SUB-OPTIMAL COVERS FOR MEASURING FRACTAL DIMENSION

Charles R. Tolle, *
Department of Electrical & Computer Engineering
Utah State University
Logan, Utah 84322

David Gorsich, †
US Army Tank-automotive and Armaments Command
Research, Development & Engineering Center
Survivability, MS263
Warren, MI 48397-5000

May 23, 1996

Abstract

This is the second paper in a series of papers [1],[2] that represent continuing work on analyzing visual signatures of ground vehicles using fractal analysis techniques. In this paper, two new dimension estimate solutions based on sub-optimal covers are introduced. These dimension estimate problems and their solutions are akin to the box dimension definition, which is the standard estimate for fractal dimension. However, they represent a dual solution problem to the standard box counting algorithm for estimating the box dimension and thus fractal dimension. These methods are discussed and compared with the standard box counting algorithm using a set of standard brownian random images.

1 Introduction

This is the second paper in a series of papers, [1] and [2], that are investigating the use of fractal dimension for future use in segmenting images into different texture regions. This segmentation process is a common method used in automatic target detection or pattern recognition systems. Image segmentation is the process of separating images into regions which are similar in some way. One possible such segmentation could be made based on texture analysis of an image. This type of segmentation is an important part of computer vision since the patterns provide important cue features to recognize objects. The paper [1] was an initial investigation of the

* TARDEC Co-op, PhD. candidate and Research Assistant for the Department of Electrical and Computer Engineering Utah State University, Utah State University Rocky Mt. NASA Space Grant Consortium Fellow and Research Engineer for the Utah State University Center for Self-Organizing Intelligent Systems.

† Research Associate for TARDEC, and Graduate Research Assistant for the Department of Mathematics Massachusetts Institute of Technology.

use of fractal dimension to segmenting images into different texture regions. Each smaller region is thus characterized by its fractal dimension. Human perception to changes in the fractal dimension of textures have been determined [3]. Moreover, Lincoln Laboratory has looked at the fractal dimension as one of the best five cue features (out of thirteen studied) in automatic target recognition of SAR imagery [4]. When the fractal dimension is used as a cue feature along with several other cue features which are orthogonal to each other (uncorrelated), these features can be mapped into a multi-dimensional space and clustered, in order to classify the regions.

A fractal is a set which has a non-integer fractal dimension. The fractal dimension is most commonly defined as the Hausdorff-Besicovitch (HB) dimension. There are a number of ways to determine the fractal dimension. These are a few of the methods: the box algorithm, the wavelet transform, the Fourier transform or power spectrum method, Hurst coefficients, and capacity dimension. Recent work has been done to determine which of the methods are efficient versus accurate [5]. There is even a web site (http://life.csu.edu.au/fractop/) which calculates the capacity dimension of a user supplied image, although we found this to give us an unreliable measure of dimension.

In our last paper, [1], we look at the box dimension algorithm and a wavelet method developed by Mallat [6]. Each method was checked in one and two dimensional standard cases along with other methods. Each method performed as expected. However, as it has been shown in [5] and Table 1, the actual calculated fractal dimension from algorithms like the box algorithm are very inaccurate for fractal textures. Because of this inaccuracy, we decided to research a new way of estimating the fractal dimension that is much closer to the actual definitions of simple box dimension than the commonly used box algorithm. It was postulated that by obtaining a better estimate of the optimal cover required in determining the box dimension the fractal dimension estimates would become more accurate. In order to achieve this end, we introduce a dual mathematical concept of solving the optimal cover problem normally stated when estimating the fractal dimension. Instead of choosing the ball size and then estimating the positions and number of balls of constant size that optimally cover the set of interest, we reverse the process. This is discussed later. Two different possible solutions are presented and discussed. One based on the Fuzzy C Means Clustering problem and the second based on the solution of a simply stated optimization problem solved using a genetic algorithm. In order to test the accuracy of these new methods a texture generation program given by Ebert [7] is used. The texture generator can give textures of a given fractal dimension with varying lacunarity, see Figure 1.

In the following sections, the Hausdorff-Besicovitch (HB) dimension is stated which leads to the more simple definition of the box dimension. Next the box algorithm is discussed, in order to clearly show the dual nature of that method with the new sub-optimal dimension estimators which are introduced subsequently. Finally, results are presented and discussed along with possible future directions of this research project.

2 Fractal Dimensions

As discussed earlier, the fractal dimension is most commonly defined as the Hausdorff-Besicovitch (HB) dimension, $D_h(A)$, where A denotes the image/signal. In general the HB dimension of A is defined in the following manner [8]:

Let

$$R^n = \{x | x = (x_1, \ldots, x_n), x_i \in R\} \tag{1}$$

for some natural number n. Then, define:

$$diam(C) = sup\{d_e(x,y) | x, y \in C\}, \tag{2}$$

where $d_e(x,y)$ denotes the euclidean distance function. Next, define an open cover of A:

$$A \subset \bigcup_{i=1}^{\infty} C_i \tag{3}$$

Then, define:

$$h_\epsilon^s(A) = \inf \left\{ \sum_{i=0}^{\infty} diam(C_i)^s \,\middle|\, \begin{array}{c} \{C_1, C_2, \ldots\} \text{ open cover} \\ \text{of } A \text{ with } diam(C_i) \leq \epsilon \end{array} \right\} \tag{4}$$

Finally, define the s-dimensional Hausdroff measure of A as:

$$h^s(A) = \lim_{\epsilon \to 0} h_\epsilon^s(A) \tag{5}$$

And it follows that,

$$D_h(A) = \inf\{s | h^s(A) = 0\} = \sup\{s | h^s(A) = \infty\} \tag{6}$$

It should be noted that calculating the HB dimension is hard in general and thus there is a need for a more easily calculated dimension, i.e. the box dimension which is discussed next.

2.1 Using The Box Counting Algorithm To Determine Fractal Dimension

The box dimension, $D_b(A)$, which is normally calculated by the box counting algorithm, is a good estimator of the HB dimension. In general the box dimension can be defined as follows [8]:

> Let $N_\delta(A)$ be the smallest number of closed balls (boxes) with size δ that cover the set A. Then it follows that:

$$D_b(A) = \lim_{\delta \to 0} \frac{log N_\delta(A)}{log \frac{1}{\delta}} \tag{7}$$

At this point it is important to note that the box dimension does not always equal the HB dimension. There are several such examples. It can be shown that, $D_b(A) = n$ for any dense subset A such that $A \subset R^n = \{x | x = (x_1, \ldots, x_n), x_i \in R\}$, likewise for the same A, $D_h(A) \leq n$, moreover $D_h(A) = 0$ for any such countable set A. Therefore, given the set A of rational numbers on $[0,1]$, the box dimension is $D_b(A) = 1$ while the HB dimension is $D_h(A) = 0$ [8]. Although the box dimension fails in some instances, the value it normally produces is a good approximation to the HB dimension.

The major problem with the box dimension is in its calculation. In order to calculate the actual box dimension, one must first find the optimal (smallest number of boxes) covering of A for a given set of boxes with sides of size δ. Note that finding such a cover is in general difficult. Therefore, the box dimension is normally estimated using the box counting algorithm. In short, the box counting algorithm places a standard set of rectangular grids (or set of boxes) upon the image/signal and counts the number of boxes that are filled by the image/signal. This count is then plotted on a *log-log* plot of the number of filled boxes verses the inverse of the box size, see Figure 2. Finally, the box dimension estimate is taken from the monotonically rising nonzero linear slope of the *log-log* plot. By examining this procedure closely one can see that the only difference between the results obtained using the box counting algorithm and the box dimension is in the choice of the cover. In other words, the box counting algorithm doesn't use the optimal cover in general. Furthermore, it has been shown that the box counting algorithm needs at least $10^{D_h(A)}$ points to determine the fractal dimension of a set with dimension $D_h(A)$ [9]. Note that

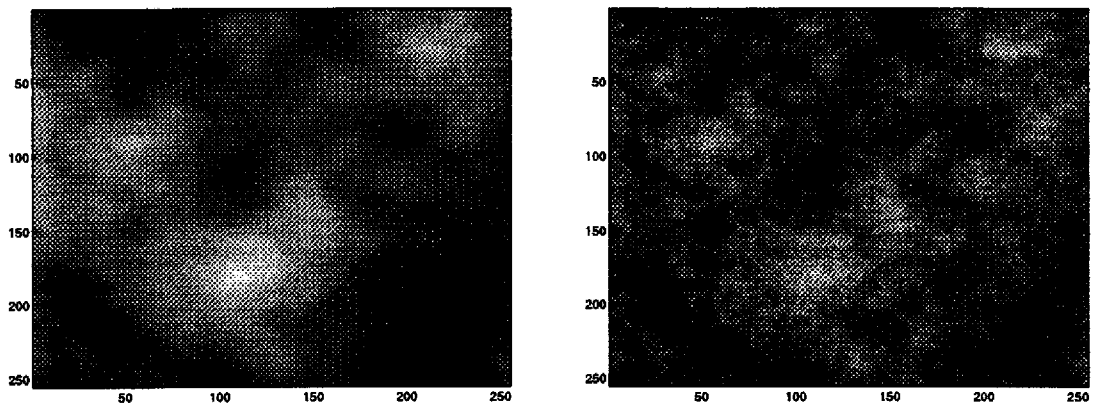

Figure 1: This is an example of two Brownian motion textures with a fractal dimension of 2.1 on the left and 2.7 on the right.

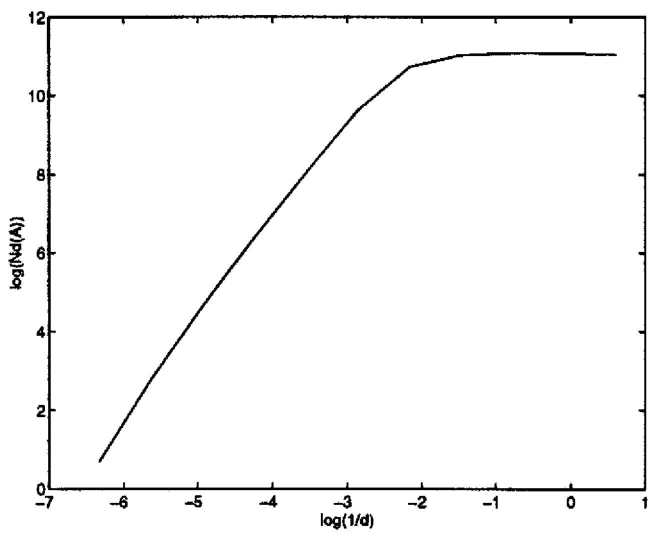

Figure 2: The log-log plot of the results from the box counting algorithm for the texture with a fractal dimension of 2.7 shown in Figure 1.

for the box counting algorithm the box size was choosen and then the number of boxes that covered the set being estimated was calculated in order to make the dimension estimate. This is the dual mathmatical solution to the next two sub-optimal cover methods that are introduced. The next two methods choose the number of boxes and then solve for the smallest box size.

2.2 Fuzzy C Means Sub-optimal Cover Fractal Dimension

As was stated earlier, the box algorithm has a dual solution. Instead of choosing the box size and then finding the number of boxes that cover the image, one might just as well choose the number of boxes and then find the smallest box size that covers the image. This is a simple clustering style problem. Therefore one might think of trying one of the standard clustering methods to produce the cover needed for the calculation of the box dimension as discussed in the last section, such as the standard Fuzzy-C Means Clustering algorithm[10]. The Fuzzy-C Means algorithm solves the following problem:

$$min \quad \sum_{i=1}^{C} \sum_{k=1}^{N} u_{ik}^2 \|x_k - v_i\|^2 \tag{8}$$

$$\begin{array}{lrl} W.C. & u_{ik} & \in [0,1] \\ & \sum_{i=1}^{C} u_{ik} & = 1 \quad \forall k = 1,2,\ldots,N \end{array} \tag{9}$$

where, C is the number of clusters, N is the number data vectors $\{ x_k \}$ being clustered, u_{ik} is the membership value of the k^{th} data vector in the i^{th} cluster, and v_i is center (mean) vector of the i^{th} cluster. For this section the $\| \bullet \|$ is assumed to be the euclidean norm. It has been shown, [10], that this problem can be solved by using the Picard iteration[11], also known as the method of successive approximations [12].

Picard iteration for the fuzzy-C means problem:

1) Choose the number of clusters(quantization levels) $C \geq 1$. Select $\epsilon > 0$, this is the ending condition. Finally, randomly initialize the centers of the clusters, v_i.

2) Solve for the memberships of each vector:

$$\begin{array}{rcl} u_{ik} & = & \frac{1}{\sum_{j=1}^{C} \left[\frac{\|x_k - v_i\|}{\|x_k - v_j\|} \right]^2} \\ where & & k = 1,2,\ldots,N \\ & & i = 1,2,\ldots,C \end{array} \tag{10}$$

3) Solve for the centers for each cluster:

$$\begin{array}{rcl} v_i & = & \frac{\sum_{k=1}^{N} u_{ik}{}^2 x_k}{\sum_{j=1}^{N} u_{ik}{}^2} \\ where & & i = 1,2,\ldots,C \end{array} \tag{11}$$

4) Repeat step 2.

5) Repeat step 3.

6) If $\Delta u_{ik} > \epsilon$ then loop back to step 4. Otherwise, stop.

Note that by using the Euclidean norm the algorithm produces a minimum distance classification of the data, which is just what we are interested in. In addition, one can assume that the box size for this cover is obtained from the largest delta of all of the clusters. However, there is a problem with this type of a definition for this clustering method. The Fuzzy-C problem doesn't penalize greatly for having a single out lying point. Therefore the maximum delta can be to large and thus a very poor estimate of the optimal cover, as was discovered. However, without mathematical justification – one can define the ball size to be the average size of each of the clusters. And using this engineering choice one can obtain relatively good results. This is the basis for the Fuzzy-C average estimates given later. This problem can be resolved by restating the problem as below.

2.3 Minimum Ball Sub-optimal Cover Fractal Dimension

If one looks closely at the definition of the box dimension one could state the problem of finding the optimal cover, using the nomenclature from the Fuzzy-C problem in the last section, as follows:

Given C, in the set of natural numbers. Find a set of C clusters centers, v_i, such that $\max_{\forall k's} (\delta_k)$ is minimized. (12)

where $\delta_k = \min_{i \in \{1,..,C\}} \|x_k - v_i\|^2$

Well this problem statement is not as easily solved numerically as the Fuzzy-C means problem earlier. However, this problem has a very well defined cost. Therefore it is a perfect candidate for the easy out solution of using a genetic algorithm for solving this problem. The genes are made up of the cluster center, v_i, and the max-min cost is easily solved with a few simple calculations and comparisons. It is clear that in solving the above problem, one does in fact obtain the desired optimal cover needed for calculating the box dimension. However, since we have decided to initially solve this problem using a genetic algorithm we are not guaranteed the optimal solution, i.e. a sub-optimal solution will be obtained in general.

3 Results

In the previous section we have introduction of the two new sub-optimal covers for estimating the box dimension and thus the fractal dimension of an image. Therefore, in giving results, only the box dimension cover estimates are used for comparison. Note that these calculations are based on massive search and clustering problems that take long run times on SGI Indigo II class machines. Some of these estimates took up to three days of solid calculating on such a computer system. Thus the results are some what limited by the run times. Furthermore, it should be noted that the number of calculations required for these types of clustering problems grow faster than the number of clusters squared. Therefore, the results presented are based on the micro level of similarities within the images. In other words, the results are based on small numbers of clusters over a narrow range for the two new sub-optimal covers. Moreover, the sample textures had a limited number of points, 100, because of the computation times required for textures on the order of a 1000 points. Note that there are much faster methods for estimating the fractal dimension such as the wavelet method that we discuss in [1] and will discuss further in [2]. However, faster methods do not provide more accurate estimates than the box dimension estimates thus far. For that reason, it is interesting to look closely at these more computationally intensive methods in order to obtain a better and more accurate estimate of the box dimension and thus the fractal dimension of the test images.

In order to test the accuracy of these methods a standard set of Brownian random virtual texture images were produced and used for the estimates. They had fractal dimensions that ranged from 2.1 to 2.9, see Figure 1. These images were produce using Ebert's texture generation program [7]. The word virtual means that these three dimensional texture surfaces were estimated using their nonscaled and only machine limited numerical values. These virtual images are hoped to minimize the problems that can be intro-

Table 1: Estimated fractal dimensions for the virtual textures.

Image Fractal Dimension	Box Algorithm Estimate	Fuzzy-C Average Estimate (100 point)	Fuzzy-C Average Estimate (400 point)	Min Ball Estimate (400 point)
2.1	2.13	2.02	nc	2.02
2.2	2.18	2.15	nc	ganc
2.3	2.24	2.30	nc	2.33
2.4	2.24	1.90	2.50	ganc
2.5	2.27	2.50	nc	ganc
2.6	2.31	2.34	nc	nc
2.7	2.56	2.5	nc	nc
2.8	2.47	2.60	nc	nc
2.9	nc	2.85	nc	nc

nc = not calculated   ganc = the genetic algorithm did not converge

duced by scaling and discretizing of the texture surface, which can result in a change in the fractal dimension of the test image. Table 1 shows the results for each of the methods that have been calculated thus far. Note that some of the results calculated from the 100 point images are promising. However, as stated earlier in order to accurately find the $D_h(A)$ dimension for images of dimension 2.1 to 2.9 these tests need to be made on texture surfaces with 1000 points. However, the computation time is on the order of weeks for each run. In addition, only one data point is included for the Minimum Ball genetic algorithm because the algorithm did not converge within the 1000 generation cutoff. Additional data points will be available subsequently.

4 Conclusions and Future Directions

By re-examining the box dimension definition a set of dual mathematical problems is shown to exist. The first is based on the box counting algorithm estimator. This problem assumes that one knows the size of the balls used to cover the image/set and only requires the number of balls. Moreover, this is what the box counting algorithm solves, in order to obtain the box dimension estimate. The second problem, or the dual of the first is to assume a known number of balls one needs to cover the image/set and then solve for the size of the balls that that correspond to that known number. In considering the later problem, two new possible solutions were discussed and a few results for those solutions were given. There are not enough results available yet in order to form a complete idea on if these new dual solution problems are more accurate than the box counting algorithm, however it does look promising. Furthermore, a numerical solution to the Max-Min ball problem should be sought. In addition, we are also investigating the much more quickly estimated wavelet method as discussed in [2].

References

[1] David Gorsich, Charles R. Tolle, Robert Karlsen, and Grant Gerhart, "Wavelet and Fractal Analysis of Ground Vehicle Signa-

tures," 7^{th} *Annual Ground Vehicle Survivability Symposium Proceedings*, Naval Postgraduate School, Monterey, CA, March 25-28, 1996.

[2] David Gorsich, Charles R. Tolle, Robert Karlsen, and Grant Gerhart, "Wavelet and Fractal Analysis of Ground Vehicle Images," *SPIE Symposium*, Denver, CO, August 5-9, 1996.

[3] Lewis F. Jardine, "Fractal-based Analysis and Synthesis of Multispectral Visual Texture for Camouflage," *Applications of Fractals and Choas*, Springer-Verlag, New York, NY, 1993, pp 101-116.

[4] Daniel E. Kreithen, Shawn D. Halversen, and Gregory J. Owirka, "Discriminating Targets from Clutter," *MIT Lincoln Laboratory Journal*, Vol. 6, No. 1, Spring, 1993, pp. 25-52.

[5] X.C. Jin, S.H. Ong, and Jayasooriah, "A practical method for estimating fractal dimension," *Pattern Recognition Letters*, Vol. 16, May, 1995, pp. 457-464.

[6] Stephane G. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. II, No. 7, July, 1989, pp. 674-693.

[7] Editor David S. Ebert, *Texturing and Modeling: A Procedural Approach*, Academic Press Professional, Cambridge, MA, 1994, pp. 256-260.

[8] Heinz-Otto Peitgen, Hartmut Jürgens, and Dietmar Saupe, *Chaos and Fractals: New Frontiers of Science*, Springer-Verlag New York, Inc., New York, NY, 1992.

[9] James B. Bassingthwaighte, Larry S. Liebovitch, and Bruce J. West, *Fractal Physiology*, American Physiological Society, New York, NY, 1994.

[10] Robert L. Cannon, Jitendra V. Dave, and James C. Bezdek, "Efficient Implementation of Fuzzy c-Means Clustering Algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8, No. 2, 1986, pp. 248-255.

[11] Peter V. O'Neil, *Advanced Engineering Mathematics*, Wadsworth Publishing Company, Belmont, California, 1987, pp. 57-61.

[12] Philip Hartman, *Ordinary Differential Equations*, John Wiley & Sons,Inc., New York, NY, 1964.