Adaptive Algorithm to enhance fast method of fractal image compression

IT (171)

BY

MSc-Huda A. Al Amawee

Dept. of Computer Science, Collage of Science, AL-Mustansiriya University
e-mail : huda_ros@yahoo.com
phone number : 07901362320

الخلاصة

أن تقسيم الصورة هو من الأمور المطلوبة في ضغط الصورة بواسطة الكسوريات. وقد استخدمنا في هذا البحث التقسيم الرباعي المهيكل والذي يُعد أساس لطريقة تطابق المعيار القياسي. وقد وجدت طريقة تطابق المعيار القياسي لتسريع طريقة التقسيم الرباعي المهيكل، وقد أثبتت هذه الطريقة نجاحها عند التطبيق، لكن الإسراع في عملية الضغط ولد صورة مرجعة رديئة أي صورة ذات نوعية مترد يه ، من اجل ذلك استحدث طريقة تطابق المعيار القياسي المحسنة لتلبية غرضين مهمين ،الأول لتقليل زمن وقت الضغط لاقل ما يمكن ، وقد والثاني هو التوصل إلى صورة مرجعة عالية الجودة جعلت هذه الطريقة تقليص وقت التأخير (الانتظار) في إرسال الصورة المضغوطة عبر الشبكة مما يؤدي إلى الإسراع في عملية الإرسال.

Abstract

In fractal image compression, a partitioning of the image is required. In this paper we used the quadtree partitioning based Global Norm Matching (GNM) method. The GNM method comes for speeding up the quadtree method, this method proves a good success, but the fast of the encoding time generated bad reconstructed image or low quality (PSNR), therefore, the adaptive GNM conducted two purpose, first is to reduce the encoding time as short as possible and second is to obtain a reconstructed image with high quality. This method make the delay time shortly for send the compression image cross the inter net due to speed up the send process.

1- *Quadtree partitioning methods*

First of all quadtree partitioning of an image is usually, described as a hierarchical operation. The root of the tree is the initial image; while each node, corresponding to square portion of the image contains four sub nodes corresponding to the quadtree. The quadtree algorithm as illustrated in Figs (1) [1] [2].

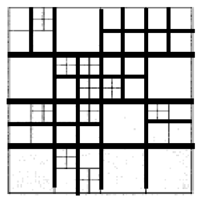


Fig (1): Quadtree partition of an image and example of the system of size 8x8 pixels of image.

2-The quadatree algorithm

The algorithm aims to finding the optimal quadtree encoding for a given compression ratio, the quadtree partition may consist of square blocks of size $2^k \times 2^k$ where $K_{min} < K_{max}$ suppose now that a code with a total rate of N bits is targeted for a block B_i of size $2^k \times 2^k$, $K_{min} < K < K_{max}$ it can be define the ratio (λ) [3].

$$\lambda(B_i) = \left| \frac{d_k - d_{k-1}}{T_k - T_{k-1}} \right|$$

Where d_k and T_k are the distortion and the rate for optimum encoding of block B_i, respectively, and d_{k-1} and T_{k-1} are the sum of distortions and the sum of the rates, respectively, for the optimum encoding of the sub blocks obtained by splitting block B_i in 4 sub blocks of size $2^{k-1} \times 2^{k-1}$. Thus, $\lambda(B_i)$ denotes the reduction in distortion per additional bit for the code when block B_i is subdivided into 4 quadrants. Now start by finding the optimal code for the image partitioned into blocks of maximum size. Then successively split that block with the highest ratio λ until the total rate of code becomes larger than N or all blocks in the quadtree partition have minimum size [4].

3- ***Partitioning the image into:***

I-The Range

An image is partitioned into a set of non-overlapping uniform square regions; these are the ***range*** regions [5]. The range regions may have different blocks size (e.g., 4, 8, and 16) the resulted blocks are called small blocks (SBs) or range block, in this work the minimum range block is (4 and 8) and the maximum range block is (8 and 16)

II -The Domain

By stepping through the image with a step size of (i) pixels, horizontally and vertically list of domain blocks will be created from the image it will call them big blocks (BBs), which are twice the range size by averaging each adjacent four pixels, the sized ***domain*** (D) block will shrink to match the size of range (R) block [6].

III-The Domain pool or Codebook

The domain pool or the codebook can be produced from the domain blocks and their corresponding blocks produced by applying the reflection and rotations by $(90°, 180°, 270°)$ angles operations on the D block [6].

The encoding process is beginning with partition the image into (range, domain and domain pool). The range are selected as flows:

After some initial number of quadtree partitions is made (corresponding to minimum tree depth in this works is 4) the squares at nodes are compared with domains block from the domain library (or domain pool), which are twice the range size. The pixels in the domain blocks are averaged in groups of four so that the domain is reduce to the size of range block, and the code coefficient (s and o) (Scaling (s) and Offset (o)) is found that minimizes the Root Mean Square Error (RMSE) difference between the transformed domain pixel values and the range pixel value. All the potential domains are compared with range blocks [1]. If the resulting optimal RMSE value is above a prèselected threshold and the depth of the quadtree is less than preselected maximum depth, than the range squares is subdivided into four quadrants (which means adding for sub nods to the node corresponding to the range block), and the tree process is repeated. While if the RMSE value is below the threshold, the best match location at the range block and domain pool and the code parameter values (i.e. s and o) should stored. The constitutes one map W_i, however, the collection of all such maps in eq (1) will be constitute the encoding process.

$$w = \bigcup_{i=1}^{N} w_i \qquad (1)$$

The search

For each Range block (R) an optimal approximation must be obtained from the affine transformation:

$$R \approx sD + o \qquad (2)$$

The computation; needs the following steps:

1- For each codebook (D_i) compute an optimal approximation eq.(2)is determined as follows.

(i) calculate the value of scaling (s) and Offset (o) coefficients by using the least square optimization i.e. ;

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

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

r: is the range blocks ,

d: is the domain blocks

n: is the number of pixels .

The above equations yield real values for the coefficients (s and o).

(ii) To store the encoding compactly, the constants (s) and brightness (o) setting have non-uniform distribution, if the values are to be quntized and stored in a fixed numbers or bits, however, a significant improvement in fidelity can be obtained.

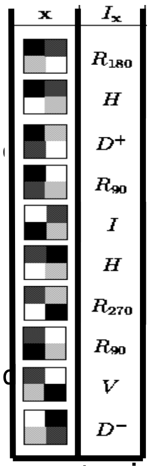
(iii) Use the quantized coefficients (s) and (o) to compute the error E (R,D) from the flowing equation.

$$E\,(R,D)^2 = \frac{1}{n}\Big[\sum_{i=1}^{n} r_i^2 + s\,\Big(s\sum_{i=1}^{n} d_i^2 - 2\sum_{i=1}^{n} d_i r_i + 2o\sum_{i=1}^{n} d_i\Big) + o\,\Big(on - 2\sum_{i=} r \qquad (5)$$

2- Among codebook (D) find the block D_k with minimal error (i.e. $E(R,D_k) < E(R,D_i) \quad \forall i \neq K$)

3- Output the code for the current (R), such code will consists the indices of quantized coefficients (s & o), the symmetry index and the domain codebook number [7]. Compression technique begins by partitioning the image into SBs for each SB, the program chose the BB that can be used to approximate that SB, most closely using one of subtabile set of affine transformation table (1) show the eight symmetries of affine transformations which applied in 1st block to match another one in the search process [3].

Table (1): A set of affine transformation



R_{180} = rotation by 180

H = reflection at the is onal

D^- = reflection at

D^+ = reflection at the

R_{90} = rotation by 90

I = identity

V = reflection at the vertical axis

R_{270} = rotation by 270 degrees

4- Partition Process Algorithm

The algorithm pscudo – code for partition is:

1- Define uniformity threshold for partitioning image and measure the Root Mean Square Error (RMSE), E (R, D) / $\sqrt{\#\,pixel}$ in R of the collage image and the minimum and maximum range sizes (i.e. 4-8, 4-16, 8-16) start with an image of maximal range size.

2-Initialize a stack of range by pushing the initial range on to it.

3-While the stack is not- empty, carryout the following steps,[8]

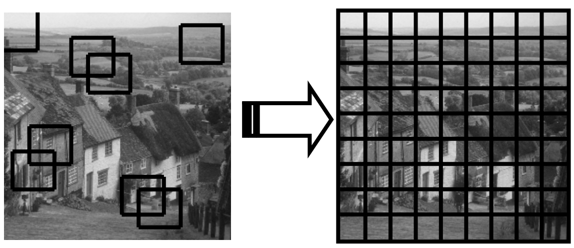
(I) Pop a range blocks R from the stack and search the corresponding codebook (i.e. domain pool) yielding an optimal approximated version of range blocks, (R $\approx s_i D_i + o_i$) with least square error E (R_i, D_i)

(II) if the (RMSE) is less than the decided size, then save the address of D_i.

(III) Otherwise partition of R_i into four quadatree and push them into the stack.

5- Encoding Process Algorithm

The aim of this process is to model a given image as a set of fractal elements. Therefore, the output will be a set of fractal codes that represent the input image. An image is partitioned into a set of non-overlapping uniform square regions; these are the range regions [5]. The range blocks may have different blocks size (e.g, 4, 8, 16) the resulted range block (SB), this image also partitioned into another set of uniform square regions. These are the Domain regions (BB), each adjacent four pixels; the sized will shrink to match the size of (SB_s). Fig (2) shows how image is partitioned.



Domain Regions   Range Regions

The algorithm of compression process is:

1- Load an input image into buffer.
2- Partitioning the image into small blocks with non-overleap call range (SB).
3- Choose the big blocks with overlap called Domain (D).
4- For each block, do step 5 -
5- Loop on all BBs for each one calculates and quantization the value of (s&o), perform all the affine transformations,($R_i \approx s_i D_i + o_i$) mapping.
6- Choose the block that resembles the R-block with the (RMSE), compute the encoding parameters that satisfy the mapping. Those parameters represent one fractal element.
7- Pack the parameters into a more compact shape in order to chive more compression.
8- Store the compact bit stream in the output file

6- Encoding control parameter

The encoding control parameters are:

1- The threshold partitioning parameters, inclusion factor (β) and acceptance ratio (α).

2- The maximum block size of quadtree partition (i.e. 8 and 16).

3-The minimum block size of quadtree partition (i.e. 4 and 8).

4-The number of blocks (i.e. 10 blocks).

5-Height and width of image (256 x 256) pixels.

7- Encoding output data

1-The quadtree-partitioning list of the image.

2-The quantized (s_i & o_i) values for each range.

3-The addressed of the domains used to encode the range.

4-The position (x, y) of the range block.

5-The symmetry index of the orientation operation used to map the (D-block) on to (R-block)

8- Global Norm Matching (GNM) method

GNM method is new technique in fractal image compression. This method will be devoted to satisfy two purposes, first to speeding up the encoding time beside loosing quality in reconstructed image , the second one to enhance the reconstruction image beside to increase the encoding time. Also, this method based in quadatree partitioning. Quadtree scheme will be speed up the encoding process. In our work reach up to 3- levels (i.e.4,8, and 16) in partitioning .

The inherent problem associated with the IFS- based encoding methods lies in finding the proper domain that can be best transformed into a given range. For example, image of size 256×256 pixles has several thousand ranges and domains; therefore searching- matching process will be very comparison requirement. Therefore, GNM method suggested to reduce the encoding time.

9- The Encoding process of GNM method

The basic idea of GNM method based on utilizing of the norm of the domain and range blocks in computing the offset – scale transform coefficient values (i.e. $R_i \approx s_i D_i + o_i$).Moreover higher order norms will also be used to inspect the degree of fitness between the range – domain couples [9]. To derive the mathematical concepts that GNM method based on it, it can be start from eq.(2), by taking the average of eq (2) we get

$$\overline{R} = s\overline{D} + o \qquad (6)$$

The relationship between 2^{nd} order norm of the domain (D) and range block (R) is:

$$R^2 - \overline{R}^2 = (sD + o) - (s\overline{D} + o)^2 \qquad (7)$$

Appling averaging on both sides of equation (7), we have

$$\overline{R^2} - \overline{R}^2 = s^2(\overline{D^2} - \overline{D}^2)) \qquad (8)$$

Similarly, higher order norm lead to:

$$\overline{R^3} - \overline{R}^3 = s^3\left(\overline{D^3} - \overline{D}^3\right) + 3s^2 o\left(\overline{D^2} - \overline{D}^2\right) \qquad (9)$$

$$\overline{R^5} - \overline{R}^5 = s^5\left(\overline{D^5} - \overline{D}^5\right) + 5s^4 o\left(\overline{D^4} - \overline{D}^4\right) + 10s^3 o^2\left(\overline{D^3} - \overline{D}^3\right) + 10s^2 o^3\left(\overline{D^2} - \overline{D}^2\right)$$

In GNM method equations (6-10) have been used to determine the affine transformation coefficients (i.e. s&o). While equations (9 and 10) has been utilize to evaluate the degree of affine fitness between (R&D) couples. Such mechanism will greatly reduce the matching requirement because: -

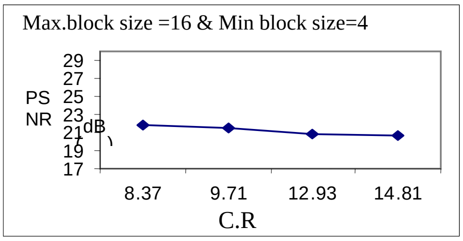
1- only few blocks (say 10 or 15) associated in matching process, (i.e. only few number satisfy the norm and higher order norm condition associated with encoding image). Also, GNM method show that these time savings in fact do provide a considerable acceleration of encoding and, moreover, allow an enlargement of the domain pool yielding improved image fidelity.

2- The domain blocks don't need to rotate and reflect to check the degree of matching because equations (6 -10) don't involve with the symmetry process.

10- Results:-

The performance of GNM algorithm in terms of compression and fidelity is the best possible it serves as a good test for our experiments in which we are aiming at evaluating the capabilities of few number of blocks (i.e. 10 – 25 blocks) in search in comparison to traditional complexity reduction attempts. In that respect GNM methods are excellent because it contains an advanced classification, this classification works as follows. A square range is subdivided into four quadtants (upper left, upper right, lower left, and lower right). In the quadtrants the average pixel intensities and the corresponding uniformity criteria are computed.

It can be notice from the result that C.R versus encoding parameter at different block size of quadtree partitioning, C.R increased linearly with increasing encoding parameters. Figs. (3) present that a high compression ratio obtained when the partitioning process is applied with maximum block size of 16×16 pixels and minimum block size of 8×8 pixels. Also in the same fig. (3)) show the relationship between C.R and PSNR at constant β factor. As it can be seen the C.R increased when PSNR is decreased, and the same behavior for different value of maximum and minimum blocks size. The reconstructed image at different maximum and minimum blocks size as the are illustrated in fig (4). The result shows missing block when maximum block size equal to 16 and minimum block size equal to 8, that obviously in fig (4) at constant β. These results comes due to the fast encoding time that turned to missing blocks of image.



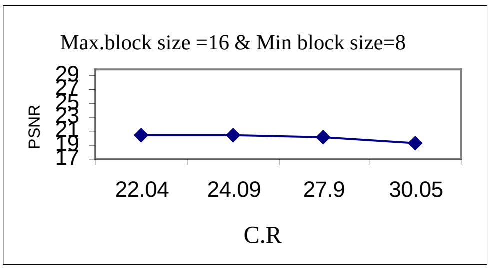


Fig (3): relationship between C.R and PSNR at different block sizes, the when, (β =0.7 and α 0.07,0.1,0.15,0.2)

Max. Block size=16
Min. Block size =4
α =0.07
C.R =7.37
SNR = 14.67
PSNR = 21.8
Time = 54.93 sec

Max. Block size=16
Min.Block size =8
α =0.07
C.R =21.04
SNR = 13.28
PSNR = 20.41
Time = 24.50 sec

Max Block size=16
Min.Block size =4
α =0.1
C.R= 8.37
SNR = 14.44
PSNR = 21.57
Time 49.05 sec

Max. Block size=16
Min Block size =8
α =0.1
C.R =22.09
SNR = 13.19
PSNR = 20.32
Time = 24.77 sec

Fig (4): The quality of reconstructed image of GNM method when β =0.7

11- Adaptive GNM method

GNM method successfully speeding the encoding time, the results shows this fact. But GNM method was failure in big blocks partitioning (i.e., maximum block size equal to 16 and minimum block size equal to 4 and 8). The reconstructing image (Lenna) has notable blocks missing, a mater which produced unclear one. This is evidence, that they're an extra speed of the encoding on the expense of the goodness of the image quality.

Adaptive method is an attempt to reach a super high quality of an image with the same block size but with some change resulting a good fruits in developing the image quality.

The quality of image is measured by criteria (PSNR). Thus, adaptive GNM method devoted to rise up the quality rate and making the reconstructed image clearest by changing the encoding parameters (β, α and block number), these changing was applied on Lenna image to show the differences between the qualities of the reconstructed images in both methods (GNM and Adaptive GNM methods).

12- Experimental results

Same image (i.e., lenna image). This method improved the quality of reconstruction image **G**NM method applied on lenna image of size (256 $\times$256) pixels, it can be notice that the reduction in the encoding time with acceptable quality of reconstruction image, but in big blocks is failure. Therefore, the adaptive GNM method applied on the, then it is improving by changing the values of number of blocks, β**,** and α testing measurements, choose the best values of Number of blocks, β and α and applied in the encoding process. To decide the best values can be applied on encoding process by using maximum block size equal to 16 x 16 pixels and minimum block size varied between 4 x 4 and 8 x 8 pixels.

The test begins with change number of blocks. Therefore, fig (5) shows high quality of reconstructed image ranging between 15 –25 blocks. Also Fig (6) show the best values of inclusion factor ranging between (0.1 to 1). Finally, Fig (7) utilized the best values of acceptance ratio ranging between (0.01 to o.1)

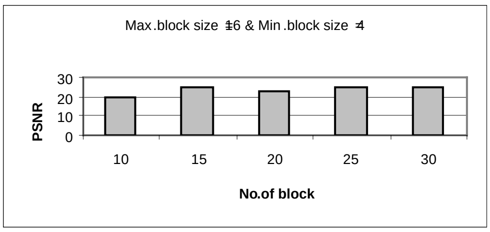


Fig (5): Relationship between No. Of blocks & PSNR

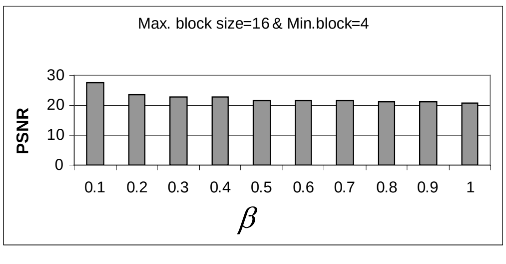
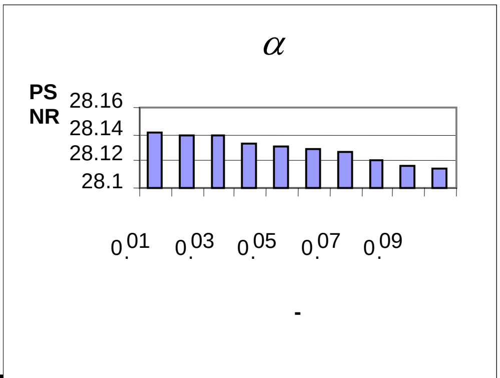


Fig (6): Relationship between β & PSNR values.

Finally, the value is changed and its test utilized in

Max. block size =16 & Min. block=4

Fig (7). It is clear that less values of α get better value of PSNR, and that is evidence more clearness of the image.



13- The best result

The aim behind adopting GNM method is to get high quality image. It is notable that with the changes about mentioned, and its application on the image, the PSNR starts raising up for each of these changes, that means the reconstruction image becomes a high quality, in our adaptive method selected the best results and applying them on the image, throughout GNM method and conducting the adopting method by

adopting three sequence changes. The changing applied on encoding parameters as below:

Step 1: **Changing the number** of blocks and testing the best value that can be raise ***PSNR*** values and then fixing the best values of (No. Block) as shows in fig (8).

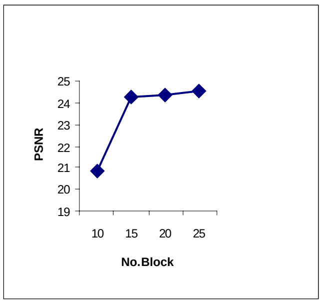
Step 2: **Changing** β **values**, and applying the best values that it can produce an extra PSNR values as are illustrated in fig (8).

Step 3: **Changing** α values. As they are illustrated in Fig (9).

The reconstruction image of GNM method

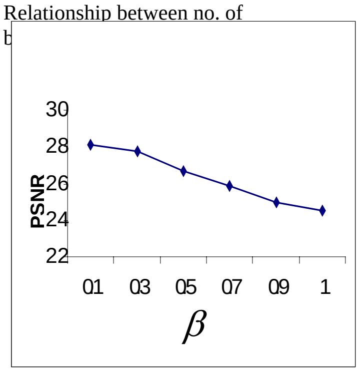
Step1

No. of blocks =25
β = 1
α = 0.1
C.R =14.50
SNR = 17.353
PSNR = 24.207



Relationship between no. of b

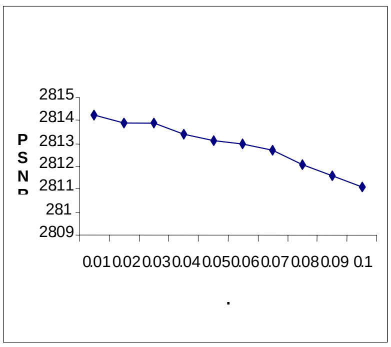
No.of blocks=25
β = 0.15
α = 0.1
C.R= 8.47
SNR = 21.08
PSNR = 28.11



Relationship between β & PSNR values.

Step3

No. Blocks =25
β = 0.15
α = 0.01
C.R=7.2
SNR = 21.11
PSNR = 28.14



Relationship between α & PSNR values

It can be seen the reconstruction image of adaptive GNM method has high quality and it approximated from the original image as shown in fig (9).

Reconstruction image of adaptive GNM method

The original image

Fig (9):The comparison between reconstructed image and by adaptive method original image.

Max block size =8
Min. block size = 4
β = 0.15
C.R =4.28
SNR = 21.77
PSNR = 28.80

Max. block size =16
Min.block size = 4
β = 0.15
C.R =7.02
SNR = 20.69
PSNR = 27.72

Max. block size = 16
Min.block size = 8
β = 0.15
C.R =18.87
SNR = 17.74
PSNR = 24.7

Max. block size = 8
Min. block size = 4
β = 0.4
C.R=6.00
SNR = 20.842
PSNR = 27.849

Max. block size = 16
Min.block size = 4
β = 0.4
C.R =7.96
SNR = 20.65
PSNR = 27.68

Max. block size = 16
Min. block size = 8
β = 0.4
C.R=21.67
SNR = 16.74
PSNR = 23.76

Fig (10): Illustrated high quality of reconstruction Lenna image of adaptive GNM method when (No. of blocks =20 and α =0.01)

14- Encoding Time

The encoding time is an important parameter in evaluating the efficiency of image compression method, thus its mentioned before, that the GNM method is created to acceleration the encoding process of quadtree method, which brings a best results for exceeding this way, but lower rate in the quality of the reconstructed image. In order to improve the quality of the reconstructed image and keep the encoding process fast, therefore, the adaptive method applied. Few differences of encoding time noticed with PSNR values, Table (2) is shown the above-mentioned relation, after applying the two methods on the computer kind of Pentium II with CPU 333.

Table (2): The results of GNM and adaptive GNM methods on Lenna image of size 256 $\times$256.

Encoding parameter				GNM		
β	α	Mix. Size	Min. Size	SNR (dB)	PSNR (dB)	Time (sec)
0.7	0.1	16	4	14.44	21.47	49.05
0.7	0.07	16	4	14.67	21.69	54.93
1	0.1	16	4	13.86	20.89	35.98
1	0.07	16	4	14.13	21.15	41.83
0.7	0.1	16	8	13.19	20.22	24.774
0.7	0.07	16	8	13.23	20.30	24.50
1	0.1	16	8	13.00	20.02	20.54
1	0.07	16	8	20.11	13.09	21.70

Encoding parameter				Adaptive GNM		
β	α	Mix. Size	Min. Size	SNR (dB)	PSNR (dB)	Time (sec)
0.4	0.1	16	4	19.931	27.05	60.32
0.4	0.02	16	4	20.79	27.92	74.94
0.2	0.02	16	4	20.93	28.063	80.32
0.2	0.1	16	4	20.86	27.99	77.06
0.4	0.1	16	8	16.68	23.81	31.47
0.4	0.02	16	8	16.93	24.06	33.77
0.2	0.02	16	8	16.950	24.07	32.81
0.2	0.1	16	8	20.86	27.99	35.27

Decoding process

The results in fig (10) illustrate the reconstruction image can reach nearly its attractor after 4 iterations of Lena image.

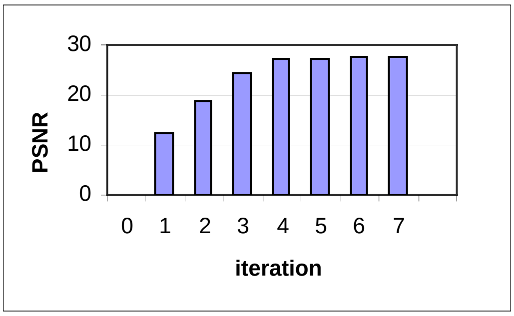


Fig (10) PSNR verses no. of iteration Lenna image with block size (4 - 8)

15- The contrastive results

According to the results, it is clear that the reconstructed image become high quality criteria when it gets a minimum block size, as in the flowing:

1-Highest number of blocks the reconstructed image has clearest it will be.

2-Less quality (i.e., less PSNR values) if the encoding parameter (α) values rising up.

3- The relationship between PSNR and C.R values are backward .

4- The relation between C.R value and the range block size is a forward.

16- Discussion and conclusion

Both GNM and adaptive GNM method showed the small range block size led to most improved in the quality of reconstructed image, therefore, minimum block of size 4 and 8 are used with maximum blocks of size 8 and 16.The results of encoding parameters (i.e., α , β, and No. Of blocks) discussed in this research.

Obviously, the adaptive GNM method improved the quality of reconstructed image with high number of blocks, but suffer long encoding time. The values of encoding parameters changing to less than values in GNM method, this effect in reconstruction image i.e., without missing blocks in the reconstructed image.

References

[1] W. Dson, John., " Compression In Video and Audio," Focal press, 1995.

[2] Y. Fisher, "Fractal image compression: Theory and Application", Springer – Verlaye, New Yourk, 1994.

[3] R. Hamzaoui, B. Ganz "Quadatree based variable rate oriented mean shape – gain vector quantization ", 2000.

[4] W.Klonowski " Signal and image analysis using chaos theory and fractal Geometry ", Lab. Of Biosignal analysis fundamental, polish Academy of Sciences, December / 2000.

[5] L.F.Anson, " Fractal image compression", Byte,October 1993.

[6] L. Thomas and F. Deravi, " Region – based Fractal image compression using heuristic search " IEEE Trans. Image Processing, June, 1995.

[7] D. Saupe, R. HamZaoui and H. Hartenstien " Fractal image compression An Introduction Over View",Lasagna, 1996.

[8] A.V. walle " Relating fractal image compression to transform methods ", M.Sc thesis, university Waterloo, Ontario, Canada, 1995.

[9] J.H. Al- A'mri " Fractal image compression ", Ph. D. thesis, Collage of science, university of Baghdad, January 2001.

[10] H.A. Al-Amawee " Adaptive new methods of Fractal image compression"M.S.C. thesis, Collage of science, AL-Mustansiriya University, October 2002