FractalNet: A Neural Network Approach to Fractal Geometry

Dr. Ronald Marsh
Computer Science Department
University of North Dakota
Grand Forks, ND 58201
rmarsh@cs.und.edu
701-777-4013

Abstract

This paper presents a multiply connected neural network designed to estimate the fractal dimension (D*f*) using the Box-counting method (BCM). Fractal analysis is a powerful shape recognition tool and has been applied to many pattern recognition problems. Additionally, the Box-Counting Method is one of the most popular methods for estimating D*f*. However, traditional methods used to estimate D*f* are sequential and have a computational cost of $O(N \log_2(N))$. A parallel method would be more efficient computationally and would suggest a possible biological realization.

The architecture presented separates the calculation of D*f* into two sections, a data sampling section and a linear regression section. The data sampling section provides the ability to dyatically sample the data. The linear regression section simply calculates the slope of the best line through the sampling results. Finally, we show that the network scales well and can be designed to analyze 1-dimensional data or 2-dimensional data.

1. Introduction

This paper presents a multiply connected neural network designed to estimate the fractal dimension (D*f*) using the Box-counting method (BCM). Calculation of D*f* is becoming a common data processing tool, and in general, the faster one can do it, the better. Traditional techniques are sequential. Not only do they not operate on the entire data vector simultaneously, they must revisit the data vector several times, performing the sequential processing each time. Hence, as the vector length increases so does the processing time. Traditional techniques have a computational cost of $O(N \log_2(N))$. A parallel (or neural network) method would be more efficient computationally and would suggest a possible biological realization.

We were led to consider a neural network approach to calculating D*f* by the possibility of the biological visual processing system's use of the Mellin transform for pattern recognition. There is evidence that the visual processing system of cats and some species of primates uses a log-polar transformation[1]. A log-polar transform followed by a Fourier transform is a known method for scale and rotation invariant pattern recognition (a Mellin transform). Therefore, an obvious question to ask is: "Does the biological visual processing system also employ a Fourier transform?" That question has not yet been answered. However, artificial neural networks have been developed which can perform a Fourier transform[2]. The evidence of the biological realization of the log-polar transform and the development of the neural network based Fourier transform, does suggest the possibility that a Mellin transform is used by the visual processing system for pattern recognition.

Given the above, we now ask: "Does the biological visual processing system employ more than one pattern recognition method?" A neural network implementation capable of estimating the fractal dimension does open that possibility.

In this paper, we present a review of fractal analysis in section 2, present our neural network implementation in section 3, and our concluding remarks in section 4.

2. Background

In the 1970s, Beneto Mandelbrot introduced a new field of mathematics to the world. Mandelbrot named this new field fractal (from the Latin term "fractus") geometry[3]. Mandelbrot claimed that fractal geometry would provide a useful tool to explain a variety of naturally occurring phenomena. A fractal is an irregular geometric object with an infinite nesting of structure at all scales (self-similarity). Fractal objects can be found everywhere in nature such as coastlines, fern trees, snowflakes, clouds, mountains, and bacteria. Some of the most important properties of fractals are self-similarity, chaos, and the non-integer D*f*, which gives a quantitative measure of self-similarity and scaling.

Fractal geometry does not concern itself with the explicit shape of objects. Instead, fractal geometry identifies the value that quantifies the shape of the object's surface, the fractal

dimension (D*f*) of the object. For example, a line is commonly thought of as 1-dimensional object, a plane as a 2-dimensional object, and a prism as a 3-dimensional object. All these dimensions have integer values. However, the surfaces of many objects cannot be described with an integer value. These objects are said to have a "fractional" dimension. Consider the following example to illustrate the concept of “fractional” dimension:

Consider a 1-D line.
Bend it.
Bend it some more.
Bend it infinitely many times.
Eventually, the line is transformed into a 2-D plane.

Note that the 1-D line did not instantly become a 2-D plane; there was a transition during which it was 1.xx-D. Essentially, with fractal geometry, the fractal dimension would be described as "1 plus the absolute value of the ratio of the log of the increase in surface to the log of the increase in volume."

The fractal theory developed by Mandelbrot[4] is based, in part, on the work of mathematicians Hausdorff and Besicovitch[5]. The Hausdorff-Besicovitch dimension, D_H, is defined as:

$$D_H = \lim_{\varepsilon \to 0^+} \frac{\ln N_\varepsilon}{\ln 1/\varepsilon} \tag{1}$$

where $N\varepsilon$ is the number of elements of ε diameter required to cover the object. Mandelbrot defines a fractal as a set for which the Hausdorff-Besicovich dimension strictly exceeds the topological dimension. D*f* can be defined as the ratio of the number of self-similar pieces (*N*) with magnification factor (*1/r*) into which a figure may be broken. D*f* is defined as:

$$D_f = \frac{\ln N}{\ln 1/r} \tag{2}$$

D*f* may be a non-integer value, in contrast to objects lying strictly in Euclidean space, which have an integer value. The figures below depict two well-known fractals and their fractal dimensions.

Enlarged: 

N : 4
r : 1/3
D*f* : ln(4)/ln(3) = 1.261859

Figure 1. Koch-curve.

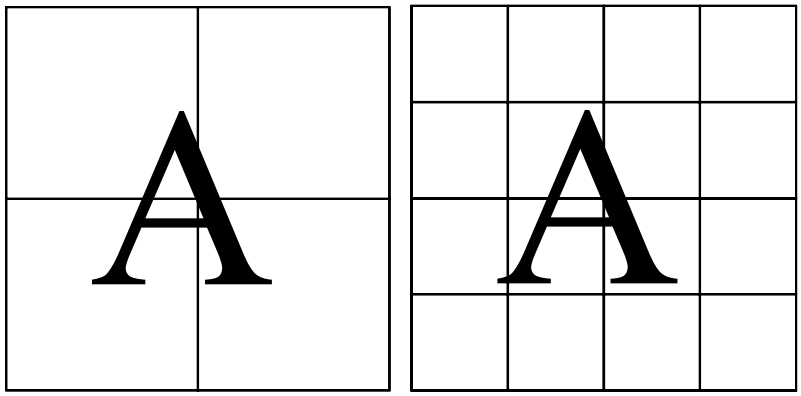
Enlarged:

N : 3
r : 1/2
D*f* : ln(3)/ln(2) = 1.58496

Figure 2. Sierpinski-Gasket.

Fractals are used to characterize, explain, and model complex objects in nature and artificial objects[6] and a considerable number of applications using fractal geometry have been studied. Fractal analysis has been successful in identifying corn roots stressed by nitrogen fertilizer [7], steer body temperature fluctuations in hot and cool chambers[7], measuring textural images[8], and measuring surface roughness[9]. In addition, fractal analysis has proven useful in medicine as medical images typically have a degree of randomness associated with the natural random nature of structure. Osman et al[10]. analyzed D*f* of trabecular bone to evaluate its potential structure. Other studies successfully used D*f* to detect micro-calcifications in mammograms[11], predict osseous changes in ankle fractures[12], diagnose small peripheral lung tumors[12], distinguish breast tumors in digitized mammograms[13], and detect brain tumors in MR images[14]. Studies have also shown that the changes in D*f* reflect alterations of structural properties.

There are a variety of algorithms for estimating D*f*, such as the Box-counting algorithm, the pixel Dilation method, the Calipher method, and the Radial Power Spectrum method. The Box-counting method (BCM)[15] is one of the most popular and simplest. With BCM, we place an arbitrary grid over the structure to be measured. We then count how many boxes in the grid are involved with the structure. The process is then repeated over and over with a grid half the size of the previous (dyatic reduction). For example: in the figure below, in the upper-left image, the letter 'A' involves 4 boxes. In the upper-right image, the letter 'A' involves 5 boxes. In the lower-left image, 12 boxes are involved. In the lower-right image, 35 boxes are involved.



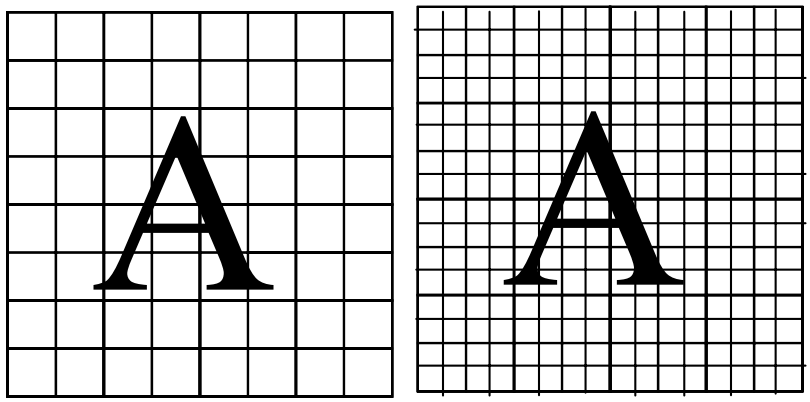


Figure 3. BCM algorithm example.

We tabulate and plot the data on a log-log plot. This is shown in table 1 and figure 4.

Table 1. BCM example data.

1/r	N
2	4
4	5
8	12
16	35

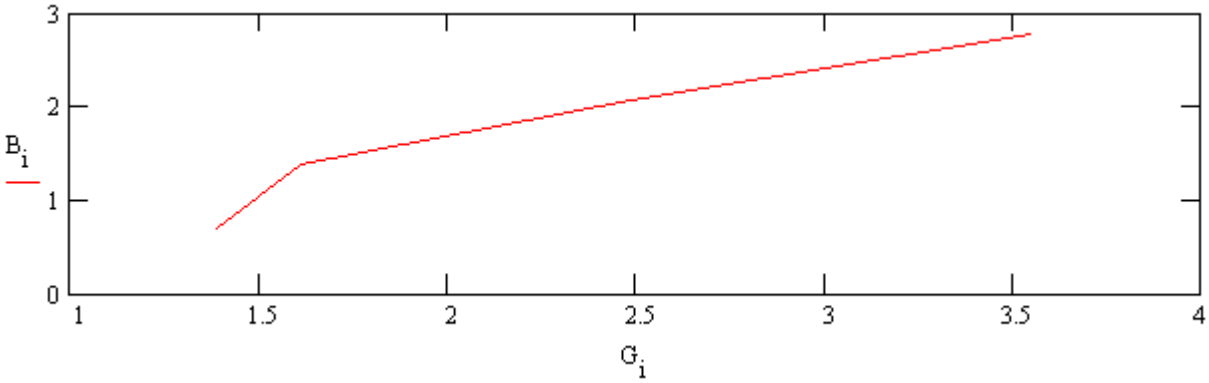


Figure 4. Log-log plot of data.

Linear regression is used to find the best fitting line. D*f* is the slope of this line. Thus, for the letter "A", D*f* is 1.139.

However, we should note that the BCM has its criticisms, especially when applied to digital images[16]. In particular: A digital image has a finite set of points, therefore there is a upper limit (image size) and lower limit (pixel) to ε, and the number of boxes applied (Nε) must be an integer value. Furthermore, there is no standard method for implementing the BCM. In our description of the BCM above, we used a grid size half the size of the previous grid. Other implementations use grid sizes of 3x3 pixels, then 5x5

pixels, then 7x7 pixels, and so on. Hence, the BCM (and most other digitally implemented methods) are only ***estimates*** of D*f* (d_B).

3. Implementation of Fractal Calculating Neural Network (FractalNet)

FractalNet consists of two sections, one section samples the data in a dyatic manner, and the second section performs the linear regression. A FractalNet designed for 8 inputs is shown in figure 5. Note that the two sections are separated by dotted lines. We discuss both sections in more detail below.

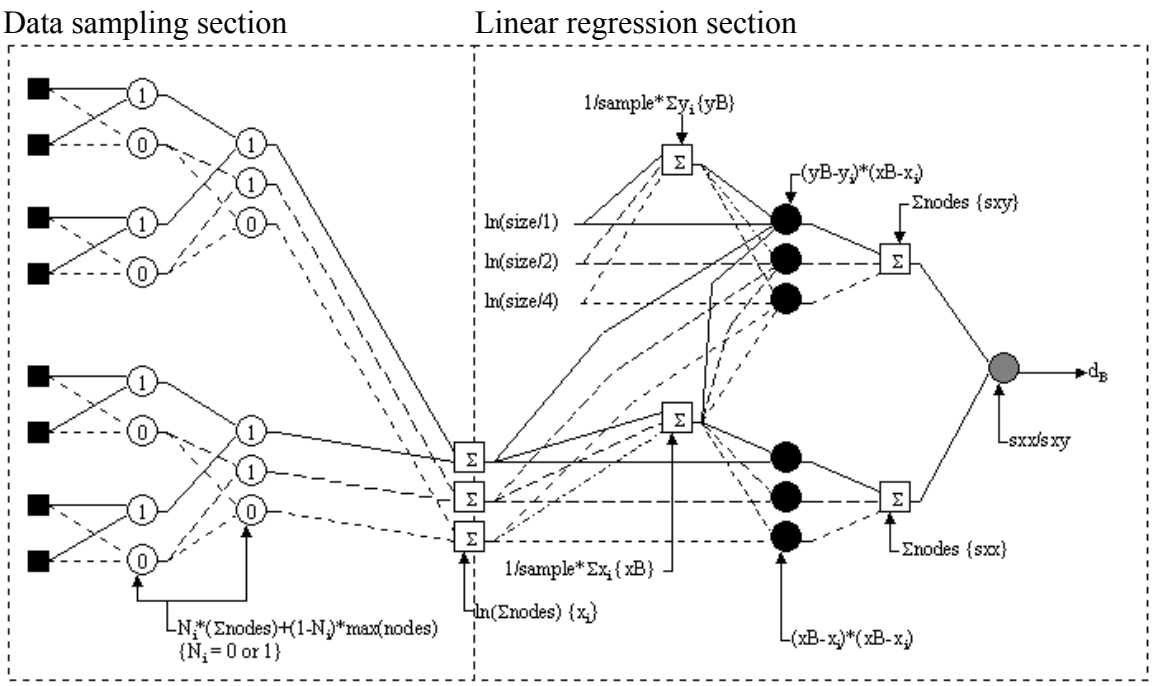


Figure 5. FractalNet for 8 inputs.

Data Sampling Section

The data sampling section is designed to sample the data in a dyatic manner depending on the value of N_i. There are two types of nodes in this section: an input node (black squares in figure 5), which segments the inputs (to 1 or 0); and a data processing node (circles in figure 5), which determines the sample rate. The neural activation function for the data processing nodes is:

$$O_j = [N_i * \sum nodes] + [(1 - N_i) * \max(nodes)] \qquad (3)$$

Note if $N_i = 1$ the nodes to the immediate left are summed. If $N_i = 0$ the nodes to the immediate left are treated as a single entity (via the maximum function). This is shown in the figure 6 where we have 4 inputs, yet the outputs have values as if we separately sampled 4, 2, and 1 input.

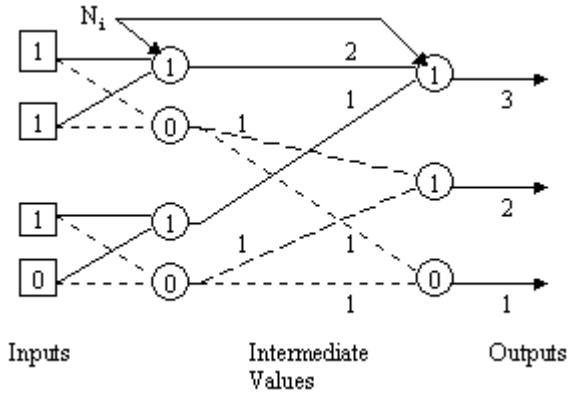


Figure 6. FractalNet example.

Given the activation function and its dependence on N_i a network becomes possible that provides parallel dyatic sampling.

Linear regression section

In keeping with the all-neural network architecture, we also developed a neural network to compute the linear regression of the data generated by the sampling section. While the network is relatively simple in its behavior, some parameters need to be defined:

- Sample - Number of inputs to the linear regression section (3 in figure 5).
- Size - Data set size (8 in figure 5).
- {xxx} - Name of result/node output.

Finally, note that this section, unlike the data sampling section, has some temporal dependencies. Specific nodes (xB and yB calculating nodes) must complete their processing before other nodes can perform their processing.

Scaling

Our second item of interest is how well FractalNet scales. As long one wishes to continue with the dyatic sampling rate, FractalNet can easily be scaled. Each doubling of the data set size results in 1 additional hidden layer in the data sampling section and four additional nodes in the linear regression section. For example, compare the network for 8 inputs (figure 5) with the network for 16 inputs (figure 7).

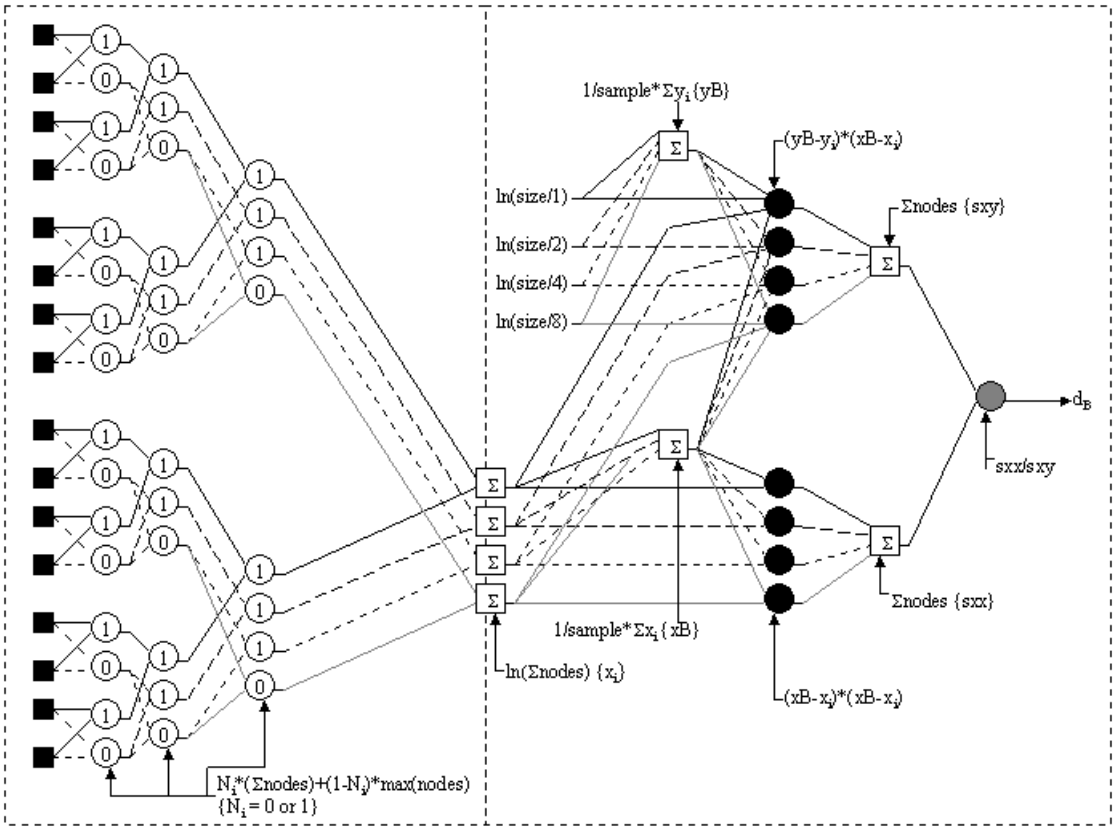


Figure 7. FracalNet for 16 inputs.

Two-Dimensional Data

Out third item of interest is the ability for FractalNet to process 2-D data. Again, only minor network changes are required. For example, figure 8 shows the data sampling section of a FractalNet that has 4 inputs and figure 9 shows the data sampling section of a FractalNet that has 8 inputs organized as a 4x2 pixel image. As can be seen, only minor changes are required to move from 1-D data to 2-D data.

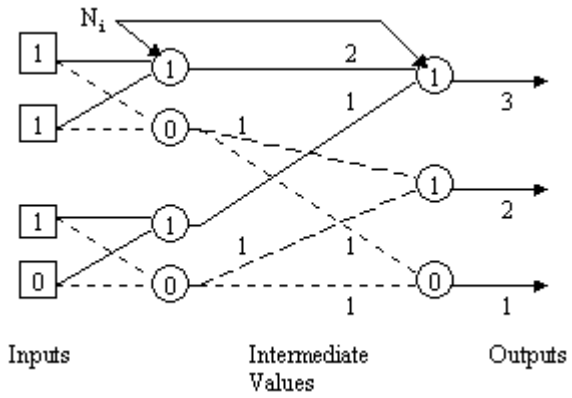


Figure 8. FracalNet for 1D (4x1) data.

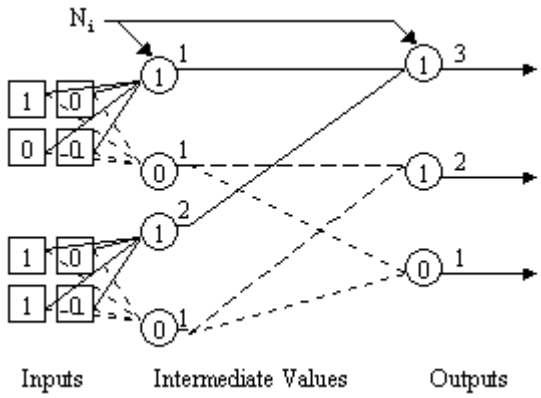


Figure 9. FracalNet for 2D (4x2) data.

4. Conclusion

This paper presents a multiply connected neural network designed to estimate the fractal dimension (D*f*) using the Box-counting method (BCM). The architecture presented is composed of two sections, a data sampling section and a linear regression section. The data sampling section employs two types of nodes: an input node, which segments the inputs (to 1 or 0); and a data processing node, which determines the data sample rate. The data processing nodes use a neural activation function that, depending on the value of N_i, is designed to sample the data in a dyatic manner. The linear regression section computes the slope of the line through the results generated by the data sampling section. The architecture presented scales well and can de modified to process 1-D or 2-D data.

While the neural network presented does generate an estimation of D*f* using the BCM, it is not entirely biologically founded. Several of the activation functions are not biologically plausible. These concerns are the focus of future work on FractalNet.

References

1. H. J. Reitboeck and J. Altmann, "A Model for Size- and Rotation-Invariant Pattern Processing in the Visual System," Biological Cybernetics, Vol. 51, pp. 113-121, 1984.
2. A. D. Culhane, M. C. Peckerar, and C. R. K. Marrian, "A Neural Network Approach to Discrete Hartley and Fourier Transforms," IEEE Transactions on Circuit and Systems, Vol. 36, No. 5, pp. 695-702, 1989.
3. B. B. Mandelbrot, The fractal geometry of nature, Freeman, San Francisco (1983).
4. B. B. Mandelbrot and J. W. Van Ness, Fractional Brownian motions, fractional noises and applications. SIAM Review 10(40), pp. 422-437, 1968.
5. K. J. Falconer, Chapter 2: Hausdorff measure and dimension, In Fractal Geometry Mathematical Foundations And Applications. Thomson Press Ltd. , 1990.

6. D. Comis, Fractals--A bridge to the future for soil science, Agricultural Research Magazine 46(4), pp. 10-13, 1998.
7. N. Sarkar and B. B. Chaudhuri, An efficient approach to estimate fractal dimension of textural images, Pattern Recognition. 23(9), pp. 1035-1041, 1992.
8. S. Davies and P. Hall, Fractal analysis of surface roughness by using spatial data, Journal of The Royal Statistical Society Series, B Statistical Methodology 61(1), pp. 3-29, 1999.
9. D. Osman, D. Newitt, A. Gies, T. Budinger, V. Truong, and S. Majumdar, Fractal based image analysis of human trabecular bone using the box counting algorithm: impact of resolution and relationship to standard measures of trabecular bone structure, Fractal. 6(3), pp. 275-283, 1998.
10. C. B. Caldwell, S. J. Stapleton, D. W. Hodsworth, R. A. Jong, W. J. Weiser, G. Cooke, and M. J. Yaffe, Characterization of mammographic parenchymal pattern by fractal dimension, Physics in Medicine & Biology 35(2), pp. 235-247, 1990.
11. R. L. Webber, T. E. Underhill, R. A. Horton, R. L. Dixon, T. L. Pope. Jr, Predicting osseous changes in ankle fractures, IEEE Engineering In Medicine And Biology Magazine 12(1), pp. 103-110, 1993.
12. N. Mihara, K. Kuriyama, S. Kido, C. Kuroda, T. Johkoh, H. Naito, and H. Nakamura, The usefulness of fractal geometry for the diagnosis of small peripheral lung tumors, Nippon Igaku Hoshasen Gakkai Zasshi 58(4), pp. 148-151, 1998.
13. S. Pohlman, K. A. Powell, N. A. Obuchowski, W. A. Chilcote, and S. Grundfest-Broniatowski, Quantitative classification of breast tumor in digitized mammograms, Medical Physics 23(8), pp. 1337-1345, 1996.
14. W. Jia, "A fractal analysis approach to identification of tumor in brain MR images", MS thesis, Department of Computer Science, NDSU, December, 1999.
15. A. Block, W. von Bloh, and H. J. Schellnhuber, “Efficient box-counting determination of generalized fractal dimensions,” Physics Review A, Vol. 42. pp. 1869-1874, 1990.
16. S. Buczkowski, S. Kyriacos, F. Nekka, and L. Cartilier, “The Modified Box-Counting Method: Analysis of Some Characteristic Parameters,” Pattern Recognition, Vol. 31, No. 4, pp. 411-418, 1998.