Anew method for speeding up Fractal Compression using different value of affine transformation and highest gray value of pixel

Asst. Prof. Hind Rostom Mohammed
University of Kufa College of Mathematical &Computer Sciences/Computer Dept.
hind_restem@yahoo.com

Hanaa Muhsen Ali Al Shemary
College of Engineering
Babylon University

Abstract: In this paper new algorithm for speeding up fractal image compression is presented. A new adapted method based on computing the highest value of the pixel of the image to reduce the computational complexity in the encoder stage *and which are led to decreasing the encoding time while the reconstructed image from the work as good as we want. For increasing the effectiveness of search stage we used another type of partitioning method* that led to increase the flexibility of range partition, this method is HV-partition. *We applied this method on images and also present a comparison of this method against other method which used to speed the fractal compression.*

Key word: fractal image compression, HV-partition, number of pixels in each block, range ,domain

1.Introduction

Increasingly, more images are acquired and stored digitally. These images may be very large in size and number and compression offers a means to reduce the cost of storage and increase the speed of transmission. Image compression is minimizing the size in bytes of a graphics file without degrading the quality of the image. The resolution in file size allows more images to be stored in a given amount of disk or memory space. It also reduces the time required for images to be sent over the Internet or download from WebPages[14]. Image compression is required to minimize the storage space and reduction of transmission cost. Medical images like MRI and CT are Special images require lossless compression as a minor loss can cause adverse effects. Prediction is one of the techniques to achieve high compression. It means to estimate current data from already known data [13]. Fractal image encoding is attractive due to its potential high compression ratio, fast decompression and multi-resolution properties. However, the key problem is the long encoding time consuming of existing methods. The main reason is that the similarity matching computation between range and domain blocks is complex. In response to this problem, an effective method is to perform image blocks classification before blocks matching. After classification, exhausting searching is replaced by partial searching, and the similarity matching computation is done only during the blocks in the same class[1].

Most of the speeding techniques cause degradation in image quality; this is due to the fact that the speeding up(reduction in the encoding time) should be on the account of image fidelity[2]. An image may be described in terms of a set of mathematical rules (Iteration Function System(IFS)) which can be represented using relatively much less number of bits. As a result, large amount of data reduction can be achieved. The value of compression ratio in this case may go up to 1000:1depending on the image to be coded. However, computational cost of encoding process is usually very high, because a large number of iterations is needed by IFS to converge. To reduce the computational cost, image is partitioned to small blocks and then fractal coding procedure is applied [3].

The remaining sections of this paper is organized as following, section-2 The Speed Problem and Encoding Time selection in section- 3 Error Metrics, section- 4 Fractal Encoding based on Highest Gray Value of Pixels (HGVP) Technique, section-5 the Compression Algorithm, section-6 Encoding Process and Decoding process, section-7 the Results and discussion and section-8Conclusions in the final.

2. The Speed Problem and Encoding Time

The essence of the compression process is the pairing of each range block to a domain block such that the difference between the two, under an affine transformation, is minimal. This involves a lot of searching.

In fact, there is nothing that says the blocks have to be squares or even rectangles. That is just an imposition made to keep the problem tractable.

More generally, the method of finding a good PIFS for any given image involves five main issues:

1. Partitioning the image into range blocks.
2. Forming the set of domain blocks.
3. Choosing type of transformations that will be considered.
4. Selecting a distance metric between blocks.
5. Specifying a method for pairing range blocks to domain blocks.

Many possibilities exist for each of these. The choices that Jacquin offered in his paper are:

1. A two-level regular square grid with 8x8 pixels for the large range blocks and 4x4 for the small ones.
2. Domain blocks are 16x16 and 8x8 pixels in size with a sub sampling step size of four. The 8 isometric symmetries (four rotations, four mirror flips) expand the domain pool to a virtual domain pool eight times larger.
3. The choices in the last point imply a shrinkage by two in each direction, with a possible rotation or flip, and then a translation in the image plane.
4. Mean squared error is used.
5. The blocks are categorized as of type smooth, midrange, simple edge, and complex edge. For a given range block the respective category is searched for the best match [6].

The objective of this project is to reduce the time complexity in the encoding step. where the time complexity arises in the encoding step the time complexity arises, and how we can reduce it. Consider an image that is 256x256 pixels in size. We partition it into 8x8 non-overlapping ranges R[7]. We also partition it into 16x16 overlapping domains D = D ,58;081. Now, for each R^i, we search through all D to find a D^i which minimizes some threshold. In other words, we try to find a part of the image that looks similar to Ri. There are eight ways to map one square to another. So for each range Ri, 464,648 domains have to be compared. Various techniques have been proposed to overcome the time complexity. One such technique is the classification of domains based on some feature, such as the edges or bright spots . A domain once discarded removes from the pool all other similar domains for a particular range. Another technique is to classify the domains as multidimensional keys. This reduces the complexity from O(N) to O(log N) . The weakness of this method is that the size of the range Ri is fixed. This is overcome in the quadtree partition method. Here, a square in the image is split up into four equal-sized sub-squares when it is not covered sufficiently by the domain. This process is repeated recursively, starting from the complete image and continued until the squares are small enough to be covered within some specified threshold[8].

3. Error Metrics

Two types of the error metrics used to compare the various image compression techniques are the Mean Square Error (MSE) and the Peak Signal to Noise Ratio (PSNR). The MSE is the cumulative squared error between the compressed and the original image, whereas PSNR is a measure of the peak error. The mathematical formulae for the two types are:

$$\text{MSE}=\frac{1}{N\times N}\sum_{X=0}^{N-1}\sum_{Y=0}^{N-1}\left[f'(x,y)-f(x,y)\right]^2 \quad (1)$$

$$\text{PSNR (dB)} = 10*\log_{10}\,(255^2/\text{MSE}) \quad (2)$$

Where f (x, y) is the original image, f (x, y) is the approximated version (which is actually the decompressed image) and N are the dimensions of the images. A lower value for MSE means lesser error, and as seen from the inverse relation between the MSE and PSNR, this translates to a high value of PSNR. Logically, a higher value of PSNR is good because it means that the ratio of signal to noise is higher. Here, the 'signal' is the original image, and the 'noise' is the error in reconstruction. So, if you find a compression scheme having a lower MSE (and a high PSNR) , it can be recognized that it is better one [5].

4. Fractal Encoding based on Highest Gray Value of Pixels (HGVP) Technique.

Fractal image encoding is capable of yielding competitive rate distortion ;however; it suffers from long time, come from search all ranges through all D to find D_i which minimize:

$$\left\|R-\left[s_i\,sym(D_{k,i})-o\right]\right\| < \left\|R-\left[s_k\,sym(D_{k,m})-o_k\right]\right\| fc \quad (3)$$

Where (s_i ,o_i) are the corresponding affine transform coefficients which map the domain block (D_i) to match the range block (D_i) to match the range block (R). The sym (D_{kj}) refers to the output obtained by applying symmetry operation (k) upon block (D_i), the scaling coefficient(s) is clamped to [-1,1] to ensure the convergence of the decoding process. In fact both (s) and (o) should uniformly quantized to yield $\bar{s}$ and The collage error for range (R) can be compute from:

$$E(R,D)=\left\|R-(\bar{s}D+\bar{o})\right\|^2 \quad (4)$$

From eq.(5) there are eight ways to map one selective point to another. So for each R_i many of domains have to be compared, i.e., eq.(4) involve the full search of domain pool. Various techniques have been proposed to overcome the time complexity. Therefore, we proposed a new technique attempt to accelerate the searching. This technique based on limit number of gray-value of pixels. In our new technique we used a very simple bit allocation scheme from a given range block. We have partition an image, such that each range is a flat partition of contour, then re arrangement of pixels depend on the gray-values. pixels arranged in a ascending arrangement, starting from the minimum value and jump, uniformity to till the maximum one . we chose fixed number of pixels(which they have the maximum values) depend on block size. Usually fixed number equal to:

Threshold condition = Min BLK Size of range block-1 (5)

Each block satisfy this threshold are used to match the domain blocks in the collage image. Next step, we used threshold condition. This condition chose fixed number of pixels[4].

5. The Compression Algorithm

The compression algorithm consist of several stages, quadtree partition in fractal, encoding process, uniformity criteria, and eight affine transformation applying on each fractal block separately to achieve minimum time in encoding stage. In our work we reduced number of range blocks by select the limit number of highest Gray level in range blocks, and used these blocks to match the domain blocks, This approach is outlined in the following algorithm:

1. Load an input image in to buffer. Partition the image into small blocks (SB) with non-overleap (i.e., range blocks).
2. Choose big blocks (BB) with overlap(i.e., domains blocks).
3. Get first SB from R-block.
4. Arrange the pixel values of the SB ascending.
5. Choose the highest values of the pixel values.
6. Determine number (n) out of (the highest values of the pixels).
7. Loop on all BBs for each one calculates and quantization the value of contrast (s) and brightness (o) by perform all the affine transformations,($R_i \approx s_i D_i + o_i$) map-ping.
8. Choose the block that resembles the R-block with (RMSE), compute the encoding parameters that satisfy the mapping. Those parameters represents fractal element.

6. Encoding Process and Decoding process

1- In fractal encoding process the original image is partitioned into range blocks (non overlapping blocks) and domain blocks (possibly overlapping blocks according to a specific step domain) using the different block size image partitioning .Compression begins by partitioning the image into R_i. for each R the Algorithm chose the D that can be used to approximate that (R), most closely using one of a small set of affine transformations.

2- Get first SB(small block) from R-block. Arrange the pixel values of the SB ascending. Choose the highest values of the pixel values. Determine number (n) out of (the highest values of the pixels).

3-The best matching domain block (D) (i.e., the more self-similar blocks) must be found for each range block (R) by applying an approximate affine transformation.

4- Compute contrast (s) and brightness (o) by using Eq.(13) and Eq.(14) respectively for each range block and then root mean square error (d_{rms}) between two blocks R(rang block) and D(domain block) by using Eq.(15).

5-The compression process is finished by storing only the descriptions of these transformations for every range block. The computation; needs for each codebook (D_i) compute an optimal approximation (i.e. s and o values) is determined by calculate the values of scaling (s) and offset (o) coefficients by using the least square optimization, i.e. :

$$S=\frac{n\sum_{i=1}^{n} d_i r_i - \sum_{i=1}^{n} d_i \sum_{i=1}^{n} r_i}{n\sum_{i=1}^{n} d_i^2 - \left(\sum_{i=1}^{n} d_i\right)^2} \quad (6)$$

$$o=\frac{1}{n}\left(\sum_{i=1}^{n} r_i - s\sum_{i=1}^{n} d_i\right) \quad . (7)$$

And

$$d_{rms}=(f,g)=\sqrt{\frac{1}{N\times N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left(f(x,y)-g(x,y)\right)^2} \quad (8)$$

Where, r is the range of block.
d is the domain of block,
n is the number of pixels.

The task of finding self-similarities (via the matching process) by full search of the domain pool is of high computational complexity and is considered to be the major lost time of the fractal image compression method. The eight affine transformations will be applied to the image with size 256 ×256 separately(select one type of transformations to operate in each processing of selection image) as explained in the table (1)and (2) we can see the results of these applications will be of high quality and highly speed up encoding time which we need as the result of this work .

One of the most remarkable features of fractal image compression is the simplicity of its decoder. The reconstruction of the image is obtained by iterating the mapping (W^i) on any initial image. This means that for each w^i, we find the domain D^i, shrink it to the size of its range R^i, multiply the pixel values by the scale factor(s^i) and add the offset factor (o^i), and put the resulting pixel values in position of R^i. typically, 10 iterations are sufficient. In this work we used only 8 iterations.

7. The Results

In this work using two medical image (T1 and T2) with size (256×256). The minimum values and maximum values of the scaling which are used in this results (from -1.5 to 1.5) and the minimum values and maximum values of the offset (from -200 to 500). The Qntscl =100.00 and Qntofs=1.000 .Eight iterations are used in decompression process for extracting the reconstruction image. A problem with the traditional search method is that more D-block is examined then actually necessary. Many of them are totally different from the R-block and it is not suitable to sent to the affine transform to match them with the considered range block. Thus only suitable, similar domain blocks (i.e., have similar characteristics) should be send for affine matching. According to this work we need to improve the values of encoding time to make the encoding time as acceptable as quality of reconstructed image. in this work used three medical Gray scale images with size 256× 256 and used the quadtree partitioning techniques with minimum block size 4 and maximum block size 8. In the following experiment results the best value of Mean(M) is 0.1 and the best value of standard deviation is 0.7 which are using in this work to make the encoding time as acceptable and high quality of reconstructed image.

8. Conclusions

In this work, the fractal image compression method was implemented . As mentioned in the previous work, this method has the disadvantages of a very long encoding time. For this reason, a new approach are suggested for speeding up the operations in the encoding process, which leads to reduce that time. And we used this approach. Many tests were performed on different images to study the behaviour of normal and speeded FIC method. From the results of these tests , the following conclusions could be summarized:

1. The encoding time is inversely proportional with each of : Standard deviation(Std) and Mean (M). PSNR is inversely proportional with each of (Std) and (M).Compression ratio is proportional with each of (M) and (Std).Number of blocks of quad tree results image are inversely proportional with each of (M) and (Std).The types of affine transformation which are used in this work make low encoding time when we used the eight type separately in the encoding process.
2. In order to keep the quality of the reconstructed image, it is not recommended to increase the block size more than 16×16 although this operation will increase the compression ratio.
3. The encoding time is inversely proportional with number of blocks, also the PSNR is inversely proportional with number of blocks , while the compression ratio C.R is proportional with number of blocks.
4. The encoding time is inversely proportional with block size, also the PSNR is inversely proportional with block size ,while the compression ratio C.R is proportional with block size.

5- Reconstructed image is influenced by Rate Distortion which also effects PSNR value since it is directly proportional with Rate Distortion on one hand, on the other hand we see that Rate Distortion is inversely proportional with encoding time.

6- PSNR is directly proportional with number of pixel. The encoding time is directly proportional with number of pixel. Compression ratio is directly proportional with number of pixel .

Reference

[1] HAN Jin-shu, 2007,***'Fast fractal image compression by pixel peaks and valleys classification' '***

[2] Fisher , Yuval, 1995, ***'Fractal Image Compression*** **theory and application' .** New York , Inc.

[3] Bhabatosh Chanda, 2009, ***'Digital Image Processing and Analysis'*** 12th , New Delhi.

[4] Jamila Harbi Saoud Al-A'mri, January 2001,'***Fractal Image Compression'***, thesis, Baghdad University

[5] Hanaa Mehsen Ali, May 2005,***'speeding up fractal image coding by using indexing block', thesis, Technology University.***

[6] La Torre, F. Mendivil and E.R. Vrscay, ***Iterated function systems on multifunction',*** (Springer-Verlag, Heidelberg,2006), '

[7]Madhuri A.Joshi, 2009,***'digital Image Processing An Algorithmic Approach***', 4th ed. New Delhi.

[8] K. U. Barthel , H. L. Cycon, DECEMBER 2003,' ***Image denoising using fractal and wavelet.***

[9] Fuzhou, Fujian, ,31-February 4, 2005,***'VSA-based Fractal Image Compression'***College of Mathematics

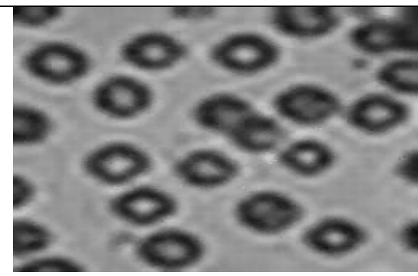
[10] A. E. Jacquin, Oct. 1993, ***' Fractal Image Coding '***: A Review, Proceedings of the IEEE, Vol.81, No.10.

[11] David Tucker, August 19th 2005,' ***Affine mode graphics on the Nintendo Virtual Boy'.***

[12] Brendt Wohlberg and Gerhard de Jager, DECEMBER 1999,***' A Review of the Fractal Image Coding Literature'.***

[13]S.M.Ramesh,.A.Shanmugam,:***Medical Image Compression using WaveletDecomposition for Prediction Method’,*** (IJCSIS) International Journal of Computer Science and Information Security, Vol. 7, No. 1, 2010.

[14] D.Vijendra ,P.Subramanian, ***Energy Efficient Wavelet Based Medical Image Compression Using Modified ROI EZW’,*** Proceedings of the Int. Conf. on Information Science and Applications ICISA 2010, 6 February 2010, Chennai, India.

Image		
CR	CR=8.46	CR=7.55
PSNR	PSNR=34.343	PSNR=33.232
NOBlk	NOBlk=2267	NOBlk=1955
Time	Time=2.33.348(m)	Time=3.78.987(m)

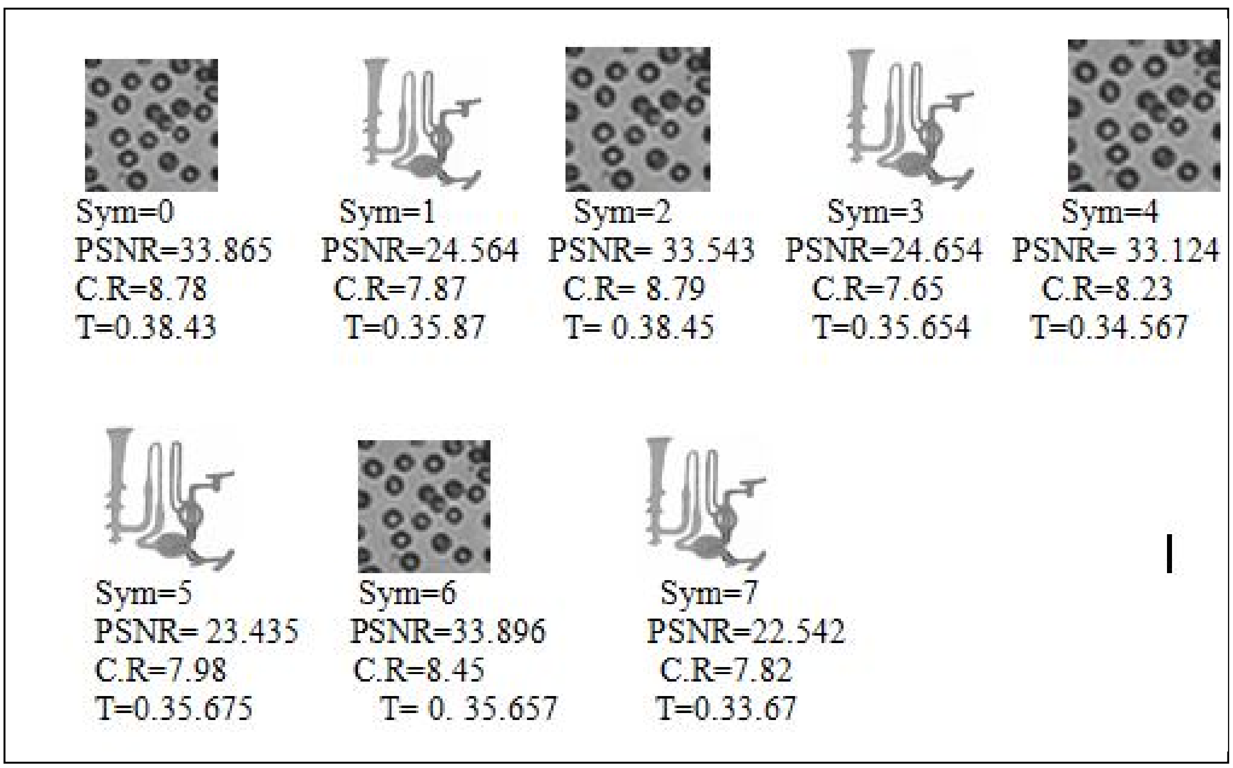
Fig(1): The Results of Suggestion Using all Eight Affine Transformations by Using medical Image (T1 and T2) with size (256×256) and Number of pixels =7.

Type of Symmetrical	Compression Ratio	PSNR	NOBLK	Time(minute)
0	7.44	23.452	2129	0.32.727
1	7.45	23.634	2129	0.32.787
2	7.37	23.443	2060	0.32.807
3	7.44	23.502	2059	0.32.757
4	7.44	23.007	2059	0.32.177
5	7.45	22.749	2059	0.33.57
6	7.54	22.129	2059	0.32.888
7	7.44	21.353	2059	0.32.597

Table (1): The Results of Suggestion Using Eight Affine Transformations Separately by Using medical Image (T1) with size (256×256) and used the value of number of pixels=3

Type of Symmetrical	CR	PSNR	NOBLK	Time(minute)
0	8.54	33.226	2332	0.35.69
1	8.55	32.986	2332	0.35.46
2	8.65	33.237	2332	0.34.59
3	8.65	33.120	2364	0.34.928
4	8.17	32.925	2538	0.33.670
5	8.19	32.960	2538	0.33.455
6	8.64	33.090	2364	0.30.855
7	8.65	33.022	2364	0.30.678

Table (2): The Results of Suggestion Using Eight Affine Transformations Separately by Using medical Image (T2) with size (256×256) and used the value of number of pixels=4



Fig(2): The Results of Suggestion Using all Eight Affine Transformations by Using medical Image (T1 and T2) with size(265×265) and with the value of number of pixels =5.

$$\left\| R - \left[s_i sym(D_{i,j}) - o \right] \right\| < \left\| R - \left[s_k sym(D_{k,m}) - o_k \right] \right\| \; for \quad \forall i \neq k \quad or \quad j \neq m$$

طريقة جديدة لتسريع الضغط ألكسوري باستخدام قيم مختلفة من تحويل الافيني وأعلى قيمة رمادية للبكسل

الخلاصة:
تم تطبيق خوارزمية جديدة لتسريع الضغط الكسوري للصور باستخدام طريقة جديدة تعتمد على حساب اعلى قيمة للبكسل للصورة لتقليل التعقيد في الحسابات في مرحلة الضغط والذي بدورة يقود الى تقليل وقت الضغط اثناء اعادة تركيب الصورة بالوضع الجديد .تم استخدام التقطيع الافقي العمودي لزيادة فعالية البحث والتي تؤدي الى زيادةالمرونة في التقطيع للصورة والذي اثبت بدورة زيادة في تسريع الضغط.