A fractal de-noising algorithm based on least absolute deviation method

Hui Guo*, Jie He

School of Information and Electronic Engineering, Wuzhou University, Wuzhou 543002, Guangxi, China

Received 1 June 2014, www.cmnt.lv

Abstract

Against the shortcoming that the traditional method of fractal image compression coding has inferior decoding quality on the original image subject to salt-and-pepper noise interference, this paper raises a least absolute deviation (LAD) method to be applied in fractal image compression, which can replace the method of least square error in computing contrast and brightness adjustment value and solve the L1-norm recursive problem using weighted median. The experimental result indicates that the LAD method has a very good anti-noise effect on the outliers introduced by salt-and-pepper noise.

Keywords: Fractal Image Compression, Least Square Error Method, Least Absolute Deviation Method, Salt-and-Pepper Noise

1 Introduction

Put forward by Mandelbrot in 1975 [1], the fractal theory was first applied in image compression by Barnsley in 1988 [2]. It was not until Jacquin raised iterated function system (IFS) and local iterated function system (LIFS) [3] that the theoretical foundation of fractal image compression technique has been laid. Afterwards, this theory was used in image retrieval [4], digital watermarking [5], image inpainting [6] and image denoising [7], etc., respectively. The algorithm raised by Jacquin is based on the concept of local self-similarity of image, in which the local blocks can identify another similar block through three adjustment methods - contrast, brightness and reversal. The LIFS can automatically convert images into affine transform coefficients, which can achieve the aim of compression only by being stored up. Fractal image compression enjoys advantages like high compression ratio, irrelevance of resolution ratio, fast decoding and more, whereas the disadvantage is the overlong compressing hour as result of using the global searching method to seek for the optimal matching blocks. Therefore, in order to speed up the encoding and strengthen the image detail compensation, some scholars combine fractal with other algorithms. He Jia, zheng-kai liu [8] used low frequency coefficients in DCT as the matching feature, and made matching more rapid and accurate.

The LAD method was put forward by Boscovich in 1757, namely the L1-norm, whose aim is to minimize the absolute deviation through estimative parameters, though the weakness of non-differentiability left it unused for complicated computation. Many parallel methods, such as the Barrodale-Roberts [9] Algorithm, the Bartels-Conn-Sinclair Algorithm and the Maximum Likelihood Estimation Algorithm [10] did not rise until Edgeworth put forward resolving the problem of non-differentiability using weighted medians [11] in 1887 and Harris solved the LAD method using the simple concept of linear programming in 1950 [12].

In traditional fractal image compression algorithms, the least square error (LSE) method is adopted to determine brightness and contrast adjustment coefficients by contrasting the range blocks to the domain blocks. It can be known from the recursive principle that the LSE method is of no robustness, so the restored image has inferior quality upon being interfered by noise after undergoing compression. Replacing the traditional LSE method with the LAD method, we take the advantage of robustness with the LAD method to remove noise from the fractal images straightforward during the compression procedure without undergoing the pre-processing prior to noise-removal so that the quality of image is not affected by noise.

2 Theory evidence

2.1 LEAST SQUARE ERROR (LSE) METHOD

Fractal encoding performs compressed encoding by exploiting the characteristic of image's local self-similarity. Each block within an image may correspond to a big similar block. Taking the two groups of blocks in Figure 1 as an example, one group is the hat edge in correspondence to the hat in the mirror, the other being the similar but larger block in correspondence to the part of shoulder. The larger block approximates the smaller counterpart within the same group after being minimized and adjusted in contrast and brightness.

* ***Corresponding author*** e-mail: 13066724@qq.com

A 256-by-256 image can fall into non-overlapped n-by-n range blocks, the set of which is a range pool. Then the original image is divided into 2n-by-2n domain blocks that can be overlapped, the set of which is called a domain pool. The procedure of fractal image compression is to pinpoint the most similar domain block from the domain pool for each range block in the range pool. This procedure can be viewed as finding a domain block through convergent operations of affine transform.

FIGURE 1 Local Self-similarity of Fractal Image Compression

In the contrasting procedure, the square error is used to estimate the difference between the range block v and the domain block u after the sub-sampling. Smaller is indicative of higher similarity between the domain block u and the range block v, as shown in Equation (1).

$$E_k = \|(p_k u_k + q_k) - v\|^2, \; k=0,1,...,7, \qquad (1)$$

where k denotes 8 directions of reversal; E_k, p_k, u_k and q_k represent the square error value, the contrast adjustment value, the block after the secondary sampling, and the brightness adjustment value, respectively, of the difference degree of the estimative block under the k^{th} reversal. $\hat{p}_k$ and $\hat{q}_k$ are the predicted values of p_k and q_k, respectively, which can derive from Equation(1)through partial differential. This method to compute p and q is just the LSE method, N in the following Equation being the size of the range block.

$$\hat{p}_k = \frac{N^2 \langle u_k, v\rangle - (\sum_{j=0}^{N-1}\sum_{i=0}^{N-1} u_k(i,j)) \cdot (\sum_{j=0}^{N-1}\sum_{i=0}^{N-1} v(i,j))}{N^2 \langle u_k, u_k\rangle - (\sum_{j=0}^{N-1}\sum_{i=0}^{N-1} u_k(i,j))^2}, \qquad (2)$$

$$\hat{q}_k = \frac{1}{N^2}\left(\sum_{j=0}^{N-1}\sum_{i=0}^{N-1} v(i,j) - \hat{p}_k \sum_{j=0}^{N-1}\sum_{i=0}^{N-1} u_k(i,j)\right). \qquad (3)$$

LAD Method

The LAD method is interpreted against the concept of linear regression, a model of which is assumed as Equation (4).

$$y_i = ax_i + b + \varepsilon_i, \; i=1......L, \qquad (4)$$

where x is the input value, y is the output value, and ε is the error term. When ε_i meets the following assumptions:

(a) $E(\varepsilon_i) = 0$;

(b) The variance number is constant;

(c) ε_i is a Gaussian distribution function whose formula of Gaussian distribution follows Equation(5), where the mean $\mu = 0$ and the variance is σ;

(d) Each follows independent distribution.

When ε_i meets the above assumption, it is the optimal choice to solve Equation (4) using the LS method. When ε_i is assumed as a Laplace distribution function whose formula of Laplace distribution follows Equation (6) where the mean $\mu = 0$ and the variance $\sigma = \frac{\sqrt{2}}{\lambda}$, and when ε_i meets the above assumptions (a), (b) and (d) with a Laplace distribution, it is the optimal choice to solve Equation (4) using the LAD method. Different methods are employed for the optimal solution according to different assumptions of ε_i.

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x)^2}{2\sigma^2}\right), -\infty < x < \infty, \qquad (5)$$

$$f(x) = \frac{\lambda}{2} \exp(-\lambda|x|), -\infty < x < \infty, \qquad (6)$$

The LAD method and the traditional LSE method are applied to Equation 4 respectively to produce Equation (7) and Equation (8).

$$E = \sum_{i=1}^{L} |y_i - (ax_i + b)| = \sum_{i=1}^{L} \mathrm{sgn}(e_i) \cdot e_i, \qquad (7)$$

$$E = \sum_{i=1}^{L} (y_i - (ax_i + b))^2 = \sum_{i=1}^{L} e_i \cdot e_i. \qquad (8)$$

In the above Equations, E is the sum of the parameters in group L, e_i is the error value of the parameters in the ith group. For the same group of parameters, e_i has identical value. Between both methods mentioned above, the difference lies in the fact that the LAD method takes the absolute value of e_i. When e_i is positive sgn (e_i) becomes 1, when e_i is negative sgn (e_i) becomes -1. While in the LSE method, the error value E will become greater after the value of e_i is squared. Under the circumstance where the LAD method is used, there is no significant difference with e_i between the interval [-1, 1] from the LSE method; but when outliers emerge to the data, e_i will be magnified at a square rate when the LSE method is employed, which leads the error E to become too great to make good judgment and estimation, so the value of e_i shall be kept from being magnified fast to reduce the effect of a few outliers on all the data, which is the robustness in inhibiting outliers, as shown in Figure 2.

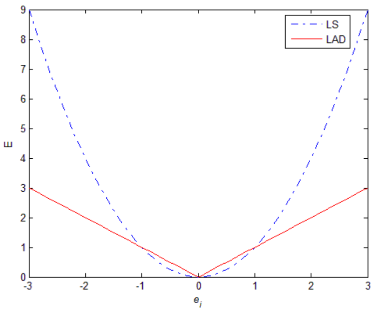


FIGURE 2 Relations between LAD, LSE and error E

2.2 MAXIMUM LIKELIHOOD ESTIMATES (MLE)

The LAD method cannot perform differential since absolute value exists, so the parameters with least absolute deviation value need to be determined through other methods. What is adopted in this paper is the maximum likelihood estimation algorithm [10], in which the weighted medians are operated. The weighted median starts from the viewpoint of geometric slope. Where the slope is greater than 0, the deviation value is minimum. It can be inferred as follows that the slopes after being sorted are on a progressive increase, turning from negative to positive as increasing to a certain value, which is the approximate minimum rising when the least deviation method is used. The proof goes as below:

$$F(a) := \sum_{i=1}^{L} |y_i - b - ax_i| = \sum_{i=1}^{L} |x| \left| \frac{y_i - b}{x_i} - a \right| \\ = \sum_{i\in I\_(a)} |x_i| (a - \frac{y_i - b}{x_i}) + \sum_{i\in I_+(a)} |x_i| (\frac{y_i - b}{x_i} - a) . \tag{9}$$

F(a) is the absolute value taken from Equation (4); *x* and *y* are the input and output, respectively, each holding the parameters in *L* Group, where:

$$I\_(a) := \left\{ i \in \{1......L\} : \frac{y_i - b}{x_i} \le a \right\}$$

$$I_+(a) := \left\{ i \in \{1......L\} : \frac{y_i - b}{x_i} > a \right\}$$

I_(a) is a set of parameters in *L* Group whose slopes are smaller than *a*; $I_+(a)$ is a set of parameters in *L* Group whose slopes are greater than *a*. Partial differential is applied to *a* in Equation (9) to get the slope as Equation (10). The minimum deviation value is determined by the LSE method when the slopes are 0. Since the LAD method is linear and continuous, it is taken that $F'(a) \ge 0$. While the parameter in correspondence to the slope turning from negative to positive reaches the minimum.

$$F'(a) := \sum_{i\in I\_(a)} |x_i| - \sum_{i\in I(a)_+} |x_i| \ge 0, \ \sum_{i\in I\_(a)} |x_i| \ge \sum_{i\in I_+(a)} |x_i| \tag{10}$$

$$2\sum_{i\in I\_(a)} |x_i| = \sum_{i\in I\_(a)} |x_i| + \sum_{i\in I\_(a)} |x_i| \ge \sum_{i\in I\_(a)} |x_i| + \sum_{i\in I_+(a)} |x_i| = \sum_{i=1}^{L} |x_i|$$

$$\sum_{i\in I\_(a)} |x_i| \ge \frac{1}{2} \sum_{i=1}^{L} |x_i| . \tag{11}$$

According to Equation (10), when 5 groups of parameters are input, the change of their slopes is shown as Figure3. The slopes are sorted as 1, 2, ..., 5 in an incremental order, whereas the slopes are $\frac{y_{[i]} - b}{x_{[i]}}$, and the corresponding parameters can be yielded. At Interval (3) $F'(a)$ is negative while becoming positive at Interval (4), so Interval (4) is the first group of data $x_{[4]}$ when $F'(a) \ge 0$. And Figure 3 indicates that the minimum should be among $x_{[3]}$. It can be known from Equation (11) that the input data are within the set of *I_(a)*. As the data $x_{[1]}$ input at the first time are accumulated to the data $x_{[i]}$ input at the ith time and when the accumulated sum is greater than one half of the sum of all input data, the input data at the ith time are the input ones of the first positive slope when the slope just turns from negative to positive. From Equation (11) it can be found the first datum where $F'(a) \ge 0$ is the one input at the ith time, so the minimum deviation value may occur when the datum is input at the (i - 1)th time.

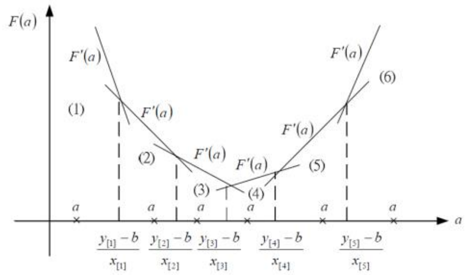


FIGURE 3 Conceptual Schematic Diagram of Weighted Median

Figure 3 reveals the minimum deviation value may occur among the data in the third group. It can be concluded from Equation (11) that, When the input data *L* group, the minimum deviation value may be among the data input at the (i - 1)th time. The weighted medians compute the slope of the data with the least deviation value. They are denoted by *MED*(•) as below:

$$a = MED(x_i \Diamond \frac{y_i - b}{x_i} |_{i=1}^{L}) . \tag{12}$$

The steps to compute the weighted medians are as follow:

Step I: Assume $w_i = \frac{y_i - b}{x_i}$, record the slope input each time.

Step II: Compute the threshold value $x' = (1/2)\sum_{i=1}^{L} x_i$.

Step III: Express all the slopes $w_1......w_L$ as $w_{[1]} \le ...... \le w_{[L]}$ after sorting them; in correspondence to $w_{[i]}$ after being sorted, the relevant input data is $x_{[i]}$ as the weight.

Step IV: Apply summation to $x_{[i]}$ one after another; record the index value j that first meets $\sum_{i=1}^{j} x_{[i]} \ge x_0$, then the index value of the minimum is j - 1.

Step V: $w_{[j-1]}$ is the slope in correspondence to the minimum; the computation for the weighted medians is just done by letting $a = w_{[j-1]}$.

The above is introduction to the concept of and steps for weighted median, whereas the following provides illustration on how MLE figures out the slope and the gap by means of the weighted medians. In order to accelerate convergence after using the weighted medians, MLE transforms the space to finely adjust the slope, thereby to get more accurate slope, reduce the times of training, and get the slope with the least error and the corresponding gap through fast convergence.

The MLE calculation method 4 offers to pinpoint the parameters a and b in correspondence to the minimum E in Equation (7).

The steps are as follow:

Step I: Set the times of recursion as k = 0 at the very beginning; b = b_k is yielded through calculation using the LSE method, as shown in Equation (13):

$$b_0 = \frac{\sum_{i=1}^{L}(x_i - \bar{x})\times(\bar{y}x_i - \bar{x}y_i)}{\sum_{i=1}^{L}(x_i - \bar{x})^2}, \tag{13}$$

$$\bar{x} = (1/L)\times\sum_{i=1}^{L} x_i \text{ , } \bar{y} = (1/L)\times\sum_{i=1}^{L} y_i$$

Step II: Compute the parameter a_0 using the weighted medians, expressed as Equation (14):

$$a_0 = MED(|x_i| \Diamond \frac{y_i - b_0}{x_i})|_{i=1}^{L}) . \tag{14}$$

Determine the index value h for existence of least deviation value. The parameter a_0 can be yielded from Equation (14), accordingly update $a_{old} = a_0$.

Step III: Let k = k + 1 to convert the input spatial coordinate $z_i = x_i - x_h$.

Step IV: Compute $a'_{k-1} = a_{k-1}$, $b'_{k-1} = b_{k-1} + a_{k-1}x_h$ and the weighted median a'_k as Equation (15):

$$a'_k = MED(|z_i| \Diamond \frac{y_i - b'_k}{z_i}|_{i=1}^{L}) . \tag{15}$$

Compute from Equation (15) the new index value m and the corresponding parameter a'_k .

Step V: Compute $a_k = a'_k$ and $b_k = b'_k$-$a'_k x_h$ by converting the original spatial coordinate.

Step VI: Let h = m. Stop where the variances of a_{old} and a_k are below the permitted value or where the time of recursion exceeds the set value; otherwise update $a_{old} = a_0$ and return to Step III. The above steps can be expressed by the flowchart as Figure 4.

3 Application of LAD in fractal image compression

The workflow of introduction of LAD in fractal image compression bears general similarity to traditional methods of fractal compression, the only difference being how to compute p and q and utilize the absolute deviation as an indicator of estimation, as shown in Figure 5.The main steps by which to replace LS with LAD are as follow:

Step I: Fall the original image into an 8-by-8 non-overlapped range block v and a 16-by-16 overlapped domain block u.

Step II: Apply secondary sampling to u and perform reversal at k.

Step III: The iteration time t = 0. Compute the initial $q_k = \hat{q}_k$

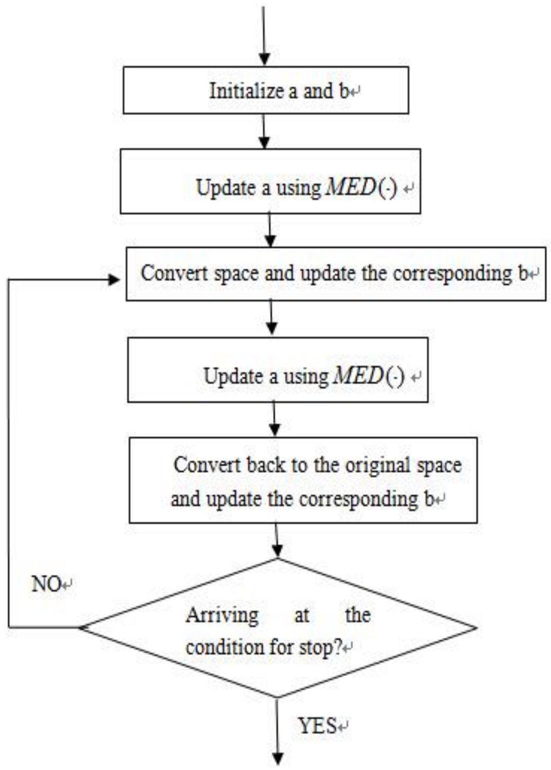


FIGURE 4 Flowchart of LAD

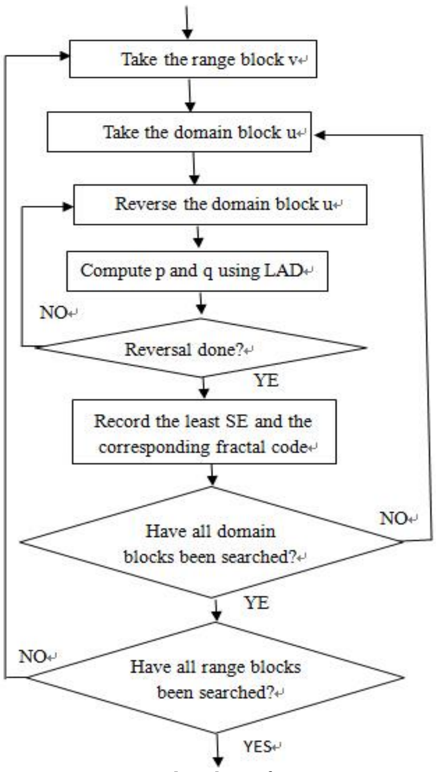


FIGURE 5 Flowchart of LAD-FIC

Step IV: Figure out pk using MED(•), expressed as Equation (16):

$$p_k = MED(|u_k(i,j)| \Diamond \frac{v(i,j)-q_k}{u_k(i,j)}) \Big|_{i,j=0}^{N-1}), \qquad (16)$$

Where N is the size of the range block v. According to the above Equation, the least deviation value within the 8-by-8 $u_k(i, j)$ is ($h1$, $h2$), whereby to update $p_{old} = p_k$.

Step V: Let $t = t + 1$. Convert the input space $z(i,j) = u_k(i,j) - u_k(h1,h2)$.

Step VI: $q_k = q_k + p_k u_k(h1,h2)$ and p_k after space conversion are shown as Equation (17):

$$p_k = MED(|z(i,j)| \Diamond \frac{y(i,j)-q_k}{z(i,j)} \Big|_{i,j=0}^{N-1}). \qquad (17)$$

According to the above Equation, the least deviation value within $z(i, j)$ is ($m1$, $m2$).

Step VII: Convert to the original space, where $q_k = q_k + p_k u_k(h1,h2)$.

Step VIII: Let $h1 = m1$, $h2 = m2$. Stop where the variances of p_{old} and p_k are below the permitted value or where the time of iteration exceeds the set value; otherwise update $p_{old} = p_k$ and return to Step IV. LAD offers to compute p_k and q_k from Step IV through Step VIII.

Step IX: Return to Step II if none of the eight directions of reversal has been used.

Step X: Record the fractal code in correspondence to the least absolute deviation in Equation (15).

$\min(E_K = |(p_k u_k + q_k) - v|)$, $k=0,1,...7$

Step XI: Apply global search to the domain pool to pinpoint the domain block u in correspondence to each range block v. Thus the LAD-FIC computation is done.

4 Experimental result

The tool for experiment in this paper is Visual C++ 6.0, the operating system is Microsoft Windows XP, the CPU is Intel Core i5 3450, and the internal memory is 4G. In this section, a comparison will be made between the traditional method of fractal compression and the LAD-introduced method of fractal image compression for multiple frames of images where different noises are included. The images in use are all 256-by-256, the size of range block being 8-by-8. The condition under which MLE stops is that the time of iteration reaches 5 or that the difference between the yielded p value and the previous p value is smaller than 0.05. PSNR, which is used to evaluate the quality of the decoded image, is defined as Equation (18), whereas MSE is defined as Equation (19).

$$PSNR(f, \hat{f}) := 10 \cdot \log_{10}\left(\frac{255^2}{MSE(f, \hat{f})}\right), \qquad (18)$$

$$MSE(f, \hat{f}) = \frac{1}{256^2} \cdot \sum_{i=1}^{256}\sum_{j=1}^{256}(f(i,j) - \hat{f}(i,j))^2, \qquad (19)$$

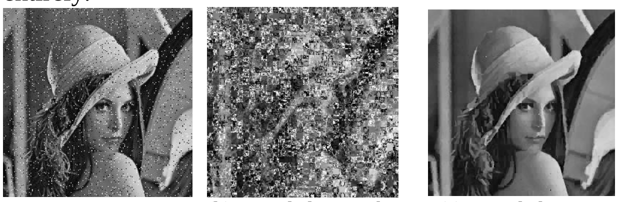
where f is the original image, $\hat{f}$ is the decoded image.

Initially, the Lena image in which no noise has been included offers as the image to test. The traditional fractal image compression (FIC) method and the fractal image compression method where LS gives way to LAD (LAD-FIC) are used separately to observe the effects of compression by both, as shown in Figure 6. Baboon is the initial image to decode. Through 9 times of iteration, Figure 6(b) is the image decoded by FIC, whose PSNR value is 28.86dB. Figure 6(c) is the image decoded by LAD-FIC, whose PSNR value is 28.27dB. The images decoded by both methods have almost the same PSNR, wherefrom both LAD-FIC and FIC methods can be verified to make the same effect in terms of compression.

(a) Original Image Lena (b) Decoded Image by FIC, PSNR = 28.86dB (c) Decoded Image by LAD-FIC, PSNR = 28.27dB

FIGURE 6 Contrasted Effects of Lena Image Tested by FIC and LAD-FIC

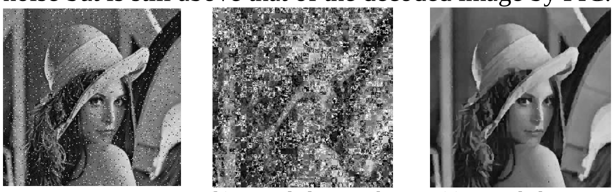
Next, it is attempted to include 5% and 10% salt-and-pepper noises in the four images (Lena, Baboon, Pepper and F16). After being compressed by both methods, all the decoded images are based on Baboon as the initial image but Baboon itself, which is based on Lena as the initial image. The decoded images are produced after 9 times of iteration. Figure 7(a) is the image by including the 5% salt-and-pepper noise in Lena. Figure 7(b) and Figure 7(c) are the decoded images after FIC and LAD-FIC are used for compression. Obviously, it can be seen that Figure 7(b) has been vulnerable to noise in that its decoded image fails to restore Lena but presents only a few lines across big blocks which cannot clearly distinguish the original image. The difference between the PSNR values of Figure 7(c) and Figure 6(c) is only 1dB, wherefrom it can be seen that LAD-FIC can almost remove the effect of the 5% salt-and-pepper noise entirely.



(a) 5% Salt-and-Pepper Noise (b) Decoded Image by FIC, PSNR = 11.87dB (c) Decoded Image by LAD-FIC, PSNR = 27.21dB

FIGURE 7 Lena Image for Testing 5% Salt-and-Pepper Noise

Next, 10% salt-and-pepper noise is then included in the test. Figure 8(a) is the image by including the 10% salt-and-pepper noise in Lena. Figure 8(b) is the decoded image by FIC whose PSNR value is 11.41dB, with only a bit more inferior effect than 7(b) though the quality of image has been so poor as to obscure the original image. Figure 8(c) is the decoded image by LAD-FIC whose PSNR value is 26.42dB, which falls below the counterpart of Figure 7(c) for the image's vulnerability to noise but is still above that of the decoded image by FIC.



(a) 10% Salt-and-Pepper (b) Decoded Image by FIC, PSNR = 11.41dB (c) Decoded Image by LAD-FIC, PSNR = 26.42dB

FIGURE 8 Lena Image for Testing 10% Salt-and-Pepper Noise

The test is conducted on the three other images by the same method. To illustrate it with Table 1, the parameter *S* in the table denotes the salt-and-pepper noise. The experiment demonstrates that the PSNR value of the decoded image produced through fractal compression on the image in which the salt-and-pepper noise has been included has decline to a considerable degree, whereas LAD-FIC remains not affected by this noise, therefore LAD-FIC has the effect of resisting the salt-and-pepper noise

TABLE 1 the PSNR (unit: dB) Values of Pepper, Baboon and F16 under 5%, 10% and 20% Salt-and-Pepper Noises

	Pepper		**Baboon**		**F16**	
	FIC	*LAD-FIC*	*FIC*	*LAD-FIC*	*FIC*	*LAD-FIC*
Original Image	29.78	29.31	20.03	19.82	25.17	24.43
S=5%	17.89	27.88	15.43	19.26	12.83	23.71
S=10%	17.96	27.21	17.56	20.12	5.53	23.23
S=20%	18.31	25.92	17.41	18.33	6.24	22.67

5 Conclusion

Traditional techniques of fractal image compression fail to discuss how to resist noise, so the quality of compression will be vulnerable to noise when the original image is interfered by noise. The LAD method of robustness used in substitution for the traditional LSE method endows the image with the capacity of resisting the salt-and-pepper noise, namely the LAD-FIC solution proposed in this paper can remove noise while compressing the image. This method makes extremely excellent effect on images vulnerable to the salt-and-pepper noise, yet there is no distinct effect for Gaussian noise and Laplace noise. Since LAD-FIC first uses FIC to estimate the rough contrast and brightness adjustment value prior to further adjustment to figure out the accurate contrast and brightness adjustment value, it is discovered through a comparison between both that the time cost for LAD-FIC is 18 times that for FIC. In the research direction in the future, it is hoped that the compressing time can be further reduced, or the contrast and brightness adjustment value can be changed to get their linear relationship, on the premise of guaranteeing its robustness, so as to achieve a higher quality of compression.

Acknowledgments

This paper is financially supported by Guangxi Natural Science Foundation Program (Grant No. 2013GXNSFBA019275, Grant No. 2013GXNSFBA019276) and Guangxi University of Science and Technology Research Program (Grant No.2013YB227, Grant No.2013YB228).

References

[1] Mandelbort B 1982 *The fractal geometry of nature* Freeman: San Francisco, California

[2] Barnsley M F 1992 *Fractals everywhere* Academic Press: New York

[3] Jacquin A E 1992 Image coding based on a fractal theory of iterated contractive image transformation *IEEE Transactions on Image Processing* **1**(1) 18-30

[4] Wang X Y A 2009 Aast fractal coding in application of image retrieval *Fractals* **17**(04) 441-50

[5] Tirkel A Z, Osbome C F, Hall T E 1998 Image and watermark registration *Signal Processing* **66**(03) 373-83
[6] Li Jin jiang, Zhang Caiming 2010 Image inpainting algorithm based on fractal theory *Acta Electronica Sinica* **38**(10) 2430-5
[7] Ghazel M, Freeman G H, Vrscay E R 2006 Fractal-Wavelet image denoising revisited *IEEE Transactions on Image Processing* **15**(09) 2669-75
[8] Hy Jia, Lшr Zhengkai 2010 A fast fractal image coding method based on the DCT transform *Acta Electronica Sinica* **29**(6) 748-50
[9] Barrodale I, Roberts F D K 1973 An improved algorithm for discrete 1 L linear approximation *SIAM Journal on Numerical Analysis*, **10**(5) 839-48
[10] Li Y B, Arce G R 2004 A maximum likelihood approach to least absolute deviation regression *EURASIP Journal on Applied Signal Processing* **20**(04) 1762-9
[11] Edgeworth F Y 1887 A new method of reducing observations relating to several quantities *Philosophical Magazine* **24** 222-3
[12] Harris T E 1950 Regression using minimum absolute deviations *The American Statistician* **4**(1) 14-5

Authors

Hui Guo, born on September 28, 1981, China

Current position, grades: Associate Professor at Wuzhou University, China.
University studies: master degree in college of electronic engineering from Guangxi Normal University, China in 2008.
Scientific interests: image compression and image representation.

Jie He, born on February 13, 1982, China

Current position, grades: Associate Professor at Wuzhou University, China.
University studies: master degree in College of Computer Science & Information Technology from Guangxi Normal University, China in 2008.
Scientific interests: image compression and image representation.