as a set of mathematical data that describes the fractal properties of the image. Fractal encoding relies on the fact that all natural, and most artificial, objects contain redundant information in the form of similar, repeating patterns called fractals.

The encoding step in fractal image compression has high computational complexity. In this paper I briefly introduce about the Fractal image compression in chapter 1 and the various image compression techniques in chapter 2. Finally the proposed image compression techniques “**Fractal Image Compression Using Hurst Edge Operator**”.

II. FRACTAL BASICS

A fractal is a structure that is made up of similar forms and patterns that occur in many different sizes. The term fractal was first used by Benoit Mandelbrot to describe repeating patterns that he observed occurring in many different structures. These patterns appeared nearly identical in form at any size and occurred naturally in all things. Mandelbrot also discovered that these fractals could be described in mathematical terms and could be created using very small and finite algorithms and data.

Let's look at a real-world example. If you look at the surface of an object such as the floor currently beneath your feet, you will notice that there are many repeating patterns in its texture. The floor's surface may be wood, concrete, tile, carpet, or even dirt, but it still contains repeating patterns ranging in size from very small to very large.

III. RELATED RESEARCHES

A. *The Hurst component and rescale analysis*

Science is often confronted with puzzles which stay unsolved for long periods of time, one of these being the Hurst effect. The effect has been initially observed in sequences of records in time, $x(t)$, of natural phenomena such as river discharges, which were expected to be independent over long periods of time. Hurst, in his analysis, first transformed the natural records in time into a new variable $X(t, N)$, the so-called accumulated departure of the natural record in time in a given year t ($t = 1, 2, \dots, N$), from the average, $\bar{x}(t)$, over a period of N years. The transformation follows the formula

$$X(t, N) = \sum_{i=1}^t (x_i - \bar{x}_N) = \{\sum_{i=1}^t x_i\} - t\bar{x}_N \quad [2.1]$$

Then, he introduced the rescaled range R/S , in which the range $R(N)$ is defined by and the standard deviation $S(N)$ by

$$R(N) = \max_{1 \leq t \leq N} X(t, N) - \min_{1 \leq t \leq N} X(t, N) \quad [2.2]$$

$$S(N) = \left\{ \frac{1}{N} \sum_{t=1}^N [x_t - \bar{x}_N]^2 \right\}^{1/2} \quad [2.3]$$

With the use of the dimensionless rescaled range, Hurst was able to compare natural phenomena of various kinds. He found, then, that the natural phenomena he studied followed the empirical law

$$R/S = (\alpha H)^N \quad [2.4]$$

Where the Hurst exponent H — called K by Hurst — was more or less symmetrically distributed about a mean 0.73 with a standard deviation (SD) of about 0.09. Although this value was slightly overestimated due to the small number of available data for certain phenomena, Hurst was able to reproduce the relation (2.4) by using simulations of biased random events. He derived the relation (2.5) below, which the rescaled range parameter should follow for truly unbiased events, and was able to show that his own simulations of unbiased random events obeyed it:

$$R/S = (\pi N/2)^{0.5} \quad [2.5]$$

This relation was also derived and discussed by Feller (1951) and later confirmed by Feder (1988) by a computer simulation of random events. Hurst's simulations of independent random processes (1951) were done by tossing $n (=10)$ coins $N (=1000)$ times and taking the random variable x to be the number of heads minus the number of tails.

Feder has, however, shown that the position in time, $x(t)$, of a particle that walks at random with steps of unit length on a line becomes asymptotically an ordinary Brownian motion with a Gaussian distribution of step lengths for time scales much larger than the time between steps and for distances much larger than the unit step length. The variable $X(t, N)$ of Equation (2.1) represents, in fact, the position of the particle which makes steps $x(t)$ away from the origin. Mandelbrot later introduced a generalized form of the Brownian motion model, the fractional Brownian motion, fBm, to model the Hurst effect. In the fBm model the Hurst exponent, as shown in relation (2.3), is a real number in the range $0 < H < 1$. There are three types of generalized fBm: (a) The persistent, for values of H in the range $0.5 < H < 1$, (b) the anti-persistent, for $0 < H < 0.5$, and (c) the case $H = 0.5$ which corresponds to the independent white noise processes of ordinary Brownian motion.

The correlation between past and future increments, C , of the accumulated Departure from the mean was estimated by Feder as

$$C = 2^{2H-1} - 1 \quad [2.6]$$

which is notably independent of time. The correlation C takes positive values for persistent fBm, negative values for anti-persistent fBm and is zero for independent random events. Since the correlation C is independent of the lag time t in time series, or the number of data N when in a computer simulation of random events, the fBm predicts an infinitely long correlation in persistent and anti-persistent fBm's.

B. Fractal dimension and Hurst exponent:

Fractal dimension (FD) has been used to characterize data texture in a large number of fields. FD separates important classes of images and characterizes information which is not characterized by other texture features. A variety of procedures have been proposed for estimating the FD of images. These measurements are frequently referred to as dimension type- e.g., Cover Dimension (CD), Box-Counting Dimension (BCD), Hausdorff-Besicovitch Dimension (HD), Wavelet based FD, and etc. We will refer to these procedures as FD-estimators. In this section, we discuss the general concept of Rescaled Range (R/S) analysis for calculating the Hurst exponent.

IV. METHODOLOGY

Rescaled Range analysis is a simple process that is highly data intensive. Here are the sequential steps 1. Start with the whole observed data set that covers and calculate the mean $\bar{A}=(1/N)\sum a_i$ 2. Next, sum the differences from the mean to get the cumulative total at each time point X_{ka} from the beginning of the period up to any time: $X_{km}=\sum (a_i - \bar{A})$, $k=1,2,3,\dots,n$. 3. Calculate the range $R(\tau)=\max(X_{ka})-\min(X_{ka})$ for $k=1,2,3,\dots,n$. 4. Calculate the standard deviation S , of the values, a_i of the observation over the period m , for which the local mean is $\bar{A}[(1/N)\sum (a_i - \bar{A})^2]^{0.5}$ 5. Calculate $R/S=R(\tau)/S(\tau)$. 6. For the next stage, partition the time interval in to two blocks of size $N/2=\tau$ and repeat the entire procedure , steps 1-5, and determined R/S for each segment of the data set of length $N/2$, then take averaged value. Repeat, using successively shorter τ 's at each stage dividing the data set into non-overlapping segments and finding the mean R/S of these segments. 7. Plot the log-log plot, that is fit Linear Regression Y on X where $Y=\log (R/S)$ and $X=\log N$. The exponent H is the slope of the regression line.

A. Example

As an example, R/S analysis has been applied to the signal shown in Figure1. The $\log\text{-}\log$ plot is shown in Figure 1. The signal produces an H value of 0.3519. Because the Hurst value is different from $H=0.50$, we say that signal exhibits the Hurst phenomena of antipersistence. This means that if the signal had been up in the previous period, it is more likely that it will be down in the next period and vice versa.

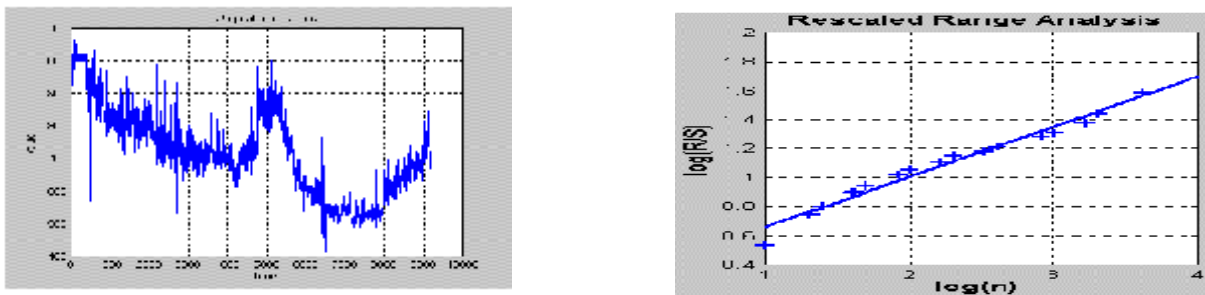


Figure 1

B. Quincunx Neighborhood Q

Multifractal analysis is a new and promising approach to texture classification and image segmentation. In this method, an image I is segmented into a finite set of parts P_1, P_2, \dots, P_s which have different parameter FD such that $I=\cup P_i$, $P_i \cap P_j = \Phi$, $i \neq j$. One of the main problems is finding the local FD. To determine such a parameter FD, it is necessary to apply a mask at each pixel. We use a special mask called Quincunx neighborhood to compute the local Hurst exponent.

C. **Definition:** A quincunx neighborhood is a set of the form: $(1/2^n)M^{-n}([0,1]^2)$ where $M=[a,b]$: $a=[1 \ 1]^{-1}$, $b=[1,-1]^{-1}$ is the quincunx matrix and for all odd values of n .

We represent the nodes by a distance vector \mathbf{d} whose length is equal to the number of different distances from its origin. For example, the distance vector for a Quincunx neighborhood of size 4 to one decimal place approximation is $\mathbf{d}=[1,1.4,2,2.2,2.8,3,3,2,4]$. The Quincunx neighborhoods of size 5 and 4 are shown in figure 2.

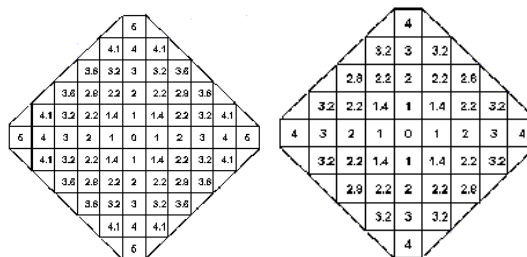


Figure 2

D. Using Hurst Edge to Compress Fractal Image:

Over the last two decades, a wide variety of image compression techniques have been proposed [13,30,31]. The standard method uses the DCT, eliminating insignificant Fourier coefficients and storing larger ones[50]. Another method, called vector quantization, uses a building block approach, breaking up images into a small number of canonical pieces and storing only a reference to where each piece did go. Fractal image compression is a method that was introduced by M.F.Barnsley[1] and A. Jacquin[2]. Its operation is based on Iterated Functions Systems and the Collage Theorem.

E. Random Walks and Long Memory

A one dimensional random walk can be generated by starting at zero and selecting a Gaussian random number. In the next step (in this case 1), add the Gaussian random number to the previous value (0). Then select another Gaussian random number and in the next time step (2) add it to the previous position, etc...

step	value
0	0
1	$0 + R_0$
1	$R_0 + R_1$
2	$R_0 + R_1 + R_2$

Here R_n is a Gaussian random number. Random walks are sometimes referred to as Brownian motion or Gaussian noise. Figure 3.2 shows a plot of this kind of simple random walk.

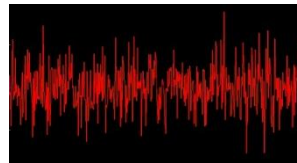


Figure 3.2, A Random Walk

Estimating the Hurst exponent for a data set provides a measure of whether the data is a pure random walk or has underlying trends. Another way to state this is that a random process with an underlying trend has some degree of autocorrelation. When the autocorrelation has a very long (or mathematically infinite) decay this kind of Gaussian process is sometimes referred to as a *long memory process*.

V. THE HURST EXPONENT AND FRACTIONAL BROWNIAN MOTION

Brownian walks can be generated from a defined Hurst exponent. If the Hurst exponent is $0.5 < H < 1.0$, the random walk will be a long memory process. Data sets like this are sometimes referred to as fractional Brownian motion (abbreviated fBm). Fractional Brownian motion can be generated by a variety of methods, including spectral synthesis using either the Fourier transform or the wavelet transform. Here the spectral density is proportional to Equation 2 (at least for the Fourier transform):

$$\text{spectral density} \propto \frac{1}{f^\beta} \text{ where } \beta = 2H + 1 \quad [3.2]$$

Fractional Brownian motion is sometimes referred to as $1/f$ noise. Since these random walks are generated from Gaussian random variables (sets of numbers), they are also referred to as fractional Gaussian noise (or fGn).

The fractal dimension provides an indication of how rough a surface is. As Equation 2 shows, the fractal dimension is directly related to the Hurst exponent for a statistically self-similar data set. A small Hurst exponent has a higher fractal dimension and a rougher surface. A larger Hurst exponent has a smaller fractional dimension and a smoother surface. This is shown in Figure 3.3.

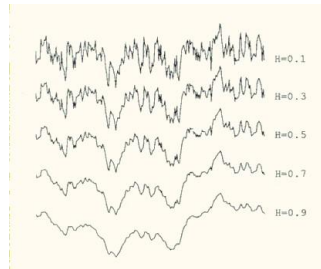


Figure 3.3, Fractional Brownian Motion and the Hurst exponent

A. Estimating the Hurst Exponent from the Rescaled Range

The Hurst exponent is estimated by calculating the average rescaled range over multiple regions of the data. In statistics, the average (mean) of a data set \mathbf{X} is sometimes written as the expected value, $E[X]$. Using this notation, the expected value of R/S , calculated over a set of regions (starting with a region size of 8 or 10) converges on the Hurst exponent power function, shown in Equation 3.3.

$$E \left[\frac{R(n)}{S(n)} \right] = Cn^H \quad \text{as } n \rightarrow \infty \quad [3.3]$$

If the data set is a random walk, the expected value will be described by a power function with an exponent of 0.5.

$$E \left[\frac{R(n)}{S(n)} \right] = Cn^{0.5} \quad \text{as } n \rightarrow \infty \quad [3.4]$$

Short range dependence should have some autocorrelation (indicating some dependence between value x_i and value x_{i+1}). If there is a Hurst exponent of 0.5, it is a random walk and there is no autocorrelation and no dependence between sequential values.

These two values are averaged, resulting in $R/S \text{ ave}_1$. In this case the process continues by dividing each of the previous sections in half and calculating the rescaled range for each new section. The rescaled range values for each section are then averaged. At some point the subdivision stops, since the regions get too small. Usually regions will have at least 8 data points.

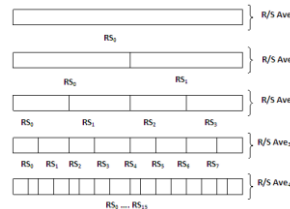


Figure 3.4, Estimating the Hurst Exponent

To estimate the Hurst exponent using the rescaled range algorithm, a vector of points is created, where x_i is the \log_2 of the size of the data region used to calculate RS_{ave_i} and y_i is the \log_2 of the RS_{ave_i} value. This is shown in Table 3.1.

Region size	RS ave.	X_i : $\log_2(\text{region size})$	Y_i : $\log_2(\text{RS ave.})$
1024	96.4451	10.0	6.5916
512	55.7367	9.0	5.8006
256	30.2581	8.0	4.9193
128	20.9820	7.0	4.3911
64	12.6513	6.0	3.6612
32	7.2883	5.0	2.8656
16	4.4608	4.0	2.1573
8.0	2.7399	3.0	1.4541

Table 3.1 : Estimating the hurst component using the rescaled range algorithm.

The Hurst exponent is estimated by a linear regression line through these points. A line has the form $y = a + bx$, where a is the y-intercept and b is the slope of the line. A linear regression line calculated through the points in Table 3.1 results in a y-intercept of -0.7455 and a slope of 0.7270. This is shown in Figure 3.5, below. The slope is the estimate for the Hurst exponent. In this case, the Hurst exponent was calculated for a synthetic data set with a Hurst exponent of 0.72.

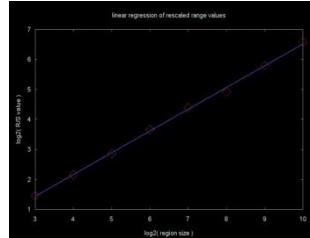


Figure 3.5 : A linear regression line calculated through the points in Table 3.1

Figure 3.6 shows an autocorrelation plot for this data set. The blue line is the approximate power curve.

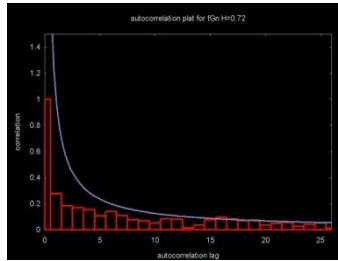


Figure 3.6: An autocorrelation plot for the data set in table 3.1

B. Estimating the Hurst Exponent using Wavelet Spectral Density

The Hurst exponent for a set of data is calculated from the wavelet spectral density, which is sometimes referred to as the scalogram. The wavelet spectral density plot is generated from the wavelet power spectrum. The equation for calculating the normalized power for octave j is shown in Equation 5.

$$P_j = \frac{1}{2^j} \sum_{i=0}^{2^j-1} c_i^2 \quad [3.5]$$

Figure 3.7 shows the normalized power density plot for the wavelet transform of 1024 data points with a Hurst exponent of 0.72. In this case the Daubechies D4 wavelet was used. Since this wavelet octave number is the \log_2 of the number of wavelet coefficients, the x-axis is a log scale.

The Hurst exponent is calculated from the wavelet spectral density by calculating a linear regression line through a set of $\{x_j, y_j\}$ points, where x_j is the octave and y_j is the \log_2 of the normalized power. The slope of this regression line is proportional to the estimate for the Hurst exponent. A regression plot for the Daubechies D4 power spectrum in Figure 3.7 is shown in Figure 3.8.

The equation for calculating the Hurst exponent from the slope of the regression line through the normalized spectral density is shown in Equation 6.

$$H = \left| \frac{(\text{slope}-1)}{2} \right| \quad [3.6]$$

VI. HURST BASED TEXTURE CLASSIFICATION AND SEGMENTATION

Fractal based texture analysis was introduced by Pentland[3], where the relation between natural surface texture and fractal dimension was demonstrated. Fractal based models are useful for image segmentation, texture classification, shape-from texture and the estimation of roughness from image data. A fractal is defined as a set for which the Hausdorff-Besicovitch dimension is strictly greater than the topological dimension [5]. If TD is the topological

dimension, then the fractal dimension FD can be estimated by $H=TD-FD$, where H is the Hurst exponent. This section, develop an efficient method for computing local Hurst exponents to measure the local roughness of an image by using the R/S technique.

The basic idea is to calculate the local Hurst exponents for an image. The local FD is then derived from the value of the Hurst exponent. A small value of FD represents a fine texture, while a large FD corresponds to a coarse texture. Based on this description, segment the image and find the edges with a simple thresholding method. Thresholding is the transformation of an input image I to an output binary image BI as follows:

$$BI(i, j) = 1 \text{ if } I(i, j) \geq T \quad BI(i, j) = 0 \text{ if } I(i, j) < T \quad [3.7]$$

where T is the threshold

A. Hurst Edge Operator

Edges are often used in image analysis for finding region boundaries. The shape of edges in images depends on many parameters: the geometrical and optical properties of the object, the illumination conditions, the point spread function of the image formation process, and the noise level in the image. There are many ways to perform edge detection. However, most may be grouped into two categories, gradient and Laplacian.

The general idea of our method is form the slope image by the Local Hurst algorithm. Then, thresholding is applied to the slope image to form a black and white image. Since thresholding is computationally inexpensive and fast, our method gives fast results from the slope image. Figure 3.13 shows the original image, the slope image and strong edges from the slope image.



Figure 3.8: Original, slope and edge images

First, start with an image to be compressed and then apply a two-dimensional fast wavelet algorithm with respect to an orthogonal or bi-orthogonal wavelet basis. For computational purposes, organize this process at level N as a vector C in the following manner:

$$C=A(N), H(N),V(N),D(N),H(N-1),V(N-1),D(N-1),\dots \\ H(1),V(1),D(1) \quad [3.10]$$

where A, H, V and D are row vectors consisting of Approximation coefficients, Horizontal detail coefficients, Vertical detail coefficients and Diagonal detail coefficients respectively, this information can be obtained by eliminating small wavelet coefficients in C .

B. Experimental Results

In most cases, variations in texture are not highly visible. This suggests that to construct the image with only edge and trend information of the image. The compression ratios obtained in this method are impressive, ranging from 24 to 49 for a 512 by 512 Lena image. The image is visually similar in quality to the original image.

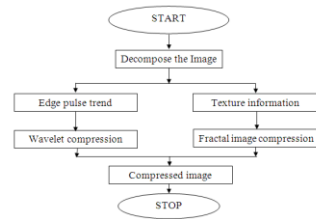


Fig 3.9: Multi Scale Image compression

However, when the original image is subtracted from the one that has been compressed, it becomes apparent that the compression process has altered details. When the compression ratio is increased to 60:1, the appearance of artifacts throughout the image is quite apparent.

Comparing restoration results requires a measure of image quality. Two commonly used measure Root Mean Square Error (RMSE) and Peak Signal-to-Noise Ratio (PSNR). The root mean square error between two images $I(x,y)$ and $CI(x,y)$ of size $M*N$ is

$$RMSE = \left[\frac{1}{MN} \sum_i \sum_j (CI(i,j) - I(i,j))^2 \right]^{0.5} \quad [3.11]$$

$$PSNR = -10 \log_{10}(RMSE/S^2) \quad [3.12]$$

where S is the maximum pixel value. Figure 3.17 shows compression ratio versus RMSE, compression ratio versus PSNR, compression ratio versus bit rate for edge trend method and compression ratio versus bit rate for texture method.

Figure 3.17, ♦, *, ○ denote threshold values 30, 20 and 10 respectively.

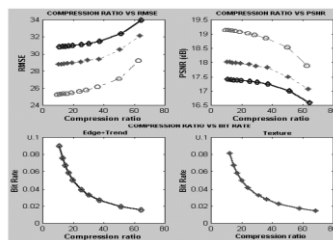


Figure 3.10 : compression ratio

V. CONCLUSION AND FUTURE WORK

In a progressively digital society, the role played by the compression methods is a vital one. The ever-growing need for saving more memory demands various new algorithms for highly compressed images. This paper have attempted to address issues such as a high improvement in the compression ratio while retaining the image quality, through developing image compression method Hurst Edge Operator

VI. SCOPE FOR FUTURE WORK

The following suggestions may be considered for future work in this area. First, higher order interpolation wavelets to measure the trade-off between smoothness of large-area triangle interpolation and computational speed can be developed.

REFERENCES

- [1]. C. M. Kung, W. S. Yang, C. C. Ku, C. Y. Wang, "Fast Fractal Image Compression Base on Block Property," icacte, pp.477-481, 2008 International Conference on Advanced Computer Theory and Engineering, 2008
- [2]. Ramyachitra Duraisamy, L. Valarmathi and Jeyalakshmi Ayyappan, Iteration Free Hybrid Fractal Wavelet Image Coder, International Journal of Computational Cognition (<http://www.ijcc.us>), Vol. 6, No. 4, December 2008
- [3]. A. E. Jacquin. Image Coding Based on a fractal theory of Iterated Contractive Image Transformation. IEEE Transaction on Image processing, 1991,1, 1.



- [4]. M. F. Barnsley and L. Hurd. Fractal Image Compression. A. K. Peters, Wellesley, MA, 1993.
- [5]. M. F. Barnsley. Fractals Everywhere. Academic Press, 1993
- [6]. D. J. Hebert and HyungJun Kim. Image Encoding with Triangulation Wavelets. Proc. SPIE, Wavelet Applications in Signal and Image Processing, 2569, 1995, 381-392.
- [7]. D.J.Hebert and Ezekiel Soundararajan. Fast Fractal Image compression with Triangulation Wavelets. Proc. SPIE Conf. on Wavelet Applications in Signal and Image Processing,1998
- [8] M.F. Barnsley, and S. Demko, Iterated function systems and the global construction of fractals," Proc. Roy. Soc. Lond., vol. A399, pp. 243-275, 1985.
- [9] 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.
- [10] Y. Fisher, Editor, Fractal Image Compression, Theory and Application. New York: Springer-Verlag, 1995.
- [11]Y. Fisher, T. P. Shen, and D. Rogovin, "Comparison of fractal methodwith discrete cosine transform (DCT) and wavelets," Proc. SPIE: Neural Stochast. Methods Image Signal Process. III, S.-S. Chen, Ed., vol. 2304, pp. 132–143, July 1994.