Fractal Decoded Image Quality Prediction

Authors: Wang, Qiang, Du, Jingjing, Wang, Qiang

Date: 2023-08-25T00:00:00+00:00

Abstract

To predict fractal decoded image quality more efficiently, an effective decoded image quality prediction method was proposed in this study. In fractal encoding process, the dynamic range of the linear correlation coefficients (LCCs) between range blocks and their best-matched domain blocks was greatly extended by several outliers which increased uncertainty and resulted in reduced prediction accuracy. To remove the interference of outliers, we introduced the effective minimum and maximum of LCCs, which provided the effective bottom and top limits of the actual percentage of accumulated collage error (EBL-APACE and ETL-APACE), respectively. Further, when EBL-APACE reached a large percentage, the average collage error (ACER) can be estimated, and the decoded image quality can be predicted directly.Experimental results show that compared with the previous method, the proposed method can provide higher prediction accuracy with fewer computations.

Full Text

Preamble

Fractal Decoded Image Quality Prediction

Qianga Wang and **Jingjingb Du**
aDalian Maritime University, College of Information Science and Technology, Linghai Road, No.1, Dalian, China, 116026
bHengshui University, College of Electronics and Information Engineering, Heping Road, No.1088, Hengshui, China, 053010
Corresponding Author, E-mail: wangqiang2011@dlmu.edu.cn

Abstract: To predict fractal decoded image quality more efficiently, this study proposes an effective decoded image quality prediction method. In the fractal encoding process, the dynamic range of the linear correlation coefficients (LCCs) between range blocks and their best-matched domain blocks is greatly extended by several outliers, which increases uncertainty and reduces prediction accuracy.

To remove the interference of outliers, we introduce the effective minimum and maximum of LCCs, which provide the effective bottom and top limits of the actual percentage of accumulated collage error (EBL-APACE and ETL-APACE), respectively. Furthermore, when EBL-APACE reaches a large percentage, the average collage error (ACER) can be estimated, and the decoded image quality can be predicted directly.

Experimental results show that compared with the previous method, the proposed method can provide higher prediction accuracy with fewer computations.

Keywords: Fractal image coding; Decoded image quality; Accumulated collage error; Average collage error

1. Introduction

With the rapid development of modern society, people are increasingly exposed to images in their daily lives, which creates substantial pressure for image transmission and storage. Efficient image compression technology represents one of the most important solutions to these challenges. Unlike conventional image compression techniques, fractal image coding offers novel concepts, potential high compression ratios, fast decoding, and resolution independence, attracting worldwide research attention [1-4].

However, fractal image coding suffers from high computational complexity during the encoding process, which seriously impedes its practical application. Consequently, fast fractal image encoding has become a key research direction. In recent years, many fast fractal encoding methods have been proposed, which can be mainly divided into two categories: 1) Local block-matching based fast fractal image encoding [5-7], which accelerates the encoding process by excluding numerous domain blocks and converting global matching into local matching; and 2) No-search fractal image coding [8-11], which assigns the best-matched domain blocks directly without search operations, enabling real-time encoding and higher compression ratios at the expense of decoded image quality. Beyond image compression, fractal image coding has been gradually applied to other image processing tasks, including image denoising [12-18], image hashing [19,20], image retrieval [21-23], watermarking [24-26], image super-resolution [27-31], head pose estimation [32, 33], and MRI image processing [34].

Predictable quality of decoded images is a novel property of fractal image coding. In previous work, the quality of decoded images was predicted using only partial encoding information without requiring the decoding process [35]. This study proposes an effective method to predict decoded image quality more accurately and quickly. First, we observed that the dynamic range of linear correlation coefficients between range blocks and their best-matched domain blocks is greatly extended by several outliers. Since the number of outliers is very small, they contribute little to the accumulated collage error (ACE) of all range blocks. Therefore, we propose a novel method to calculate the effective minimum and maximum of linear correlation coefficients to eliminate outlier interference, which provides the effective bottom and top limits of the actual percentage of accumulated collage error (EBL-APACE and ETL-APACE), respectively. Moreover, when EBL-APACE reaches a large percentage (such as 90%), EBL-APACE and ETL-APACE determine a small dynamic range for the actual percentage of accumulated collage error (APACE), allowing the average collage error (ACER) to be approximately estimated. Based on the logarithmic relationship between ACER and fractal decoded image quality, the quality of decoded images can be predicted directly. Finally, we adopted four fractal encoding methods to assess the performance of the proposed method. Experimental results demonstrate that compared with the previous method, the proposed method provides higher prediction accuracy with fewer computations.

This paper is organized as follows: Section 2 reviews conventional fractal image coding. Section 3 describes the principle of the proposed method. Section 4 presents and analyzes experimental results in detail. Section 5 provides the final conclusion.

2. Conventional Fractal Image Coding

Fractal image coding aims to establish an iterated function system whose fixed point can approximate the original image. The first block-based fractal encoding method was proposed by Jacquin [1]. First, the original $M \times N$ image f is uniformly divided into non-overlapping $B \times B$ range blocks R_i, $i = 1, 2, 3, \ldots, NumR$, where $NumR$ denotes the total number of range blocks. Then, domain blocks D_j, $j = 1, 2, 3, \ldots, NumD$, can be obtained by sliding a $2B \times 2B$ window over the original image from left to right and top to bottom, where $NumD$ denotes the total number of domain blocks. Further, domain blocks are contracted to the same size as range blocks, and an extended domain block pool (EDBP) is obtained by performing eight isometric transformations on domain blocks. For a given image, Figure 1 [Figure 1: see original paper] illustrates the eight isometric transformations, which include rotation through 0°, 90°, 180°, and 270°, reflection about the middle horizontal axis, reflection about the middle vertical axis, reflection about the main diagonal, and reflection about the secondary diagonal, in that order. Finally, the best-matched domain block can be obtained by minimizing the following function:

$$\min_{\substack{i=1,2,3,\ldots,NumR \\ j=1,2,3,\ldots,NumD}} CE(R_i)$$

where $CE(R_i)$ denotes the collage error of R_i. I denotes a $B \times B$ matrix whose elements are all ones. s_i and o_i denote scaling and offset coefficients, respectively. D_j is the best-matched domain block of R_i. Using the least squares method, s_i and o_i can be calculated as follows:

$$s_i = \frac{\langle R_i - \bar{r}I, D_j - \bar{d}I \rangle}{\langle D_j - \bar{d}I, D_j - \bar{d}I \rangle}$$

$$o_i = \bar{r} - s_i\bar{d}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product. $\bar{r}$ and $\bar{d}$ are the averages of R_i and D_j, respectively.

Fig.1 Eight isometric transformations.

In the decoding process, an arbitrary $M \times N$ image can be selected as the initial image. First, the same transformations as those in the encoding process are applied from the best-matched domain blocks to the corresponding range blocks. One iteration is completed after all range blocks are reconstructed. After approximately 10 iterations, the decoded image can be obtained.

Figure 2 [Figure 2: see original paper] selects a rotated 256×256 Bridge image as the initial image. When we encoded Bridge in the encoding process, Figures 3 [Figure 3: see original paper] (b) to (f) illustrate the first five iteration images in the decoding process. We observe that the iteration images converge gradually to Bridge.

(a) Initial Image (b) First iteration (c) Second iteration (d) Third iteration (e) Fourth iteration (f) Fifth iteration

Fig.2 Fractal decoding process.

3.1 Effective Minimum and Maximum of Linear Correlation Coefficients Between Range Blocks and Their Best-Matched Domain Blocks

For R_i, substituting (2) back into (1) yields:

$$CE(R_i) = (1 - LCC_i^2) \cdot Var(R_i)$$

where LCC_i denotes the linear correlation coefficient between R_i and D_j. Because LCC_i is between 0 and 1, Eq. (3) satisfies:

$$0 \leq CE(R_i) \leq Var(R_i)$$

In Eq. (4), the variance of R_i provides the upper limit of $CE(R_i)$, and thus only range blocks with large variances may result in large collage errors. By Eq. (3), the ACE of all range blocks can be calculated as:

$$ACE_{All} = \sum_{i=1}^{NumR} (1 - LCC_i^2) \cdot Var(R_i)$$

If we use LCC_{Max} and LCC_{Min} to denote the maximum and minimum of LCC_i, $i = 1, 2, 3, \ldots, NumR$, respectively, then by Eq. (5), we have:

$$ACE_{Min} < ACE_{All} < ACE_{Max}$$

where:

$$ACE_{Min} = \sum_{i=1}^{NumR} (1 - LCC_{Max}^2) \cdot Var(R_i)$$

$$ACE_{Max} = \sum_{i=1}^{NumR} (1 - LCC_{Min}^2) \cdot Var(R_i)$$

Fig.3 Distribution of linear correlation coefficients.

In the encoding process, we sorted the range blocks by their variances from largest to smallest and encoded them in order. For Bridge, Figure 3 shows the distribution of LCCs. In Eq. (7), LCC_{Max} and LCC_{Min} provide the bottom and top limits of ACE_{All}, respectively. However, we observe that the dynamic range of LCCs is mainly determined by several outliers that greatly extend the dynamic range of LCCs. In Fig.3, all LCCs below the red line (i.e., smaller than 0.6) are marked with " ". The outliers cause LCC_{Min} to extend below 0.6. Because the ACE of these outliers comprises only 1.47% of the total collage errors, their effect can be ignored. To remove outlier interference and effectively estimate the dynamic range of ACE_{All}, we adopt the following method to compute the effective minimum and maximum of LCCs:

For all range blocks, we establish a histogram of collage errors with respect to LCCs. If we have $NumC$ range blocks R_k, $k = 1, 2, 3, \ldots, NumC$, whose LCCs are all lcc, we use $p(lcc)$ to represent the normalized version of the sum of collage errors with the same LCC:

$$p(lcc) = \frac{\sum_{k=1}^{NumC} CE_k(lcc)}{ACE_{All}}$$

where $CE_k(lcc)$ denotes the collage error of R_k with $LCC = lcc$. From Eq. (8), we know that $p(lcc)$ actually represents the probability density function of collage errors (PDF-CE) with respect to LCCs.

Fig.4 PDF-CE versus LCCs. (a) Fitting result of the left part and its reflection. (b) Fitting result of the left part and its reflection.

Fig.5 Fitting results for the left and right parts of PDF-CE in Fig.5, respectively.

Fig.6 Fitting results of PDF-CE.

Figure 4 [Figure 4: see original paper] shows PDF-CE for Bridge, which is asymmetric. We plotted a vertical line (VL) at $lcc = T$ where PDF-CE reached its peak value. The VL divides PDF-CE into left and right parts comprising $T_{Left}\%$ and $T_{Right}\%$ of ACE_{All}, respectively, and we have $T_{Left}\% + T_{Right}\% = 100\%$. By extracting the left part and reflecting it about the VL, we obtain a symmetric curve in Fig.5 (a). Similarly, by reflecting the right part about the VL, we obtain a symmetric curve in Fig.5 (b). Then, we adopt the following two Gaussian functions, $GF_{Left}(lcc)$ and $GF_{Right}(lcc)$, to fit the two curves in Fig.5 (a) and (b), respectively:

$$GF_{Left}(lcc) = \frac{1}{\sqrt{2\pi}\alpha_3} \exp\left(-\frac{(lcc - \alpha_2)^2}{2\alpha_3^2}\right)$$

$$GF_{Right}(lcc) = \frac{1}{\sqrt{2\pi}\beta_3} \exp\left(-\frac{(lcc - \beta_2)^2}{2\beta_3^2}\right)$$

where α_3 and β_3 represent the standard deviations of the curves in Fig.5 (a) and (b), respectively. α_2 and β_2 are both equal to T. Gaussian distributions concentrate $95.5\% \times 2T_{Left}\%$ and $95.5\% \times 2T_{Right}\%$ of ACE_{All} in the intervals $[\alpha_2 - 2\alpha_3, \alpha_2 + 2\alpha_3]$ and $[\beta_2 - 2\beta_3, \beta_2 + 2\beta_3]$ in Fig.5 (a) and (b), respectively. Thus, $[\alpha_2 - 2\alpha_3, \alpha_2]$ and $[\beta_2, \beta_2 + 2\beta_3]$ concentrate $95.5\% \times T_{Left}\%$ and $95.5\% \times T_{Right}\%$ of ACE_{All}, respectively. Further, we extract the left and right parts of the fitting curves in Fig.5 (a) and (b), respectively, and combine them in Fig.6. By Eq. (9), we know that the interval $[\alpha_2 - 2\alpha_3, \beta_2 + 2\beta_3]$ concentrates 95.5% of ACE_{All}. Finally, we define $\alpha_2 - 2\alpha_3$ and $\beta_2 + 2\beta_3$ as the effective minimum and maximum of LCCs, $LCC_{Min\_Effective}$ and $LCC_{Max\_Effective}$, respectively:

$$LCC_{Min\_Effective} = \alpha_2 - 2\alpha_3$$

$$LCC_{Max\_Effective} = \beta_2 + 2\beta_3$$

$LCC_{Min\_Effective}$ and $LCC_{Max\_Effective}$ can effectively remove the effect of outliers. Because the dynamic range of LCCs is narrowed, we also have the inequalities:

$$LCC_{Min\_Effective} \leq LCC_{Min} \leq LCC_{Max} \leq LCC_{Max\_Effective}$$

3.2 Effective Bottom and Top Limits of Accumulated Collage Error

With $LCC_{Min\_Effective}$ and $LCC_{Max\_Effective}$, Eq. (7) can be modified as:

$$ACE_{Min\_Effective} < ACE_{All} < ACE_{Max\_Effective}$$

where:

$$ACE_{Min\_Effective} = \sum_{i=1}^{NumR} (1 - LCC^2_{Max\_Effective}) \cdot Var(R_i)$$

$$ACE_{Max\_Effective} = \sum_{i=1}^{NumR} (1 - LCC^2_{Min\_Effective}) \cdot Var(R_i)$$

We compared ACE_{All} with $ACE_{Min\_Effective}$:

$$ACE_{Min\_Effective} = \sum_{i=1}^{NumR} (1 - LCC^2_{Max\_Effective}) \cdot Var(R_i) = \sum_{i=1}^{NumR} CE(R_i) \cdot \frac{1 - LCC^2_{Max\_Effective}}{1 - LCC^2_i}$$

The range blocks can be divided into two categories: those with LCCs above $LCC_{Max\_Effective}$ and those below. We use $NumAbove$ and $NumBelow$ to denote the numbers of range blocks whose LCCs are above and below $LCC_{Max\_Effective}$, respectively. Then, Eq. (14) can be rewritten as:

$$ACE_{Min\_Effective} = \sum_{i=1}^{NumAbove} CE(R_i) \cdot \frac{1 - LCC^2_{Max\_Effective}}{1 - LCC^2_i} + \sum_{i=1}^{NumBelow} CE(R_i) \cdot \frac{1 - LCC^2_{Max\_Effective}}{1 - LCC^2_i}$$

Only the LCCs of several outliers satisfy $LCC_i > LCC_{Max\_Effective}$, while the others satisfy $LCC_i < LCC_{Max\_Effective}$. For Bridge, $NumAbove = 39$, $NumBelow = 4,057$, and $ACE_{Min\_Effective} = 529,480 > 0$. Generally, the number of LCCs below $LCC_{Max\_Effective}$ is more than ten times larger than those above. Thus, in Eq. (15), when $NumBelow \gg NumAbove$, $ACE_{Min\_Effective} < ACE_{All}$, and we have:

$$ACE_{All} > ACE_{Min\_Effective}$$

Similarly, we compared ACE_{All} with $ACE_{Max\_Effective}$:

$$ACE_{Max\_Effective} = \sum_{i=1}^{NumR} (1-LCC^2_{Min\_Effective}) \cdot Var(R_i) = \sum_{i=1}^{NumR} CE(R_i) \cdot \frac{1 - LCC^2_{Min\_Effective}}{1 - LCC_i^2}$$

Analogous to Eq. (15), we divided range blocks into two categories: those with LCCs above and below $LCC_{Min\_Effective}$. We use $NumAbove$ and $NumBelow$ to denote the numbers of range blocks whose LCCs are above and below $LCC_{Min\_Effective}$, respectively. Then, Eq. (17) can be rewritten as:

$$ACE_{Max\_Effective} = \sum_{i=1}^{NumAbove} CE(R_i) \cdot \frac{1 - LCC^2_{Min\_Effective}}{1 - LCC_i^2} + \sum_{i=1}^{NumBelow} CE(R_i) \cdot \frac{1 - LCC^2_{Min\_Effective}}{1 - LCC_i^2}$$

Only the LCCs of several outliers satisfy $LCC_i < LCC_{Min\_Effective}$, while the others satisfy $LCC_i > LCC_{Min\_Effective}$. For Bridge, $NumAbove = 4,026$, $NumBelow = 70$, and $ACE_{Max\_Effective} > ACE_{All} = 760,276 > 0$. Thus, in Eq. (18), when $NumAbove \gg NumBelow$, we have:

$$ACE_{All} < ACE_{Max\_Effective}$$

By Eqs. (16) and (19), we obtain:

$$ACE_{Max\_Effective} > ACE_{All} > ACE_{Min\_Effective}$$

Thus, the effective minimum and maximum of LCCs, $LCC_{Min\_Effective}$ and $LCC_{Max\_Effective}$, can provide the effective top and bottom limits of ACE_{All}, $ACE_{Max\_Effective}$ and $ACE_{Min\_Effective}$, respectively.

3.3 Decoded Image Quality Prediction

In the encoding process, range blocks are divided into two categories: coded and uncoded range blocks. For coded blocks, if their number is sufficiently large, we can use the effective minimum and maximum of the LCCs of coded range blocks, $LCC_{Coded\_Min\_Effective}$ and $LCC_{Coded\_Max\_Effective}$, to approximate those of all range blocks, $LCC_{Min\_Effective}$ and $LCC_{Max\_Effective}$:

$$LCC_{Coded\_Min\_Effective} \approx LCC_{Min\_Effective}$$

$$LCC_{Coded\_Max\_Effective} \approx LCC_{Max\_Effective}$$

Moreover, the effective minimum and maximum of the LCCs of uncoded range blocks, $LCC_{Uncoded\_Min\_Effective}$ and $LCC_{Uncoded\_Max\_Effective}$, can be represented by those of all range blocks:

$$LCC_{Uncoded\_Min\_Effective} \approx LCC_{Min\_Effective}$$

$$LCC_{Uncoded\_Max\_Effective} \approx LCC_{Max\_Effective}$$

By Eqs. (21) and (22), we can use $LCC_{Coded\_Min\_Effective}$ and $LCC_{Coded\_Max\_Effective}$ to approximate $LCC_{Uncoded\_Min\_Effective}$ and $LCC_{Uncoded\_Max\_Effective}$:

$$LCC_{Coded\_Min\_Effective} \approx LCC_{Uncoded\_Min\_Effective}$$

$$LCC_{Coded\_Max\_Effective} \approx LCC_{Uncoded\_Max\_Effective}$$

Further, by Eq. (3), the collage error of an arbitrary uncoded range block can be expressed as:

$$CE_{Uncoded}(R_i) = (1 - LCC^2_{Uncoded}) \cdot Var(R_i), \quad i = 1, 2, 3, \ldots, NumUncoded$$

where $NumUncoded$ denotes the number of uncoded range blocks. $LCC_{Uncoded}$ denotes the LCC of uncoded range blocks. Then, the ACE of uncoded range blocks can be described as:

$$ACE_{Uncoded} = \sum_{i=1}^{NumUncoded} (1 - LCC^2_{Uncoded}) \cdot Var(R_i)$$

We assume that the result in Subsection 3.2 also applies to uncoded range blocks, i.e., the effective minimum and maximum of the LCCs of uncoded range blocks, $LCC_{Uncoded\_Min\_Effective}$ and $LCC_{Uncoded\_Max\_Effective}$, can provide the effective top and bottom limits for $ACE_{Uncoded}$:

$$ACE_{Uncoded\_Min\_Effective} < ACE_{Uncoded} < ACE_{Uncoded\_Max\_Effective}$$

where:

$$ACE_{Uncoded\_Min\_Effective} = \sum_{i=1}^{NumUncoded} (1 - LCC^2_{Uncoded\_Max\_Effective}) \cdot Var(R_i)$$

$$ACE_{Uncoded\_Max\_Effective} = \sum_{i=1}^{NumUncoded} (1 - LCC^2_{Uncoded\_Min\_Effective}) \cdot Var(R_i)$$

By Eq. (23), Eq. (27) can be rewritten as:

$$ACE_{Uncoded\_Min\_Effective} = \sum_{i=1}^{NumUncoded} (1 - LCC^2_{Coded\_Max\_Effective}) \cdot Var(R_i)$$

$$ACE_{Uncoded\_Max\_Effective} = \sum_{i=1}^{NumUncoded} (1 - LCC^2_{Coded\_Min\_Effective}) \cdot Var(R_i)$$

Moreover, if the actual percentage of accumulated collage error (APACE) is defined as the ratio of ACE_{Coded} to ACE_{All}:

$$APACE = \frac{ACE_{Coded}}{ACE_{All}} = \frac{ACE_{Coded}}{ACE_{Coded} + ACE_{Uncoded}}$$

By Eqs. (26) and (29), we obtain the effective bottom and top limits of the actual percentage of accumulated collage error, EBL-APACE and ETL-APACE:

$$EBL\text{-}APACE = \frac{ACE_{Coded}}{ACE_{Coded} + ACE_{Uncoded\_Max\_Effective}}$$

$$ETL\text{-}APACE = \frac{ACE_{Coded}}{ACE_{Coded} + ACE_{Uncoded\_Min\_Effective}}$$

From Eq. (30), we know that $ETL\text{-}APACE < 1$ and $EBL\text{-}APACE < ETL\text{-}APACE$. When $EBL\text{-}APACE$ reaches a large percentage (such as 90%), $EBL\text{-}APACE$ can approach $ETL\text{-}APACE$ sufficiently, implying that APACE lies in a rather small dynamic range and can be approximately estimated as:

$$APACE_{Estimated} = \frac{EBL\text{-}APACE + ETL\text{-}APACE}{2}$$

Moreover, the average collage error (ACER) of all range blocks can be computed as:

$$ACER = \frac{ACE_{All}}{NumR}$$

By Eqs. (32), (29), and (31), we can estimate ACER as:

$$ACER_{Estimated} = \frac{ACE_{Coded}}{APACE_{Estimated} \cdot NumR}$$

The numerator of Eq. (33) represents the estimated version of ACE_{All}. In addition, the peak signal-to-noise ratio (PSNR) is used to measure decoded image quality:

$$PSNR = 10\log_{10}\left(\frac{255^2}{\frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}[f(x,y) - f_{Decoded}(x,y)]^2}\right)$$

where f and $f_{Decoded}$ represent the input and decoded images, respectively. Moreover, a logarithmic relationship exists between ACER and the PSNR of the decoded image [35]:

$$PSNR = \beta_1 + \beta_2 \log_{10}(ACER)$$

where β_1 and β_2 are constant values. Finally, by Eqs. (33) and (35), we can directly predict the PSNR of the decoded image.

4. Experiments

We adopted six 256 × 256 images (shown in Fig.7): Peppers, Airplane, Boat, Bridge, House, and Lake. Four fractal image coding methods were used to assess the performance of the proposed method [1,5-7]. The scaling and offset coefficients, s and o, were quantized using 5 and 7 bits, respectively. The procedures of the proposed method are as follows:

Step 1: Divide the input image into range and domain blocks, and sort the range blocks by their variances from largest to smallest.

Step 2: Take one uncoded range block and encode it. If $EBL\text{-}APACE$ is smaller than 90%, return to Step 2. Otherwise, calculate $ETL\text{-}APACE$ and proceed to Step 3.

Step 3: Calculate ACER using Eq. (33) and predict the decoded image quality using Eq. (35).

(a) Peppers (b) Airplane (c) Boat (d) Bridge (e) House (f) Lake

Fig.7 Six test images.

When the range block size is set to 4×4 and 8×8, Tables 1 , 2, 3, and 4 list the experimental results for Jacquin's, Chaurasia's, Zheng's, and Gupta's methods, respectively [1,5-7]. In Table 1, when the range block size is 4×4, the second row shows the PSNRs of decoded images for Jacquin's method. In the third row, we encode all range blocks, perform block-matching 4,096 times for all test images, and consider the percentage of computations as 100%. For the previous method, the fourth to sixth rows list the predicted PSNRs, deviations between predicted and actual PSNRs, and the percentage of computations relative to Jacquin's method, respectively. Similarly, the seventh to ninth rows show the counterparts for the proposed method.

From Table 1, we observe that when the range block size is 4×4, the average deviations of the previous and proposed methods are 0.08 dB and 0.05 dB, respectively, and the average percentages of computations are 52.99% and 47.76%, respectively. When the range block size is 8×8, the average deviations of the previous and proposed methods are 0.10 dB and 0.09 dB, respectively, and the required percentages of computations are 52.62% and 50.73%, respectively. These results demonstrate that compared with the previous method, the proposed method provides higher prediction accuracy with fewer computations. The analysis is as follows:

In the previous method, LCC_{Coded_Min} and LCC_{Coded_Max} were used to approximate $LCC_{Uncoded\_Min}$ and $LCC_{Uncoded\_Max}$, providing the bottom and top limits of APACE as:

$$APACE_{Min} = \frac{ACE_{Coded}}{ACE_{Coded} + ACE_{Uncoded\_Max}}$$

$$APACE_{Max} = \frac{ACE_{Coded}}{ACE_{Coded} + ACE_{Uncoded\_Min}}$$

where:

$$ACE_{Uncoded\_Max} = \sum_{i=1}^{NumUncoded} (1 - LCC^2_{Coded\_Min}) \cdot Var(R_i)$$

$$ACE_{Uncoded\_Min} = \sum_{i=1}^{NumUncoded} (1 - LCC^2_{Coded\_Max}) \cdot Var(R_i)$$

Similar to Eq. (12), for coded range blocks, we also have:

$$LCC_{Coded\_Min\_Effective} \leq LCC_{Coded\_Min} \leq LCC_{Coded\_Max} \leq LCC_{Coded\_Max\_Effective}$$

By comparing Eq. (28) and (37), we obtain:

$$APACE_{Min} < EBL\text{-}APACE, \quad ETL\text{-}APACE < APACE_{Max}$$

By Eq. (39), $APACE_{Min}$ is always smaller than $EBL\text{-}APACE$. When $EBL\text{-}APACE$ reaches 90% in the proposed method, $APACE_{Min}$ is smaller than 90%, implying that in the previous method, extra range blocks must be encoded to make $APACE_{Min}$ reach 90%. Thus, the proposed method saves a significant number of computations. When the range block size is 4×4, we have 4,096 range blocks in total. For Peppers, in the previous method, when $APACE_{Min}$ reaches 90%, we need to encode 2,327 range blocks (represented by red and white boxes in Fig.8 (a)), comprising 56.81% of total range blocks. In the proposed method, when $EBL\text{-}APACE$ reaches 90%, we only need to encode 2,002 range blocks (represented by white boxes in Fig.8 (a)), comprising 48.88% of total range blocks. This implies that computations for 325 range blocks (represented by red boxes) are saved. When the range block size is 8×8, we have 1,024 range blocks in total. In the previous method, when $APACE_{Min}$ reaches 90%, we need to encode 529 range blocks (represented by red and white boxes in Fig.9 (a)), comprising 51.66% of total range blocks. In the proposed method, when $EBL\text{-}APACE$ reaches 90%, we need to encode 487 range blocks (represented by white boxes in Fig.9 (a)), comprising 47.56% of total range blocks. This implies that computations for 42 range blocks (represented by red boxes) are saved. Moreover, by Eq. (39), we also know that compared with the range from $APACE_{Min}$ to $APACE_{Max}$, the range from $EBL\text{-}APACE$ to $ETL\text{-}APACE$ provides a smaller dynamic range for estimating APACE, and the proposed method is expected to provide higher prediction accuracy.

In Table 2, for Chaurasia's method, when the range block size is 4×4, the average deviations of the previous and proposed methods are 0.07 dB and 0.05 dB, respectively, and the average percentages of computations are 54.14% and 48.12%, respectively. When the range block size is 8×8, the average deviations of both methods are 0.17 dB, and the average percentages of computations are 52.44% and 50.31%, respectively.

In Table 3, for Zheng's method, when the range block size is 4×4, the average deviations of the previous and proposed methods are 0.14 dB and 0.10 dB, respectively, and the average percentages of computations are 53.87% and 47.70%, respectively. When the range block size is 8×8, the average deviations are 0.11 dB and 0.09 dB, respectively, and the average percentages of computations are 52.60% and 50.68%, respectively.

In Table 4, for Gupta's method, when the range block size is 4×4, the average deviations of the previous and proposed methods are 0.08 dB and 0.04 dB, respectively, and the average percentages of computations are 53.24% and 48.13%, respectively. When the range block size is 8×8, the average deviations are 0.15 dB and 0.14 dB, respectively, and the average percentages of computations are 52.80% and 50.90%, respectively.

In summary, compared with the previous method, the proposed method can predict decoded image quality more accurately with fewer computations.

(a) Peppers (b) Airplane (c) Boat (d) Bridge (e) House (f) Lake

Fig.8 When the range block size is 4×4, white and red boxes represent range blocks encoded in the previous method, while white boxes represent range blocks encoded in the proposed method.

(a) Peppers (b) Airplane (c) Boat (d) Bridge (e) House (f) Lake

Fig.9 When the range block size is 8×8, white and red boxes represent range blocks encoded in the previous method, while white boxes represent range blocks encoded in the proposed method.

Table 1 Performance comparison between the previous and proposed methods for Jacquin's method [1].

Test images	Peppers	Airplane	Bridge	House	Average
Jacquin's [1]					
Decoded (dB)					
Computations (%)	100	100	100	100	100
Previous [35]					
Predicted (dB)					
Deviation (dB)					
Computations (%)					
Proposed					
Predicted (dB)					
Deviation (dB)					
Computations (%)					

Table 2 Performance comparison between the previous and proposed methods for Chaurasia's method [5].

Test images	Peppers	Airplane	Bridge	House	Average
Chaurasia's					
Decoded (dB)					
Computations (%)	100	100	100	100	100
Previous [35]					
Predicted (dB)					
Deviation (dB)					
Computations (%)					
Proposed					
Predicted (dB)					
Deviation (dB)					
Computations (%)					

Table 3 Performance comparison between the previous and proposed methods for Zheng's method [6].

Test images	Peppers	Airplane	Bridge	House	Average
Zheng's [6]					
Decoded (dB)					
Computations (%)	100	100	100	100	100
Previous [35]					
Predicted (dB)					
Deviation (dB)					
Computations (%)					
Proposed					
Predicted (dB)					
Deviation (dB)					
Computations (%)					

Table 4 Performance comparison between the previous and proposed methods for Gupta's method [7].

Test images	Peppers	Airplane	Bridge	House	Average
Gupta's [7]					
Decoded (dB)					
Computations (%)	100	100	100	100	100
Previous [35]					
Predicted (dB)					
Deviation (dB)					
Computations (%)					
Proposed					
Predicted (dB)					
Deviation (dB)					
Computations (%)					

5. Conclusion

This paper proposes an effective method for predicting fractal decoded image quality. First, we introduced the effective minimum and maximum of LCCs, which provide EBL-APACE and ETL-APACE for APACE. Then, decoded image quality can be directly predicted when EBL-APACE reaches a large percentage. Experimental results demonstrate that compared with the previous method, the proposed method provides higher prediction accuracy with fewer computations. Future work will focus on two aspects: (a) Maintaining prediction accuracy. By removing the interference of LCC outliers, we have obtained satisfactory prediction accuracy, and we will strive to maintain this accuracy in future work. (b) Shortening the prediction process. Compared with the previous method, the proposed method has reduced the number of computations, and we will attempt to further reduce computations in future work.

Acknowledgments

This work was partially supported by the State Scholarship Fund of China (Grant No. 201806575003) and the Special Project of Collaborative Innovation of Hengshui University (Grant No. 2022XJZX19).

References

1. A. K. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Trans. on Image Process.* 1(1), 18-30 (1992).
2. B. Wohlberg and B. G. Jager, "A review of the fractal image coding literature," *IEEE Trans. on Image Process.* 8(12), 1716-1729 (1999).
3. V. Chaurasia and V. Chaurasia, "Statistical feature extraction based technique for fast fractal image compression," *J. Vis. Commun. Image R.* 41, 87-95 (2016).
4. Y. P. Zheng, X. P. Li, and M. Sarem, "Fast fractal image compression algorithm using specific update search," *IET Image Process.* 14(9), 1733-1739 (2020).
5. R. Gupta, D. Mehrotra, and R. K. Tyagi, "Hybrid edge-based fractal image encoding using K-NN search," *Multimed. Tools Appl.* 81, 21135-21154 (2022).
6. F. R. Shen and H. Osamu, "A fast no search fractal image coding method," *Signal Process. Image Commun.* 19(5), 393-404 (2004).
7. X. Y. Wang and S. G. Wang, "An improved no-search fractal image coding method based on a modified gray-level transform," *Comput. Graph.* 32(4), 445-450 (2008).
8. X. Y. Wang, Y. X. Wang, and J. J. Yun, "An improved no-search fractal image coding method based on a fitting plane," *Image Vision Comput.* 28, 1303-1308 (2010).
9. S. Bi and Q. Wang, "Fractal image coding based on a fitting surface," *J. Appl. Math.* 2014, 634848 (2014).
10. M. Ghazel, G. H. Freeman, and E. R. Vrscay, "Fractal image denoising," *IEEE Trans. on Image Process.* 12(12), 1560-1578 (2003).
11. M. Ghazel, G. H. Freeman, and E. R. Vrscay, "Fractal-wavelet image denoising revisited," *IEEE Trans. on Image Process.* 15(9), 2669-2675 (2006).
12. J. Lu, Z. X. Ye, Y. Y. Zou, and R. S. Ye, "An enhanced fractal image denoising algorithm," *Chaos, Soliton. Fract.* 38, 1054-1064 (2008).

13. J. H. Jeng, C. C. Tseng, and J. G. Hsieh, "Study on huber fractal image compression," *IEEE Trans. on Image Process.* 18(5), 995-1003 (2009).

14. J. Lu, Z. X. Ye, and Y. Y. Zou, "Huber fractal image coding based on a fitting plane," *IEEE Trans. on Image Process.* 22(1), 134-145 (2013).

15. Y. Y. Zou, H. X. Hu, J. Lu, X. X. Liu, Q. T. Jiang, and G. H. Song, "A nonlocal low-rank regularization method for fractal image coding," *Fractals* 29(5), 2150125 (2021).

16. C. Xu, Y. T. Ye, and Z. W. Hu, "A primal-dual algorithm for robust fractal image coding," *Fractals* 27(7), 1950119 (2019).

17. S. M. Abdullahi, H. X. Wang, and T. Li, "Fractal coding-based robust and alignment-free fingerprint image hashing," *IEEE Trans. on Inf. Foren. Sec.* 15, 2587-2601 (2020).

18. F. Khelaifi and H. J. He, "Perceptual image hashing based on structural fractal features of image coding and ring partition," *Multimed. Tools Appl.* 79(27), 19025-19044 (2020).

19. M. H. Pi, M. K. Mandal, and A. Basu, "Image retrieval based on histogram of fractal parameters," *IEEE Trans. on Multimedia* 7(4), 597-605 (2005).

20. X. Y. Wang and Z. Chen, "A fast fractal coding in application of image retrieval," *Fractals* 17(4), 441-450 (2009).

21. X. Huang, Q. Zhang, and W. Liu, "A new method for image retrieval based on analyzing fractal coding characters," *J. Vis. Commun. Image R.* 24(1), 42-47 (2013).

22. M. H. Pi, C. H. Li, and H. Li, "A novel fractal image watermarking," *IEEE Trans. on Multimedia* 8(3), 488-498 (2006).

23. F. Daraee and S. Mozaffari, "Watermarking in binary document images using fractal codes," *Pattern Recogn. Lett.* 35, 120-129 (2014).

24. J. Lu, Y. Y. Zou, C. Y. Yang, and L. J. Wang, "A robust fractal color image watermarking algorithm," *Math. Probl. Eng.* 2014, 638174 (2014).

25. K. H. Chung, Y. H. Fung, and Y. H. Chan, "Image enlargement using fractal," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, 273-275 (2003).

26. C. M. Lai, K. M. Lam, Y. H. Chan, and W. C. Siu, "An efficient fractal based algorithm for image magnification," in *Proceedings of International Symposium on Intelligent Multimedia, Video and Speech Processing*, Hong Kong, 571-574 (2004).

27. Z. P. Chen, Z. L. Ye, S. X. Wang, and G. H. Peng, "Image magnification based on similarity analogy," *Chaos, Soliton. Fract.* 40, 2370-2375 (2009).

28. Y. C. Wee and H. J. Shin, "A novel fast fractal super resolution technique," *IEEE Trans. Consum. Electron.* 56(3), 1537-1541 (2010).

29. Z. Hua, H. C. Zhang, and J. J. Li, "Image super resolution using fractal coding and residual network," *Complexity* 2019, 9419107 (2019).

30. C. Bisogni, M. Nappi, C. Pero, and S. Ricciardi, "FASHE: A fractal based strategy for head pose estimation," *IEEE Trans. on Image Process.* 30, 3192-3203 (2021).

31. C. Bisogni, M. Nappi, C. Pero, and S. Ricciardi, "PIFS scheme for head pose estimation aimed at faster face recognition," *IEEE Trans. on Biometrics, Behavior, and Identity Science* 4, 173-184 (2022).

32. S. Liu, W. L. Bai, N. Y. Zeng, and S. H. Wang, "A fast fractal based compression for MRI images," *IEEE Access* 7, 62412-62420 (2019).

33. Q. Wang and S. Bi, "Improved method for predicting the peak signal-to-noise ratio quality of decoded images in fractal image coding," *J. Electron. Imaging* 26(1), 013024 (2017).