Applied Mechanics and Materials Vols. 241-244 (2013) pp 3034-3039
Online available since 2012/Dec/13 at www.scientific.net
© (2013) Trans Tech Publications, Switzerland
doi:10.4028/www.scientific.net/AMM.241-244.3034

Fast Fractal Image Coding Method Based on RMSE and DCT Classification

Xianqiang Lv[1, a], Song Yang[1, b], Xin Zhang[1, c], Ying Wang[1, d], Yunfeng Shi[1, e], Liu Wei[1, f]

[1]Educational Technology & Computing Center, Dalian Ocean University, China

[a]lxq@dlou.edu.cn, [b]yangapple@dlou.edu.cn, [c]realxin@dlou.edu.cn, [d]wying@dlou.edu.cn, [e]shiyunfeng@dlou.edu.cn, [f]liuwei@dlou.edu.cn

Key words: ractal image coding, root mean square error, discrete cosine transform, classification.

Abstract. To solve the problem of long time consuming in the fractal encoding process, a fast fractal encoding algorithm based on RMSE (Root mean square error) and DCT (Discrete Cosine Transform) classification is proposed. During the encoding process, firstly, the image is divided into range blocks and domain blocks by quadtree partition according to RMSE, then, according to DCT coefficients of image block, three classes of image blocks are defined, which are smooth class, horizontal/vertical edge class, diagonal/sub-diagonal class. At last, every range block is limited to search the best matched block in the corresponding domain block class, and the fractal coding are recorded until the process is completed. When searching the best matched block, the nearest neighbor block will be found in the sense of RMSE in the ordered codebook, and the best matched block will be further found in the vicinity of the nearest neighbor block. The experimental results show that the proposed algorithm can efficiently reduce the search space and shorten the encoding time, while achieving the same reconstructed image quality as that of the full search method.

Introduction

In the last decade, fractal image coding is developed as a new image compression method with fast decompression, resolution-independent and high compression ratio, etc[1], however, the long encoding time limits its application. Many scholars have proposed a fast local searching-based fractal coding algorithm[2-4], which can be summarized as the use of segmentation method[5-7], classification method[8-13] and limiting the search space method [14], which used to reduce the complexity of fractal coding, and improve the speed of the fractal encoding. Image partition methods include quadtree partition [5-6], triangular partition, HVS partition[7] and regional partition, and quadtree partition becomes the most popular partition method because of its advantages of flexibility, high compression ratio and simplicity. Classification methods include clustering method[8], moment invariants method [9-10], and so on, and the principle is that ranges and domain blocks are classified by certain properties in the encoding process, each range block can find the best matching block in the same or adjacent domain blocks pool, which can reduce effectively the number of comparisons and shorten the encoding time. The methods of Limiting search space include K neighborhood searching method [14], which can further narrow the search space.

This article presents a fast fractal image coding method based on RMSE and DCT classification. The proposed method utilizes a quadtree to partition images while simultaneously classify images according to their edge properties. It enables the range block to find a satisfactory self-similar block in the corresponding domain blocks poor during the encoding process. Hence, on the premise of the reconstructed image quality, the encoding speed is improved considerably. Further, the classification threshold value can be self-adapted according to the range blocks, and the compression ratio is also stable.

Basic Fractal Image Coding Algorithm

During the basic fractal graphics coding, first, the graphics are divided into $B \times B$ size range blocks (hereinafter, referred as R Blocks), which do not overlap with each other, and $2B \times 2B$ size domain blocks (hereinafter, referred as D Blocks), which can overlap with each other; the step is δ. Then, each domain block is compressed by sampling into smaller range blocks via average of 4-neighbourhood pixel grey value. After eight equidistant transformations, all the newly generated blocks are considered as codebook Ω for the following matching operation. During the process of encoding, for any range block, the best matching block and self-affine transformation is searched in the codebook Ω for reaching a minimum value of the mean square deviation *E(R,D)*, i.e., formula (1) is calculated as follows:

$$E(R,D) = \min\left\{ \min_{s,o\in R,|s|<1} \left\| R - sD_m - oI \right\|^2 \right\} \tag{1}$$

In the formula(1), s and o represent the zoom and translation operations for the domain block respectively, m represents the order number of best matching domain block. First, s and o are calculated for satisfying the minimal values of formula (1) via formula (2). Second, the external minimized value of the sub-blocks required for balancing formula (1) are searched in the codebook Ω by using the global search method. The above-mentioned operations are repeated for all range blocks. The corresponding locations, equidistance transformations, gray scaling, and translation information of all domain blocks are recorded to complete the entire image encoding. The value of s ($0 \le s \le 1$) is guaranteed theoretically to ensure convergence of formula (1). During the iterative decoding process and the unsatisfactory values are discarded.

$$s = \frac{\langle R - \overline{R}I, D - \overline{D}I \rangle}{\left\| D - \overline{D}I \right\|^2} \qquad o = \overline{R} - s \cdot \overline{D} \tag{2}$$

To decode the compressed image, we firstly choose any arbitrary image as initial image, perform the affine transform on the image to obtain a new image according to the compression codes, the process is proceeded recursively. According to Partitioned Iteration Function Theorem (PIFS), the results of iteration can be converged; the final image is the decoded image of fractal coding.

The quadtree partition based on root mean square error

In fractal coding, image segmentation is very important aspect which can improve the encoding speed and compression ratio. The paper uses top-down recursive partitioning of the quadtree, and the process is as follows:

Step 1: Set the size of R block and root mean square error threshold τ.

Step 2: Calculate the root mean square error τ' of the current image block C.

Step 3: If $\tau' < \tau$, this block C is a smooth block, and store the location information and the mean value; otherwise, to determine the block size is greater than D block size, if larger than the D block, the image block is divided into the same four parts, and then recursively followed by the steps; if equal to size of the D block, then stop the partition to record the location of this block and root mean square error, this block is D block.

Step 4:After the original image is partitioned recursively accordance to the above steps, information of all of the D block and smooth blocks which are greater than or equal to D block have been stored, and then all of the D blocks are divided into four parts, each part calculate the root mean square error, if less than τ, it is the smooth blocks, and add it to the smooth block set; otherwise, it is R block, record the location and root mean square error information, and add it to the R block set. After the partition, you can set the R blocks set, D blocks set and smooth blocks set, and it will be ready for the fractal coding.

The classification Method based on DCT coefficients

It is difficult to find the R block and the D block in the space domain. Doing so requires a considerable amount of calculation to find the best matched block. It is comparatively easy to find a self-similar matched block using DCT transformation coefficients because the shape in the frequency domain is comparatively similar: the amplitude is large at the low-frequency end and small at the high-frequency end. Hence, this study utilizes the DCT transformation coefficients to perform block classification. It can make the R blocks find the best matched block only among the corresponding classification.

According to the geometrical features of the blocks (including the R block and the D block), the blocks can be classified to three types:

(1) Smooth block S: It is smooth and has no significant change in gradient.

(2) Edge block H: It has a significant gray scale change along the horizontal and vertical directions.

(3) Diagonal block E: It has a significant gray scale change along the diagonal and small diagonal directions.

Let f be the image of a given size $N \times N$. The DCT definition of f is F, and then (showed as formula(3))

$$F(m,n) = \frac{2}{N} C_m C_n \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} f(i,j) \cos(\frac{(2i+1)m\pi}{2N}) \cos(\frac{(2j+1)n\pi}{2N}) \ (m,n = 0,1,\cdots,N-1) \tag{3}$$

Here,(showed as formula(4)):

$$C_k = \begin{cases} 1/\sqrt{2} & k = 0 \\ 1 & otherwise \end{cases} \tag{4}$$

We use $F_R(m,n)$ and $F_D(m,n)$ to present the coefficient of the range block and the domain block after DCT transformation respectively. $F_R(0,0)$ is the direct current component, and $F_R(0,1), F_R(1,0)$ and $F_R(1,1)$ are the alternating components of a comparatively low frequency.

Setting a threshold to indicate whether current block is a simple or complex block or not, there

$$|F_R(0,1)| + |F_R(1,0)| + |F_R(1,1)| = \begin{cases} < \sigma_N, \ F_R \ \ is \ \ simple \ \ block \\ \geq \sigma_N, \ F_R \ \ is \ \ complex \ \ block \end{cases}$$

If the block is a simple block, the block belongs to smooth type; otherwise the block is a complex block, which belongs to the edge block H or diagonal blocks E. Then it can use the DCT classification coefficients, if $\left\| F_R(0,1) \right| - \left| F_R(1,0) \right\| \leq \sigma_D$ is met (σ_D is pre-set threshold for diagonal block E), the block belongs to the diagonal block E, otherwise it belongs to the edge block H.

The basic steps of classification based on DCT coefficients are as follows:

Step 1: For each R block to be calculated vertical and horizontal DCT coefficients: $F_R(0,1), F_R(1,0), F_R(1,1)$ and the sum of the them, $F^+ = F_R(0,1) + F_R(1,0) + F_R(1,1)$.

Step 2: If $F^+ \le \sigma_N$, the block is a smooth block , just to keep the information, otherwise the block is a complex block, if $\left\| F_R(0,1) \right| - \left| F_R(1,0) \right\| \le \sigma_D$ is met, the block is the diagonal block , otherwise, the edge is edge block .

Step 3: for each block must calculate the minimum vertical and horizontal DCT coefficients and also to calculate the maximum value of the two, $\overline{F} = \max\{F_R(0,1), F_R(1,0)\}$

Step 4: sort all $\overline{F}$ in increasing order. Find $\overline{F}^* = \max\{F_R(0,1), F_R(1,0)\}$ located at the first one-third of R blocks. Set the threshold $\sigma'_N = \overline{F}^*$.

Step 5: For the rest two-third of the R blocks, calculate $\overline{\overline{F}} = \left\| F_R(0,1) \right| - \left| F_R(1,0) \right\|$.

Step 6: sort the rest $\overline{\overline{F}}$ in increasing order. Find $\overline{\overline{F}}^*$ located at the middle of the rest blocks. Set the threshold $\sigma_D = \overline{\overline{F}}^*$.

Step 7: According to the threshold σ'_N and σ_D, the D blocks are classified.

The K Neighborhood Search Based on Root Mean Square Error

To speed up the search in the corresponding codebook, three types of codebooks are sorted in the ascending order according to the root mean square error. For each R block, first find its corresponding codebook, and use binary search to find the D block which is close to mean square error of this R block. Then 8 isometric transformations are done in the k-nearest neighbor, then choose the D block which has the smallest matching error, and record the location and other information to constitute coding information for R block.

Results and Analyses of Experiment

In the experiment, we take the a lot of standard testing images for simulation, and the following lists include only three standard gray-scale image lena, goldhill and peppers as an example to illustrate the encoding speed of this method and the reconstructed image quality. The given image is a 256 × 256 gray scale image; further, we use VC+6.0 programming language. The experiment is performed on a computer that is configured with Celeron 2.0 GHz, 1GB memory and runs Microsoft Windows XP operation system. R block size is 4×4, and D block size is 8×8. Make a comparison among full search method，Duh method and crosstrace method, performance testing parameters are PSNR(db) and encoding time(sec).

(a)Lena image(τ =1,PSNR=32.26) (b) Lena image(τ =10,PSNR=29.87) (c) Lena image(τ =4,PSNR=31.50)

Fig.1 The image comparison of different τ value

This encoding time and the quality of decoding images depend on two parameters, and the effect of the parameter τ for the experimental is obvious. If τ is too big, more image blocks will be considered as the smooth blocks. Although it can accelerate image encoding speed, the decoded image will appear blocking effects, and seriously affect the image quality. Through the experiments, we can know, if $\tau = 4$, the image will appear slightly partial blocking effects. The greater the τ value, the more obvious blocking effects (see Fig.1(c)), and then, if $\tau = 1$ blocking effects disappeared, the eye can not see the block effects.

This time encoding and decoding image quality depends on two parameters, and the impact of parameter τ for the experiment is obvious, if τ is too large, there will be more image blocks are considered as constant block, and can accelerate image encoding speed and obtain higher compression ratio, but blocking effects appear and seriously affect the image quality. Through the experiments, we can know, if τ is equal to 4, the image will appear slightly partial blocking effects (see Fig.1(c)), the greater the value, the more obvious blocking effects (see Fig.1(b)), and if τ is less than 4, the blocking effects disappeared basically. If τ is equal to 1, blocking effects is not visible (see Fig.1(a)).

Tab.1 The comparison of image compression ratio with different τ

Image	$\tau=1$			$\tau=10$			$\tau=4$		
	Smooth	R	CR	Smooth	R	CR	Smooth	R	CR
Lena	40	4056	4.46	1087	1302	11.29	999	2359	6.86
Goldhill	2	4094	4.41	1092	2005	8.69	221	736	4.84
Peppers	9	4087	4.42	1218	1348	10.74	839	2885	5.80

Tab.1 The experimental data of different k neighbor

K	PSNR(dB)	Time(s)	PSNR(dB)	Time(s)	PSNR(dB)	Time(s)
10	29.87	0.2831	27.80	0.3844	28.99	0.2719
20	30.50	0.5288	28.35	0.8109	29.74	0.5175
50	31.23	1.1813	28.84	1.6744	30.35	1.0469
100	31.41	2.3813	29.27	3.0312	30.65	2.5719
image	Lena		Goldhill		Peppers	

The compression ratios of image coding are shown in Table 1, if τ is different, compression ratio is different, too. If τ is a small, the corresponding compression ratio is small, the quality of decoded image is good. If τ is great, the corresponding compression ratio is great, and the quality of decoded image is not good.

Another parameter K can affect the quality of decoded image and the encoding time. If the value of K is small, the neighborhood range is small; the probability of finding the best match block is low, the quality of decoded image is not good, but encoding time is short. If the value of K is large, the probability of finding the best matched block is great, the quality of decoded image is good, but encoding time is long.

The Comparison between Several methods

Table.2 presents the encoding time and the quality of decoded image comparison between several traditional methods and the proposed method. Because the proposed method classifies the image blocks, the classification calculation job is not large and only the two smallest horizontal and vertical coefficients of DCT need to be calculated. The Range block can find the best matching domain block in the corresponding classification. Hence, from Table 1, we can see that compared to Jacquin's full search method, this proposed method reduces the encoding time significantly, and the encoding speed improves by 100 times. The reconstructed image quality is almost the same.

Table 3 shows comparison of three standard testing images using the traditional method and the proposed method about encoding time the quality of decoded image. As image blocks are classified, and the complexity of classification computation is low, R block only find the best matched block in the corresponding D block set. Compared to the full search method, this method greatly reduces the encoding time, and encoding speed increased nearly 77 times, image quality is similar to full search method or better. Compared to crosstrace algorithm, the proposed method acquires the higher PSNR in the similar encoding time, and image encoding quality is also improved.

Tab.3 Encoding results of full searching, cross trace and the proposed method

Image	Full searching method		Cross trace		The proposed method	
	PSNR(db)	Encoding time	PSNR	Encoding time	PSNR	Encoding time
Lena	31.24	285.68	31.14	3.6744	31.50	3.6563
Goldhill	29.38	312.46	28.94	4.6506	29.32	4.8688
Peppers	30.58	298.52	30.07	3.9656	30.73	3.8852

Conclusions

This paper presents a fast fractal image coding method which combines quadtree partition, DCT coefficients classification and K neighborhood search. Through the simulation of three experimental mages, the results show that compared with the basic fractal algorithm, the peak signal to noise ratio in the same or increase in the case of the algorithm to speed up encoding times. In the same encoding quality of the premise that the algorithm with two other fast fractal coding algorithms, with faster encoding speed, but the characteristics of longer decoding time. Fractal coding is limitative distortion coding method, in the premise of keeping the image quality unchanged, how to select a better feature as the basis of classification or search and achieve faster encoding is an important issue in fractal image coding research.

References

[1] Barnsley M F, Demko S. Iterated function systems and the global construction of fractals[J]. Proc Roy Soc 1985, A399: 243-275.

[2] Jacquin A E. Image encoding based on a fractal theory of iterated contractive image transformations[J]. IEEE Trans Image Process,1992,1(1):18-30.

[3] Wohlberg B, Jager G. A review of the fractal image encoding literature. IEEE transaction on Image Proccessing, 1999, 8(12):1716-1729.

[4] M.Hassaballah,M,M.Makky,Youssef B.Mahdy.A fast fractal image compression method based entropy[J].Electronic letters on Computer Vision and Image Analysis 5(1):30-40,2005.

[5] Zhang L B,Fan S.New method for fractal image compression based on adaptive threshold value quadtree[J].Computer engineering and Design,2006,27(13):2322-2323,2337.[In Chinese]

[6] Novel quadtree-based algorithm for fast fractal image compress[J].Journal of Chinese Computer System, 2007,28(6):1465-1469.[In Chinese]

[7] Yi C J.Fast fractal image encoding based on novel quadtree partition[J].Computer&Digital Engineering, 2009, (7):151-153.[In Chinese]

[8] Yang H Y, Wang X Y, Yu Y C. Fast fractal image compression based on fuzzy C-mean clustering[J].Journal of Chinese Computer System.2009,3(3):493-497.[In Chinese]

[9] Chen z p,Ye Z L,Gao X F,Meng F. A fast method for fractal image coding based on invariant quadrature[J]. Computer Engineering and Applications.2004,33:62-66.[In Chinese]

[10] Wu Y Q,Guo J J,Sun Z Y. Fast fractal coding method based on krawtchouk moments and kernel-induced fuzzy clustering[J].Journal of Optoelectronics•Laser,2010,21(9):1412-1416.[In Chinese]

[11] Fisher Y. Fractal image compression, theory and application[M]. New York: Springer-Verlag, 1994.

[12] Hütgen B. Fast Hierachical codebook search for fractal encoding of still images[J]. Proc. EOS/SPIE Visual Comm. PACS Medical APP. Berlin, Germany,1993.

[13] Wang Z, Zhang D, Yu Y. Hybrid image encoding based on partial fractal mapping[J]. Signal Processing: Image Communication, 2000, 15(9):767-779.

[14] Wang Q,Liang D Q,Bi S.Nearest neighbour for fast fractal image encoding based on correlation information feature[J].Journal of Chinese Computer System,2011,32(6):1108-1112.[In Chinese]

Industrial Instrumentation and Control Systems
10.4028/www.scientific.net/AMM.241-244

Fast Fractal Image Coding Method Based on RMSE and DCT Classification
10.4028/www.scientific.net/AMM.241-244.3034