

# A Box-Counting Method with Adaptable Box Height for Measuring the Fractal Feature of Images

Min LONG<sup>1</sup>, Fei PENG<sup>2</sup>

<sup>1</sup> College of Computer and Communication Engineering, ChangSha University of Science and Technology, 410004, Changsha, People's Republic of China

<sup>2</sup> School of Information Science and Engineering, Hunan University, 410082, ChangSha, People's Republic of China

longm@tom.com, pengfei@hnu.edu.cn

**Abstract.** Most of the existing box-counting methods for measuring fractal features are only applicable to square images or images with each dimension equal to the power of 2 and require that the box at the top of the box stack of each image block is of the same height as that of other boxes in the same stack, which gives rise to inaccurate estimation of fractal dimension. In this paper, we propose a more accurate box-counting method for images of arbitrary size, which allows the height of the box at the top of each grid block to be adaptable to the maximum and minimum gray-scales of that block so as to circumvent the common limitations of existing box-counting methods.

## Keywords

Fractal dimension, image feature, fractal features, box-counting method.

## 1. Introduction

A fractal is generally a rough or approximate geometric shape that can be broken down into smaller pieces; each is similar to the original. Mandelbrot suggested [1] that, given a bounded set  $I$ , in an Euclidean space, the set is self-similar if  $I$  is the union of  $N_r$  different non-overlapping duplicates of  $I$ , each of which is similar to  $I$  scaled down by a ratio of  $r$  of  $I$ . In fractal geometry [1], the fractal dimension,  $D$ ,

$$D = \lim_{r \rightarrow 0} \frac{\log(N_r)}{\log(1/r)} \quad (1)$$

is a statistical quantity that indicates how completely a fractal fills the space when viewed at finer scales. It is an effective index to measure the feature of complex objects and surfaces, such as coastlines, mountain, clouds and texture.

Pentland has proved that the image of a fractal object is also a fractal [2], which has popularized the research on the methods of estimating the fractal dimensions of images. Since the establishment of fractal geometry theory, many researchers have put great efforts into this field, and many

methods for estimating fractal dimensions of certain objects have been proposed. Typical methods include spectral analysis and box-counting. Spectral analysis method generally implements FFT (Fast Fourier Transformation) to image and obtain the coefficients and mean spectral energy density. The fractal dimension can be estimated by analyzing the power-law dependence of spectral energy density and the square size [3]. Box-counting is one of the most frequently used methods for determining fractal dimension, which considers, if the 3-D space containing a specific object is partitioned into boxes of a certain size, how many boxes could fill up the object. Using the ratio  $r$  in (1) to decide the box size, the task of box counting method is to count the total number of boxes (i.e.,  $N_r$  of (1)) that are needed to form the object. Then counting  $N_r$  for different scaling ratio  $r$ , the fractal dimension of  $D$  of (1) can be estimated from the least square linear fit of  $\log(N_r)$  versus  $\log(1/r)$ .

Many traditional box-counting methods have been proposed for the calculation of the fractal dimensions of images, such as the reticular cell counting method [4], Keller's approach [5], differential box-counting (DBC) method [6], Feng's method [7], etc. Among these box-counting methods, DBC has been proved to have better performance than the rest [8]. After the publication of [6], many analyses and improvements have been done to DBC methods [9-12]. Nevertheless, only the calculation of fractal dimensions of images of  $M \times M$  and  $2^m \times 2^n$  pixels are considered in the above methods. How to calculate the fractal dimensions of images with arbitrary size of  $M \times N$  pixels (i.e.  $M$  is not necessarily equal to  $N$ ) is a challenge facing the existing methods. Moreover, these methods require that the box at the top of the box stack of each image block is of the same height as that of other boxes in the same stack, which gives rise to inaccurate estimation of fractal dimension.

This paper aims to propose a more accurate box-counting method, which is applicable to images with arbitrary sizes and improves the accuracy of fractal dimension estimation by allowing the height of the box at the top of each grid block to be locally adaptable to the maximum and minimum gray-scales of that block. The remainder of the paper is organized as follows. A traditional DBC

method and its improved versions are introduced in Section 2. In Section 3, the proposed box-counting method is described in detail. The bound of the box size of the proposed method is discussed in Section 4. Experimental results and the performance analysis of various methods are conducted in Section 5. Finally, conclusions are drawn in Section 6.

## 2. Differential Box-Counting (DBC) Methods and Their Limitations

Given a square image  $I$  of  $M \times M$  pixels. We can superpose on top of it a grid of blocks of  $s \times s$  pixels and place a stack of boxes of size  $s \times s \times p$  on top of each block, where  $p = s \times \Lambda / M$ , and  $\Lambda$  is the total number of gray-scales (e.g.,  $\Lambda = 256$  if 8 bits are used to represent the intensity/gray-scale of each pixel). The boxes in each stack are indexed in the ascending order, starting from 1 at the bottom. Differential Box-Counting (DBC) methods [6], [9], with  $r = s / M$ , count the number of boxes on top of the  $(i, j)$ th block by using

$$n_r(i, j) = l - k + 1 \tag{2}$$

where  $l$  and  $k$  are the  $l$ -th and  $k$ -th boxes where the maximum and the minimum gray-scales of the  $(i, j)$ th square fall in, respectively, when the image is seen as a 3-D landscape with the vertical dimension representing the gray-scales.

Taking contributions from all blocks, we can say that the *volume* of the image, a 3-D landscape, is equivalent to  $N_r$  boxes, where

$$N_r = \sum_{i,j} n_r(i, j). \tag{3}$$

Since the values of  $N_r$  are dependent on  $r$ , the fractal dimension of image  $I$  can be estimated from the least square linear fit of  $\log(N_r)$  and  $\log(1/r)$ , and the slope of the fit line will be  $D$ .

Paper [6] indicated that the bound of box size  $s$  is  $2 \leq s \leq M / 2$ , while paper [9] found that the number of boxes may be over counted if  $\Lambda / p$  is greater than  $s \times s$  and suggested the bound of the box size  $s$  should be

$$\sqrt[3]{M} \leq s \leq M / 2.$$

Moreover, some improvements [10], [11], [12] have also been done to the DBC methods. In [10], unlike those proposed in [6] and [9], a common border is shared between adjacent boxes and the height of the box is

$$p(i, j) = \begin{cases} \frac{s}{0.5[\max I(i, j) - \min I(i, j)]}, & \text{if } \max I(i, j) \neq \min I(i, j) \\ 1 & \text{if } \max I(i, j) = \min I(i, j) \end{cases} \tag{4}$$

where  $\max I(i, j)$  and  $\min I(i, j)$  are the highest and lowest gray-scale appearing in the  $(i, j)$ th grid block.

The number of the boxes in the  $(i, j)$ th block is

$$p(i, j) = \begin{cases} \frac{\max I(i, j) - \min I(i, j)}{p(i, j)}, & \text{if } \max I(i, j) \neq \min I(i, j) \\ 1 & \text{if } \max I(i, j) = \min I(i, j) \end{cases} \tag{5}$$

By performing the ceiling function in (5), no matter how insignificant the remainder of  $(\max I(i, j) - \min I(i, j)) / p(i, j)$  is, a box of the same height as others has to be placed at the top of the box stack of each image block to contain the remainder, which gives rise to inaccurate estimation of fractal dimension.

An approach for estimating the fractal of corrosion images with sizes of  $2^m \times 2^n$  is proposed in [11]. With an image seen as a 3-D space, the image can be partitioned into  $r \times r \times r$  boxes, with each side length determined by the image dimensions and the range of gray-scales, where  $r = 2, 4, 8, \dots, 2^k$ . The number of the boxes on top of the  $(i, j)$ th block is

$$n_r(i, j) = \text{card}\{I(i, j)^l \mid \overline{H(I(i, j)^l)} < \overline{H(I)} \mid l = 1, 2, \dots, r\} \tag{6}$$

where  $I(i, j)^l$  represents the  $l$ -th box in the  $(i, j)$ -th block,  $\overline{H(I(i, j)^l)}$  represents the average gray-scale of the pixels that fall in the  $I(i, j)^l$ , and  $\overline{H(I)}$  represents the average gray-scale of the whole image,  $\text{card}\{\cdot\}$  is the function which returns the cardinality (i.e., the size of a set). The condition,  $\overline{H(I(i, j)^l)} < \overline{H(I)}$ , decides whether or not to add a box of the same height as others at the top of the box stack of each image block without taking into account the real “volume” that is actually needed. Again this introduces inaccuracy into the estimation of fractal dimension.

Another improved box-counting method for image fractal dimension estimation is proposed in paper [12]. This method is only applicable to square images of  $M \times M$  pixels. Similar to [10], a common border is shared between adjacent boxes and the height of the box is

$$p = \frac{s - 1}{1 + 2\alpha\sigma} \tag{7}$$

where  $s$  is the side lengths of the box,  $\alpha$  ( $=3$ ) is a parameter of the method, and  $\sigma$  is the standard deviation of the image. The number of the boxes in the  $(i, j)$ th square is calculated using (5) as well.

Results reported in [10], [11], [12] indicate that, compared to (2), equations (4), (5) and (7) provide more accurate counts of the boxes, thus help to measure the self-similarity of the images more accurately. However, in summary, they still have some inherent limitations.

- The methods proposed in [6], [9], [10], [12] are only applicable to square images of  $M \times M$  pixels, while

the method of [11] is only suitable for images of  $2^m \times 2^n$  pixels, which is shown in Fig. 1.

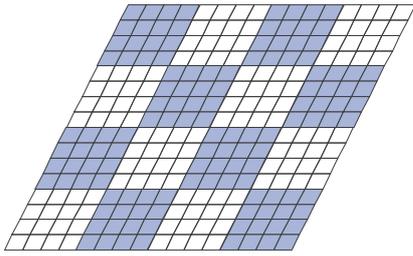


Fig. 1. Illustration of image partition.

- Using a box of the same height as that of other at the top the box stack introduces estimation error.
- The methods of [10] and [12] allow adjacent blocks to share common borders. As a result, the pixels in the common borders contribute twice in the calculation of the number of boxes, which introduce more estimation errors.
- As for the method in [11], the number of boxes is solely decided by a global threshold  $\overline{H(I)}$ , which ignores the local characteristics within each grid block. This will inevitably inflict negative effect on the accuracy of the box counting.

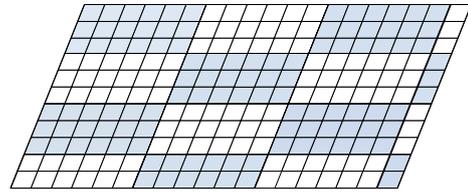
### 3. The Integer Ratio Based Box-Counting Method

To circumvent the afore-mentioned limitations, we propose an accurate box-counting method with adaptable box height for measuring the fractal dimension of Images. Given an image  $I$  of  $M \times N$  pixels, for any ratio  $r$ , ( $r \geq 2, r \in \mathbb{Z}^+$ ), we superpose a grid of blocks of  $m \times n$  pixels, where  $m = M/r, n = N/r$ . There are four situations to deal with when superposing the grid,

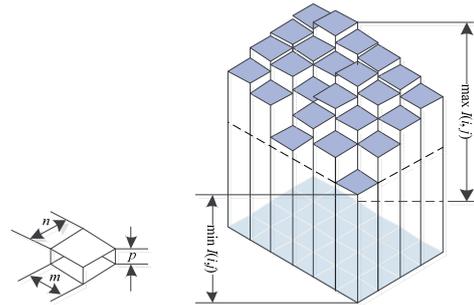
- $M = mr$  and  $N = nr$ : The image plane is evenly partitioned into  $r \times r$  blocks of  $m \times n$  pixels;
- $M = mr$  and  $N > nr$ : The image plane is partitioned into  $r \times (r + 1)$  blocks. Among these blocks, there are  $r \times r$  blocks of  $m \times n$  pixels and  $r \times 1$  blocks of  $m \times (N - nr)$  pixels;
- $M > mr, N = nr$ : The image plane is partitioned into  $(r + 1) \times r$  blocks. Among these blocks, there are  $r \times r$  blocks of  $m \times n$  pixels and  $1 \times r$  blocks of  $(M - mr) \times n$  pixels;
- $M > mr, N > nr$ : The image plane is partitioned into  $(r + 1) \times (r + 1)$  blocks. Among these blocks, there are  $r \times r$  blocks of  $m \times n$  pixels,  $r \times 1$  blocks of  $m \times (N - nr)$  pixels,  $1 \times r$  blocks of  $(M - mr) \times n$  pixels and 1 block of  $(M - mr) \times (N - nr)$ .

For example, the size of the image plane in Fig. 2(a) is  $11 \times 19$  pixels. Let the ratio  $r = 3$ , then we can get  $m = 3$ ,

$n = 6$ , which confirms to the fourth situation. So we can partition the image plane into  $3 \times 3$  blocks of  $3 \times 6$  pixels,  $3 \times 1$  blocks of  $3 \times 1$  pixels,  $1 \times 3$  blocks of  $2 \times 6$  pixels and 1 block of  $1 \times 2$ , as illustrated in Fig. 2 (a).



(a) Illustration of partition of an image plane.



(b) Illustration of a box (c) Illustration of a grid

Fig. 2. Partition of an image and box counting.

After partitioning the image plane, we treat the image as a 3-D landscape, with the vertical dimension representing the gray-scale and count the number of boxes in each stack based on each block. The height of the box is  $p = A/r$ , where  $A$  is the total number of gray-scales. In order to contain the image tightly and accurately, we allow the number of the boxes to be real, rather than integer, i.e., we allow the height of box at the top of each block to be a fraction of the height of the boxes below it, as seen in Fig. 2(b) and Fig. 2(c).

The number of boxes in the  $(i, j)$ -th block is counted as

$$n_r(i, j) = (\max I(i, j) - \min I(i, j)) / p + 1 \times \frac{S(i, j)}{mn}, \quad (8)$$

where  $S(i, j)$  is the area of the  $(i, j)$ -th, which for most blocks is  $m \times n$ , but for the blocks along the borders on the right-hand side and bottom of images whose plane dimensions are not power of 2,  $S(i, j)$  is less than  $m \times n$ , as can be seen in Fig. 2. Moreover, unlike (5), we do not perform the ceiling function in (8). This allows  $n_r$  to take real values rather than just integer values. That is to say that the proposed method is able to adaptively place box of arbitrary height at the top of the box stack according to the values of  $\max I(i, j)$  and  $\min I(i, j)$ , thus improving the estimation accuracy. The total number of boxes of the entire image/landscape with a ratio of  $r$  is:

$$N_r = \sum_{i,j} n_r(i, j). \quad (9)$$

Calculating various values of  $N_r$  for different values of  $r$ , the fractal dimension  $D$  can be calculated by finding

the slopes of a log-log curve at a series of points in the  $(\log(1/r), \log(N_r))$  space.

### 4. Bound of the Box Size

According to the principle described in [9], the number of pixels covered by a particular grid block should be no less than the maximum number of the boxes on top of that block. This principle also applies to the proposed method, i.e.,

$$m \times n = M / r \times N / r \geq r. \tag{10}$$

Approximately, we get  $r \leq \sqrt[3]{MN}$ . Meanwhile,  $m$  and  $n$  should be large enough to form blocks, thus

$$m \times n = M / r \geq 1 \text{ and } n = N / r \geq 1, \tag{11}$$

i.e.,  $r \leq M$  and  $r \leq N$ .

From (10) and (11), we can get the upper limit of the box size  $Q$  according to

$$Q = \min\{\sqrt[3]{MN}, M, N\}. \tag{12}$$

Moreover, to avoid the situation where there is only one block in the entire image,  $r$  should be chosen such that  $r \geq 2$ . Thus  $2 \leq r \leq Q$  is the closed range of the box size when the proposed method is used.

### 5. Experimental Results and Analyses

In this section, experiments and analyses are done to evaluate the performance of the proposed method. The experiments are carried out on a system with a 1.8G CPU, a RAM of 1 GBytes and Matlab V7.0 (R14). Color images are converted to the gray-scale format before the box-counting methods are applied.

As indicated in [9], theoretically the smoother an image is, the closer its fractal dimension is to 2. On the contrary, the rougher an image is, the closer its fractal dimension is to 3. In our experiments, as can be seen in Fig. 3, Image (a) and Image (b) are two extreme cases. Image (a), with gray-scale of 0 for all pixels, has a perfectly smooth surface, while Image (b), with two alternating colors, black and white, represents a highly “rough” area. The difference between Image (a) and (c), and Image (b) and (d), is their sizes. The fractal dimensions,  $D$ , of the 4 images estimated with different methods are shown in Tab. 1. From Tab. 1, we can see that the method proposed in [11] cannot estimate the fractal dimension of Image 1 because equations (6) and (7) return 0. Because of the drawbacks described in Section 2, the DBC method and all of the ones in [10 - 12] cannot calculate the fractal dimensions of Image 3 and Image 4 either because the images are not square or because their dimensions are not power of 2. On the contrary, our method can calculate the fractal dimensions of all images. The fractal dimensions of Image 1

and Image 3 calculated with our method are very close to the ideal value of 2 and the fractal dimensions of Image 2 and Image 4 calculated with our method are very close to the ideal value of 3. These indicate that our method can indeed more accurately reflect the roughness of the images than the DBC method and the methods of [10-12] and is applicable to images of any sizes. As for the fractal dimensions estimated from spectral based method [3], it can be seen from Tab. 1 that the fractal dimensions are all out of the interval of [2, 3], which indicates the limitation of the spectral based method.

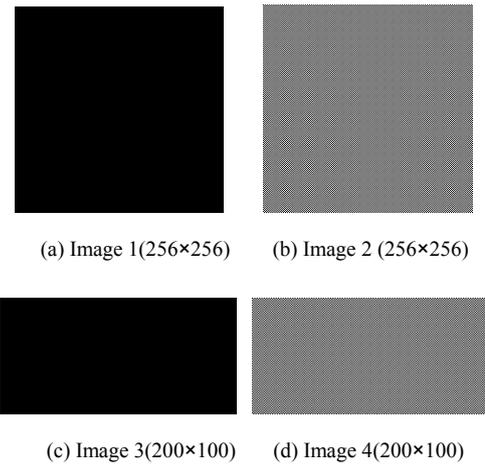


Fig. 3. Images used in our experiments.

Images	Methods					
	DBC	Method [3]	Method [10]	Method [11]	Method [12]	Our Method
Image 1	1.8022	0	2.2591	—	2.1192	<b>2.0602</b>
Image 2	2.7035	0.2656	3.2585	2.0000	3.1188	<b>2.9927</b>
Image 3	—	0	—	—	—	<b>2.0800</b>
Image 4	—	0.3033	—	—	—	<b>2.9812</b>

Tab. 1. Fractal dimensions of images calculated with different methods (“—” denotes that the fractal dimension of the corresponding image cannot be measured by the corresponding method).

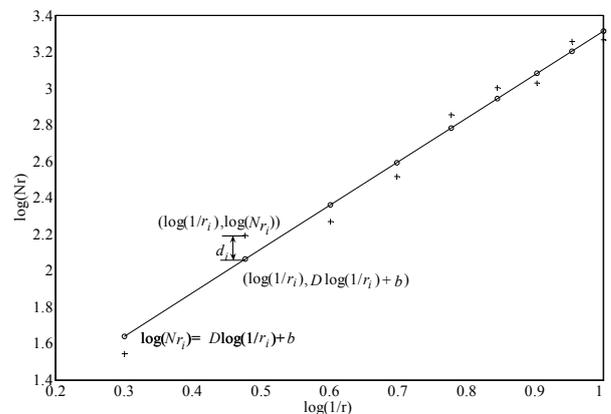


Fig. 4. Illustration of fractal dimension estimation using least square linear fit.

Since the fractal dimension of an image is estimated from the least square fit, fitting errors are therefore a good

performance indicator of box-counting methods. As shown in Fig. 4, the fractal dimension,  $D$ , is the slope of the least square linear fit of  $\log(N_r)$  and  $\log(1/r)$ . The straight line, with a slope-intercept form of  $\log(N_r) = D \times \log(1/r) + b$ , represents the real relationship between  $\log(N_r)$  and  $\log(1/r)$ . The actual deviation of  $(\log(1/r_x), \log(N_{r_x}))$  from the straight line is  $d_r = \log(N_{r_x}) - D \times \log(1/r_x) - b$ , thus the square deviation from the straight line is

$$d_r^2 = (\log(N_{r_x}) - D \times \log(1/r_x) - b)^2. \tag{13}$$

The mean square deviation  $\delta$  can therefore be formulated as

$$\delta = \frac{1}{X} \sum_{x=1}^X d_x^2 = \frac{1}{X} \sum_{x=1}^X ((\log(N_{r_x}) - D \times \log(1/r_x) - b)^2) \tag{14}$$

where  $X$  is the number of different values of ratio  $r$ . Experiments have been carried out on 100 images of  $256 \times 256$  pixels to compare the performance of different methods. Fig. 5 shows some examples of the 100 test images. Note the reason we choose images of  $256 \times 256$  pixels is because all the methods, except ours, are only applicable to either square images or images whose dimensions are power of 2. The distribution of the mean square deviation,  $\delta$ , is illustrated in Fig. 6.

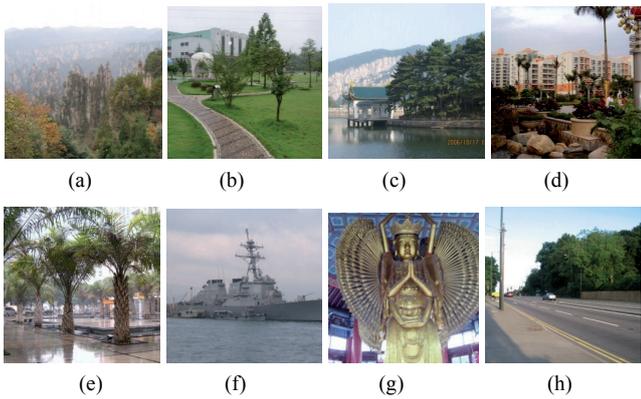


Fig. 5. Samples of the 100 test images.

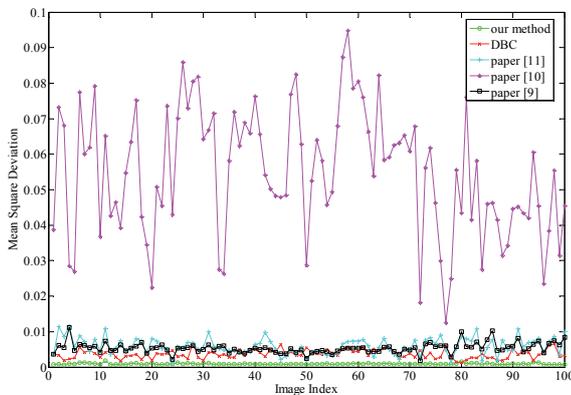


Fig. 6. The mean square deviation of 100 images of different methods.

From Fig. 6 we can see that, for any given image, the values of  $\delta$  calculated with our method is lower than the other methods. The average of the mean square deviation  $\bar{\delta}$  of the 100 images can be calculated using

$$\bar{\delta} = \frac{1}{100} \sum_{y=1}^{100} \delta_y \tag{15}$$

where  $\delta_y$  represents the mean square deviation of the  $y$ -th image of the 100 images. The list of the fractal dimension of some sample images is in Tab. 2, and the  $\bar{\delta}$  of the 100 images calculated with different methods are listed in Tab. 3, which indicates that our method performs significantly better than other methods.

	DBC	Method [3]	Method [10]	Method [11]	Method [12]	Proposed method
a	2.1814	2.2848	2.2680	2.0536	2.2848	2.2418
b	2.2807	2.5993	2.3780	2.1460	2.5993	2.3126
c	2.2158	2.6089	2.3310	2.0505	2.6089	2.2734
d	2.4220	2.5022	2.5506	2.1276	2.5022	2.5129
e	2.4491	2.2917	2.6484	2.1311	2.2917	2.5536
f	2.1175	2.5807	2.3212	1.9724	2.5807	2.2104
g	2.3866	2.3458	2.6149	2.1369	2.3458	2.5617
h	2.2245	2.8994	2.2482	2.0785	2.8994	2.2372

Tab. 2. The fractal dimensions of sample images in Fig. 4.

Parameter	DBC	Paper [10]	Paper [11]	Paper [12]	Our method
$\bar{\delta}$	37	54	549	59	9

Tab. 3. The average of the mean square deviation of 100 images of different methods ( $\times 10^{-4}$ ).

## 6. Conclusion

A new accurate box-counting method for measuring images' fractal dimensions is proposed in this paper. Higher accuracy of fractal dimension estimation is achieved by allowing the height of the box at the top of each grid block to be adaptable to the maximum and minimum gray-scales of that block and the method is applicable to images with arbitrary sizes. Experimental results have shown that the fractal dimensions calculated with the proposed method can better reflect the roughness of images and the calculated fractal dimension is significantly more accurate than other methods in terms of mean square deviation from the fit line.

## Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (Grant No. 61070195, 61001004), Hunan Provincial Natural Science Foundation of China (Grant No. 12JJA006) and Young Teacher's Growth Program of Hunan University.

## References

- [1] MANDELBROT, B. B. *The Fractal Geometry of Nature*. W.H. Freeman and Company, 1982.
- [2] PENTLAND, A. P. Fractal-based description of nature scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1984, vol. 6, no. 6, p. 315-326.
- [3] HUANG, J., TURCOTTE, D. L. Fractal image analysis: application to the topography of Oregon and synthetic images. *Journal of Optical Society of America A-Optics Image Science and Vision*, 1990, vol. 7, no. 6, p. 1124-1130.
- [4] GANGEPAIN, J., ROQUES-CARMES, C. Fractal approach to two dimensional and three dimensional surface roughness. *Wear*, 1986, vol. 109, p. 119-126.
- [5] CHEN, S. S., KELLER, J. M., CROWNOVER, R. M. On the calculation of fractal features from images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1993, vol. 15, no. 10, p. 1087-1090.
- [6] SARKAR, N., CHAUDHURI, B. B. An efficient differential box-counting approach to compute fractal dimension of image. *IEEE Transactions on Systems, Man and Cybernetics*, 1994, vol. 24, no. 1, p. 115-120.
- [7] FENG, J., JIN, W.-C., CHEN, C.-T. Fractional box-counting approach to fractal dimension estimation. In *Proceedings of International Conference on Pattern Recognition*. 1996, p. 854 - 858.
- [8] XIE, W., XIE, W. Fractal-based analysis of time series data and features extraction. *Signal Processing*, 1997, vol. 13, no. 2, p. 98 to 104.
- [9] BISOI, A. K., MISHRA, J. On calculation of fractal dimension of images. *Pattern Recognition Letters*, 2001, vol. 22, p. 631-637.
- [10] LI, J., SUN, C., DU, Q. A new box-counting method for estimation of image fractal dimension. In *Proceedings of IEEE International Conference on Image Processing*. 2006, p. 3029-3022.
- [11] XU, S., WENG, Y. A new approach to estimate fractal dimensions of corrosion image. *Pattern Recognition Letters*, 2006, vol. 27, p. 1942-1947.
- [12] LI, J., DU, Q., SUN, C. An improved box-counting method for image fractal dimension estimation. *Pattern Recognition*, 2009, vol. 42, p. 2460-2469.

## About Authors

**Min LONG** was born in 1977. She received her Ph.D from South China University of Science and Technology in 2006. Her research interests include chaos and fractal theory, chaos-based encryption and digital image forensics.

**Fei PENG** was born in 1977. He received his Ph.D from South China University of Science and Technology in 2006. His research interests include digital image forensics, digital watermarking, and chaos-based encryption.