

Article

An Approximated Box Height for Differential-Box-Counting Method to Estimate Fractal Dimensions of Gray-Scale Images

Chinmaya Panigrahy ¹, Angel Garcia-Pedrero ² , Ayan Seal ¹,
Dionisio Rodríguez-Esparragón ³ , Nihar Kumar Mahato ¹ and Consuelo Gonzalo-Martín ^{2,*}

¹ PDPM Indian Institute of Information Technology, Design and Manufacturing, Jabalpur 482005, India; panigrahy.chinmaya@gmail.com (C.P.); ayanseal30@ieee.org (A.S.); nihar@iiitdmj.ac.in (N.K.M.)

² Center for Biomedical Technology, Universidad Politécnica de Madrid, Campus de Montegancedo, 28233 Pozuelo de Alarcón, Spain; angel.garcia@ctb.upm.es

³ Instituto de Oceanografía y Cambio Global, IOCAG, Campus de Tafira, Edificios de Telecomunicaciones, 35017 Las Palmas, Spain; dionisio.rodriguez@ulpgc.es

* Correspondence: consuelo.gonzalo@upm.es; Tel.: +34-91-336-4663

Received: 7 June 2017; Accepted: 30 September 2017; Published: 10 October 2017

Abstract: The Fractal Dimension (FD) of an image defines the roughness using a real number which is highly associated with the human perception of surface roughness. It has been applied successfully for many computer vision applications such as texture analysis, segmentation and classification. Several techniques can be found in literature to estimate FD. One such technique is Differential Box Counting (DBC). Its performance is influenced by many parameters. In particular, the box height is directly related to the gray-level variations over image grid, which badly affects the performance of DBC. In this work, a new method for estimating box height is proposed without changing the other parameters of DBC. The proposed box height has been determined empirically and depends only on the image size. All the experiments have been performed on simulated Fractal Brownian Motion (FBM) Database and Brodatz Database. It has been proved experimentally that the proposed box height allow to improve the performance of DBC, Shifting DBC, Improved DBC and Improved Triangle DBC, which are closer to actual FD values of the simulated FBM images.

Keywords: Fractal Dimension; Differential Box Counting; box height; Fractal Brownian Motion; Brodatz Database

1. Introduction

A natural scene could be treated as complex objects that could be represented using Fractal Geometry, which was introduced by Mandelbrot [1] in 1983. Fractal Geometry is used to define self-similar elements independent of scale, known as fractal set. FD is a measure to quantify the roughness or irregularities present in that set. Nowadays, researchers have been using FD in various computer vision applications such as texture, image segmentation [2], shape identification [3], palm print recognition [4], time series analysis [5], classification [6] and spatial variability analysis [7]. Interested readers can find more application areas in [8].

Mandelbrot [1] first gave an idea to compute FD which was further advanced by Peleg et al. [9], Pentland [10], Keller et al. [11], Gagnepain et al. [12] etc. In [13], Sarkar et al. proposed a method known as DBC to compute FD using box counting (BC) technique, which became popular in research community because it is simple, easy to interpret and implement. However, it suffers from various drawbacks like over-counting of boxes along z-direction (quantization-computation), over-counting of boxes along xy-direction, lack of a proper box height, inappropriate use of grid sizes. Several

modifications of DBC have been done to yield accurate FD values. Jin et al. [14] restricted the allowed range of grid sizes to avoid distortions occurred during line fitting to compute FD. Chen et al. [15] noticed that DBC suffers from over-counting of boxes along z-direction and proposed a new formula to compute the number of boxes required to cover rough surface over a grid (n_r), but authors used cubes to represent boxes. In [16,17], Li et al. used a new xy -plane partitioning mechanism to consider the gray-level variations between two adjacent grids that overlaps two adjacent grids by one row and one column. Authors also proposed a new formula to compute n_r and two finer box heights. However, these box heights are too small, which increase the number of box counts significantly. As a result, more memory is required to store the large number of boxes. Liu et al. in [18] proposed a novel xy -plane shifting mechanism to solve the problem of under-counting. Authors used the grid sizes as only the divisors of the image size. In [19], Kaewaramsri et al. partitioned the square grid into two triangular grids to obtain better precision of box count. Interested readers can find various modifications of DBC in [20–23].

Out of different parameters of DBC, box height, which is directly related to computing the number of boxes required to represent the gray-level variations over a grid and badly affects the estimation of FD values. Larger box height incurs high error. While lower box heights require more computation. Some box heights depend only on the grid size on the contrary some depend on the image characteristics, whose values change according to the degree of roughness present in an image. It was observed that the box height is always directly proportional to grid size in order to preserve the scale of boxes. Different box heights used by different authors are defined in Section 2.4. Box height used by DBC method is directly proportional to number of gray-levels and inversely proportional to image size [13]. Authors in [15] used cubes to represent a box. It means that the height is directly proportional to grid size. Some of the authors used maximum and minimum gray-level of an image as box height in their works [16,23] respectively. Authors in [17] used standard deviation (σ) of an image as box height. Box heights proposed in [16,17] are very small and accurate in estimating FD but are not dependent on image size. The impact of box heights have been studied thoroughly and proposed a new box height as a function of image size based using Brodatz Database [24], which is also validated using simulated FBM Database. Results show that the new box height allows to better estimate FD value with a less Distance Error (DE) than the box height used by DBC method. Rest of the paper is organized as follows: Section 2 discusses the traditional DBC method and its variants. Section 3 depicts experimental results, which help to propose a new box height as a function of image size. The validation of the proposed box height has been done on gray-scale images from simulated FBM Database and Brodatz Database in Section 3. The conclusion has been drawn in Section 4.

2. Materials and Methods

2.1. Databases

All the experiments have been performed on one publicly available Brodatz Database [24] and one generated FBM Database using random mid-point displacement algorithm [25]. The Brodatz Database contains 112 textured images of size 640×640 pixels. This database is used to propose a novel box height for calculating FD values because this database is very much popular and used by most of the researchers in this context. Some of the textured images from Brodatz Database and generated FBM images with their actual FD values that could be generated using Hurst component (H) by 3-H are shown in Figures 1 and 2 respectively. The H varies from 0 to 1 with step size 0.1 and it is used to validate the proposed method in this work.

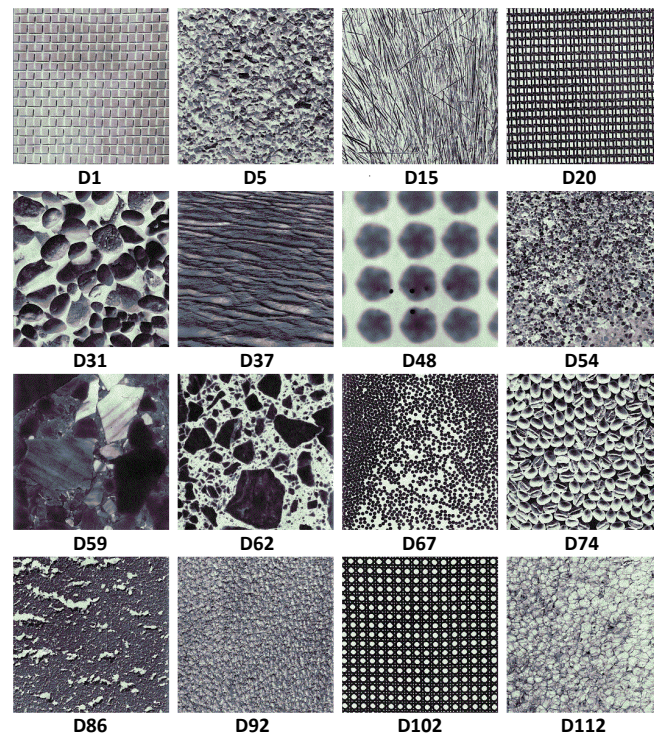


Figure 1. Some of the sample textured images from Brodatz Database.

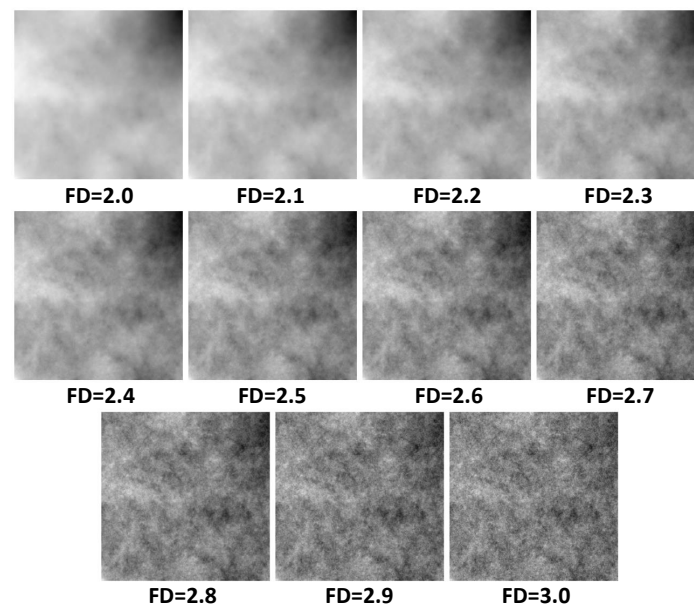


Figure 2. Generated FBM images with their actual FD values.

2.2. Fractal Dimension

Standard formula for computing FD of an ideal bounded fractal set, B , using BC method is expressed by Equation (1) [1].

$$FD_B = \frac{\log(N_r)}{\log\left(\frac{1}{r}\right)}, r \neq 1 \quad (1)$$

where, N_r is the number of distinct boxes of scale r that are required to cover the B . If the B is not an ideal fractal set then Equation (1) will give different FD values for different values of r . In that case

the slope of the line obtained by fitting the points $(\log N_r, \log \frac{1}{r}) \forall r$ is known as the FD of the non-ideal fractal set. The detail description of the DBC method is given in Section 2.3.

2.3. The DBC Method

In 1992 Sarkar et al. [13] proposed an straightforward approach to estimate FD of a gray-scale image. Their methodology is as follows: a square image (I) of size $M \times M$ pixels is partitioned into non-overlapping grids of size $s \times s$ pixels where s is an integer and s varies from 2 to $\frac{M}{2}$. The related scale (r) of a grid with size $s \times s$ pixels with respect to an image size M is denoted by an expression, $r = \frac{s}{M}$. If s is not a divisor of M then non-image pixels of the grids on the boundary are treated as zero.

On each grid, there are a number of boxes of size $s \times s \times h$, to cover the rough gray-level image intensity surface over that grid and these boxes are assigned with a number as shown in Figure 3, where, $\lfloor \cdot \rfloor$ denotes the floor function and

$$\begin{aligned} \lfloor \frac{G}{h} \rfloor &= \lfloor \frac{M}{s} \rfloor \\ \Rightarrow h &= \frac{s \times G}{M} \end{aligned} \quad (2)$$

The G is the total number of gray-levels in a gray-scale image i.e., 256. The total number of boxes of size $s \times s \times h$, required to represent the rough surface over a (i, j) th grid of size $s \times s$ pixels, $n_r(i, j)$ is

$$n_r(i, j) = \left\lceil \frac{g_{max}}{h} \right\rceil - \left\lceil \frac{g_{min}}{h} \right\rceil + 1 \quad (3)$$

where, $\lceil \cdot \rceil$ denotes the ceiling function. The g_{max} and g_{min} are the maximum and minimum gray-levels present on (i, j) th grid respectively. The $\lceil \frac{g_{max}}{h} \rceil$ represents the box number, which contains g_{max} . On the other hand, g_{min} located in a box, which is denoted by box number $\lceil \frac{g_{min}}{h} \rceil$.

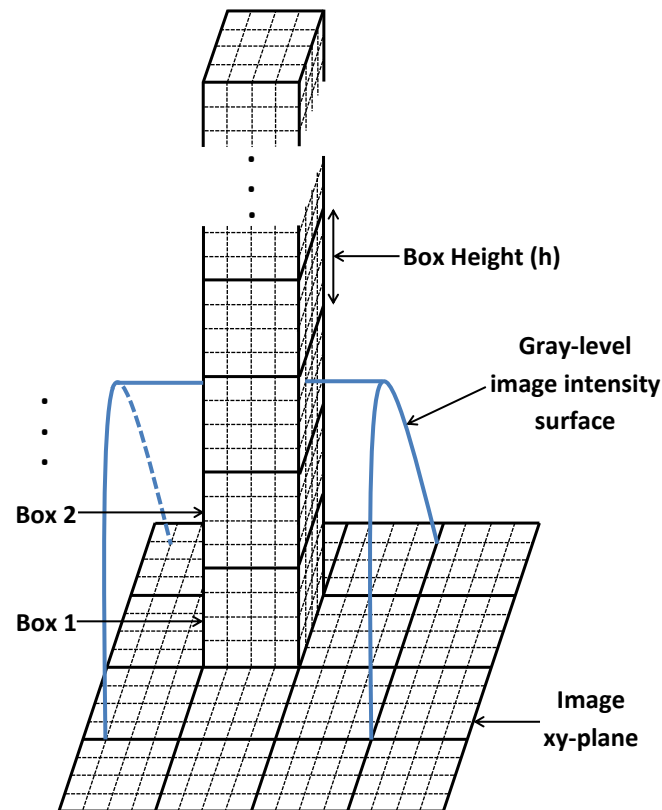


Figure 3. Sketch for determining the number of boxes, n_r , by DBC method.

The total number of boxes, N_r , at scale r is required to cover the rough intensity surface of an image can be computed by Equation (4).

$$N_r = \sum_{i,j} n_r(i,j) \text{ where, } \frac{2}{M} \leq r \leq \frac{1}{2} \quad (4)$$

Then the FD of an image or the slope of a line is computed by fitting all the points $(\frac{1}{r}, N_r)$ using Linear Least Squares as shown in Figure 4.

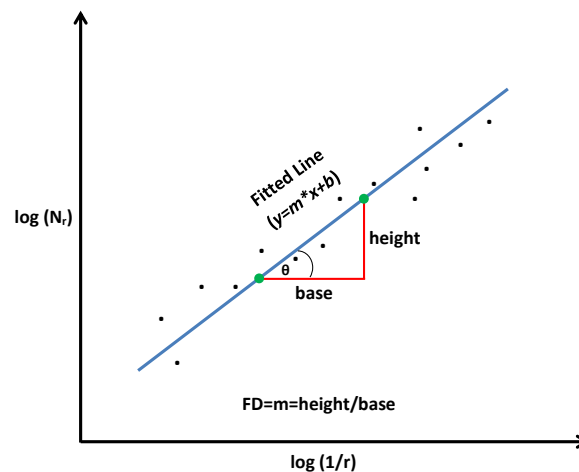


Figure 4. The log-log plot for the computation of FD.

The corresponding Distance Error (DE) incurred while computing FD can be estimated using Equation (5) defined as follows:

$$DE = \frac{1}{N} \sqrt{\frac{\sum_{i=1}^N (mx_i + b - y_i)^2}{1 + m^2}} \quad (5)$$

where, m is slope of the line i.e., FD, b is the y -intercept and N is the total number of points used in fitting.

Box height is directly related to an image gray-level intensity variations which is characterized by z -axis in a 3D plane. It is clear that a larger value of box height is unable to compute accurate number of boxes. Whereas a lower value of box height is capable of finding out FD of an image accurately. However, it requires more memory to store box counts. Therefore, extensive study on the impact of different box heights have been done in this work to compute FD values.

2.4. Box Height

Box height is directly proportional to grid size s . It means that the ratio of box height and grid size is equal to a constant value. The box height should not exceed the size of a grid otherwise, it leads to an inaccurate FD. Moreover, box height should not be constant because in that case the scale of boxes (r) on different grids will not be maintained. Let us assume that 23 gray-levels are required to represent the rough surface over a particular grid. If the box height is 11 then 3 boxes are required to represent the rough surface. Out of these three boxes, first two boxes are completely filled up by 11 gray-levels and the third box contains only one gray-level. But, if the box height is 2 then 12 boxes are required with 11 full boxes and 1 box is added for one gray-level count. Again if the box height is 0.5 then a total of 46 full boxes are required to cover the rough image intensity surface. Therefore, $h = 0.5$ is more accurate and $h = 11$ is less accurate. Hence, a smaller box height is preferable and we have proposed a method to compute box height in Section 3.2. Different existing definitions for computing box heights are as follows:

- **A:** In [13,14,18–22], authors used a box height, $h = \frac{s \times G}{M}$, for their work. When $M \leq G$ then this box height becomes very large hence fails in providing accurate FD.
- **B:** Chen et al. considered $h = s$ in their work [15] to compute FD. Sometimes it may happen that the box height is greater than the number of gray-levels in an image ($h \geq G$); in that case only one box is required to represent the rough surface of a particular grid. Consequently, it leads to an inaccurate FD.
- **C:** Li et al. proposed $h = \frac{s}{0.5 \times (I_{max} - I_{min})}$ to be their box height in [16], where I_{max} , I_{min} are the maximum and minimum gray-level intensities present in an image respectively. The box height will be very small when an image is having a large texture variation. On the other hand, when gray-level differences ($I_{max} - I_{min}$) will be small then the box height will be high as a result inaccurate FD value will be produced.
- **D:** Same authors proposed another formula for box height, $h = \frac{s}{1+2a\sigma}$, where σ is the standard deviation of an image and $a = 3$ is a constant in [17]. But, the proposed box height is too small and it requires more memory space for storage. The value of this box height is almost same to the box height used in [16] and like **C** this box height is dependent on image gray-levels.
- **E:** In [23], Lai et al. used $h = \frac{s \times (I_{max} - I_{min} + 1)}{M}$ as box height. Sometimes $(I_{max} - I_{min} + 1) \approx G$, which is almost same to the box height used in [13,14,18–22].

3. Experimental Results and Discussion

3.1. Impact of Box Heights

Firstly, three box heights namely, **A:** $\frac{s \times G}{M}$, **B:** $h = s$ and **D:** $\frac{s}{1+2a\sigma}$ (**B** > **A** > **D**) are selected for further experiment from **A**, **B**, **C**, **D** and **E** described in Section 2.4 because box height **A** is almost same to **E** for texture images and box height **C** is nearly equal to **D**. These box heights have been considered separately with DBC method by keeping other parameters same. The FD values along with DE values are then computed on Brodatz Database and they have been depicted in Figures 5 and 6. However, all the images are scaled down to standard 512×512 pixels before computation. Because for $M = 256$, **A** = **B**, hence these two box heights cannot be distinguished and simultaneously $M = 640$ is not selected since it is not a standard image size i.e., not a power of 2. The list of parameters of DBC method are reported in Table 1. It is clear from both Figures 5 and 6 that FD value increases when box height decreases. On the other hand, DE values increases while box height increases.

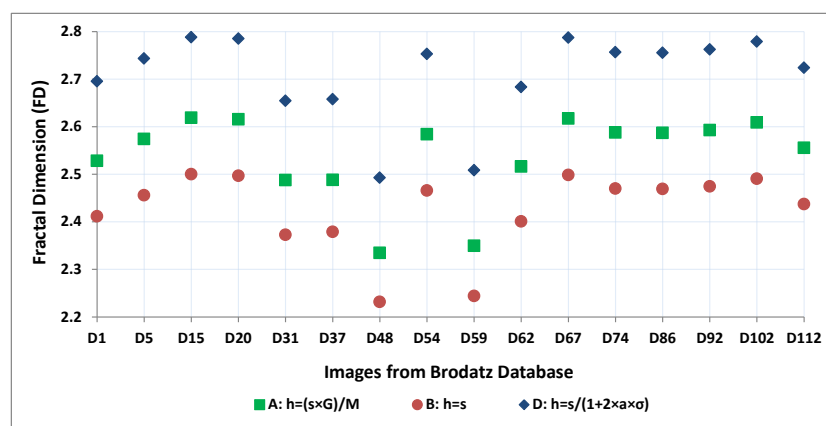


Figure 5. Different FD values for images from Brodatz Database using DBC method together with box heights **A**, **B** and **D** while image size is 512×512 .

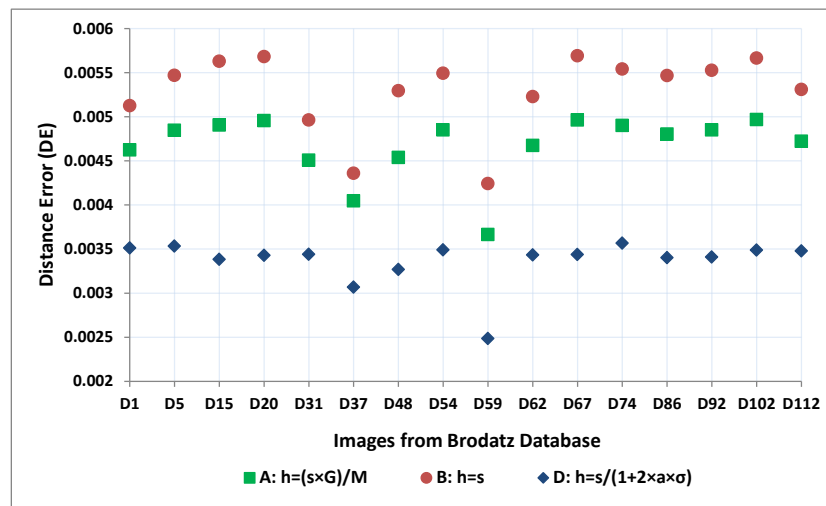


Figure 6. Different DE values for images from Brodatz Database using DBC method together with box heights **A**, **B** and **D** while image size is 512×512 .

3.2. A Novel Method to Estimate Box Height

We can conclude from Section 2.4 that the standard expression for box height is as follows:

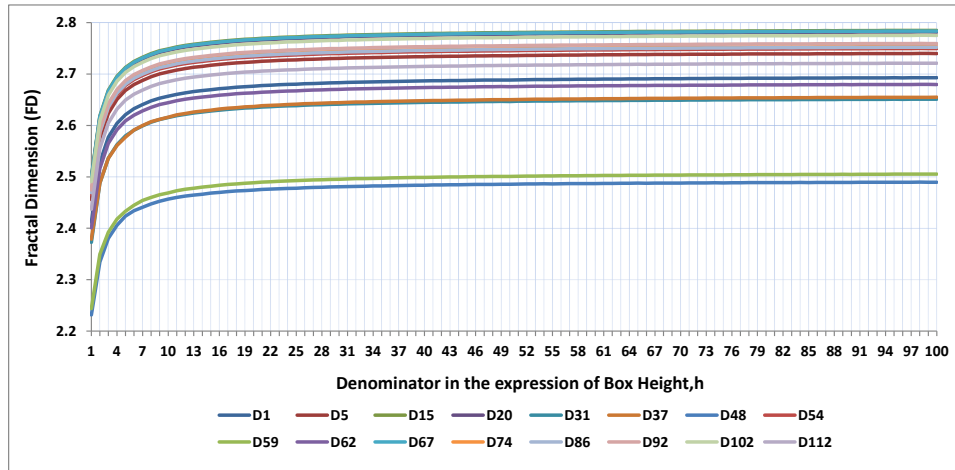
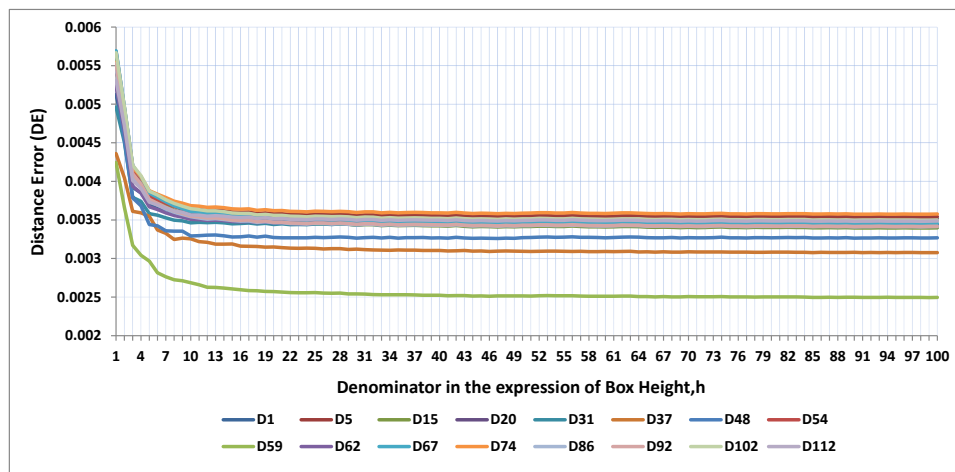
$$h = \frac{s}{R} \quad (6)$$

where, R is any real number ($R \in \mathbb{R}$) and R must be greater than or equal to one ($R \geq 1$) because if the value of R is less than one and the value of s is large then h will be more than G in that case the obtained value of FD will be inaccurate. Most of the authors have considered R as a positive natural number for their works. Thus, a series of natural numbers 1 to 100 have been chosen for the value of R to compute FD value in order to find out a R value for an image, which satisfies Equation (7) in this work. Bicubic interpolation method is applied on each of these images of size 640×640 pixels to resize them into 128×128 , 256×256 , 512×512 and 1024×1024 pixels for further experiments. FD of ideal fractal sets are invariant to size of the set i.e., for any value of s in Equation (1), the value of FD_s will remain same. Since texture images are not ideal fractals, their FD value changes with the size of the image though the difference in FD values are less but not equal to zero. Hence we have used different image sizes in-order to relate their FD value and using their relation a new box height is proposed in this paper.

The Equation (6) is applied as box height with other parameters of DBC described in Table 1 on 112 textured images from Brodatz Database at different image sizes 128×128 , 256×256 , 512×512 , 640×640 and 1024×1024 pixels. Figures 7 and 8 show different FD values and DE values for different values of R respectively, which has started from 1 to 100 with step size 1. Figure 7 contains the FD values of images, which are shown in Figure 2. The similar experiments have been performed on 112 images with different image sizes 128×128 , 256×256 , 640×640 and 1024×1024 pixels. It is observed that the same trend persist between these experiments as shown in Figure 7. Moreover, it is clear from Figure 7 that for all images when box height decreases i.e., R in Equation (6) increases then the computed FD value increases, but after some higher value of R the estimated FD value almost remains the same as R increases. Similarly Figure 8 shows that as box height decreases, incurred error i.e., DE, decreases. Though decreasing of DE values are not fully monotonic but eventually after some lower box height, i.e., for higher R values, the DE values remain almost the same. So, we have to find out a proper value of R for a particular image size in such a way that there is no significant change in the difference of two successive FD values for two successive values of R .

Table 1. Different parameters of DBC, SDBC, IDBC and ITDBC methods.

	DBC [13]	SDBC [15]	IDBC [18]	ITDBC [22]
s_{min}	2	2	2	2
s_{max}	$\frac{M}{2}$	$\frac{M}{2}$	$\frac{M}{2}$	$\frac{M}{2}$
Allowed Grid sizes, s	$s_{max} \leq s \leq s_{max}$	$s_{max} \leq s \leq s_{max}$	$2^i, i = 1, 2, \dots, \log_2 M - 1$	$s_{max} \leq s \leq s_{max}$
Grid shifting mechanism	No shifting	No shifting	Each grid is shifted by one pixel along South-East direction	No shifting
Grid partitioning mechanism	No grid partitioning	No grid partitioning	No grid partitioning	Each grid is partitioned into unequal triangular grids in four different ways
n_r	$\left\lceil \frac{g_{max}}{h} \right\rceil - \left\lceil \frac{g_{min}}{h} \right\rceil + 1$	$\left\lceil \frac{g_{max} - g_{min} + 1}{h} \right\rceil$	$\begin{cases} \left\lceil \frac{g_{max} - g_{min} + 1}{h} \right\rceil, & \text{if } g_{max} \neq g_{min} \\ 1, & \text{if } g_{max} = g_{min} \end{cases}$	$\left(\left\lceil \frac{t g_{max}}{h} \right\rceil - \left\lceil \frac{t g_{min}}{h} \right\rceil + 1 \right) \times \left(\frac{A_t}{s \times s} \right)$ A_t is the number of image pixels in the t th triangular partition
xy -plane partitioning	Partition image plane into non overlapping square grids.	Partition image plane into non overlapping square grids.	Partition image plane into non overlapping square grids.	Partition image plane into non overlapping square grids.
Box Height, h	$\frac{s \times G}{M}$	s	$\frac{s \times G}{M}$	$\frac{s \times G}{M}$

**Figure 7.** Different FD values for images from Brodatz Database using DBC method along with box height, $h(= \frac{s}{R})$, while image size is 512×512 .**Figure 8.** Different DE values for images from Brodatz Database using DBC method along with box height, $h(= \frac{s}{R})$, while image size is 512×512 .

Approximating R

Let, FD_R^{pM} be a FD value of an image numbered as p from Brodatz Database having $M \times M$ pixels with a particular value of R where, R starts from one and $M \in [128, 256, 512, 640, 1024]$. Now, smallest integer value, R , has to be found that satisfies Equation (7) for each image p at a particular image size $M \times M$.

$$|FD_R^{pM} - FD_{(R+1)}^{pM}| \leq 0.0001 \quad (7)$$

It is observed from the experiments that the value of FD up-to four decimal places is enough to distinguish textures. Moreover, floating point arithmetic is very expensive. So, a threshold value 0.0001 is chosen for our work. Table 2 shows the number of images having the same R value for different image size that satisfy Equation (7). Here, R_{max}^M is marked using red color in Table 2, which represents the smallest R value for an M , after which FD value does not change significantly for all the images.

Table 2. Finding out the smallest values of R that satisfy Equation (7) for different image sizes of Brodatz Database.

$M = 128$		$M = 256$		$M = 512$		$M = 640$		$M = 1024$	
R	No. of Images	R	No. of Images	R	No. of Images	R	No. of Images	R	No. of Images
8	53	24	61	29	5	33	1	35	3
10	1	26	33	34	1	35	4	38	2
12	1	28	2	39	69	38	2	40	2
14	3	29	6	43	33	40	8	45	2
15	1	31	1	47	2	42	44	47	1
16	7	32 (R_{max}^{256})	9	49 (R_{max}^{512})	2	43	1	49	3
17	15	Total Number of Images = 112		Total Number of Images = 112		46	1	50	1
18	1					48	4	51	1
19	10					50	17	52	9
21	4					52	4	54	1
24	2					53	1	56	15
25	2					54	2	57	3
26	10					56	10	58	2
27	1					57	2	59	2
28 (R_{max}^{128})	1					60	9	60	9
Total Number of Images = 112						62 (R_{max}^{640})	1	63	1
						Total Number of Images = 112		64	6
								65	1
								66	35
								70	7
								71	2
								74	1
								76	2
								78 (R_{max}^{1024})	1
								Total Number of Images = 112	

Then all the points (M, R_{max}^M) are plotted into a 2D plane to get a line using linear least square, where x -axis represents M i.e., image size and y -axis denotes R_{max}^M of a particular image size M . Here, (M, R_{max}^M) corresponds to $(128, 28)$, $(256, 32)$, $(512, 49)$, $(640, 62)$ and $(1024, 78)$ that have been obtained from Table 2. The expression of the plotted line is represented by Equation (8).

$$R = 0.0589M + 19.667 \quad (8)$$

Now, this line is validated by putting different values of M (128, 256, 512, 640 and 1024) and the obtained R values are shown in Table 3. It is clear from Table 3 that the obtained R value is not always greater than or equal to R_{max}^M . So, the next task is to find out an exact equation, which can replace Equation (8). In other words, an exact equation is expected in this work, which can find out R after putting M , where the value of R will be always greater than or equal to R_{max}^M without performing

the above analysis. However, this analysis helps us to propose a novel box height, which depends on image size. The expression of R in Equation (8) is approximated by $\log_2(M)$, which is shown in Equation (9). Initially we started with $R = \lceil \log_2 M \times (\log_2(M) - 3) \rceil$ because it is the nearest multiple of $\log_2(M)$ to satisfy for least image size $M = 128$ and its $R_{max}^{128} = 28$.

$$R = \lceil \log_2 M \times (\log_2(M) - 3) \rceil \quad (9)$$

Table 3. Approximating R for all image sizes.

Image Size (M)	R_{max}^M	R		
		$0.0589M + 19.667$	$\lceil \log_2 M \times (\log_2(M) - 3) \rceil$	$\lceil \log_2 M \times (\log_2(M) - 2) \rceil$
128	28	27.21 (X)	28 (✓)	35 (✓)
256	32	34.75 (✓)	40 (✓)	48 (✓)
512	49	49.82 (✓)	54 (✓)	63 (✓)
640	62	57.36 (X)	59 (X)	69 (✓)
1024	78	79.98 (✓)	70 (X)	80 (✓)

This expression is again validated using different values of M (128, 256, 512, 640 and 1024). Unfortunately, it fails for image size 640 and 1024 as shown in Table 3. Equation (9) has two parts and 1 has been added to the second part, $(\log_2(M) - 3)$. New formula is shown in Equation (10).

$$R = f(M) = \lceil \log_2(M) \times (\log_2(M) - 2) \rceil \quad (10)$$

Again, Equation (10) has been validated for all image size M (128, 256, 512, 640 and 1024) and this time Equation (10) works fine for all image sizes M . The ceil function, $\lceil \cdot \rceil$ is used to deal with the image sizes, M which are not a power of 2 like when $M = 640$ i.e., to obtain an integer value. Hence, Equation (10) is used to fix the value of R . Therefore using Equations (6) and (10), the new box height proposed in this paper for DBC [13] is as follows:

$$h = \frac{s}{\lceil \log_2(M) \times (\log_2(M) - 2) \rceil} \quad (11)$$

In Equation (11):

- If $\log_2(M) < 2 \implies R < 0 \implies h < 0$.
- If $\log_2(M) = 2 \implies R = 0 \implies h = \infty$.
- If $2 < \log_2(M) < 3 \implies R < \log_2(M) \implies h$ becomes relatively higher.

Hence when $\log_2(M) < 3$, we need to restrict R . Hence if $\log_2(M) < 3$ then $R = \log_2(M)$. Therefore the final proposed formula to compute box height is as follows:

$$h = \begin{cases} \frac{s}{\lceil \log_2(M) \times (\log_2(M) - 2) \rceil} & \text{if } \log_2(M) \geq 3 \\ \lceil \log_2(M) \rceil & \text{Otherwise} \end{cases} \quad (12)$$

Eleven generated FBM images as shown in Figure 2 have been used for the first set of experiment to validate the proposed box height as denoted in Equation (12). The FBM images are generated using random mid-point displacement algorithm with different Hurst components. The value of Hurst component varies from 0 to 1 with step size 0.1 in this work. So, we have altogether 11 Hurst values, which help to create eleven FBM images. Moreover, Hurst component defines the actual FD value of an image. The different Hurst values and their actual FD values are shown in 1st column and 2nd column of Table 4 respectively. The 3rd column of Table 4 depicts the FD values obtained using original DBC method. Table 1 shows different parameters of DBC, SDBC [15], IDBC [18] and ITDBC [22]. Now, the

proposed box height have been used and the achieved FD values for all the above mentioned images are shown in column number 4 of Table 4. The 5th and 6th columns of Table 4 show the difference of the FD values obtained by DBC method and new box height from actual FD values respectively. The DE values using both the methods are also reported in Table 4. It is clear from the columns 5th and 6th of Table 4 that the proposed box height allows to better estimate FD values, which are closer to actual FD values with less DE values. The same set of experiments using both the methods have been done on sixteen textured images from Brodatz Database as shown in Figure 1 and the results are shown in Table 5. We do not have actual FD values for these textured images. So, the actual FD values are missing in Table 5. However, the obtained FD values using the proposed box height of DBC are larger than the achieved FD values by the original DBC method. Thus, we can conclude that new box height of DBC estimates better FD values than the original DBC method.

Table 4. Computed FD and DE values using DBC and the proposed box height for generated FBM images.

FBM Images with		Fractal Dimension (FD)					Distance Error (DE)		
Hurst Parameter (H)	Actual FD (3-H) (P)	DBC (Q)	DBC + Proposed Box Height (R)	Absolute Difference in FD Values		Change in FD Values (R-Q)	DBC (X)	DBC + Proposed Box Height (Y)	Difference in DE Values (Y-X)
				$ P - Q $	$ P - R $				
1.0	2.0	1.9098	1.9857	0.0902	0.0143	0.0759	0.0045	0.0047	0.0002
0.9	2.1	1.9374	2.0248	0.1626	0.0752	0.0874	0.0042	0.0042	0.0000
0.8	2.2	1.9690	2.0757	0.2310	0.1243	0.1067	0.0042	0.0039	-0.0003
0.7	2.3	2.0141	2.1369	0.2859	0.1631	0.1228	0.0042	0.0036	-0.0006
0.6	2.4	2.0666	2.2056	0.3334	0.1944	0.1390	0.0041	0.0034	-0.0007
0.5	2.5	2.1237	2.2779	0.3763	0.2221	0.1542	0.0042	0.0033	-0.0009
0.4	2.6	2.1917	2.3514	0.4083	0.2486	0.1597	0.0041	0.0032	-0.0009
0.3	2.7	2.2575	2.4172	0.4425	0.2828	0.1597	0.0040	0.0032	-0.0008
0.2	2.8	2.3205	2.4742	0.4795	0.3258	0.1537	0.0038	0.0032	-0.0006
0.1	2.9	2.3785	2.5233	0.5215	0.3767	0.1448	0.0036	0.0032	-0.0004
0.0	3.0	2.4192	2.5612	0.5808	0.4388	0.1420	0.0037	0.0032	-0.0005

Table 5. Computed FD and DE values using DBC and the proposed box height for images from Brodatz Database as shown in Figure 1.

Images from Brodatz Database with Image Name	Fractal Dimension (FD)			Distance Error (DE)		
	DBC (P)	DBC + Proposed Box Height (Q)	Difference in FD Values (Q-P)	DBC (X)	DBC + Proposed Box Height (Y)	Difference in DE Values (Y-X)
D1	2.5281	2.6903	0.1622	0.0046	0.0035	-0.0011
D5	2.5740	2.7375	0.1635	0.0048	0.0036	-0.0012
D15	2.6188	2.7823	0.1635	0.0049	0.0034	-0.0015
D20	2.6154	2.7793	0.1639	0.0050	0.0035	-0.0015
D31	2.4877	2.6486	0.1609	0.0045	0.0034	-0.0011
D37	2.4882	2.6521	0.1639	0.0040	0.0031	-0.0009
D48	2.3346	2.4874	0.1528	0.0045	0.0033	-0.0012
D54	2.5839	2.7471	0.1632	0.0049	0.0035	-0.0014
D59	2.3496	2.5028	0.1532	0.0037	0.0025	-0.0012
D62	2.5164	2.6773	0.1609	0.0047	0.0034	-0.0013
D67	2.6172	2.7810	0.1638	0.0050	0.0035	-0.0015
D74	2.5878	2.7508	0.1630	0.0049	0.0036	-0.0013
D86	2.5868	2.7496	0.1628	0.0048	0.0034	-0.0014
D92	2.5929	2.7569	0.1640	0.0049	0.0034	-0.0015
D102	2.6090	2.7728	0.1638	0.0050	0.0035	-0.0015
D112	2.5554	2.7185	0.1631	0.0047	0.0035	-0.0012

In the next set of experiment, original SDBC [15] and SDBC with proposed box height have been considered on eleven generated FBM images and sixteen textured images from Brodatz Database and the obtained results are shown in Tables 6 and 7 respectively.

Table 6. Computed FD and DE values using SDBC and the proposed box height for generated FBM images.

FBM Images with		Fractal Dimension (FD)				Distance Error (DE)			
Hurst Parameter (H)	Actual FD (3-H) (P)	SDBC (Q)	SDBC + Proposed Box Height (R)	Absolute Difference in FD Values		Change in FD Values (R-Q)	SDBC (X)	SDBC + Proposed Box Height (Y)	Difference in DE Values (Y-X)
				$ P - Q $	$ P - R $				
1.0	2.0	1.9364	2.0272	0.0636	0.0272	0.0908	0.0038	0.0042	0.0004
0.9	2.1	1.9595	2.0605	0.1405	0.0395	0.1010	0.0038	0.0039	0.0001
0.8	2.2	2.0037	2.1056	0.1963	0.0944	0.1019	0.0039	0.0037	−0.0002
0.7	2.3	2.0675	2.1611	0.2325	0.1389	0.0936	0.0041	0.0035	−0.0006
0.6	2.4	2.1376	2.2251	0.2624	0.1749	0.0875	0.0040	0.0033	−0.0007
0.5	2.5	2.2078	2.2935	0.2922	0.2065	0.0857	0.0038	0.0032	−0.0006
0.4	2.6	2.2767	2.3640	0.3233	0.2360	0.0873	0.0036	0.0031	−0.0005
0.3	2.7	2.3408	2.4273	0.3592	0.2727	0.0865	0.0037	0.0031	−0.0006
0.2	2.8	2.3951	2.4828	0.4049	0.3172	0.0877	0.0037	0.0032	−0.0005
0.1	2.9	2.4347	2.5306	0.4653	0.3694	0.0959	0.0037	0.0032	−0.0005
0.0	3.0	2.4679	2.5678	0.5321	0.4322	0.0999	0.0034	0.0032	−0.0002

Table 7. Computed FD and DE values using SDBC and the proposed box height for images from Brodatz Database as shown in Figure 1.

Images from Brodatz Database with Image Name	Fractal Dimension (FD)			Distance Error (DE)		
	SDBC (P)	SDBC + Proposed Box Height (Q)	Difference in FD Values (Q-P)	SDBC (X)	SDBC + Proposed Box Height (Y)	Difference in DE Values (Y-X)
D1	2.6238	2.6958	0.0720	0.0046	0.0035	−0.0011
D5	2.6693	2.7427	0.0734	0.0048	0.0035	−0.0013
D15	2.7138	2.7871	0.0733	0.0047	0.0034	−0.0013
D20	2.7100	2.7841	0.0741	0.0048	0.0034	−0.0014
D31	2.5828	2.6544	0.0716	0.0046	0.0034	−0.0012
D37	2.5896	2.6583	0.0687	0.0037	0.0031	−0.0006
D48	2.4281	2.4944	0.0663	0.0043	0.0033	−0.0010
D54	2.6790	2.7522	0.0732	0.0048	0.0035	−0.0013
D59	2.4478	2.5101	0.0623	0.0033	0.0025	−0.0008
D62	2.6102	2.6829	0.0727	0.0047	0.0034	−0.0013
D67	2.7115	2.7859	0.0744	0.0048	0.0034	−0.0014
D74	2.6822	2.7558	0.0736	0.0048	0.0036	−0.0012
D86	2.6817	2.7547	0.0730	0.0047	0.0034	−0.0013
D92	2.6880	2.7619	0.0739	0.0048	0.0034	−0.0014
D102	2.7034	2.7777	0.0743	0.0048	0.0035	−0.0013
D112	2.6516	2.7238	0.0722	0.0047	0.0035	−0.0012

Then IDBC [18] method and the proposed box height have been used for the next set of experiments on eleven generated FBM images and sixteen textured images from Brodatz Database and the achieved results are shown in Tables 8 and 9 respectively.

Table 8. Computed FD and DE values using IDBC and the proposed box height for generated FBM images.

FBM Images with		Fractal Dimension (FD)				Distance Error (DE)			
Hurst Parameter (H)	Actual FD (3-H) (P)	IDBC (Q)	IDBC + Proposed Box Height (R)	Absolute Difference in FD Values		Change in FD Values (R-Q)	IDBC (X)	IDBC + Proposed Box Height (Y)	Difference in DE Values (Y-X)
				$ P - Q $	$ P - R $				
1.0	2.0	2.0921	2.1543	0.0921	0.1543	0.0622	0.0114	0.0062	−0.0052
0.9	2.1	2.1222	2.1891	0.0222	0.0891	0.0669	0.0155	0.0073	−0.0082
0.8	2.2	2.1744	2.2352	0.0256	0.0352	0.0608	0.0163	0.0095	−0.0068
0.7	2.3	2.2420	2.2896	0.0580	0.0104	0.0476	0.0174	0.0117	−0.0057
0.6	2.4	2.2717	2.3496	0.1283	0.0504	0.0779	0.0111	0.0137	0.0026
0.5	2.5	2.3462	2.4099	0.1538	0.0901	0.0637	0.0134	0.0152	0.0018
0.4	2.6	2.3986	2.4711	0.2014	0.1289	0.0725	0.0133	0.0167	0.0034
0.3	2.7	2.4654	2.5232	0.2346	0.1768	0.0578	0.0147	0.0174	0.0027
0.2	2.8	2.5182	2.5699	0.2818	0.2301	0.0517	0.0151	0.0180	0.0029
0.1	2.9	2.5712	2.6080	0.3288	0.2920	0.0368	0.0165	0.0182	0.0017
0.0	3.0	2.6091	2.6392	0.3909	0.3608	0.0301	0.0171	0.0181	0.0010

Table 9. Computed FD and DE values using IDBC and the proposed box height for images from Brodatz Database as shown in Figure 1.

Images from Brodatz Database with Image Name	Fractal Dimension (FD)			Distance Error (DE)		
	IDBC (P)	IDBC + Proposed Box Height (Q)	Difference in FD Values (Q-P)	IDBC (X)	IDBC + Proposed Box Height (Y)	Difference in DE Values (Y-X)
D1	2.7045	2.7051	0.0006	0.0290	0.0282	−0.0008
D5	2.7618	2.7625	0.0007	0.0301	0.0295	−0.0006
D15	2.8487	2.8495	0.0008	0.0235	0.0231	−0.0004
D20	2.8359	2.8370	0.0011	0.0271	0.0267	−0.0004
D31	2.6590	2.6594	0.0004	0.0285	0.0275	−0.0010
D37	2.6898	2.7124	0.0226	0.0209	0.0209	0.0000
D48	2.5668	2.5882	0.0214	0.0171	0.0148	−0.0023
D54	2.7797	2.7804	0.0007	0.0291	0.0285	−0.0006
D59	2.6008	2.6121	0.0113	0.0122	0.0088	−0.0034
D62	2.6973	2.6979	0.0006	0.0289	0.0280	−0.0009
D67	2.8378	2.8389	0.0011	0.0264	0.0261	−0.0003
D74	2.8378	2.8389	0.0011	0.0264	0.0261	−0.0003
D86	2.8378	2.8389	0.0011	0.0264	0.0261	−0.0003
D92	2.8378	2.8389	0.0011	0.0264	0.0261	−0.0003
D102	2.8378	2.8389	0.0011	0.0264	0.0261	−0.0003
D112	2.8378	2.8389	0.0011	0.0264	0.0261	−0.0003

Again, similar experiments using ITDBC method [22] and proposed box height have been considered on both set of images i.e., on generated FBM, textured images and the gained results are reported in Tables 10 and 11 respectively.

To know the impact of the new box height for different methods such as DBC, SDBC, IDBC and ITDBC, we have also plotted the results of these methods obtained using their original parameters and the results obtained with the proposed box height as shown in Figure 9. While plotting, the actual FD values are represented in x -axis and the corresponding computed FD values in y -axis. It is clear from Figure 9 that for all the cases the proposed box height allows the four existing methods to compute better FD values compare to their original methods. Moreover, we can conclude from Figure 9 that the proposed box height together with ITDBC estimate better FD values for maximum set of experiments.

We have also plotted a similar graph for Brodatz Database in Figure 10. In x -axis of Figure 10 represents image number and y -axis denotes computed FD values using all the above mentioned

methods. We can easily determine that the proposed box height along with ITDBC outperforms than other methods shown in Figure 10.

FD is the main factor that indicates image roughness. DE is merely a secondary component of FD values and if less DE is generated than it is good but slightly high DE is acceptable and that does not mean that the method is entirely bad. The main priority lies in better estimating FD values, Hence we relaxed DE value while defining proposed box height i.e., not all DE values generated by proposed box height for DBC, SDBC, IDBC and ITDBC are less when compared with their original box heights as shown in last columns of Tables 8–11.

Table 10. Computed FD and DE values using ITDBC and the proposed box height for generated FBM images.

FBM Images with		Fractal Dimension (FD)				Distance Error (DE)			
Hurst Parameter (H)	Actual FD (3-H) (P)	ITDBC (Q)	ITDBC + Proposed Box Height (R)	Absolute Difference in FD Values		Change in FD Values (R-Q)	ITDBC (X)	ITDBC + Proposed Box Height (Y)	Difference in DE Values (Y-X)
				$ P - Q $	$ P - R $				
1.0	2.0	2.0811	2.1286	0.0811	0.1286	0.0475	0.0027	0.0022	−0.0005
0.9	2.1	2.1232	2.1786	0.0232	0.0786	0.0554	0.0027	0.0021	−0.0006
0.8	2.2	2.1659	2.2395	0.0341	0.0395	0.0736	0.0021	0.0019	−0.0002
0.7	2.3	2.2094	2.3063	0.0906	0.0063	0.0969	0.0018	0.0018	0.0000
0.6	2.4	2.2457	2.3759	0.1543	0.0241	0.1302	0.0014	0.0017	0.0003
0.5	2.5	2.2988	2.4446	0.2012	0.0554	0.1458	0.0012	0.0017	0.0005
0.4	2.6	2.3582	2.5097	0.2418	0.0903	0.1515	0.0014	0.0017	0.0003
0.3	2.7	2.4257	2.5672	0.2743	0.1328	0.1415	0.0015	0.0017	0.0002
0.2	2.8	2.4890	2.6159	0.3110	0.1841	0.1269	0.0015	0.0017	0.0002
0.1	2.9	2.5494	2.6570	0.3506	0.2430	0.1076	0.0016	0.0017	0.0001
0.0	3.0	2.5903	2.6904	0.4097	0.3096	0.1001	0.0016	0.0017	0.0001

Table 11. Computed FD and DE values using ITDBC and the proposed box height for images from Brodatz Database as shown in Figure 1.

Images from Brodatz Database with Image Name	Fractal Dimension (FD)			Distance Error (DE)		
	ITDBC (P)	ITDBC + Proposed Box Height (Q)	Difference in FD Values (Q-P)	ITDBC (X)	ITDBC + Proposed Box Height (Y)	Difference in DE Values (Y-X)
D1	2.6618	2.8231	0.1613	0.0024	0.0030	0.0006
D5	2.7075	2.8705	0.1630	0.0023	0.0026	0.0003
D15	2.7600	2.9236	0.1636	0.0021	0.0020	−0.0001
D20	2.7602	2.9232	0.1630	0.0022	0.0022	0.0000
D31	2.6176	2.7765	0.1589	0.0024	0.0032	0.0008
D37	2.6186	2.7808	0.1622	0.0015	0.0020	0.0005
D48	2.4861	2.6272	0.1411	0.0019	0.0025	0.0006
D54	2.7183	2.8812	0.1629	0.0023	0.0025	0.0002
D59	2.4726	2.6228	0.1502	0.0008	0.0014	0.0006
D62	2.6520	2.8118	0.1598	0.0023	0.0028	0.0005
D67	2.7603	2.9234	0.1631	0.0022	0.0021	−0.0001
D74	2.7603	2.9234	0.1631	0.0022	0.0021	−0.0001
D86	2.7603	2.9234	0.1631	0.0022	0.0021	−0.0001
D92	2.7603	2.9234	0.1631	0.0022	0.0021	−0.0001
D102	2.7603	2.9234	0.1631	0.0022	0.0021	−0.0001
D112	2.7603	2.9234	0.1631	0.0022	0.0021	−0.0001

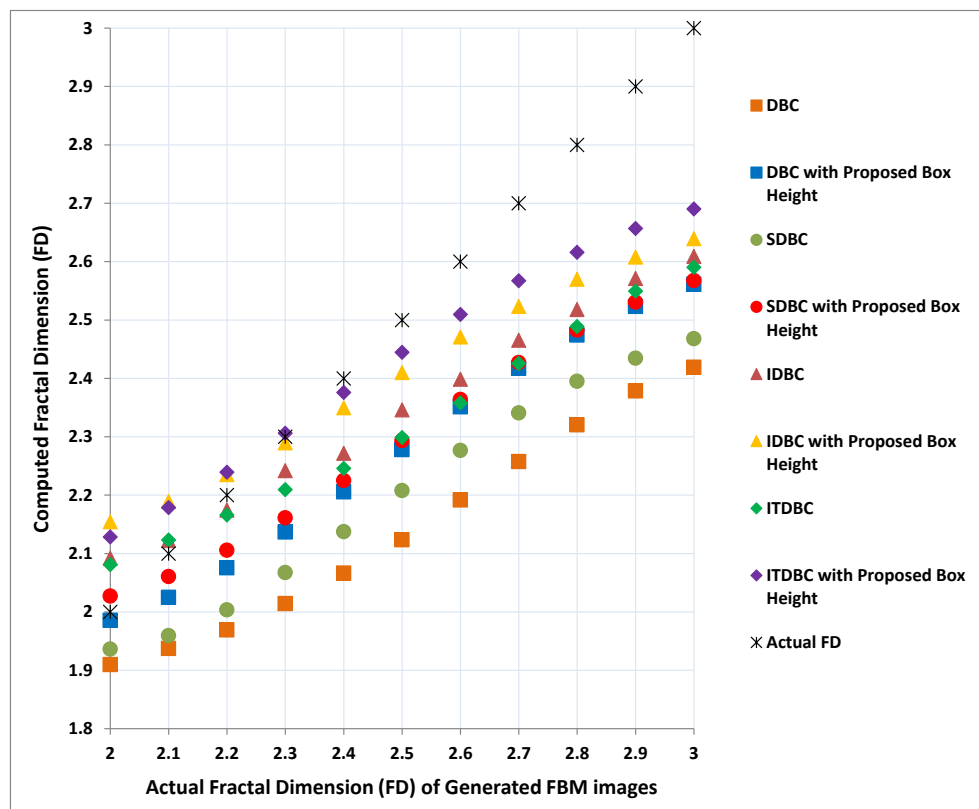


Figure 9. Computed and actual FD values for generated FBM images using DBC, SDBC, IDBC, ITDBC along with the original and the proposed box height.

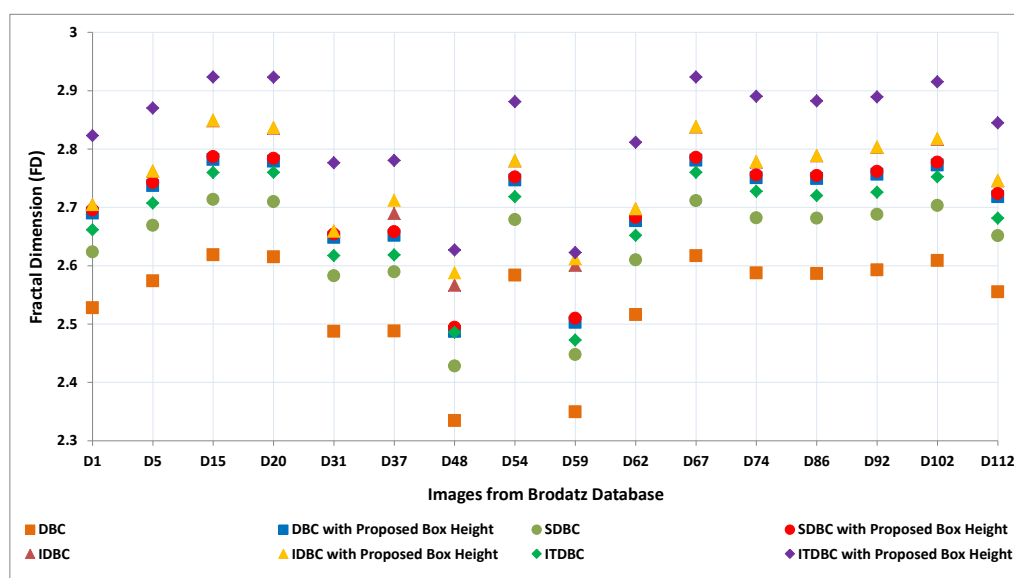


Figure 10. Computed FD values for images of Brodatz Database as shown in Figure 1 using DBC, SDBC, IDBC, ITDBC along with the original and the proposed box height.

4. Conclusions

In this work, the impact of box height in estimating FD values using DBC method has been studied thoroughly. It is clear from the literature that a larger box height is unable to compute FD value correctly. In other words, a lower box height produces better FD value with relatively less error. Thus, an equation

for a new box height is proposed in this work based on 112 textured images of Brodatz Database. However, the new box height depends on image size and it generates relatively smaller box height compare to some of the existing methods in literature. The new box height is also validated on generated FBM images, which have been created using random mid-point displacement algorithm.

Several experiments have been performed on generated FBM images and images from Brodatz Database using DBC, SDBC, IDBC and ITDBC along with proposed box height. It was observed from the obtained results that the proposed box height allows to estimate better FD values with less errors in most of the cases with DBC, SDBC, IDBC and ITDBC methods. Moreover, it was observed that the proposed box height with ITDBC method outperforms other methods for all but only two FBM tested images. There are still some aspects of different parameters of DBC method that deserve further study. We have tested our proposed algorithm on gray-scale images only. So, we would like to work on color images. In future works, the behavior of the proposed method will be studied. We would also like to focus in the application areas of FD such as image segmentation, object recognition.

Acknowledgments: Chinmaya Panigrahy would like to thank Erasmus+ Programme (KA107) for financial support through project 2015-1-ES01-KA107-015460, which makes this paper possible. This work has been partially funded by the Agencia Estatal de Investigación (AEI) of Spain and the European Regional Development Fund (ERDF) through the project ARTEMISat-2: Advanced Processing of Remote Sensing Data for Monitoring and Sustainable Management of Marine and Terrestrial Resources in Vulnerable Ecosystems (CTM2016-77733-R). Ayan Seal is grateful to Media Lab Asia, Ministry of Electronics and Information Technology, Government of India for providing him young faculty research fellowship. Finally, the authors would like to thank the anonymous reviewers for their constructive advice.

Author Contributions: Authors make substantial contributions such as data collection, conception, design, analysis and interpretation of data to the work reported in this manuscript. Authors participate in initial drafting of article and revising it critically for important intellectual content. Authors also give final approval of the version to be submitted.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mandelbrot, B.B. *The Fractal Geometry of Nature/Revised and Enlarged Edition*; WH Freeman and Co.: New York, NY, USA, 1983; Volume 495, p. 1.
2. Chaudhuri, B.B.; Sarkar, N. Texture segmentation using fractal dimension. *IEEE Trans. Pattern Anal. Mach. Intell.* **1995**, *17*, 72–77.
3. Neil, G.; Curtis, K.M. Shape recognition using fractal geometry. *Pattern Recognit.* **1997**, *30*, 1957–1969.
4. Hong, D.; Pan, Z.; Wu, X. Improved differential box counting with multi-scale and multi-direction: A new palmprint recognition method. *Opt. Int. J. Light Electron Opt.* **2014**, *125*, 4154–4160.
5. Cervantes-De la Torre, F.; González-Trejo, J.; Real-Ramírez, C.; Hoyos-Reyes, L. Fractal dimension algorithms and their application to time series associated with natural phenomena. *J. Phys.* **2013**, *475*, 012002.
6. Yu, L.; Zhang, D.; Wang, K.; Yang, W. Coarse iris classification using box-counting to estimate fractal dimensions. *Pattern Recognit.* **2005**, *38*, 1791–1798.
7. Lillo-Saavedra, M.; Gonzalo, C.; Lagos, O. Toward reduction of artifacts in fused images. *Int. J. Appl. Earth Obs. Geoinf.* **2011**, *13*, 368–375.
8. Edrees, M. Application of fractal dimension. *Iraqi J. Stat. Sci.* **2006**, *10*, 54–73.
9. Peleg, S.; Naor, J.; Hartley, R.; Avnir, D. Multiple resolution texture analysis and classification. *IEEE Trans. Pattern Anal. Mach. Intell.* **1984**, *6*, 518–523.
10. Pentland, A.P. Fractal-based description of natural scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* **1984**, *6*, 661–674.
11. Keller, J.M.; Chen, S.; Crownover, R.M. Texture description and segmentation through fractal geometry. *Comput. Vis. Graph. Image Process.* **1989**, *45*, 150–166.
12. Gagnepain, J.; Roques-Carmes, C. Fractal approach to two-dimensional and three-dimensional surface roughness. *Wear* **1986**, *109*, 119–126.
13. Sarkar, N.; Chaudhuri, B. An efficient differential box-counting approach to compute fractal dimension of image. *IEEE Trans. Syst. Man Cybern.* **1994**, *24*, 115–120.

14. Jin, X.; Ong, S.; Jayasooriah. A practical method for estimating fractal dimension. *Pattern Recognit. Lett.* **1995**, *16*, 457–464.
15. Chen, W.S.; Yuan, S.Y.; Hsieh, C.M. Two algorithms to estimate fractal dimension of gray-level images. *Opt. Eng.* **2003**, *42*, 2452–2464.
16. Li, J.; Sun, C.; Du, Q. A new box-counting method for estimation of image fractal dimension. In Proceedings of the 2006 IEEE International Conference on Image, Atlanta, GA, USA, 8–11 October 2006; pp. 3029–3032.
17. Li, J.; Du, Q.; Sun, C. An improved box-counting method for image fractal dimension estimation. *Pattern Recognit.* **2009**, *42*, 2460–2469.
18. Liu, Y.; Chen, L.; Wang, H.; Jiang, L.; Zhang, Y.; Zhao, J.; Wang, D.; Zhao, Y.; Song, Y. An improved differential box-counting method to estimate fractal dimensions of gray-level images. *J. Vis. Commun. Image Represent.* **2014**, *25*, 1102–1111.
19. Kaewaramsri, Y.; Woraratpanya, K. Improved Triangle Box-Counting Method for Fractal Dimension Estimation. In *Recent Advances in Information and Communication Technology 2015*; Springer: New York, NY, USA, 2015; pp. 53–61.
20. Liu, S. An improved differential box-counting approach to compute fractal dimension of gray-level image. In Proceedings of the IEEE International Symposium on Information Science and Engineering (ISISE'08), Shanghai, China, 20–22 December 2008; Volume 1, pp. 303–306.
21. Long, M.; Peng, F. A box-counting method with adaptable box height for measuring the fractal feature of images. *Radioengineering* **2013**, *22*, 208–213.
22. Nunsong, W.; Woraratpanya, K. Modified differential box-counting method using weighted triangle-box partition. In Proceedings of the IEEE 7th International Conference on Information Technology and Electrical Engineering (ICITEE), Chiang Mai, Thailand, 29–30 October 2015; pp. 221–226.
23. Lai, K.; Li, C.; He, T.; Chen, L.; Yu, K.; Zhou, W. Study on an improved differential box-counting approach for gray-level variation of images. In Proceedings of the IEEE 10th International Conference on Sensing Technology (ICST), Nanjing, China, 11–13 November 2016; pp. 1–6.
24. Original Brodatz Texture—Universite de Sherbrooke. Available online: http://multibandtexture.recherche.usherbrooke.ca/original_brodatz.html (accessed on 7 October 2017).
25. Saupe, D. Algorithms for random fractals. In *The Science of Fractal Images*; Springer: New York, NY, USA, 1988; pp. 71–136.



© 2017 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).