

## Research Article

# Prediction of the PSNR Quality of Decoded Images in Fractal Image Coding

**Qiang Wang and Sheng Bi**

*College of Information Science and Technology, Dalian Maritime University, Dalian 116026, China*

Correspondence should be addressed to Sheng Bi; [dmsupi@hotmail.com](mailto:dmsupi@hotmail.com)

Received 8 October 2015; Revised 22 January 2016; Accepted 24 January 2016

Academic Editor: Rafael Morales

Copyright © 2016 Q. Wang and S. Bi. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With many observations, we find that there exists a logarithmic relationship between the average collage error (ACER) and the PSNR quality of decoded images. By making use of ACER in the encoding process, the curve fitting result can help us to predict the PSNR quality of decoded images. Then, in order to reduce the computational complexity further, an accelerated version of the prediction method is proposed. Firstly, a low limit of percentage of accumulated collage error (LLPACE) is proposed to evaluate the actual percentage of accumulated collage error (APACE). If LLPACE reaches a large value, such as 90%, the corresponding APACE can be proved to be limited in a small range (90%–100%) and the APACE can be estimated approximately. Thus, the remaining range blocks can be neglected and the corresponding computations can be saved. With the approximated APACE and the logarithmic relationship, the quality of decoded images can be predicted directly. Experiments show that, for four fractal coding methods, the quality of decoded images can be predicted accurately. Furthermore, the accelerated prediction method can provide competitive performance and reduce about one-third of total computations in the encoding process. Finally, the application of the proposed method is also discussed and analyzed.

## 1. Introduction

Fractal image coding was firstly proposed by Barnsley; then, Jacquin introduced the first practical fractal coding method [1–3]. Since then, it has attracted much attention of the researchers worldwide. In the early years, fractal coding was mainly considered as an image compression method. The researchers worldwide showed great interest in it and many aspects of fractal coding have been explored comprehensively such as the mathematical background concerning its limits and prospects, relationship with vector quantization, encoding of video sequences, and compressing ECG signals [4]. Moreover, many excellent characteristics, such as its novel idea, independence of resolution, fast decoding, and potential of high compression ratio, have been gradually found [5]. Although fractal coding has many advantages, the high computational complexity in fractal encoding is a main obstacle for its practical applications. Many fast fractal coding methods have been proposed and they can be mainly divided into three categories: classification techniques, feature vector techniques, and the no-search based fractal coding methods.

For the former ones, the range blocks and domain blocks are first divided into different categories and the block matching is only carried out within the same category [6, 7]. For the secondary ones, the block matching process is carried out in the feature space [8–10]. Due to the lower dimension of features and more effective searching strategies, such as kd-tree method [11], the fractal encoding process can be finished in a short time. Finally, no-search fractal coding method is another way to provide faster fractal encoding and higher compression ratio at the expense of poor quality of the decoded image [12–15]. Furthermore, during the past two decades, it has been successfully applied in many other image processing applications such as image denoising [16, 17], image magnification [18–20], image retrieval [21, 22], and digital watermarking [23–25].

Since the fractal coding method aims to reduce the redundancy existing among different local parts of the input image, only the images with good self-similarity can get high quality of decoded images and the quality of decoded images is determined by the self-similarity of input images. This provides the possibility of predicting the quality of decoded images for

us. In our research, we introduce the average collage error (ACER) to represent the self-similarity of the input image. By many observations, we find that there exists a logarithmic relationship between the ACER and the PSNR quality of the decoded image. Thus, we can predict the quality of decoded images by making use of ACER in the encoding process. The proposed method stated above does not need the fractal decoding process, but all the encoding process is necessary to compute the ACER. In order to accelerate the prediction method, on one hand, we propose a low limit of percentage of accumulated collage error (LLPACE) which provides an effective way to estimate the percentage of the accumulated collage error (ACE) in the encoding process. When LLPACE reaches a large value, such as 90%, the corresponding ACE can be limited in a small range (90%–100%). That is to say, the ACE can be estimated approximately. On the other hand, by theoretical derivations and analyses, we find that the collage error mainly comes from the range blocks with large variances. We firstly sort the range blocks by their variances from largest to smallest. Then, the range blocks are encoded one by one subject to the same order. When the LLPACE reaches a threshold (90%), we can get the approximated ACE and predict the quality of decoded images directly by the above logarithmic relationship. Thus, the remaining range blocks can be neglected and the corresponding computations can be saved. Finally, we assume the practical application for our proposed method. For example, the quality of decoded images is demanded to be above a certain value, such as 30 dB. If fractal coding method can satisfy this practical demand, without the decoding process, only the encoding process is needed to predict this result. If not, our prediction method can get the final result only with a small part of encoding process and much time-consumption can be saved. Experiments show that, for the basic fractal coding and the other three fast fractal coding methods, the quality of decoded images can be predicted accurately by making use of the proposed logarithmic relationship. Moreover, the accelerated prediction method can provide competitive performance and reduce about one-third of total computations in the fractal encoding. For their practical applications, the conditions of setting different practical demands are simulated. In particular, when the threshold of the practical demand is set to be a large value, only a small number of range blocks are needed to be encoded to judge whether fractal coding can satisfy the practical requirements.

## 2. Review of Fractal Image Coding

The theoretical basis of fractal encoding method is the collage theorem which can be described as follows: let  $(X, d)$  denote a metric space with a given metric  $d$ . For any  $x \in X$ , we try to find a contractive transform  $T$  that makes the collage distance  $d(x, T(x))$  as small as possible, thereby making the approximation error  $d(x, \bar{x})$  small:

$$d(x, \bar{x}) \leq \frac{d(x, T(x))}{1 - s}, \quad (1)$$

where  $s$  and  $\bar{x}$  denote the contractivity factor and fixed point of  $T$ , respectively. For the practical fractal coding method,

the aim of fractal encoding process is to construct a set of transformations whose fixed point can approximate the original image well [3].

Following Jacquin's method, the input image is first partitioned uniformly into nonoverlapping range blocks; generally, the size of domain blocks is two times larger in each direction than that of the range blocks. The domain blocks can be obtained by sliding a window over the original image with a search step  $\delta$ . Then, a domain block pool can be established by contracting domain blocks to the same size of range blocks. Furthermore, the domain block pool is extended by eight isometric transformations. For partial Lena image, Figure 1 illustrates the eight isometric transformations. In order to make the domain block match the range block better, an affine transformation is also needed and the best matched domain block can be obtained by searching the domain block pool and minimizing the following matching error function:

$$CE(\mathbf{R}) = \min_{\alpha, \beta} \|\mathbf{R} - \alpha\mathbf{D} - \beta\mathbf{I}\|^2, \quad (2)$$

where  $\mathbf{I}$  denotes a matrix whose elements are all ones.  $\alpha$  and  $\beta$  denote the scaling coefficient and offset coefficient for the affine transformation, respectively.  $CE(\mathbf{R})$  is the collage error for the range block  $\mathbf{R}$ .  $\mathbf{D}$  denotes the contracted domain block. Figure 2 illustrates the fractal encoding process. The red block and green block indicate the range block and domain block, respectively. The yellow arrow shows that, after contractive, isometric, and affine transformations, the best matched domain block can be obtained by searching the domain block pool.

At the fractal decoding phase, the same transformations as the encoding process are recursively performed on arbitrary initial image. After about ten iterations, the decoding process will converge to the decoded image. While the black image is selected as the initial image, Figure 3 illustrates the first eight iterations for partial Lena image.

## 3. Prediction of the Decoded Image Quality

For real world images with different extent of self-similarity, the decoded image quality of fractal image coding is different. Figures 4(a) and 4(c) illustrate two input images. Figures 4(b) and 4(d) are their decoded images, respectively. In this paper, we characterize the quality of decoded images in terms of the Peak Signal-to-Noise Ratio (PSNR) and compute it as follows:

$$PSNR = 10 \log_{10} \left( \frac{255^2}{(1/MN) \sum_{i=1}^M \sum_{j=1}^N (x_{ij} - y_{ij})^2} \right), \quad (3)$$

where  $M$  and  $N$  are the height and width of the input image.  $x$  and  $y$  denote the original image and the decoded image, respectively. The PSNRs of the two decoded images are 35.95 dB and 27.73 dB, respectively.

From Section 2 and Figure 2, we know that fractal coding method aims to reduce the redundancy existing among different local parts of the input image and the quality of decoded images is determined by the self-similarity of input

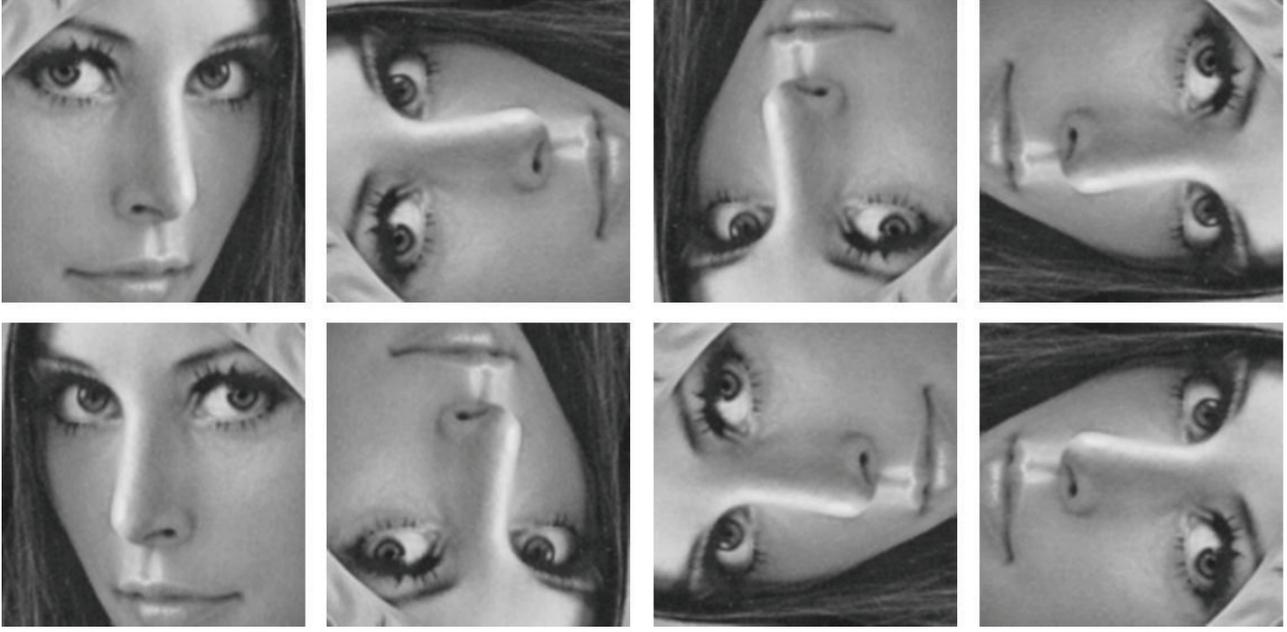


FIGURE 1: Eight isometric transformations.

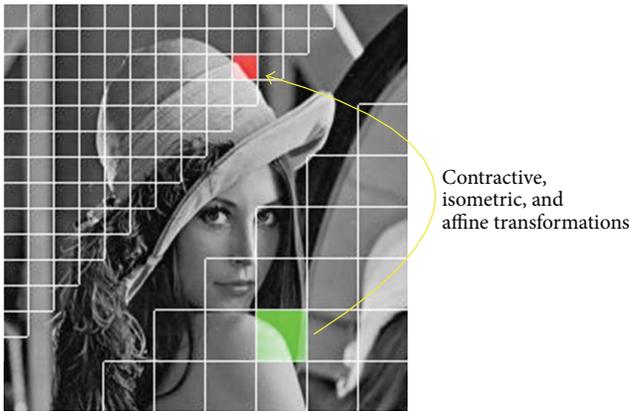


FIGURE 2: Illustration of the fractal encoding process.

images. Moreover, self-similarity could be measured by histogram of collage error distributions [26]. From Figures 4(e) and 4(f), we can see that, compared with Figure 4(c), the collage error distribution of Figure 4(a) demonstrates noticeable peaking which is closer to zero and this implies that Figure 4(a) can be considered to be more self-similar. This can be also verified by the PSNR comparison between Figures 4(b) and 4(d). In our research, in order to measure the self-similarity quantitatively, the average collage error (ACER) is proposed to represent the self-similarity of the input image and can be represented as follows:

$$\text{ACER} = \frac{1}{\text{Num}} \sum_{i=1}^{\text{Num}} \text{CE}(\mathbf{R}_i), \quad (4)$$

where  $\text{CE}(\mathbf{R}_i)$  is the collage error of range block  $\mathbf{R}_i$  and Num denotes the total number of range blocks.

In order to reveal the relationship between the ACERs and PSNRs of decoded images, ninety-seven images in the USC-SIPI database and one hundred images in the Vision Texture database are used and all the test images are encoded and decoded by fractal coding method [27, 28]. In the experiment, each input image is partitioned uniformly into the range blocks whose sizes are  $4 \times 4$ ,  $8 \times 8$ , and  $16 \times 16$ , respectively. The sizes of domain blocks and search step  $\delta$  are both two times larger than that of range blocks. Figure 5(a) illustrates the PSNR versus ACER for all the tests. We can observe that there exists a fixed relationship between ACERs and PSNRs of decoded images and we can use (4) to represent the relationship between ACERs and PSNRs of decoded images:

$$f(x) = \beta_1 + \beta_2 \log_{10}(x), \quad (5)$$

where  $x$  and  $f(x)$  denote the ACER and the PSNR of the decoded image, respectively.  $\beta_1$  and  $\beta_2$  are the coefficients needed to be determined.

For the  $i$ th image, let  $x_i$  and  $f(x_i)$ ,  $i = 1, 2, \dots, C$ , denote the corresponding ACER and PSNR of the decoded image, respectively;  $C$  is the total number of data points. Based on the above representations, we can get the system of equations as follows:

$$\begin{aligned} \beta_1 + \beta_2 \log_{10}(x_1) &= f(x_1), \\ \beta_1 + \beta_2 \log_{10}(x_2) &= f(x_2), \\ &\vdots \\ \beta_1 + \beta_2 \log_{10}(x_C) &= f(x_C). \end{aligned} \quad (6)$$

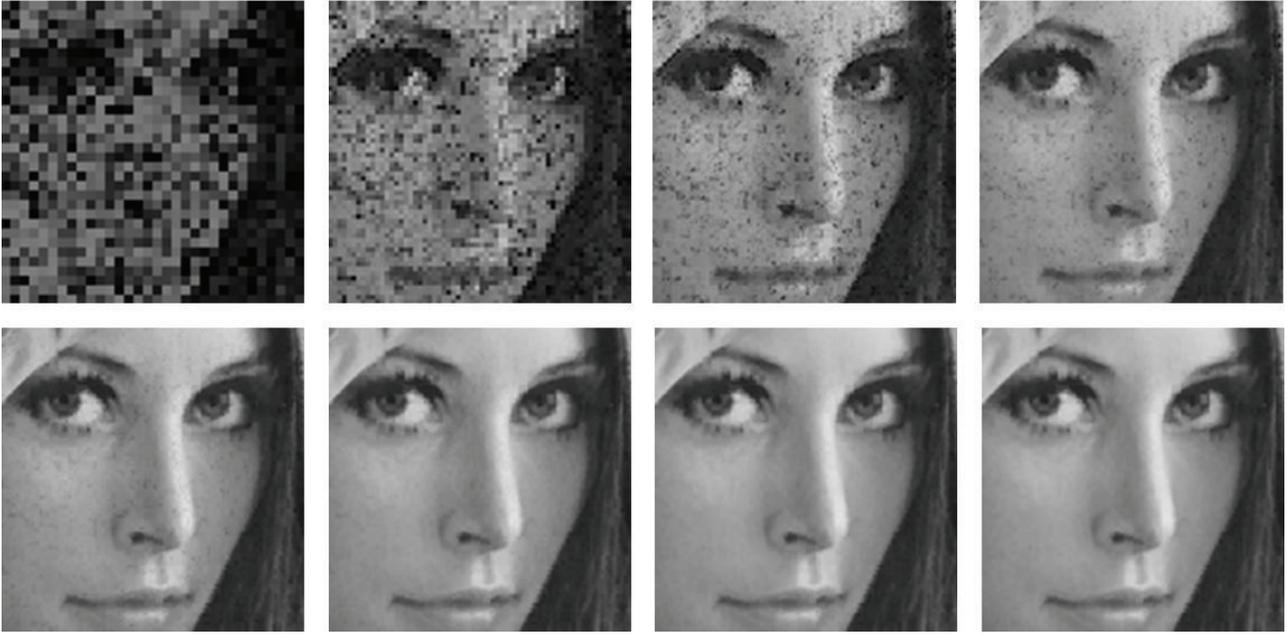


FIGURE 3: The first eight iteration images.

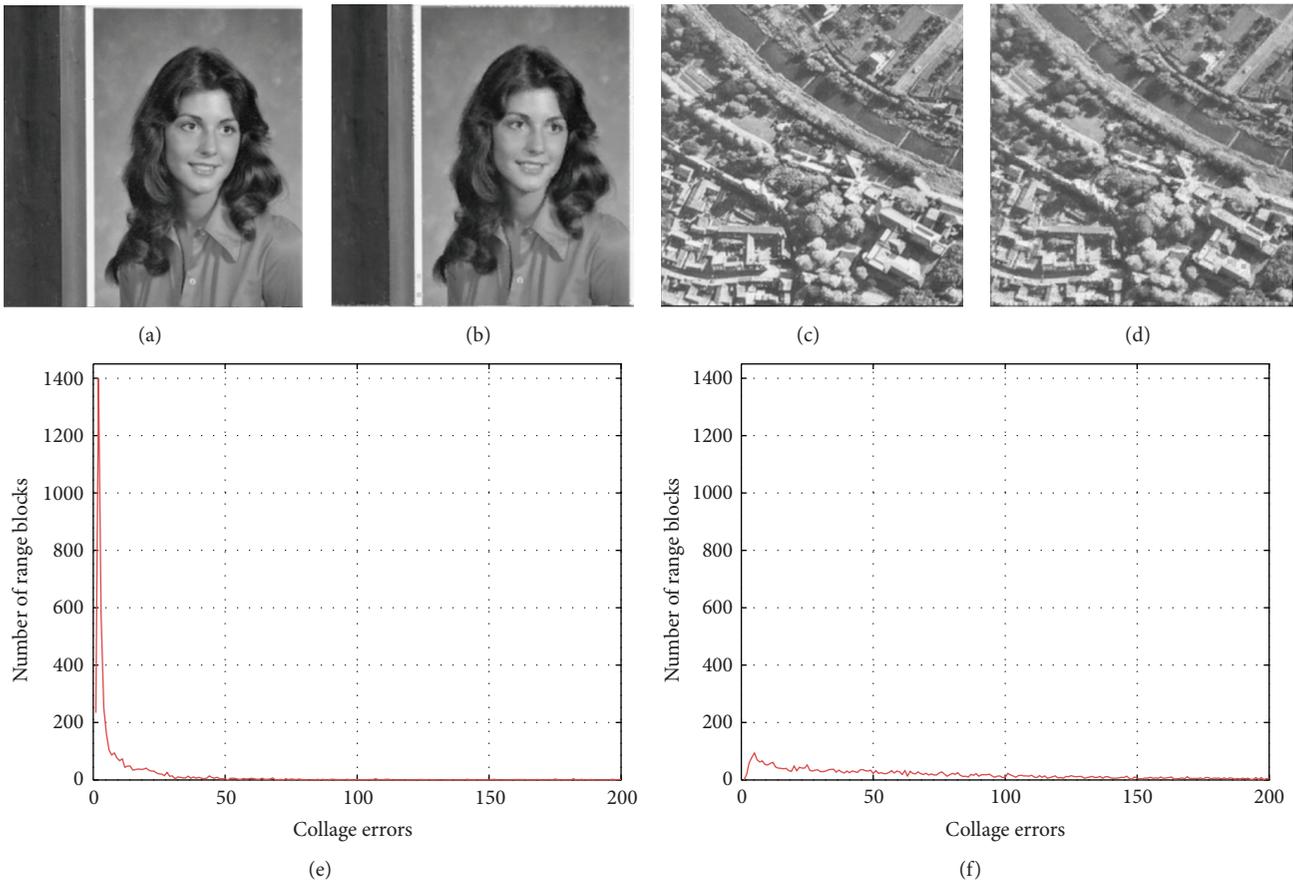


FIGURE 4: The PSNRs of the decoded images for different images. (a) and (c) Original images. (b) and (d) Decoded images of (a) and (c) and their PSNRs are 35.95 dB and 27.73 dB, respectively. (e) and (f) Distributions of collage errors for (a) and (c), respectively.

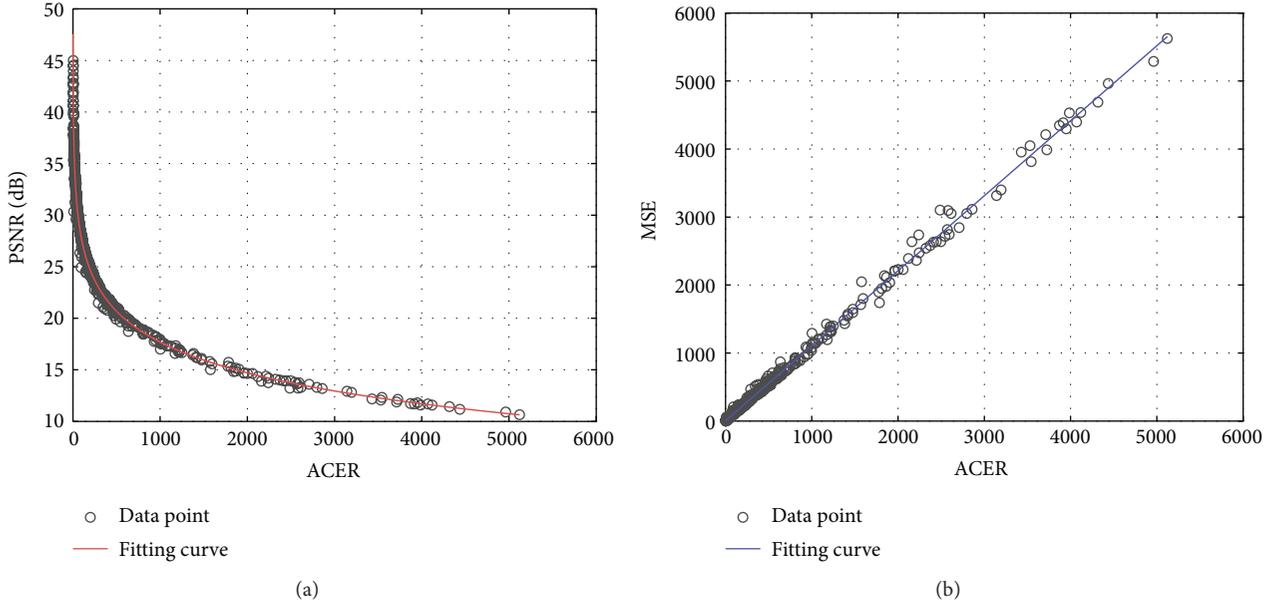


FIGURE 5: Relationship between two image quality measures and ACER. (a) PSNR versus ACER. (b) MSE versus ACER.

Assume

$$\mathbf{A}^T = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \log_{10}(x_1) & \log_{10}(x_2) & \cdots & \log_{10}(x_C) \end{pmatrix},$$

$$\mathbf{X} = \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}, \quad (7)$$

$$\mathbf{b}^T = (f(x_1) \ f(x_2) \ \cdots \ f(x_C)),$$

where  $T$  denotes the transpose operation. Then, we can express (6) as

$$\mathbf{A}\mathbf{X} = \mathbf{b}. \quad (8)$$

The least squares solution of (8) can be obtained by solving the system of equations of  $\mathbf{A}^T\mathbf{A}\mathbf{X} = \mathbf{A}^T\mathbf{b}$ , where  $\mathbf{X} = (\beta_1, \beta_2)^T$  is the coefficient estimation of (4). In Figure 5(a), the fitting curve is plotted with a red color. For all the test images, 92% of the deviations between the PSNRs and their corresponding values on the fitting curve are within the range of  $-0.5$  dB to  $0.5$  dB. With the fitting curve and ACER, we can predict the PSNR of the decoded image directly. Similar to PSNR versus ACER, we can easily derive that there will exist a linear relationship between mean squared error (MSE) and ACER which is also illustrated in Figure 5(b).

#### 4. Accelerated Version of the Prediction Method

According to the above statements, we know that although the fractal decoding is not needed, we need to perform the entire fractal encoding to get the ACER. In order to accelerate the method in Section 3, one way is to get the majority of collage errors with fewest range blocks encoded. In order

to facilitate the exposition, we firstly derive the following inequality. In order to minimize the matching error function (2), we set the derivatives of (2) with respect to  $\alpha$  and  $\beta$  to zeros as follows:

$$\frac{\partial \text{CE}}{\partial \alpha} = 0,$$

$$\frac{\partial \text{CE}}{\partial \beta} = 0. \quad (9)$$

By solving the above system of equations, we can obtain the optimal coefficients

$$\alpha = \frac{\langle \mathbf{R} - \bar{r}\mathbf{I}, \mathbf{D} - \bar{d}\mathbf{I} \rangle}{\|\mathbf{D} - \bar{d}\mathbf{I}\|^2},$$

$$\beta = \bar{r} - \alpha\bar{d}, \quad (10)$$

where  $\langle \cdot, \cdot \rangle$  denotes the inner product over vectors that are obtained from the  $n \times n$ -pixel range and contracted domain blocks by rearranging them as  $n^2$  vectors.  $\bar{d}$  and  $\bar{r}$  are the mean values of the contracted domain block and the range block, respectively. By substituting (10) back into (2), (2) can be simplified as

$$\begin{aligned} \text{CE}(\mathbf{R}) &= \|\mathbf{R} - \alpha\mathbf{D} - \beta\mathbf{I}\|^2 = \|\mathbf{R} - \alpha\mathbf{D} - (\bar{r} - \alpha\bar{d})\mathbf{I}\|^2 \\ &= \|(\mathbf{R} - \bar{r}\mathbf{I}) - \alpha(\mathbf{D} - \bar{d}\mathbf{I})\|^2 \\ &= \|\mathbf{R} - \bar{r}\mathbf{I}\|^2 + \alpha^2 \|\mathbf{D} - \bar{d}\mathbf{I}\|^2 - 2\alpha^2 \|\mathbf{D} - \bar{d}\mathbf{I}\|^2 \\ &= \|\mathbf{R} - \bar{r}\mathbf{I}\|^2 - \alpha^2 \|\mathbf{D} - \bar{d}\mathbf{I}\|^2 \leq \|\mathbf{R} - \bar{r}\mathbf{I}\|^2; \end{aligned} \quad (11)$$

from (11), we see that the collage error is smaller than its variance. In addition, we can also understand (11) as follows:

The variance is the squared collage error obtained in the special case when one sets the fractal parameter  $\alpha = 0$ ; that is, one finds the best constant approximation of the range block  $\mathbf{R}$ . This, of course, is the mean value  $\bar{r}$  of the range block. In other words, when  $\alpha = 0$ ,  $\beta = \bar{r}$ . Since only one fractal parameter is used, one expects the collage error to be equal to or higher than when two fractal parameters are used [26].

From (11), we know that the variances of range blocks provide an upper limit for collage errors and the range blocks with large collage errors should have corresponding large variances. Only the range blocks with large variances may have corresponding large collage errors. If the range blocks with large collage errors can contribute to the majority of total collage errors, the remaining range blocks with few collage errors can be neglected and the corresponding computational cost can be saved.

Based on the above statements, ahead of the fractal encoding procedures, the range blocks are sorted by their variances from largest to smallest and a range block sequence is established. Then, each range block in the sequence is encoded in order one by one and we can get the accumulated variance (AV) and accumulated collage error (ACE). By (11), we know that AV is larger than ACE. In order to estimate the percentage of ACE in the encoding process, we propose the following inequality:

$$\frac{\overline{ACE}}{\overline{ACE + \overline{AV}}} \leq \frac{\overline{ACE}}{\overline{ACE + \overline{ACE}}}, \quad (12)$$

where  $\overline{ACE}$  denotes the accumulated collage errors of the remaining range blocks except for the range blocks encoded. Thus,  $\overline{ACE + \overline{ACE}}$  represents the total collage errors and the right part of (12) represents the actual percentage ACE; we call it APACE.  $\overline{AV}$  belongs to the same range blocks as  $\overline{ACE}$ . Since each range block's variance is larger than its corresponding collage error,  $\overline{ACE}$  is smaller than  $\overline{AV}$  and the right part of the inequality will be always larger than the left part. Since it provides the lower limit of the percentage of ACE, we call it LLPACE.

In the practical encoding process, if we want to get the majority of total collage errors, we only need to guarantee that LLPACE is a large value, such as 90%; by (12), we can know that the corresponding APACE will account for more than 90% of total collage errors. Then, we can get the approximated ACER as follows:

$$ACER \approx \frac{ACE/Num}{((T_{LLPACE} + 100\%)/2)} \quad (13)$$

while LLPACE is a large value,

where Num denotes the total number of range blocks. Since the APACE is within the range of  $T_{LLPACE} - 100\%$ , the denominator of (13) is used to estimate the percentage of ACE. Here dividing Num aims to convert the partial ACE to the partial ACER. By making use of the approximated ACER and (5), the PSNR of the decoded image can be predicted directly.

We take four  $256 \times 256$  images, Lena, Bridge, Zelda, and Bird, for example, to illustrate the above proposed method.

In the encoder, firstly, the variances of range blocks are computed and sorted from the largest to smallest. As the number of range blocks encoded, proportional to the area encoded, increases, the corresponding LLPACE and APACE are plotted in Figure 6. In order to make the above procedures intuitive, Figure 7 illustrates the corresponding range blocks encoded in white boxes. We can observe the following key points:

- (i) Generally, while the range blocks with large variances account for a small proportion of total area, their corresponding LLPACE and APACE account for a large proportion.
- (ii) In accordance with (12), LLPACE is always smaller than APACE. In particular, while LLPACE reaches 90%, the corresponding APACE will be limited in a small dynamic range, that is, 90%–100%.

(i) shows us that we can get the majority of collage errors with only part of the range blocks encoded. (ii) tells us that LLPACE provides us with an effective way to estimate APACE. Thus, if some range blocks reach a large LLPACE, the corresponding APACE will be limited in a small dynamic range. For example, while some range blocks with large variances satisfy LLPACE = 90%, by (12), their corresponding ACE will account for 90%–100% of the total collage errors. By making use of the majority of total collage errors and (13), we can predict the quality of the decoded image by (5) directly. Simultaneously, if the range blocks only account for a relatively small area, the prediction method in Section 3 can be accelerated.

## 5. Applications of Our Proposed Method

In some practical applications, such as the production of videos with high quality, the PSNRs of decoded images are required to be above a certain value  $T_{PSNR}$ . Given one input image, we want to know whether fractal coding can satisfy people's demand. Generally speaking, we need to perform the encoding and decoding process entirely to determine whether fractal coding is suitable for it. In our research, we propose a more effective method to achieve this target. If the decoded image quality is required to be above a threshold  $T_{PSNR}$ , such as 30 dB, firstly, we need to get the threshold  $T_{ACER}$  of ACER by (14) which is the inverse function of (5). The corresponding threshold  $T_{ACE}$  of ACE can be obtained by  $T_{ACE} = T_{ACER} \times Num$ . Here Num also denotes the total number of range blocks. Then, in the fractal encoding process, after one range block is encoded, we compute the ACE and LLPACE. The ACE just exceeding the threshold  $T_{ACE}$  or LLPACE reaching 90% is considered as the terminating position. If the ACE exceeds the  $T_{ACE}$ , we know that fractal coding can not satisfy the practical demand. If the ACE is smaller than  $T_{ACE}$ , but the LLPACE reaches 90%, we can estimate the ACER with (13) and predict the final PSNR. We can also know whether fractal coding method is suitable in this case. In order to shorten the encoding process, the range blocks are sorted by their variances from largest to smallest and the range blocks are encoded one by one

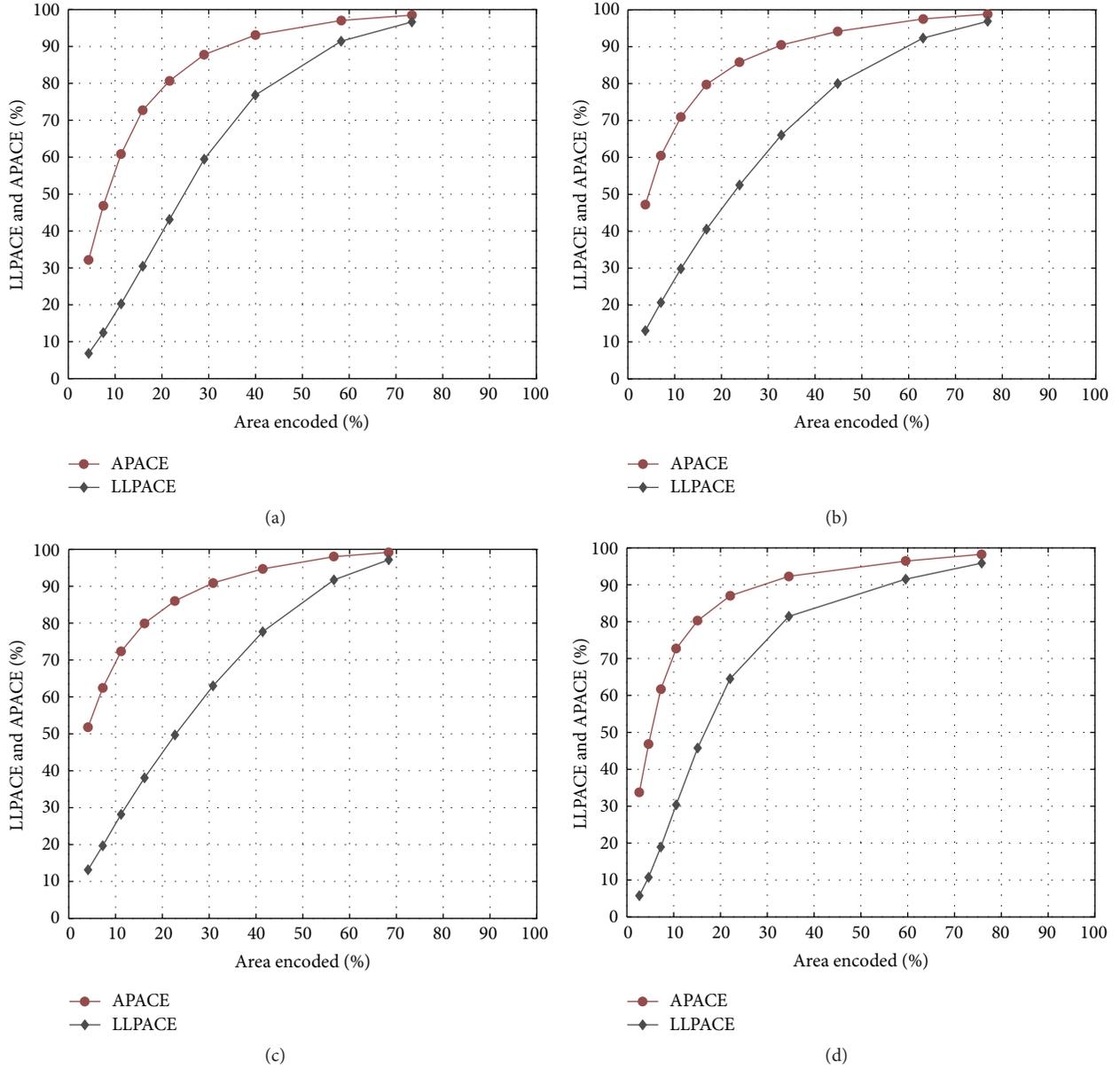


FIGURE 6: LLPACE and APACE versus encoded area. (a) Lena. (b) Zelda. (c) Couple. (d) Bird.

following this order. Note that the fractal decoding process is always not needed for our proposed method. The flow chart of our proposed method is shown in Figure 8:

$$T_{ACER} = 10^{(T_{PSNR} - \beta_1) / \beta_2}. \quad (14)$$

Figure 9 illustrates the above steps for a  $256 \times 256$  Lena image. While the size of range blocks is set to be  $8 \times 8$ , the actual PSNR of the decoded image is 25.54 dB. When  $T_{PSNR}$  is set to be 30 dB, the right part in Figure 9 illustrates the procedures from  $T_{PSNR}$  to  $T_{ACER}$ . By performing the encoding process in Figure 8, the left part of Figure 9 shows us the range blocks we need to encode. Only 5.27% of the total range blocks are needed to determine that fractal coding is not suitable for Lena image in this condition. Moreover, the fractal decoding process is not needed.

## 6. Experiment

In this section, five  $512 \times 512$  images, Lena, Barbara, Boat, Peppers, and Baboon, are used. The sizes of domain blocks and search step  $\delta$  are all set to be two times larger in each direction than that of the range blocks. The coefficients,  $\alpha$  and  $\beta$ , are quantized by 5 and 7 bits, respectively. The experiments are carried out on a Pentium Dual-Core 2.93 GHz PC and programmed using MATLAB software. We will use Jacquin's method and the other three fast fractal coding algorithms to assess the performance of our proposed method.

6.1. Predict the Quality of Decoded Images by Making Use of the Relationship between ACER and PSNR of the Decoded Image. In this part, besides Jacquin's fractal coding method [1], we also use Lai's [8], He's [9], and Wang's [10] fast fractal coding



FIGURE 7: Illustration of the percentages of APACE, LLPACE, and encoded area with different numbers of range blocks in the white boxes encoded. For Lena, (a) LLPACE: 30.46%, APACE: 72.77%, and encoded area: 15.92%. (b) LLPACE: 59.45%, APACE: 87.79%, and encoded area: 29.03%. (c) LLPACE: 91.43%, APACE: 97.03%, and encoded area: 58.30%. For Zelda, (d) LLPACE: 40.53%, APACE: 79.76%, and encoded area: 16.82%. (e) LLPACE: 66.05%, APACE: 90.44%, and encoded area: 32.81%. (f) LLPACE: 92.33%, APACE: 97.51%, and encoded area: 63.09%. For Couple, (g) LLPACE: 38.09%, APACE: 79.98%, and encoded area: 16.21%. (h) LLPACE: 62.99%, APACE: 90.90%, and encoded area: 30.83%. (i) LLPACE: 91.71%, APACE: 98.00%, and encoded area: 56.66%. For Bird, (j) LLPACE: 30.39%, APACE: 72.70%, and encoded area: 10.55%. (k) LLPACE: 64.53%, APACE: 86.99%, and encoded area: 22.07%. (l) LLPACE: 91.53%, APACE: 96.43%, and encoded area: 59.59%.

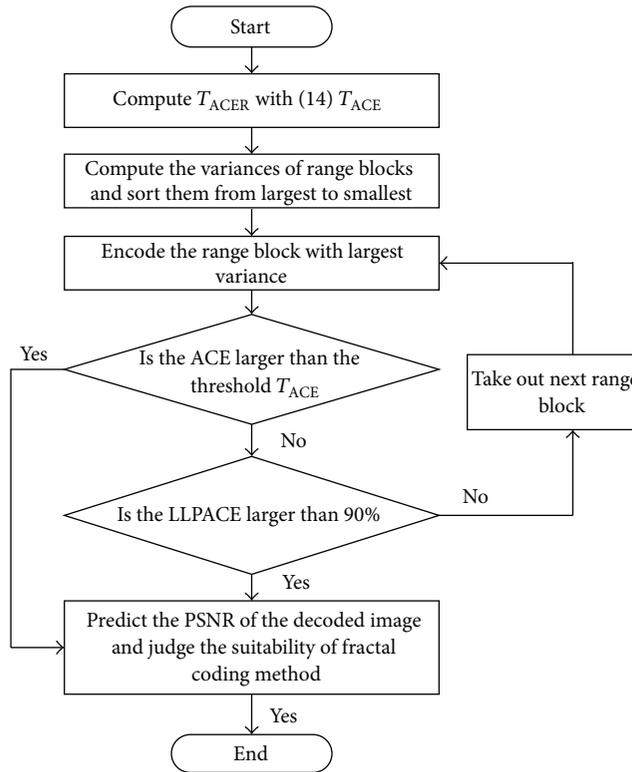


FIGURE 8: Flow chart of the proposed method.

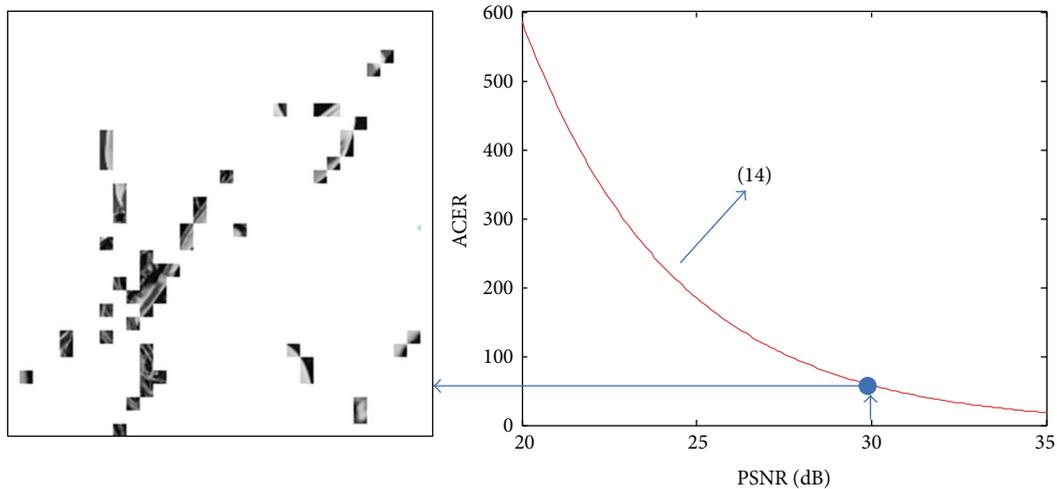


FIGURE 9: Terminating position in the fractal encoding process for Lena.

method to assess the accuracy of the method in Section 3. The detailed procedures can be described as follows.

*Step 1.* For one input image, perform Jacquin’s method and Lai’s, He’s, and Wang’s fast fractal coding method on it and get the actual PSNRs of the decoded images, respectively.

*Step 2.* In the above encoding processes, compute the ACER and predict the PSNR of the decoded image by (5).

*Step 3.* Compare the predicted PSNR with the actual PSNR of the decoded image.

Following the above steps, in the case of  $4 \times 4$  and  $8 \times 8$  range blocks, Tables 1 and 2 show the experimental results for four fractal coding methods, respectively. For each method, the first row to the fourth row list the predicted PSNRs, the actual PSNRs, their corresponding deviations, and the encoding times, respectively. In Table 1, for Jacquin’s

TABLE 1: Performance of the proposed prediction method for  $4 \times 4$  range blocks.

Images		Lena	Barbara	Boat	Peppers	Baboon
Jacquin's [1]	Predicted (dB)	36.43	29.86	33.31	35.79	26.77
	Decoded (dB)	36.67	29.47	33.41	35.95	26.52
	Deviation (dB)	0.24	0.39	0.10	0.16	0.25
	Time (s)	64.39	64.25	63.66	63.92	64.45
Lai's [8]	Predicted (dB)	35.62	28.60	31.86	34.38	25.30
	Decoded (dB)	35.77	28.45	31.79	34.51	25.06
	Deviation (dB)	0.15	0.15	0.07	0.13	0.24
	Time (s)	7.28	7.29	7.26	7.21	7.15
He's [9]	Predicted (dB)	35.60	28.45	32.43	34.76	25.79
	Decoded (dB)	35.81	28.28	32.50	35.04	25.33
	Deviation (dB)	0.21	0.17	0.07	0.28	0.46
	Time (s)	8.52	8.47	8.48	8.53	8.39
Wang's [10]	Predicted (dB)	35.87	28.84	32.58	34.98	26.02
	Decoded (dB)	36.16	28.69	32.69	35.16	25.54
	Deviation (dB)	0.29	0.15	0.11	0.18	0.48
	Time (s)	8.14	8.34	8.27	8.19	8.47

TABLE 2: Performance of the proposed prediction method for  $8 \times 8$  range blocks.

Images		Lena	Barbara	Boat	Peppers	Baboon
Jacquin's [1]	Predicted (dB)	29.88	25.12	26.74	29.67	21.67
	Decoded (dB)	30.03	24.80	26.97	29.64	21.64
	Deviation (dB)	0.15	0.32	0.23	0.03	0.03
	Time (s)	10.69	10.69	10.66	10.67	10.66
Lai's [8]	Predicted (dB)	29.42	24.49	25.99	29.46	21.39
	Decoded (dB)	29.49	24.50	26.00	29.47	21.53
	Deviation (dB)	0.07	0.01	0.01	0.01	0.14
	Time (s)	4.36	4.31	4.20	4.30	4.17
He's [9]	Predicted (dB)	29.45	24.44	26.32	29.38	21.45
	Decoded (dB)	29.53	24.36	26.39	29.49	21.49
	Deviation (dB)	0.08	0.08	0.07	0.11	0.04
	Time (s)	4.53	4.58	4.44	4.50	4.50
Wang's [10]	Predicted (dB)	29.36	24.62	26.24	29.34	21.47
	Decoded (dB)	29.47	24.46	26.39	29.24	21.49
	Deviation (dB)	0.11	0.16	0.15	0.10	0.02
	Time (s)	4.56	4.64	4.58	4.61	4.63

method, the average deviation is 0.23 dB. For the other three fast fractal coding methods, their average deviations are 0.15 dB, 0.24 dB, and 0.24 dB, respectively. In Table 2, the corresponding average deviations for the four methods are 0.15 dB, 0.05 dB, 0.08 dB, and 0.11 dB, respectively. They are all below 0.3 dB.

*6.2. Assess the Performance of the Accelerated Prediction Method.* In this part, we will compare the performance of the prediction method with its accelerated version. Based on

the statements in Section 4, the corresponding experimental procedures can be described as follows.

*Step 1.* For one input image, perform Jacquin's method and Lai's, He's, and Wang's fast fractal coding method on it and get the actual PSNRs of the decoded images, respectively.

*Step 2.* Partition the input image uniformly into range blocks and compute the variances of the range blocks. Sort the range blocks by their variances from largest to smallest.

TABLE 3: Performance of the accelerated prediction method for  $4 \times 4$  range blocks.

	Images	Lena	Barbara	Boat	Peppers	Baboon
Jacquin's [1]	Predicted (dB)	36.53	29.73	33.27	35.75	26.67
	Decoded (dB)	36.67	29.47	33.41	35.95	26.52
	Deviation (dB)	0.14	0.26	0.14	0.20	0.15
	Computations needed	69.73%	49.03%	72.15%	73.32%	69.56%
	Time (s)	45.31	31.97	46.38	47.39	45.30
Lai's [8]	Predicted (dB)	35.73	28.50	31.86	34.38	25.24
	Decoded (dB)	35.77	28.45	31.79	34.51	25.06
	Deviation (dB)	0.04	0.05	0.07	0.13	0.18
	Computations needed	66.23%	45.70%	66.64%	67.98%	64.90%
	Time (s)	4.92	3.41	4.88	4.97	4.70
He's [9]	Predicted (dB)	35.91	28.37	32.49	34.94	25.73
	Decoded (dB)	35.81	28.28	32.50	35.04	25.33
	Deviation (dB)	0.10	0.09	0.01	0.10	0.40
	Computations needed	66.92%	45.37%	69.15%	70.21%	66.52%
	Time (s)	5.69	3.89	5.88	6.02	5.59
Wang's [10]	Predicted (dB)	36.04	28.72	32.59	35.03	25.94
	Decoded (dB)	36.16	28.69	32.69	35.16	25.54
	Deviation (dB)	0.12	0.03	0.10	0.13	0.40
	Computations needed	67.83%	46.29%	69.65%	70.57%	67.21%
	Time (s)	5.59	3.97	5.81	5.86	5.70

*Step 3.* Following the order established in Step 2, take one range block, encode it, and compute the LLPACE by (12). If LLPACE is smaller than the threshold  $T_{LLPACE}$ , such as 90%, take the next range block and continue encoding it; if not, record the ACE and go to Step 4.

*Step 4.* Estimate the ACER by (13) and predict the PSNR of the decoded image by (5).

*Step 5.* Compare the predicted PSNR with the actual PSNR of the decoded image.

Similar to Tables 1 and 2, Tables 3 and 4 illustrate the experimental results of the accelerated prediction method for  $4 \times 4$  and  $8 \times 8$  range blocks, respectively. Moreover, the theoretical amounts of computations needed are also added in the fourth row for each method. With respect to the fourth row in Tables 1 and 2 and the fifth row in Tables 3 and 4, the percentage of computations needed in row four of Tables 3 and 4 can be verified by the actual encoding times, for example, for  $4 \times 4$  range blocks, performing Jacquin's method on Lena,  $45.31/64.39 = 70.37\% \approx 69.73\%$ , and for  $8 \times 8$  range blocks, performing He's method on Boat,  $2.52/4.44 = 56.76\% \approx 56.93\%$ . In Table 3, the average deviations of the four fractal coding methods are 0.18 dB, 0.09 dB, 0.14 dB, and 0.16 dB, respectively. They are also below 0.3 dB. Their average percentages of the computations needed are 66.76%, 62.29%, 63.63%, and 64.31%, respectively. In Table 4, the average deviations of the four fractal coding methods are 0.16 dB, 0.12 dB, 0.18 dB, and 0.21 dB, respectively. Their percentages of the computations needed are 56.06%, 54.66%, 55.25%, and 55.19%, respectively. As a summary, compared with the

prediction method in Section 3, the accelerated prediction method can provide competitive performance and about one-third of the computations in fractal encoding can be saved.

*6.3. Application of the Proposed Prediction Method.* In this part, the size of range blocks is set to be  $4 \times 4$ . We take  $512 \times 512$  Lena image as the test image, perform Jacquin's method on it, and simulate the practical applications with different thresholds  $T_{PSNR}$ s. The detailed procedures are listed as follows.

*Step 1.* According to the practical demands, set the threshold  $T_{PSNR}$  as the lower limit of the quality of decoded images, such as 30 dB.

*Step 2.* Corresponding to the threshold  $T_{PSNR}$  in Step 1, compute the  $T_{ACER}$  by (14), multiply it by the total number of range blocks, and get the threshold  $T_{ACE}$ . For arbitrary input image, partition it uniformly into range blocks, sort them from largest to smallest by their variances, and encode them one by one.

*Step 3.* Take out one range block, encode it, and add its collage error to the ACE. If the ACE is larger than  $T_{ACE}$ , the decoded image quality will be smaller than 30 dB, and stop the encoding process. If not, go to Step 4.

*Step 4.* Compute the LLPACE. If LLPACE is smaller than 90%, return to Step 3 and encode the next range block. If not, stop the encoding process and compute the corresponding ACER with (13) and predict the PSNR of the decoded image. Whether fractal coding method is suitable can be determined.

TABLE 4: Performance of the accelerated prediction method for  $8 \times 8$  range blocks.

	Images	Lena	Barbara	Boat	Peppers	Baboon
Jacquin's [1]	Predicted (dB)	29.95	25.07	26.78	29.82	21.70
	Decoded (dB)	30.03	24.80	26.97	29.64	21.64
	Deviation (dB)	0.08	0.27	0.19	0.18	0.06
	Computations needed	52.20%	49.81%	57.72%	57.59%	62.96%
	Time (s)	5.61	5.36	6.17	6.17	6.74
Lai's [8]	Predicted (dB)	29.60	24.54	26.17	29.66	21.44
	Decoded (dB)	29.49	24.50	26.00	29.47	21.53
	Deviation (dB)	0.11	0.04	0.17	0.19	0.09
	Computations needed	50.76%	48.44%	55.27%	56.76%	62.09%
	Time (s)	2.28	2.17	2.36	2.50	2.63
He's [9]	Predicted (dB)	29.83	24.61	26.59	29.63	21.51
	Decoded (dB)	29.53	24.36	26.39	29.49	21.49
	Deviation (dB)	0.30	0.25	0.20	0.14	0.02
	Computations needed	51.69%	48.66%	56.93%	56.62%	62.33%
	Time (s)	2.36	2.28	2.52	2.59	2.86
Wang's [10]	Predicted (dB)	29.75	24.75	26.51	29.59	21.53
	Decoded (dB)	29.47	24.46	26.39	29.24	21.49
	Deviation (dB)	0.27	0.29	0.12	0.35	0.04
	Computations needed	51.54%	49.02%	56.59%	56.45%	62.35%
	Time (s)	2.42	2.34	2.66	2.66	2.95

For Lena image, while  $T_{\text{PSNR}}$  varies from 35 dB to 41 dB, Figure 10 illustrates the corresponding percentages of the computations needed in fractal encoding. Since the actual PSNR of the decoded image for Lena image is 36.67 dB, if  $T_{\text{PSNR}}$  is equal to or smaller than 36 dB, the fractal coding method can satisfy the practical demand. According to Table 3, we know that we always need 69.73% of the computations in fractal encoding. If  $T_{\text{PSNR}}$  is set to be above 37 dB, the fractal coding method can not satisfy the practical demand. As  $T_{\text{PSNR}}$  increases, the percentage of the computations needed decreases. If  $T_{\text{PSNR}}$  is set to be 41 dB or larger, the percentage of the computations in fractal encoding approximates 0%. That is to say, in this case, if we want to know whether fractal coding method can satisfy the practical demand or not, we only need to encode a small number of range blocks.

## 7. Conclusion

Based on the logarithmic relationship between the ACER and PSNR of the decoded image, a prediction method is proposed to estimate the PSNR of the decoded image in fractal encoding process. Furthermore, by theoretical derivations and analyses, an accelerated version of the above prediction method is also introduced. Experiments show that the proposed method can predict the quality of decoded images accurately. The accelerated prediction method can provide competitive performance while about one-third of total computations in fractal encoding process can be saved. In addition, the proposed method can be also applied in practical applications and simulations verify its effectiveness.

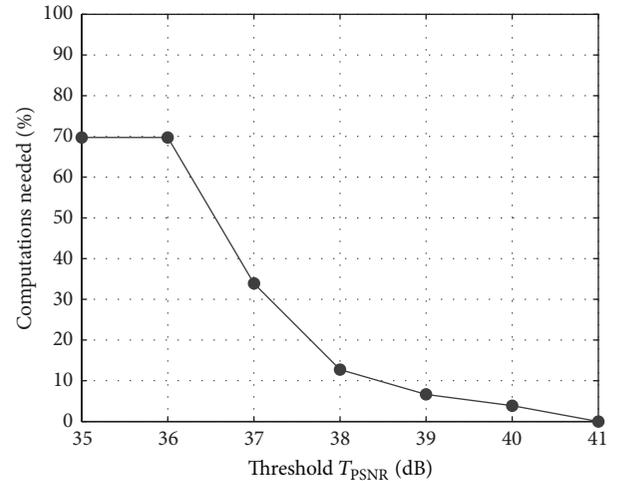


FIGURE 10: Illustration of the amounts of computations with different  $T_{\text{PSNR}}$ s.

In the future research, we will attempt to explore the relationship between the quality of decoded images and some unknown features of the input image, so that we can predict the quality of decoded image more accurately. It really has theoretical as well as practical meanings.

## Conflict of Interests

The authors declare that they have no conflict of interests regarding the publication of this paper.

## Acknowledgments

This work is partially supported by the Fundamental Research Funds for the Central Universities (Grant no. 3132014090) and the National 863 Program of China (Grant no. 2011AA110201).

## References

- [1] A. E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Transactions of Image Processing*, vol. 1, no. 1, pp. 18–30, 1992.
- [2] M. F. Barnsley, *Fractals Everywhere*, Academic Press, New York, NY, USA, 1993.
- [3] A. E. Jacquin, "Fractal image coding: a review," *Proceedings of the IEEE*, vol. 81, no. 10, pp. 1451–1465, 1993.
- [4] Y. Fisher, Ed., *Fractal Image Encoding and Analysis*, vol. 159 of NATO ASI Series F, Springer, Heidelberg, Germany, 1998.
- [5] B. Wohlberg and G. de Jager, "A review of the fractal image coding literature," *IEEE Transactions on Image Processing*, vol. 8, no. 12, pp. 1716–1729, 1999.
- [6] Y. Fisher, *Fractal Image Compression: Theory and Application*, Springer, New York, NY, USA, 1995.
- [7] B. Hurtgen and C. Stiller, "Fast hierarchical codebook search for fractal coding of still images," in *Video Communications and PACS for Medical Applications*, vol. 1977 of *Proceedings of SPIE*, pp. 397–408, Berlin, Germany, October 1993.
- [8] C.-M. Lai, K.-M. Lam, and W.-C. Siu, "Improved searching scheme for fractal image coding," *Electronics Letters*, vol. 38, no. 25, pp. 1653–1654, 2002.
- [9] C. He, X. Xu, and J. Yang, "Fast fractal image encoding using one-norm of normalised block," *Chaos, Solitons & Fractals*, vol. 27, no. 5, pp. 1178–1186, 2006.
- [10] Q. Wang, D. Liang, and S. Bi, "Fast fractal image encoding based on correlation information feature," in *Proceedings of the 3rd International Congress on Image and Signal Processing (CISP '10)*, pp. 540–543, IEEE, Yantai, China, October 2010.
- [11] J. H. Freidman, J. L. Bentley, and R. A. Finkel, "An algorithm for finding best matches in logarithmic expected time," *ACM Transactions on Mathematical Software*, vol. 3, no. 3, pp. 209–226, 1977.
- [12] S. Furaio and O. Hasegawa, "A fast no search fractal image coding method," *Signal Processing: Image Communication*, vol. 19, no. 5, pp. 393–404, 2004.
- [13] X.-Y. Wang and S.-G. Wang, "An improved no-search fractal image coding method based on a modified gray-level transform," *Computers & Graphics*, vol. 32, no. 4, pp. 445–450, 2008.
- [14] X.-Y. Wang, Y.-X. Wang, and J.-J. Yun, "An improved no search fractal image coding method based on a fitting plane," *Image and Vision Computing*, vol. 28, no. 8, pp. 1303–1308, 2010.
- [15] S. Bi and Q. Wang, "Fractal image coding based on a fitting surface," *Journal of Applied Mathematics*, vol. 2014, Article ID 634848, 11 pages, 2014.
- [16] M. Ghazel, G. H. Freeman, and E. R. Vrscay, "Fractal image denoising," *IEEE Transactions on Image Processing*, vol. 12, no. 12, pp. 1560–1578, 2003.
- [17] M. Ghazel, G. H. Freeman, and E. R. Vrscay, "Fractal-wavelet image denoising revisited," *IEEE Transactions on Image Processing*, vol. 15, no. 9, pp. 2669–2675, 2006.
- [18] Y. C. Wee and H. J. Shin, "A novel fast fractal super resolution technique," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp. 1537–1541, 2010.
- [19] C.-M. Lai, K.-M. Lam, Y.-H. Chan, and W.-C. Siu, "An efficient fractal-based algorithm for image magnification," in *Proceedings of the International Symposium on Intelligent Multimedia, Video and Speech Processing (ISIMP '04)*, pp. 571–574, IEEE, Hong Kong, China, October 2004.
- [20] K.-H. Chung, Y.-H. Fung, and Y.-H. Chan, "Image enlargement using fractal," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 273–275, IEEE, Hong Kong, April 2003.
- [21] X.-Y. Wang and Z.-F. Chen, "A fast fractal coding in application of image retrieval," *Fractals*, vol. 17, no. 4, pp. 441–450, 2009.
- [22] H. Xiaoqing, Z. Qin, and L. Wenbo, "A new method for image retrieval based on analyzing fractal coding characters," *Journal of Visual Communication and Image Representation*, vol. 24, no. 1, pp. 42–47, 2013.
- [23] J. Lu, Y. Zou, C. Yang, and L. Wang, "A robust fractal color image watermarking algorithm," *Mathematical Problems in Engineering*, vol. 2014, Article ID 638174, 12 pages, 2014.
- [24] M. H. Pi, C. H. Li, and H. Li, "A novel fractal image watermarking," *IEEE Transactions on Multimedia*, vol. 8, no. 3, pp. 488–499, 2006.
- [25] F. Daraee and S. Mozaffari, "Watermarking in binary document images using fractal codes," *Pattern Recognition Letters*, vol. 35, no. 1, pp. 120–129, 2014.
- [26] S. K. Alexander, E. R. Vrscay, and S. Tsurumi, "A simple, general model for the affine self-similarity of images," in *Proceedings of the International Conference on Image Analysis and Recognition (ICIAR '08)*, A. Campilho and M. Kamel, Eds., vol. 5112 of *Lecture Notes on Computer Sciences*, pp. 192–203, 2008.
- [27] USC-SIPI image database, University of Southern California, <http://sipi.usc.edu/services/database>.
- [28] Vision Texture Database, Media Laboratory at MIT, <http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html>.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

