An Adaptive Algorithm for Improving the Fractal Image Compression (FIC)

Taha Mohammed Hasan
School of Computer Science and Technology, Harbin Institute of Technology (HIT)
Harbin 150001, China
taha_alzaidy@yahoo.com

Xiangqian Wu
School of Computer Science and Technology, Harbin Institute of Technology (HIT),
Harbin 150001, China
xqwu@hit.edu.cn

***Abstract*—In this paper an adaptive algorithm is proposed to reduce the long time that has been taken in the Fractal Image Compression (FIC) technique. This algorithm worked on reducing the number of matching operations between range and domain blocks by reducing both of the range and domain blocks needed in the matching process, for this purpose, two techniques have been proposed; the first one is called Range Exclusion (*RE*), in this technique variance factor is used to reduce the number of range blocks by excluding ranges of the homogenous or flat regions from the matching process; the second technique is called Reducing the Domain Image Size (*RDIZ*), it is responsible for the reduction of the domain by minimizing the Domain Image Size to 1/16th instead of 1/4th of the original image size used in the traditional FIC. This in turn will affect the encoding time, compression ratio and the reconstructed image quality. For getting best results, the two techniques are coupled in one algorithm; the new algorithm is called (*RD-RE*). The tested (256x256) gray images are partitioned into fixed (4x4) blocks and then compressed using visual C++ 6.0 code. The results show that *RE* technique is faster and gets more compression ratio than the traditional FIC and keeping a high reconstructed images quality while *RD-RE* is faster and it gets higher compression ratio than *RE* but with slight loss in the reconstructed image quality.**

***Index Terms*—Fractal, range block, variance, image compression, encoding time**

I. Introduction

Compression and decompression technology of digital image has become an important aspect in the storing and transferring of digital image in information society [1]. Recently fractal compression of digital images has attracted much attention [2]. M. Barnsley introduced the fundamental principle of fractal image compression in 1988 [3]. Fractal theories are totally different from the others. Fractal image compression is also called as fractal image encoding because compressed image is represented by contractive transforms and mathematical functions required for reconstruction of original image, instead of any data in pixel form [4]. One of the most important characteristics of fractal image coding is its unsymmetrical property of encoding and decoding processing. Coding time is rather long for domain codebook generation and domain-range matching operation, while decoding algorithm is relatively simple and fast. This weak aspect makes the fractal compression method not widely used as standard compression, although it has advantage of fast decompression as well as very high compression ratios [5].

Mathematically, FIC is based on the theory of Iterated Function System (IFS) and its performance relies on the presence of self-similarity between the regions of an image. Since most images possess high degree of self-similarity, fractal compression contributes an excellent tool for compressing them [6]. FIC consists of finding a set of transformations that produces a fractal image which approximates the original image [7].

In IFS coding scheme, many main processes must be done. First, range creating, the image must be partitioned into blocks (ranges) with non-overlapping [8]. Second, domain creating, the domain is created through taking the average of every four (2 x 2) adjacent elements in range to be one pixel in the domain, that means the size of domain image will be quarter size of the range. Fig. 1 shows an example of range and domain blocks size. Third, matching process, for every range block, a similar domain block is found using IFS mapping. The data of blocks of the compressed image are represented using the IFS mapping coefficients [9].

FIC suffers from the length of time spent in the compression process because there are a huge number of corresponding mapping operations, as all over the range compared with all domains for each of eight cases of (8 symmetries) [10].

In decoder process, the compressed image must be reconstructed from IFS-code, which is saved in codebook file. The reconstructed image starting from an arbitrary image and iterates these affine transformation parameters, according to the contractive mapping theory, the

© 2011 ACADEMY PUBLISHER
doi:10.4304/jmm.6.6.477-485

reconstructed image converges to attractor after 8 iterations [11]. Fig.2 shows the main process of the FIC model.

Large efforts have been undertaken to speed up the encoding process. Most of the proposed techniques attempt to accelerate the searching and are based on some kind of feature vector assigned to ranges and domains. A different route to increased speed can be chosen by less searching as opposed to faster searching [12]. In this work the proposed algorithm worked to speed-up the FIC by less searching through excluding homogenous ranges from the search process and also through minimizes the domain pool.

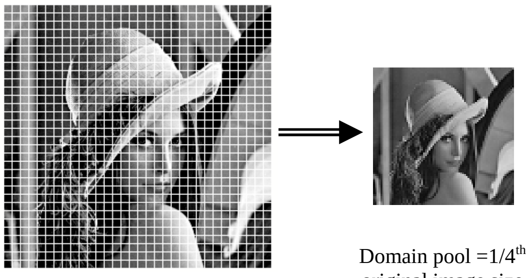


Figure 1. Construction domain block from the range block

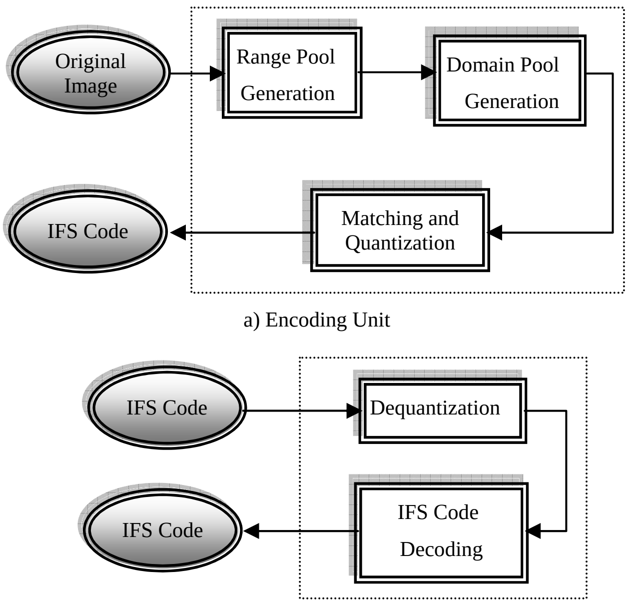


Figure 2. Fractal Image Compression System Model

II. Self Similarity

In mathematics, a self-similar object is exactly or approximately similar to a part of itself (i.e. the whole has the same shape as one or more of the parts). Many objects in the real world, such as coastlines, are statistically self-similar: parts of them show the same statistical properties at many scales. Self-similarity is a typical property of fractals. Scale invariance is an exact form of self-similarity where at any magnification there is a smaller piece of the object that is similar to the whole. For instance, a side of the Koch snowflake is both symmetrical and scale-invariant; it can be continually magnified 3x without changing shape [13].

Natural images are not exactly self-similar, natural images can be partially constructed from affine transformations of small parts of themselves. Self-Similarity indicates that small portions of the image resemble larger portions of the same image. The search for this resemblance forms the basis of fractal compression scheme [15]. Therefore the image must be partitioned into blocks to find self-similarity in other portion of the same image. This is intrinsic of fractal encoding techniques.

Fig. 3 shows that self–similar portion of the image can be found, there is a reflection of the hat in the mirror. The reflected portion can be obtained using an affine transformation of a small portion of her hat. Parts of her shoulder are almost identical [11].

Figure 3. An example shows the self-similarity in Lenna image

This paper constrains on the idea that the self-similarity in images depends on the image features because in FIC compression, the matching process search for self-similar portions in the image but in different scales, so if the partition of the image to be search has many details, it is hard to find a suitable matching part of image for it and vice versa.

III. Image Features

Images contain many regions of different details, some of these regions have significant detailed information and others have not (flat or homogenous) those regions where there is no gradation or that the gradation is not recognize with the naked eye, see Fig.4. The homogeneous regions are easy detectable in the images that are manufactured or composed by human, such as personal photos, which often contain areas of constant color, smooth as it is in the background. On the contrary, it is difficult to obtain a homogeneous 100% in natural images (landscape, etc.) such images may be reflected areas appear to the naked eye as a homogeneous or with a single color, but the arithmetic is not, the current research focuses on the exploitation of this principle , i.e., the exploitation of the regions that appear homogeneous to the naked eye and using them to speed up the compression process of the

© 2011 ACADEMY PUBLISHER

FIC to the point that do not affect the quality of the image recovered after compression.

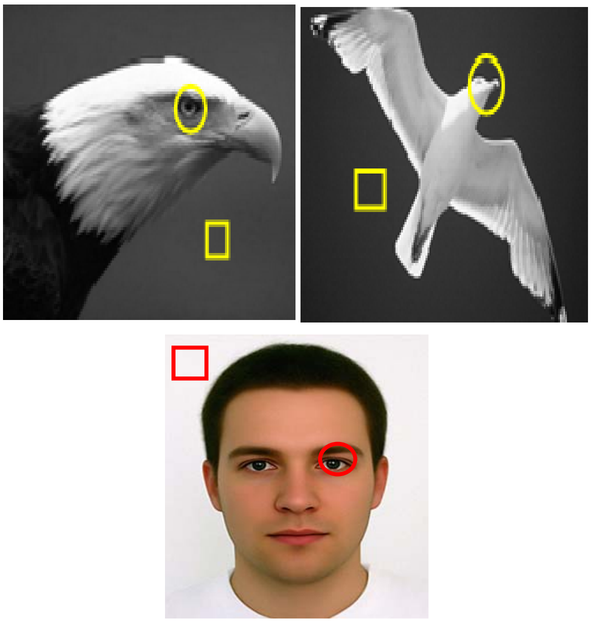


Figure 4. Area inside the box is a homogenous region, while the region inside the circle contains different combinations

IV. Proposed Method

In this work an adaptive algorithm for improving the basic fractal image coding process is proposed. The algorithm works to enhance the performance of FIC in terms of speeding up the encoding process and increasing the compression ratio while keeping a high reconstructed image quality. The proposed algorithm concentrates on reducing the number of matching operations between each rang block and domain pool needed to produce the FIC code by the following two techniques:

A. *Range Exclusion (RE) technique:*

This technique works to reducing the number of range blocks required in matching process by extracting a few numbers of features that characterize the range images. Then a number of ranges are excluding from matching process with the all 8 symmetries. The mean and standard variance features of all range blocks will be extracted. The mean value of range gives the measure of average gray level of range (*Mr*); standard variance (*Vr*) value of range defines the dispersion of its gray level from the mean. Variance has been used to check the range area whether it is homogenous region or contains details. After partitioning, the contents of each range will be checked before starting the search operation to decide if it is a homogenous region (flat) or not using the variance criteria, in homogenous region the value of variance is about zero while it is increase in the areas with more details. The flat region means that all pixels of this region have the same value or are close to each other. During the matching process, the homogeneous ranges will be excluded. So, the matching operation will be limited on the detailed regions only and this leads to reduce a huge amount of complex calculations, which results as fast coding process. In order to achieve the greatest benefit, the areas of homogeneous are controlled by using several values of the variance; these values were named Homogenous Permittivity (*HP*) which represents the amount of homogeneity allowed. If the variance of any part of the image (Range) is zero or is less than the *HP* value, this means that all pixels of that part is equal or so closely , then it will not enter in the search and matching operation, and this range will be encoding only by saving its mean value. So this process will speed-up FIC significantly also it will increase the compression ratio because each range excluded from the matching operation will require only one byte (8 bits) to store its mean value (*Mr*) as its fractal code instead of the 25 bits required to store its IFS code parameters (*s* = 7bits, *O*= 5bits, *x*= 5 bits, *y*= 5bits and *sym*= 3bits) [11].

B. *Reducing the Domain Image Size (RDIZ) technique:*

This technique works on minimizing the domain pool; as mentioned previously in traditional FIC, the encoding process is computationally intensive. A large number of sequential searches through a list of domains are carried out while trying to find a best match for a range block. A large domain pool will increase the number of comparisons that have to be made to find the best domain block and this where most of the computing time is used [13]. The proposed method significantly reduces the encoding time. The key of the idea is to reduce the number of domain blocks searched for each range block. This can be done by reducing the domain image size to $1/4^{th}$ the traditional domain image size so this process called Reducing the Domain Image Size (*RDIZ*). The domain pool will be created as $1/16^{th}$ of the original image size instead of the conventional domain size ($1/4^{th}$ of the original size) by down sampling every 4x4 (instead 2x2) pixels in the original image (using the average method) to one pixel in the reduced domain image as illustrated in Fig. 5,6 . In this case, for example if the original image was (256 x 256) pixels, the range block size was 4 and the domain jump step was 4, the number of the domain blocks needed in the match process for each range block will be reduced from 1024 (in the traditional FIC) to the 256 domain blocks. So, the computations needed in the encoding process will be reduced to be (4096 x 256) =1,048,576 instead of (4096 x 1024) = 4,194,304 and this will decrease the encoding time significantly. Also reducing the domain size will reduce the number of bits required to encode the original image because the number of bits needed for storing each of *x* and *y* coordinates of the best matched domains will be decreased. In our example when the domain pool of size (64 x 64) pixels , the maximum value for each *x* and *y* coordinates will be (60) by dividing it on the jump step (60/4 =15) then the encoder will need 4 bits to store each of *x* and *y* coordinates instead of 5 bits in the traditional FIC. Accordingly, this will lead to remarkable increase in the compression ratio.

© 2011 ACADEMY PUBLISHER

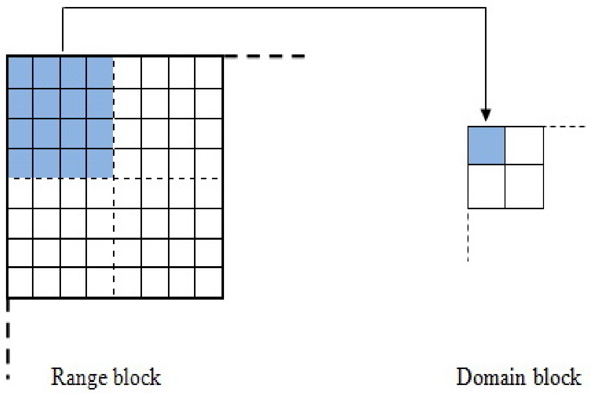
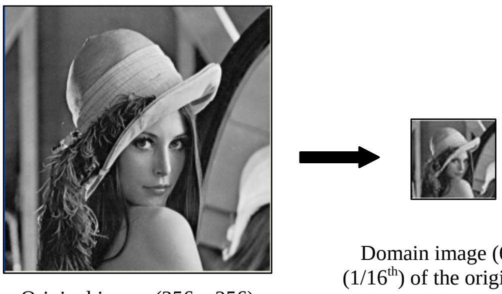


Figure 5. Down sampling method using the average of (4x4) pixels



Original image (256 x 256)

Domain image (64 x 64) (1/16th) of the original image size

Figure 6. Down sampling Lenna image using the average of (4x4) pixels

The following algorithm utilized steps required to perform the proposed method.

Algorithm

Input: Image, *HP*

Output: IFS code ((*x*, *y*, *s*, *o*, and *sym*) or *Mr*)

Step1: Load the image into buffer

Step2: Partitioning the image into fixed blocks size with non-overleap ($R_1 \ldots R_n$)

Step3: Generate the domain pool blocks ($D_1 \ldots D_m$) from the original image using 4x4 averaging method.

Step 4: Compute the mean for the current range block R_i according to (1):

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

Step 5: Compute its variance according to (2):

$$Vr = \frac{1}{X_{size} \times Y_{size}} \sum_{i=0}^{X_{size}-1} \sum_{j=0}^{Y_{size}-1} \left[X_{ij} - Mr\right]^2 \quad (2)$$

Step 6: If *Vr* <= *HP* then save range's mean (*Mr*) and excluding this range from the mapping operation (jump to step 10), else jump to step 7.

Step 7: Do the mapping operation by:

- Compute the scale *s* and offset *o* coefficients according to (3) and (4):

$$S = \frac{\left[n\left(\sum_{i=1}^{n} d_i r_i\right) - \left(\sum_{i=1}^{n} d_i\right)\left(\sum_{i=1}^{n} r_i\right)\right]}{\left[n\sum_{i=1}^{n} d_i^2 - \left(\sum_{i=1}^{n} d_i\right)^2\right]} \quad (3)$$

And

$$O = \frac{\left[\sum_{i=1}^{n} r_i \sum_{i=1}^{n} d_i^2 - \sum_{i=1}^{n} d_i \sum_{i=1}^{n} d_i r_i\right]}{\left[n\sum_{i=1}^{n} d_i^2 - \left(\sum_{i=1}^{n} d_i\right)^2\right]} \quad (4)$$

- Quantize the *s* and *o* values
- Compute the approximation error $E(R_i, D_i; s, o)$ according to (5):

$$E(R,D) = \frac{1}{n}\left[\sum_{i=1}^{n} r_i^2 + S\left(S\sum_{i=1}^{n} d_i^2 - 2\sum_{i=1}^{n} d_i r_i + 2O\sum_{i=1}^{n} d_i\right) + O\left(nO - 2\sum_{i=1}^{n} r_i\right)\right] \quad (5)$$

- Compare the computed error with the minimum registered error (E_{min}): if $E(R_i, D_i; s, o) > E_{min}$ then jump to step 8, else
- Replace the E_{min} and store the current IFS code (i.e. *x*, *y*, *s*, *o*, and symmetry).

Step8: Repeat the step (7) for all symmetry versions of tested domain blocks.

Step9: Repeat step (7) to (9) for all domain blocks listed in the domain pool.

Step10: Get the next range.

Step11: Repeat step (4) to (9) for all range blocks listed in the range pool.

In order to get high speed FIC with more compression ratio and keep as much as possible the reconstructed image quality, different values of the variance are adopted in the present research.

V. Results and Discussion

In order to show the effects of each of the *RE* and *RDIZ* on the traditional FIC, *RE* will be tested and its results will be discussed at first and then we will discuss the tests and results of coupling *RE* and *RDIZ* together in one algorithm.

A. Results and Discussion of RE

The program of FIC was applied to many images without check whether the image contains homogenous region or not (i.e. *HP*=0, the normal state), then the *RE* technique is applied to the same images with different *HP* values. Table I shows the effects of applying different values of *HP* on Peppers Image, it shows that when *HP*=2, the compression ratio been increased to reach (5.534) and the time required has been reduced from (34 seconds) to (18 seconds), when *HP* = 4 the compression ratio increased to (5.65) and the encoding time decreased to (14 seconds), when *HP* = 10, the compression ratio increased to (5.747) and the time required for the compression has decreased to (11 seconds), when the

© 2011 ACADEMY PUBLISHER

value of *HP* = 30 the compression ratio increased to (5.82) with the loss in image quality recovered, but still a high quality (30.74) and the encoding time been decreased to (9 seconds). Compared to time-spent when HP = 0 (34 seconds), we have obtained the proportion of speed about 73.5% of the compression process.

TABLE I.
THE EFFECTS OF USING DIFFERENT HP VALUES ON C.R, E.T, AND THE QUALITY OF PEPPER IMAGE

HP value	C.R.	E.T. (second)	PSNR (dB)
0	5.12	34	32.69
2	5.534	18	32.45
4	5.65	14	32.15
10	5.747	11	31.65
30	5.82	9	30.74

Fig. 7 shows the results of applying 0, 4 and 30 *HP* values on Pepper image. Fig. 8 and 9 show some results obtained when applying *HP* = 0, 6 and 20 on Bird and Lenna images respectively.

In the experiments, *HP* values ranged from 1 to 30 have been applied to all images. The data obtained from experiments, is plotted the relationship between the values of *HP* and the encoding time (see Fig. 10) as well as the relationship between *HP* and the quality of recovered image (see Fig. 11) and also the relationship between *HP* and the compression ratio (see Fig. 12).

The Original Image PEPPER image (File size: 64 kb)

The Reconstructed Image

(a) Traditional method (b) (c)

(*HP*=0) *HP*=4 *HP*=30

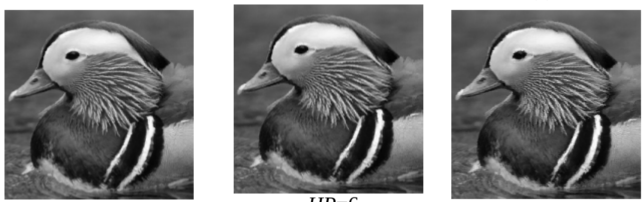
(*HP*=0)	*HP*=4	*HP*=30
Enco. time: 34 Sec.	Enco. time: 14 Sec.	Enco. time: 9 Sec.
PSNR 32.69 DB	PSNR 32.15 DB	PSNR 30.74 DB
C.R:5.12	C.R:5.65	C.R:5.82
File size: 12.5 kb	File size: 11.33 kb	File size: 10.99 kb

Fig. 7 Results of the impact of HP on the PEPPER image

The Original Image BIRD image (File size: 64 kb)

The Reconstructed Image

(a) Traditional method (b) (c)



(*HP*=0)	*HP*=6	*HP*=20
Enco. time: 34 Sec.	Enco. time: 13 Sec.	Enco. time: 9 Sec.
PSNR 30.30 DB	PSNR 29.92 DB	PSNR 28.69 DB
C.R:5.12	C.R:5.68	C.R:5.80
File size: 12.5 kb	File size: 11.27 kb	File size: 11.03 kb

Fig. 8 Results of the impact of *HP* on the BIRD image

The Original Image LENNA image (File size: 64 kb)

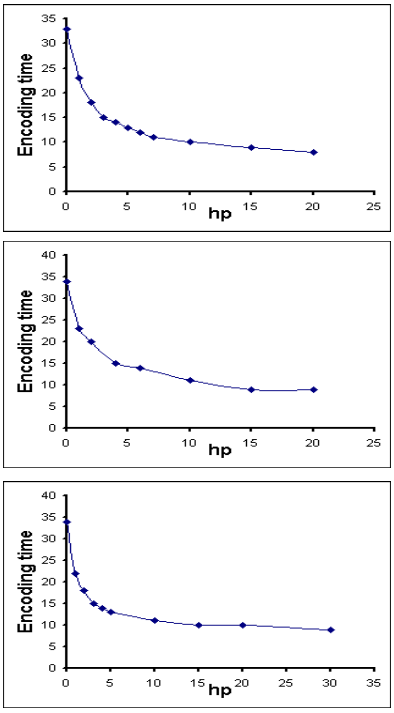
The Reconstructed Image

(a) Traditional method (b) (c)

(*HP*=0)	*HP*=6	*HP*=20
Enco. time: 35 Sec.	Enco. time: 12 Sec.	Enco. time: 9 Sec.
PSNR 32.22 DB	PSNR 31.51 DB	PSNR 30.64 DB
C.R:5.12	C.R:5.72	C.R:5.81
File size: 12.5 kb	File size: 11.18 kb	File size: 11.01 kb

Fig. 9 Results of the impact of *HP* on the LENNA image

© 2011 ACADEMY PUBLISHER

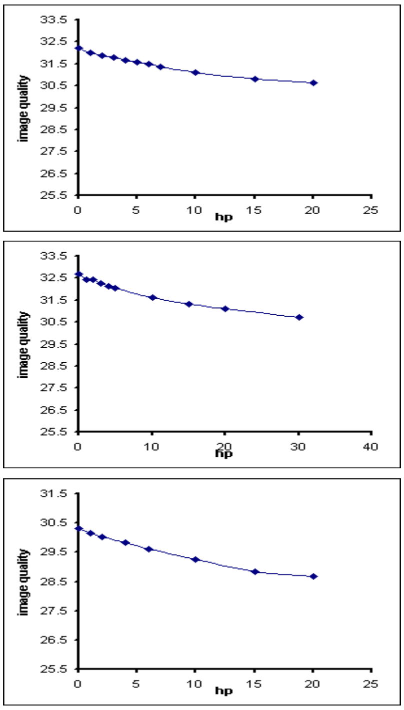


LEENA image

PEPPER image

BIRD image

Fig. 10 The effect of *HP* on the Encoding Time

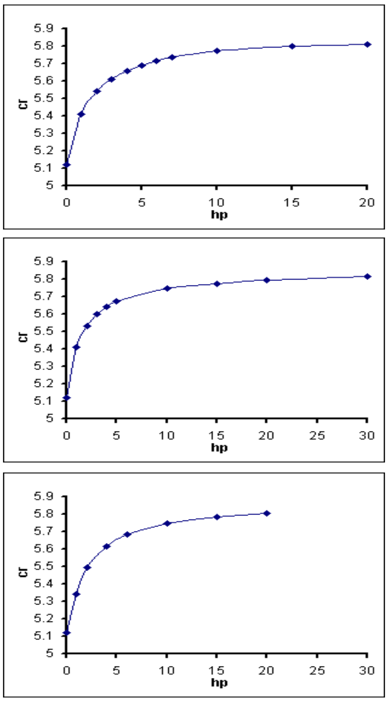


LEENA image

PEPPER image

BIRD image

Fig. 11 The effect of *HP* on the image quality



LEENA image

PEPPER image

BIRD image

Fig. 12 The effect of *HP* on the Compression Ratio (C.R.)

B.Results and Discussion of coupling RDIZ with RE

To achieve more speeding-up and more compression ratio, *RDIZ* is coupled with *RE* in one algorithm. The new algorithm is called *RD-RE*. *RD-RE* is applied to same tested images with same previous *HP* values. The results show that; there is a reasonable increase in compression ratio, the encoding time is reduced significantly and the reconstructed image quality is still acceptable. Table II shows the effects of applying *RD-RE* with different values of *HP* on Peppers Image.

TABLE II.
THE EFFECTS OF APPLYING *RD-RE* WITH DIFFERENT *HP* VALUES ON C.R, E.T, AND THE QUALITY OF PEPPER IMAGE

HP value	C.R.	E.T. (second)	PSNR (dB)
0	5.56	9	31.9
2	5.89	4.5	31.3
4	6.1	3.6	31.04
10	6.29	2.7	30.56
30	6.375	2.1	29.68

© 2011 ACADEMY PUBLISHER

From table II, it can be seen that when *HP*=0, that means there no effects of *RE*, so the results is affected only by the *RDIZ*, the C.R. is increased from 5.12 to 5.56 (increasing about 8.6%) and time is reduced from 34 to 9 seconds (decreasing about 73.5%) but with a little lose in PSNR (about 2.4%). The other values of the *HP* show the full effects of *RD-RE* on the FIC results. From comparing the results achieved from tables I and II, it obvious that using *HP* value = 30 in *RE* method gives an acceptable PSNR (about 30), increasing C.R. from 5.12 to 5.82 (about 14% increasing) and reduction in the E.T. from 34 to 9 seconds (about only 26% of the time required when *HP*= 0), and in *RD-RE* the same PSNR (about 30) can be achieved when *HP*=10 but the C.R. will be increased to 6.29 and this mean that this method get higher compression ratio (about 22.85% increasing) than the traditional FIC and also the encoding time will be significantly reduced to 2.7 seconds meaning that the time will be decreased about 93.82% of the time required in traditional FIC. Fig. 13-15 show the effects of applying *RD-RE* with different *HP* values on Pepper, Bird and Lenna images respectively.

Fig. 16-18 show comparison of the results achieved from applying traditional FIC, *RE* technique, *RDIZ* technique and the *RD-RE* algorithm on the tests images (the results of traditional FIC in the figures are the results with *HP*=0 in the *RE* columns and the results of *RDIZ* are with *HP*=0 in the *RD-RE* column). The figures showed that the *RDIZ* can reduce the encoding time to about 1/4th the time required in the traditional FIC and can get about 8.6% C.R. more than traditional but with a little loss in PSNR differs from one image to another but it is about 1.78% in average (for the test images). Also the figures showed that the effects of each of the *RE* and *RD-RE* with different *HP* values on the test images, the results are differ from one image to another depending on how many homogenous regions are there? but if we take the value of *HP*=10 as a compression value, we can see that the encoding time is reduced to about 62% and 91% in average and the C.R. can be increased to about 12.2% and 22% in average by applying *RE* and *RD-RE* respectively in comparison with the result achieved by applying the traditional FIC on the test images but the PSNR is reduced slightly to about 2.77% and 5.71% respectively.

The Original Image PEPPER image (File size: 64 kb)

The Reconstructed Image

(a) Traditional method

(HP=0)

Enco. time: 9 Sec.
PSNR 31.9 dB
C.R:5.56
File size: 11.51 kb

(b)

HP=4

Enco. time: 3.6 Sec.
PSNR 31.04 dB
C.R:6.1
File size: 10.49 kb

(c)

HP=30

Enco. time: 2.1 Sec.
PSNR 29.68 dB
C.R:6.375
File size: 10.03 kb

Fig. 13 Results of applying RE-RD with different HP on the PEPPER image

The Original Image BIRD image (File size: 64 kb)

The Reconstructed Image

(a) Traditional method

(*HP*=0)

Enco. time: 9 Sec.
PSNR 29.75 dB
C.R:5.56
File size: 11.51 kb

(b)

HP=6

Enco. time: 3.27 Sec.
PSNR 28.99 dB
C.R:6.13
File size: 10.44 kb

(c)

HP=20

Enco. time: 2.24 Sec.
PSNR 27.74 dB
C.R:6.34
File size: 10.09 kb

Fig. 14 Results of applying RE-RD with different *HP* on the Bird image

© 2011 ACADEMY PUBLISHER

The Original Image LENNA image (File size: 64 kb)

The Reconstructed Image

(a) Traditional method	(b)	(c)
		
(*HP*=0)	*HP*=6	*HP*=20
Enco. time: 9 Sec. PSNR 31.85 dB C.R:5.56 File size: 11.51 kb	Enco. time: 3.2 Sec. PSNR 30.53 dB C.R:6.19 File size: 10.33 kb	Enco. time: 2.21 Sec. PSNR 29.62 dB C.R:6.31 File size: 10.14 kb

Fig. 15 Results of applying *RE-RD* with different *HP* on the Lenna image

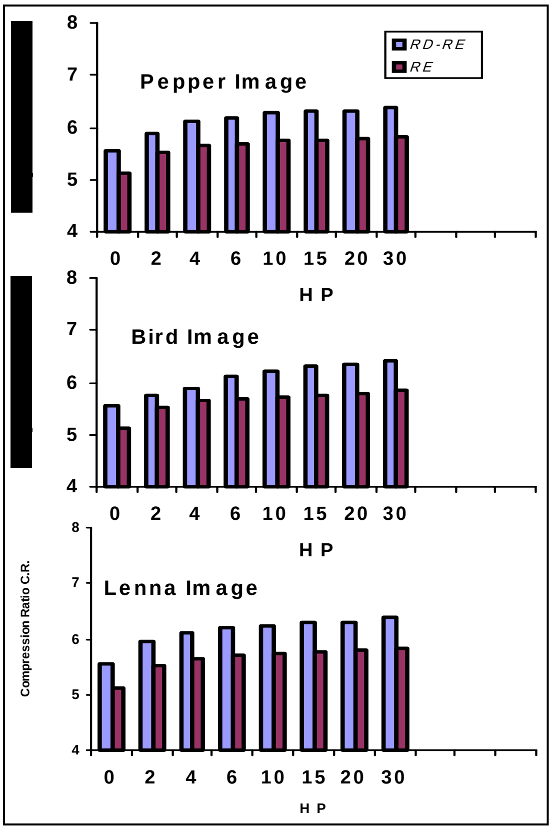


Fig. 17 The effects of applying *RE-RD* and *RD* on the C.R. with different *HP*

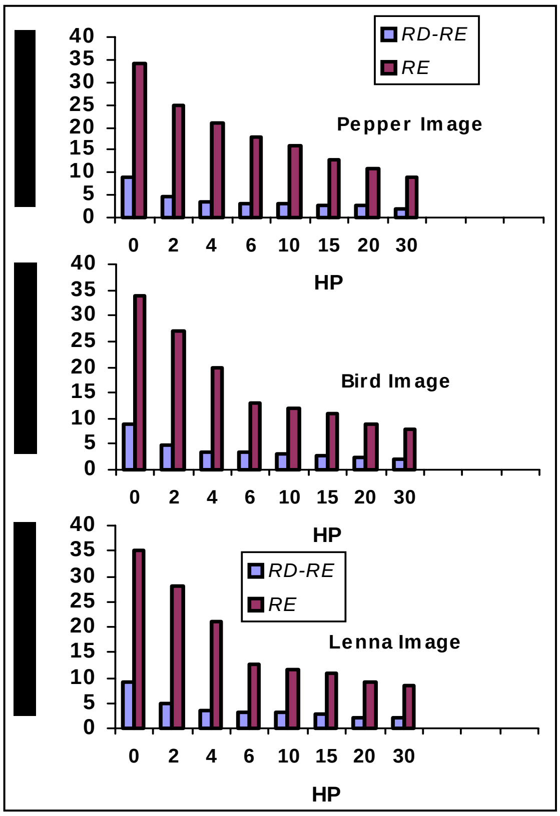


Fig. 16 The effects of applying *RE-RD* and *RD* on the E.T. with different *HP*

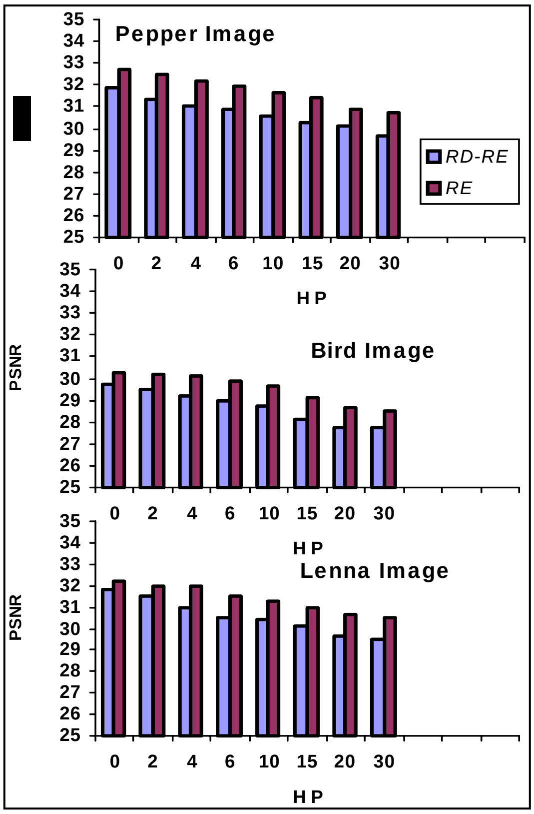


Fig. 18 The effects of applying *RE-RD* and *RD* on the PSNR with different *HP*

© 2011 ACADEMY PUBLISHER

VI. CONCLUSIONS

1. The effects of *HP* values on the results of FIC are different from one image to another depending on the image composition. Best results can be achieved in the images that have many homogeneous regions.
2. Experience has shown that if the value of *HP*=0, the results represent the normal situation for the FIC process.
3. High values of the *HP* lead to exclude more number of range areas from the process of searching and matching in the compression algorithm and this will provide high speed and more compression ratio but on the expense of the quality.
4. Experiments showed (in test images) that values of *HP* greater than 30 in *RE* technique and greater than 10 in *RD-RE* technique may lead to poor quality.
5. When *HP*=0 in *RD-RE* mean that the results are affected only by *RDIZ* technique, and the results showed that *RDIZ* lead to reduce E.T. significantly and increase the C.R. remarkably but with a slightly loos in PSNR.
6. There is an inverse relationship between *HP* and the time spent in the encryption process.
7. There is an inverse relationship between *HP* and the quality of recovered image.
8. The relationship direct correlation between *HP* and the compression ratio.
9. *HP*=10 is the best value in *RD-RE* algorithm because it made a best tradeoff among E.T., C.R. and PSNR (for the test images).

ACKNOWLEDGMENT

This work was supported by the Natural Science Foundation of China (NSFC) (No. 60873140, 61073125 and 61071179), the Program for New Century Excellent Talents in University (No. NCET-08-0155 and NCET-08-0156), and the Fok Ying Tong Education Foundation (No. 122035).

REFERENCES

[1] Y.Chakrapani and K.Soundera Rajan, "Hybrid Genetic-Simulated Annealing Approach for Fractal Image Compression," *International Journal of Information and Mathematical Sciences* vol.4, no.4, pp. 308-313, 2008.

[2] Gonzalez R. C., and Wintz, P., *Digital Image Processing*, 2nd ed., Addison-Wesley publication company, 1987.

[3] M. Barnsley, *Fractals Everywhere*. New York: Academic, 1988.

[4] V. Chaurasia and A. Somkuwar, "Speed up Technique for Fractal Image Compression," *International Conference on Digital Image Processing icdip*, pp.319-323, 2009, "doi. 10.1109/ICDIP.2009.66".

[5] Y. Fisher, *Fractal Image Compression Theory and Application*, Springier Verlage, New York, 1994.

[6] Ghada K., "Adaptive Fractal Image Compression", *M.Sc. thesis*, National computer Center /Higher Education Institute of computer and Informatics, 2001.

[7] Sumathi Poobal, and G. Ravindran, "Arriving at an Optimum Value of Tolerance Factor for Compressing Medical Images," *World Academy of Science, Engineering and Technology*, vol. 24, pp. 169-173, 2006.

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

[9] Auday A., "Fractal Image Compression with Fasting Approaches", *M.Sc. thesis*, College of Science, Saddam University, 2003.

[10] Jamila H. S., "Fractal Image compression", *Ph.D. thesis*, College of Science, University of Baghdad, January, 2001.

[11] S. Abdul-Khalik, "Fractal Image Compression Using Shape Structure," *M.Sc. thesis*, College of Science, Al-Mustansiriya University, Iraq, 2005.

[12] Dietmar Saupe, "Accelerating Fractal Image Compression by Multi-Dimensional Nearest Neighbor Search," *CC'95 Data Compression Conference, J. A. Storer, M. Cohn (eds.), IEEE Computer Society Press*, March 1995.

[13] http://en.wikipedia.org/wiki/Self-similarity, 1/6/2011.

[14] Mario Polvere and Michele Nappi, "Speed-Up In Fractal Image Coding Comparison of Methods," *IEEE TRANSACTIONS ON IMAGE PROCESSING,* VOL. 9, NO. 6, JUNE 2000.

[15] E. Vrcasy and L. Colin, , "Image Compression Using Fractals," *IBM Journal of Research and Development*, vol. 65, no.19, pp. 121-134 1995.

Taha M. Hasan received his BSc, MSc in computer science from Mansour University College and The University of Mustansiriyah, Baghdad, Iraq in1992 and 2006 respectively. He is currently pursuing the Ph.D. degree at the Harbin Institute of Technology (HIT), Harbin, China. His research interests is the image processing.

Xiangqian Wu received his BSc, MSc and PhD in computer science from Harbin Institute of Technology (HIT), Harbin, China in 1997, 1999 and 2004, respectively. Currently he is a professor in HIT. His research interests include image processing, pattern recognition and biometrics, etc.

© 2011 ACADEMY PUBLISHER