Speeding-Up Fractal Image Compression by Using Classification Range Blocks

Dr.Jamila Harbi
Al-Mustensiriyh University,college of science,Computer Science Department/Baghdad
Email: jamela1232002@yahoo.com

Received on: 24/9/2012 & Accepted on: 4/4/2013

ABSTRACT

In fractal compression technique, an image is partitioned into sub blocks called range blocks, each of which is encoded by matching it (after an appropriate affine transformation) with a block chosen from a large pool of domain blocks, which is constructed from the image itself. The problem is that the encoding is very time consuming because of the need to search in a very large domain pool.

Our proposed approach presents a speed algorithm to reduce the encoding time called Classification Range Blocks. This technique will be reducing the size of the domain pool. The proposed method yields superior performance over conventional fractal encoding.

In our proposed speeding technique, we partitioned the image by using fixed block size partitioning and computing the mean and variance for each blocks. The blocks have the variance ranging from (250, 500, 750, 1000, and 1250) only used in matching process between pair range-domain blocks.

Keywords: Fractal Encoding, Classification, Partitioning.

تسريع ضغط الصورة كسوريا باستخدام تصنيف كتل المجال

الخلاصة

تقطع الصورة في تقنية الضغط الكسوري الى كتل فرعية تسمى بكتل المجال و ترمز كل واحدة بالمطابقة (بعد اختيار تحويل افيني مناسب) مع كتلة المجال المجال المقابل و التي تبنى من الصورة نفسها. مشكلة الترميز في الضغط الكسوري تحتاج الى وقت طويل في عملية البحث في مستودع كبير Domain Pool من كتل المجال المقابل.

قدمنا في البحث المقترح خوارزمية مسرعة لتقليل وقت الترميز وتدعى الخوارزمية المقترحة بخوارزمية تصنيف كتل المجال. هذه الخوارزمية ستقلل من حجم مستودع كتل المجال المقابل. الخوارزمية المقترحة اعطت نتائج افضل من الطرق التقليدية للضغط الكسوري.

https://doi.org/10.30684/etj.31.6B.8
2412-0758/University of Technology-Iraq, Baghdad, Iraq

اعتمدت الخوارزمية المقترحة على تقطيع الصورة الى حجم الكتل الثابت و حساب معدل و التغاير لكل كتل المجال. الكتل التي لها تغاير ضمن المدى (1250، 1000، 750، 500، 250) فقط التي تستخدم في عملية المطابقة بين زوج كتل المجال و المجال المقابل.

الكلمات المرشدة: ترميز الكسوريات، التصنيف، التقطيع

INTRODUCTION

The fractals theory has proved to be suitable in many fields and particularly interesting in various applications of image compression. First important advances are due to M. F. Barnsley who introduces for the first time the term of Iterated Function Systems (IFS) [1][2] based on the self-similarity of fractal sets. Many objects can be closely approximated by self-similarity objects that might be generated by use of IFS simple transformations. From this assumption, the IFS can be seen as a relationship between the whole image and its parts, thus exploiting the similarities that exist between an image and its smaller parts [3].

In Fractal coding methods based on partitioning iterated function system (PIFS), the main idea is to take different parts of the image at different scales are similar. As a matter of fact, they are block-based algorithms that intend to approximate blocks of a determined size with contractive transformations applied on bigger blocks [4]. However, in theory the shape of the segments to encode is not restricted. An image partition into blocks, for each block of the partition (called range block) an address for a similar image block, called domain, and a small set of real quantized transform parameters [5] . These parameters determine an affine mapping of the intensity values of the domain block that is shaped like the range block. These mappings are chosen such that the intensities in each range block are approximated by the transformed intensities of the corresponding domain block. The fractal transform operates on an arbitrary image by mapping domains to ranges as specified in the fractal code. Contractively of the transform may be provided by a simple condition on one of the parameters of the affine mappings [6].

In this paper we design and implement speeding–up fractal image encoding depended on the idea of reducing the mapping search operations. In this method we suggested a new search mechanism for determining the IFS-codes between the range and the domain blocks, achieving significant reduction in the encoding time.

FRACTAL IMAGE ENCODING

Fractal image compression (FIC) is also called as fractal image encoding because compressed image is represented by contractive transforms and mathematical functions required for reconstruction of original image. In fractal encoding, the image is compressed by using the partial self-similarity of the images as redundancy, self-similarity well approximating the block to be encoded is extracted from the image and the transform parameter for a contraction transform representing the self-similarity is used as code [7]. Therefor, an image is partitioned into a set of ranges blocks. The encoding of each range block search through all of Domain pool (D) to find a best domain D_i (i.e. $D_i \in D$) which minimizes the collage error:

$$E(D,R) = \|R - sD - o\|^2 \dots\dots\dots. (1)$$

Where, s is the contrast coefficient and o is the offset coefficient.
That is, find the part of image that most looks like the image above R_i. There are 8 ways of symmetry operation to map one block onto another. Minimizing eq. (1) means two things. First, it means finding a good choice for D_i, second, it means finding good contrast s_i and brightness o_i for w_i [8][9]. The procedure of finding a fractal model for a given image is called encoding, compression, or searching for a fractal image representation [10].
Fractal image encoding consists of these steps:

Step1: Partitioning an image into ranges blocks (R) and domains blocks (D),

Step2: Search for an appropriate D block for each R block, and

Step3: Find an affine transformation that adjusts the intensity values in the D to those in the R [11].

After partitioning a given image into ranges *R* and domains *D*, blocks D_i and maps w_i should be found so that when w_i applied a to the part of the image over D_i, the minimum error between R_i and D_i is obtained (i.e. D_i covered R_i).

$$E^2(R,D) = \frac{1}{n}\left[\sum_{i=0}^{n-1} r_i^2 + s(s\sum_{i=0}^{n-1} d_i^2 - 2\sum_{i=0}^{n-1} d_i r_i + 2o\sum_{i=0}^{n-1} d_i + o(no - 2\sum_{i=0}^{n-1} r_i)\right] \dots\dots (2)$$

At each mapping instance one determines the mapping coefficients (s and o) which called IFS coefficients:

$$s = \frac{n\sum_{i=0}^{n-1} r_i d_i - \sum_{i=0}^{n-1} r_i \sum_{i=0}^{n-1} d_i}{n\sum_{i=0}^{n-1} d_i^2 - \left(\sum_{i=0}^{n-1} d_i\right)^2} \dots\dots\dots\dots (3)$$

$$o = \frac{\sum_{i=0}^{n-1} r_i \sum_{i=0}^{n-1} d_i^2 - \sum_{i=0}^{n-1} r_i d_i \sum_{i=0}^{n-1} d_i}{n\sum_{i=0}^{n-1} d_i^2 - \left(\sum_{i=0}^{n-1} d_i\right)^2} \dots\dots\dots\dots (4)$$

From the overall, the encoding process implies finding the blocks R_i and corresponding D_i by minimizing distances between them using eq.2, which is the goal of the problem [12][13].
This is why fractal compression is slow. Each range block must be compared to all domain blocks. For each of these comparisons, all 8 orientations must be checked. This allows the best match to be found. This is also illustrating, why fractal compression is lossy compression method, since sometimes an exact match cannot be found [14].

FRACTAL DECODING PROCESS

The encoding process of FIC is time-consuming, for each range block is compared with all domain blocks among the domain block pool. Most of the research on fractal image compression concentrates on speeding up encoding. But the decoding scheme of FIC is simple and fast. The decoding of an image using iterative decoding consists of any types of partitioning to determine all the ranges in the image. For each range *Ri*, the domain *Di* that maps to it is shrunk by two in each dimension by averaging non-overlapping groups of 2 x 2 pixels. The shrunken domain pixel values are then multiplied by *si*, added to *oi*, and placed in the location in the range determined by orientation information (i.e., apply affine transformation equation $R_{i=} s_iD_i + o_i$) [50] . This represents the first iteration. Normally about 10 iterations are sufficient to give Fractal image compression an appropriate approximation for the fixed point. Further iteration does not generally improve the image. Hence, the decoding time is significantly shorter. In addition, a better peak-to-peak signal-to-noise ratio (PSNR), defined for a 8-bit grey-scale image is expected [12][15].

SPEEDING-UP FRACTAL IMAGE COMPRESSION

Fractal encoding algorithm consuming long encoding time, this weak aspect makes fractal compression method still not widely used as standard compression although it achieves a high compression performance, since time is one of the most considerable factors in any compression method. In order to speed up the encoding process in the part of matching process from eq.2 (i.e., $\sum_{i=1}^{n} r_i d_i$ in this part the encoding process suffered long time to compare each domain block with all range blocks in range image to find the best approximation between domain block and range block), therefore to reduce the encoding time we must reduce the number of range blocks in matching process. In this work we proposed a new approach to reduce the number of domain blocks.

CLASSIFICATION RANGE BLOCKS APPROACH

Fractal encoding in our proposed Algorithm is done by finding a linear transformations of our domain block to arrive to the best approximation of a given range block, i.e., for each R_i search through all of D (i.e., domain pool) to find $D_i \in D$, which minimizes eq.1. This domain is said to cover range (i.e. the part of the image that most looks like the above R_i).

The classification of range blocks and domain blocks serves the purpose of reducing the encoding time (i.e., in each search only domains for a particular class need to be examined). If the original image contains an edge running through a range blocks, then domain in which contain only ‘flat’ pieces of the image can be safely discarded when searching for a good match for that range.

The flat region means that all pixels of this region have the same value or close to each other. There is no effect of symmetry operation on a region like this. So there is no matter to neglect the symmetry operation just for the flat region. This is the base of our proposed approach called Classification Range Blocks.

Before beginning with the mapping operation, the range will be examined in which contain an “edge” or a “flat” region of the image by using the variance as a criteria, a flat region of image has variance = 0. Algorithm 1 utilized steps requirement to perform our Classification Range Blocks approach, in which modified the encoding module.

Algorithm 1: Classification Range Blocks steps

Step1: Compute the mean for the current range block:

$$Mr = \frac{1}{X_{size} \times Y_{size}} \sum_{i=0}^{X_{size}-1} \sum_{j=0}^{Y_{size}-1} X_{ij} \quad ...(4)$$

Where: X_{size} represent the width of the range.
Y_{size} represent the High of the range.
X_{ij} represents pixel (i,j) in the range.
M_r represents the mean of all pixels in the range block.

Step2: Compute its variance:

$$Vr = \sqrt{\frac{1}{X_{size} \times Y_{size}} \sum_{i=0}^{X_{size}-1} \sum_{j=0}^{Y_{size}-1} (X_{ij} - Mr_{ij})^2} \quad ....(5)$$

Step3: If Vr = 0 then do the mapping operation with identical case only
Else do the mapping operation with eight cases of symmetry.

In our approach must be finding the best domain block for every range block by using the variance as a criteria but without need any symmetry transformation, therefore the eight transformations for only one specific domain block that gives the best IFS match with the considered range block with no need for transforming the other domain blocks (see Fig.1). This will lead to avoid number of search operations equal to the number of domain blocks multiplied by seven, the number of discarded operation is: $1000_{range} \times 900_{domain} \times 7_{symmetries} = 6{,}300{,}000$. Also, in this case, the three bits of symmetry will be saved: $1000range \times 3_{bit\,/}\ 1024 = 2.93$ KB.

THE RESULTS OF PROPOSED APPROACH

Our Classification Range Blocks approach is applied on the same image which are used in the traditional schema, Table(1) illustrates the results of applied our approach on the pepper and Lena images.

Our approach gives higher quality image in case of small block size (i.e. more than 2500 range blocks), and about 0.4dB in loss in the other cases. Figure (2) (A& B) show some reconstructed images.

The Effect Of Encoding Process Parameters

Also, we study the effect of fractal parameters on encoding process by using our new approach that will showing in these sections:

Quantization Test

We study the effect of quantization on the reconstructed image. Quantization tests study the effect of the number of bit for scale coefficient, the number of bit for offset coefficient, and the total bits (Scale bits+ Offset bits).

In our work we used the block size 4x4, domain size=128x128, domain step size=2, Max. Scale =3, Min. scale=-1.

The reconstructed image compressed with and without used quantization process is shown in Figure (3) and Table (2). Also, we can see from the comparison between them (i.e., quantization and without used quantization) that when the quantization increased the compression ratio values and causes a decrease in PSNR values.

Block Size Effect

The effect of block size by using speeding up approach is studied. Figure (4) showed the reconstructed images with different block size and V_r varying values.

3. Step Size Tests

To perform the step size tests with 5 bits for scale and 7 bits for offset after quantization. The reconstructed images by using Classification Range Blocks approach can be seen in Figure (5).

THE RECONSTRUCTED IMAGE

In our approach we can reconstruct image in decoding process the same way of its in traditional Fractal Image Compression technique, but in Classification Range Blocks technique we need a few iteration to reach the attractor. Fig.6 shows the decoding process.

CONCLUSIONS

In this work, a new approach is suggested for speeding up the operations in the encoding process, which leads to reduce that time. The Classification Range Block approach is suitable for use with any image. The experiment results show that the compression it achieves with fractal encoding is a good reconstructed images with high compression ratio reach to rate of 70% , but the encoding time accelerated much than the traditional methods. Encoding time decreasing with rate of 85% than in the traditional method. The reconstructed image in a new approach needs 4^{th} iteration to reach an attractor. When we applied a new approach the symmetry will be neglected.

REFERENCES

[1]. Barnsley, M. F. "Methods and apparatus for Image compression by iterated function systems", United States Patent Number 4, 941, 193, 1990.

[2]. Barnsley and S. Demko, M. F. "Iterated function systems and the global construction of fractals", Proceedings of the Royal Society of London, A399: 243-275, 1985

[3]. Barnsley and L. P. Hud,M. F. "Fractal Image Compression", AK Peters, Ltd., Wellesley, Massachusetts, 1993.

[4]. Jacquin A., "Image Coding Based on a fractal Theory of Iterated Contractive Image Transformations", *IEEE Transactions on image processing*, Vol1, pp 18-30, January 1992.

[5]. Sofia Douda , Abdelhakim El Imrani, and Abdallah Bagri, " A new approach for improvement of fractal image encoding", IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 04, 1387-1394,2010.

[6]. Fisher Y.," Fractal Image Compression: Theory and Application", Spinger Verlag Edition, New York, 1995.

[7]. Robert J., “Combining Fractal Compression with Sampling and Interpolation”, Internet paper, January, 10, 2003.

[8]. Gonzalez R. C., and Wintz P., “Digital image processing”, 2nd ed., Addison-Wesley publication company, 1987.

[9]. Henry Xiao, "Fractal Compression", Queen's University, Canada, April, 2004.

[10]. Vijayshri Chaurasia and Ajay Somkuwar, "Review of a novel technique: fractal image compression", International Journal on Emerging Technologies 1(1): 53-56, 2010.

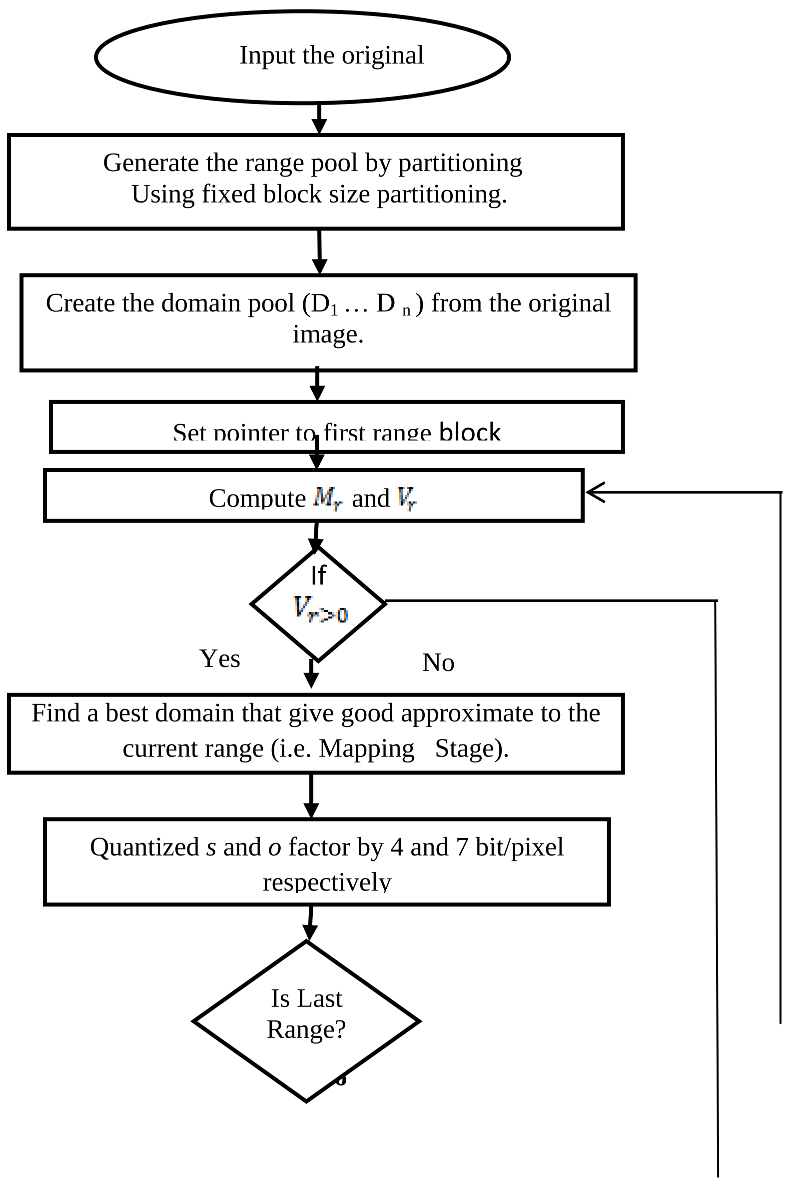
[11]. Mahadevaswamy H.R., "New Approaches to Image Compression", Ph.D. thesis, Regional Engineering College, university of Calicut, December, 2000.

[12].Satish K., "An Introduction to Image Compression", http://www.debugmode.com/imagecmp, 22, Oct, 2001.

[13]. Boukelif Aoued," Accelerating Fractal Image Compression by Domain Pool Reduction Adaptive Partitioning and Structural Block classification ", 3rd International Conference: **S**ciences of **E**lectronic, **T**echnologies of **I**nformation and telecommunications– TUNISIA , March 27-31, 2005.

[14]. Furao and O. Hasegawa,S. " A fast no search fractal image coding method', Signal Processing: Image Communication, pp. 393–404, 19 ,2004.

[15]. Shraddha Viraj Pandit 1 , M.V. Kulkarni 2, M.L.Dhore," Image Comparison Search Engine based on Traditional and Improved Fractal Encoding Techniques", International Journal of Recent Trends in Engineering, Vol 2, No. 3, November 2009.



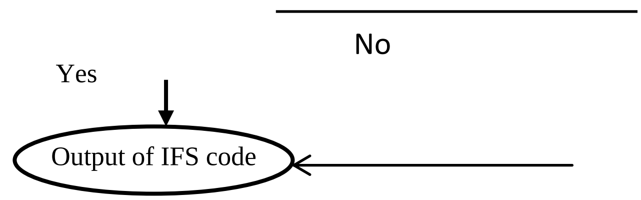


Figure (1) the block diagram of proposed approach.

Table (1) the result of performs the speed up by using the Classification Range Blocks approach applied on the pepper and Lena images.

Domain Step=4 Minimum Scale =-1 and Maximum Scale =1 S = 5 and O = 7 Minimum Offset = -256 and Maximum Offset =256						
Item	V_r	***No. of Blocks***	***CR***	***SNR (dB)***	***PSNR***	***ET(sec)***
Lena	250	3035	6.80	24.28	34.45	10.45
	500	2668	7.70	23.77	33.64	10.04
	750	2363	8.65	22.98	32.86	8.96
	1000	1983	10.26	21.61	31.50	7.34
	1250	1337	11.18	18.91	28.83	5.33
Pepper	250	2627	7.80	28.80	34.17	9.08
	500	2323	8.77	28.11	33.48	8.55
	750	2095	9.69	27.09	32.46	7.11
	1000	1854	10.92	25.97	31.34	5.92
	1250	1381	14.63	23.84	28.86	4.36

$V_r = 250$ CR=7.80 PSNR=34.17 ET=9.08 Sec

$V_r = 750$ CR=9.69 PSNR=32.46 ET=7.11 Sec

$V_r = 1250$ CR=14.63 PSNR=28.86 ET=4.36 Sec

(A)

$V_r = 250$ CR=6.80 PSNR=34.45 ET=10.45 Sec

$V_r = 750$ CR=8.65 PSNR=32.86 ET=8.96 Sec

$V_r = 1250$ CR=11.18 PSNR=28.83 ET=5.33 Sec

(B)

Figure (2) The reconstructed images by using Classification Range Blocks approach for two images of size 256x256 pixels A) pepper image ad B) LENA image .

Without Quantization
PSNR=33.98,CR=4.71
$V_r = 250$, ET=10.03sec

Original Image

With Quantization
PSNR=31.23,CR=7.65
$V_r = 250$,ET=12.09 sec

Figure (3) Show the reconstructed images with and without using quantization process.

Table (2) the effect of total number of bits on the Compression ratio values and PSNR values when the $V_r = 250$.

ScaleBits	*OffsetBits*	*PSNR*	*CR*	*ET (sec)*
3	5	33.45	13.11	13.09
3	6	32.68	11.63	13.01
3	7	32.23	10.74	12.46
4	5	33.63	10.27	12.40
4	6	32.91	9.52	12.23
4	7	31.23	7.65	12.09
5	5	32.62	8.52	12.25
5	6	32.02	7.81	12.14
5	7	30.19	7.32	11.98
6	5	33.32	12.81	13.06
6	6	32.22	11.74	12.78
6	7	31.41	10.03	12.47
7	5	33.56	7.17	12.82

7	6	33.66	6.77	12.90
7	7	33.72	6.12	12.98
Without quantization		33.98	4.17	10.03

No.blocks=2980,C.R.=4.72
SNR=15.55dB,PSNR=25.18
E.T.=7.04 Sec

No.blocks=1500,C.R.=8.72
SNR=14.28dB,PSNR=23.75
E.T.=6.27 Sec

No.blocks=1000,C.R.=14.75
SNR=13.24dB,PSNR=21.72
E.T.=4.94 Sec

Figure (4) The reconstructed images with different block size (4x4, 8x8, and 16x16) respectively and the V_r value is 750.

StepSize=1 | StepSize=2 | StepSize=3 | StepSize=4

CR=5.38
PSNR=31.72
ET=12.52 sec

CR=5.38
PSNR=30.06
ET=7.83 sec

CR=5.38
PSNR=28.59
ET=4.82 sec

CR=5.38
PSNR=27.62
ET=2sec

Figure (5) the reconstructed images by using V_r=500.

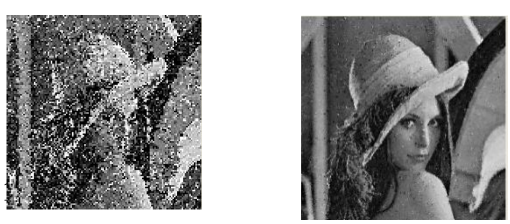


Figure (6) the reconstructed image.