A Proposed Block-Coding Technique of an Image based on Fractal Compression

Shimal Das
Assistant professor
Tripura Institute of Technology,
Narsingarh, Tripura, India,

Dibyendu Ghoshal, PhD
Associate professor
National Institute of Technology
Agartala, India

ABSTRACT
Now a day the fractal image compression technique models a natural image using a contractive mapping called fractal mapping in the image space. Due to reducing the search complexity of matching between range block and domain block in fractal image compression is one of the most active research areas lately. There are two main characteristics of this approach are (a) It relies on the assumption that image redundancy can be efficiently captured and exploited through piecewise serf-transformability on a block-wise basis, and (b) It approximates an original image by a fractal image, obtained from a finite number of iterations of an image transformation called a fractal code. This paper proposed to this approach as Fractal Block Coding. For such an application, the general problem statement is the following. For any given original discrete image specified by an array of pixels, how can a computer construct a fractal image, the coded image-which is both visually close to the original one, and has a digital representation which requires fewer bits than the original image? The proposed coding scheme carried out an approach to image coding based on a fractal theory contractive transformations defined piecewise. In experimental results show that compared with Jacquin coding scheme and our proposed coding scheme achieves an average of 89% reduction in encoding time and improves the efficiency of search. Simultaneously the compression ratio and quality of decoded images are guaranteed to the same as Jacquin coding scheme for the same images.

General Terms
Digital image processing, Pattern Recognition

Keywords
Fractals Coding, Image Encoding, IFS, LIFS, Fractal Compression

1. INTRODUCTION
The concept of fractal was introduced by Mandelbrot [1] as an alternative to the traditional Euclidean geometry mainly for dealing with shapes generated by nature. Currently, the interest of applying this theory has been steadily growing. Now a day in computer graphics and image processing has been to use iterated function system (IFS) to generate and describe both man-made fractal- like structures and natural images. Barnsley et al. were the first to present the concept of fractal image compression using IFS [1]. Deterministic Fractals have the intrinsic property of having extremely high visual complexity while being very low in information content, as they can be described and generated by simple recursive deterministic algorithms [2]. Those are mathematical objects with a high degree of redundancy in the sense that they are recursively made of transformed copies of either themselves or parts of themselves. These objects, which arise from the mathematical theory of Iterated Sequences, were first labelled mathematical "curiosities" or "monsters" by mathematicians in the beginning of the 20th century who lacked the tools to properly analyze and understand those [3], [4], [5]. After falling into nearly complete oblivion for a while, they were "rediscovered" by the mathematical research community in the 1970's, thanks to the pioneering work of Mandelbrot who also coined their name [1]. It is indisputable that this rediscovery was also triggered by the availability of computers and automatic graphic tools which made it possible for the first time to render and visualize them as complex, beautiful, often realistic-looking objects or scenes [1], [6]. In the past twenty years, fractals have also been part of a set of tools in a variety of fields in Physics, where they are closely related to Chaos Theory [7], [8]. They have recently emerged in various fields of Electrical Engineering, as attested by the contents of this Special Section. Fractal-based techniques have been applied in several areas of digital image processing, such as image segmentation [9], image analysis [10], [11], image synthesis and computer graphics [14], [7], [18], [19], [20], [21], [22], and texture coding [23], [24]. Barnsley was the first to propose the notion of Fractal Image Compression, by which real-life objects or images would be modelled by deterministic fractal objects-attractors of sets of two-dimensional affine transformations [12]-[14]. The mathematical theories of Iterated Function Systems (IFS) and Recurrent Iterated Function Systems [15], [16], along with the important Collage Theorem, constitute the broad foundations of fractal image compression. However, these theories alone do not provide any constructive procedure for the "encoding" of a gray-tone image-as understood by the image coding community- i.e., in an automated way. That particular task can be performed by defining piecewise affine contractive transformations which make use of only the partial self-transformability of images. The rest of the paper arranged thus: section 2 presents Theoretical Foundations, section 3 Presents Overview of Fractal Image Coding Method Overview of Edge Detection Method Based on Fractal Image Compression, section 4 Presents Overview Of Encoding Images, section 5 presents Our Proposed Algorithm, section 6 Experimental Result And Discursion, section 7 presents Conclusion and last section 8 presents references.

2. THEORETICAL FOUNDATIONS

2.1 Self-affine and self-similar transformation

Fractal image compression algorithm is based on the fractal theory of self-similar and self-affine transformations [25].

Definition 1. A self-affine transformation $w: R^n \rightarrow R^n$ is a transformation of the form $w(x) = T(x) + b$, where T is a linear transformation on R^n and $b \in R^n$ is a vector.

Definition 2. A mapping $w: D \rightarrow D;\ D \subseteq R^n$ is called a contraction on D if there is a real number $c \in (0,1)$, such that $d(w(x), w(y)) \leq cd(x, y)$ for $x, y \in D$ and for a metric d on R^n. The real number c is called the contractility of w.

Definition 3. If $d(w(x), w(y)) = cd(x, y)$, then w is called a similarity. A family $\{w_1, \ldots, w_m\}$ of contractions is known as a local iterated function system (LIFS). If there is a subset $F \subseteq D$ such that for a LIFS $\{w_1, \ldots, w_m\}$,

$$F = \bigcup_{i=1}^{m} w_i(F) \qquad (1)$$

then F is said to be invariant for that LIFS. If F is invariant under a collection of similarities, F is known as a self-similar set.

Let S denote the class of all non-empty compact subsets of D. The δ-parallel body of $A \in S$ is the set of points within distance δ of A, i.e.

$$A_\delta = \{ x \in D: \text{there exists } \alpha \in A \text{ such that } |x-a| \leq \delta \}. \quad (2)$$

Let us define the distance $d(A,B)$ between two sets A, B to be

$$d(A,B) = \inf\{ \delta: A \subset B_\delta \text{ and } B \subset A_\delta \} \qquad (3)$$

The distance function is known as the Hausdorff metric on S (other distance functions can also be used).

Given a LIFS $\{ w_1, \ldots, w_m \}$, there exists an unique compact invariant set F, such that

$$F = \bigcup_{i=1}^{m} w_i(F) \qquad (4)$$

This F is known as the attractor of the system. If E is a compact non-empty subset such that $w_i(E)$ and $W_i(E) \subset E$ and

$$W(E) = \bigcup_{i=1}^{m} w_i(F) \qquad (5)$$

The proposed method define the k-th iteration of w, $w^k(E)$, to be

$$w^0(E) = E,\ w^k(E) = w(w^{k-1}(E)) \quad \ldots (6)$$

For $k \geq 1$, then got $F = \bigcap_{i=1}^{\infty} w^k(E)$ (7)

The sequence of iteration $w^k(E)$ converges to the attractor of the system for any set E. This means that it may carry out a family of contractions that approximate complex images and, using the family of contractions, the images can be stored and transmitted in a very efficient way. Another present method is a LIFS; it is straightforward to obtain the encoded image. If any one wants to encode an arbitrary image in this way, they will have to find a family of contractions so that its attractor is an approximation to the given image. Barnsley's Collage Theorem states how well the attractor of a LIFS can approximate of any given images.

2.2 Collage theorem

Let $\{ w_1, .., w_m \}$ be contractions on R^n so that for any $x,y \in R^n$ and any i, $|w_i(x) - w_i(y)| \leq c$ ……………(8)

Where $c \in (0, 1)$ is a constant. Let $E \subset R^n$ be any non-empty compact set. Then

$$d(E,F) \leq 1/(1-c)\ d(E, \bigcup_{i=1}^{m} w_i(E)) \qquad (9)$$

where F is the invariant set for the w_i, and d is the Hausdorff metric [9].

As a consequence of this theorem, any subset of R^n can be approximated within an arbitrarily tolerance by a self-similar set, i.e., given $\delta > 0$, there exist contracting similarities $\{ w_1, \ldots, w_m \}$ with invariant set F satisfying $d(E,F) < \delta$. Therefore, the problem of finding a LIFS $\{ w_1, \ldots, w_m \}$ whose attractor F is arbitrary close to a given image I is equivalent to minimize the distance

$$d(I, \bigcup_{i=1}^{m} w_i(I))$$

2.3. Contractive transformations

A transformation w is said to be contractive if for any two points $P1$, $P2$, the distance

$$d(w(P1),w(P2)) < sd(P1,P2) \qquad (10)$$

for some $s < 1$, where d = distance. This formula says the application of a contractive map always brings points closer together (by some factor less than 1).

3. THE CONTRACTIVE MAPPING FIXED POINT THEOREM

The fractal image coding makes good uses of image self-similarity in space by ablating image geometric redundant. This coding process is quite complicated but decoding process is very simple, which makes use of potentials in high compression ratio. The theory of fractal image coding is based on iterated function system, attractor theorem, and collage theorem. Regard original compressible image as attractor, how to get LIFS parameters is main problem of fractal coding [26] . For conventional fractal image coding technique, the image is partitioned into a number of non-overlapping blocks called range blocks. In spite of storing the information of the range blocks as such, only the parameters defining the affine transformations are stored. All these parameters are obtained by mapping each range block to a closely resembling block called the domain block on which the transformations are applied. Domain blocks are selected from the same image, and they can overlap. Fundamentally the size of the domain block will be twice the size of the range block. Any grayscale image can be coded by mapping the domain block D to the range block R with the contractive affine transformation, fig 1(a) shows the fractal image coding procedure, fig (b) shows the domain and Range block of "Lena" image and fig 1(c) shows the mapping of intensity value in fractal transform.

I. A given image I is divided into non-overlapping M range blocks of size $B \times B$ and into arbitrarily located N domain blocks of size $2B \times 2B$. The range blocks are numbered from 1 to M, and represented by $R_i\ (1 \leq i \leq M)$. Similarly, the domain blocks are from 1 to N, and represented by $D_j\ (1 \leq j \leq N)$.

II. For each range block R_i, the best matched domain block $D_k\ (1 \leq k \leq N)$ and an appropriate contractive affine transformation τ_{ik} which satisfy the following equation are found.

$$D(R_i, \tau_{ik}(D_k)) = \min d(R_i, \tau_{ik}(D_j)) \qquad (11)$$

$$d(R_i, \tau_{ij}(D_{ij})) = \frac{\sum\limits_{0 \le N \le B} (A_{i,k} - \bar{A}_{i,k})^2}{B \times B}, \quad (o \le l, k \le B\text{-}1) \qquad (12)$$

Where τ_{ij} is an contractive affine transformation from the domain block *Dj* to the range block R_i; the distortion measure $d(R_i, \tau_{ij}(D_j))$ is the mean square error (MSE) between the range block R_i and the contractive domain block $\tau_{ij}(D_j)$. The contractive affine transformation τ_{ij} is composed of two mappings ϕ_j and ϕ_{ij} as follows: $\tau_{ij} = \phi_{ij} \circ \phi_j$ ……………(13)

The former mapping ϕ_j ***is*** the transformation of domain-block size to the same size as range block's. This transformation is achieved as follows: The domain block D_j is divided into non-overlapping unit blocks of size 2 × 2; And each pixel value of the transformed block $\phi_j(D_j)$ is an average value of four pixels in each unit block in D_j. The latter mapping ϕ_{ij} consists of two steps: The first step transforms the block $\phi_j(D_j)$ a way of the eight transformations: rotation around center of the block $\phi_j(D_j)$, through 0°, +90°, +180°, and +270°, and each rotation after orthogonal reflection about mid-vertical axis of the block $\phi_j(D_j)$. Those eight transformations are called *isometries.* The second step is the transformation (p_{ij}) of pixel values of a block obtained by the first step. This transformation p_{ij} is defined as

$$p_{ij}(v) = \alpha_{ij} v + g_{ij} \ \ldots\ldots\ldots\ldots \ (14)$$

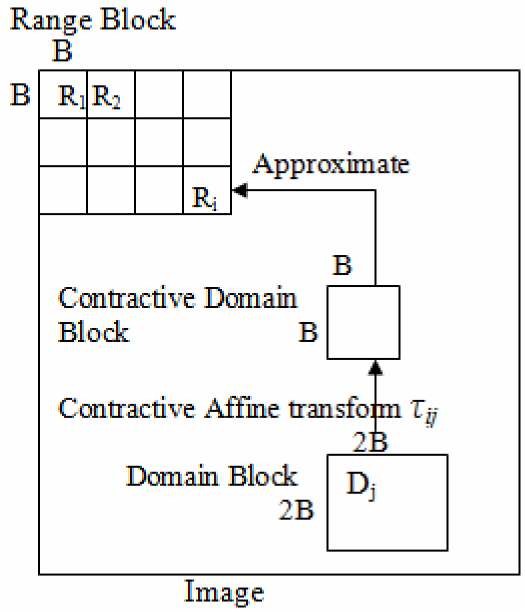

Fig. 1(a) Fractal image coding

Where *v* is a pixel value of the block obtained by the first step, and the parameters α_{ij} and g_{ij} are computed by the least square analysis of pixel values of the range block R_i and the block obtained by the first step. These parameters α_{ij} *and* g_{ij}, a scaling coefficient and an offset and the IFS Parameters are as (a) Parameters to indicate a location of the best matched domain block; (b) A parameter to indicate an *isometry* on the best matched domain block; (c) A scaling coefficient and an offset. Proposed method quantizes these LIFS parameter.

Fig.1 (b) Domain and Range block of "Lena"

4. OVERVIEW OF ENCODING IMAGES

From the above mentioned theorem it may carried out that transformation *W* will have a unique fixed point in the space of all images. These transformations repeatedly apply to the images until it will converge to a fixed image.

Let any given an image *f* that we wish to encode.

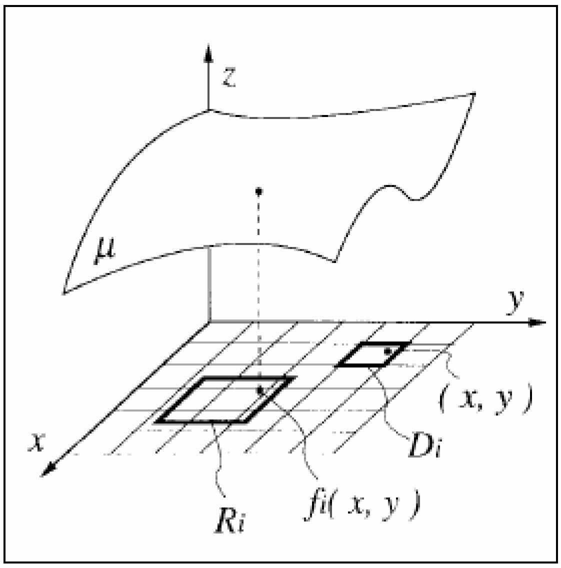

Fig. 1 (c) Point *(x, y)* ∈ D_i is mapped to f_i *(x, y)* ∈ R_i and the intensity value at f_i *(x, y)*: $\mu(f_i(x, y))$ is sampled in the fractal transform

The meaning is want to find a collection of transformations w_1, w_2, ...,w_N and *f* to be the fixed point of the map *W* (from fixed Point Theorem) and another way to partition *f* into pieces to which apply the transformations w_i, and get back the original image *f*. A typical image of a face, does not contain the type of self-similarity like the fern in Figure 2. The image does contain other type of self-similarity. Figure 3 shows regions of Lena identical, and a portion of the reflection of the hat in the mirror is similar to the original. These distinctions form the kind of self-similarity shown in Figure 2; rather than having the image be formed by whole copies of the original (under appropriate affine transformations), here the image will be formed by copies of properly transformed parts of the original. These types of transformed parts do not fit together, in general, to form an exact copy of the original image, and so it must allow some error in this representation of an image as a set of transformations.

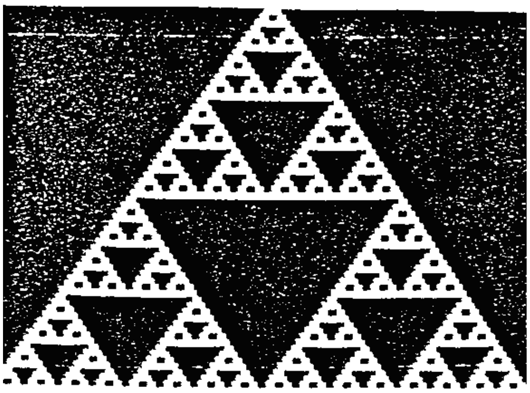

Figure 2: Fractal Fern

Fig. 3. Self similar portions of Lena

5. OUR PROPOSED ALGORITHM

5.1 Encoding

From the following example suggests how the Fractal Encoding can be done. Now it may deal with a 128×128 image in which each pixel can be one of 256 levels of gray, called this picture Range Image. Then it reduce by averaging (down sampling and lowpass-filtering) the original image to 64×64, called this new image Domain Image.Then partitioned both images into blocks 4 x 4 pixels shown in Figure 4 and performed the following affine transformation to each block as follows

$(D_{ij}) = \alpha D_{ij} + t_o$ ……………….. (15)

Where α = [0, 1], $\alpha \in \Re$ and $t_o \in$ [-255, 255], $t_o \in Z$

So in this case trying to find linear transformations of Domain Block to arrive to the best approximation of a given Range Block. For each Domain Block is transformed and then compared to each Range Block $R_{k,l}$. Then the exact transformation on each domain block, i.e. the determination of α and t_o is found minimizing

$$\text{Min} \sum (R_{k,l})_{m,n} - (\Gamma (D_{i,j}))_{m,n} \qquad (16)$$

With respect to α and t_o

$$\alpha = \frac{N_s^2 \sum_{m,n} (D_{i,j})_{m,n} (R_{k,l})_{m,n} - (\sum_{m,n} (D_{i,j})_{m,n})(\sum_{m,n} (R_{k,l})_{m,n})}{N_s^2 \sum_{m,n} ((D_{i,j})_{m,n})^2 - (\sum_{m,n} (D_{i,j})_{m,n})^2} \qquad (17)$$

$$t_0 = \frac{(\sum_{m,n} (D_{i,j})_{m,n})^2 - (\sum_{m,n} (R_{i,j})_{m,n})^2}{N_s^2 \sum_{m,n} ((D_{i,j})_{m,n})^2 - (\sum_{m,n} (D_{i,j})_{m,n})^2} \qquad (18)$$

Here m, n, N_s = 2 or 4 (blocks size). For every transformed domain block $\Gamma(Di,j)$ is compared to each range block $R_{k,l}$ in order to find the closest domain block to each range block using the following distortion measure.

$$d_{l_2} (\Gamma(D_{i,j}), R_{k,l}) = \sum_{m,n} ((\Gamma(D_{i,j})-(R_{k,l})_{m,n})^2 \qquad (19)$$

Every distortion is stored and the minimum is chosen. Then the transformed domain block which is found to be the best approximation for the current range block is assigned to that range block, i.e. the coordinates of the domain block along with its α and t_o are saved into the file describing the transformation. For this why is called the Fractal Code Book.

$$\Gamma(D_{i,j})_{best} \Rightarrow R_{k,l} \qquad (20)$$

5.2 Decoding

In decoding process the reconstruction for the original image consists on the applications of the transformations describe in the fractal code book iteratively to some initial image $\Omega init$, until the encoded image is retrieved back. The whole image transformation can be described as follows:

$$\Omega n = \eta(\Omega n\text{-}1) \qquad (21)$$

Where $\Omega 1 = \eta(\Omega init)$, $\Omega 2 = \eta(\Omega 1)$, $\Omega 3 = \eta(\Omega 2)$, ….. = ...

Here η can be expressed as two distinct transformations:

$$\eta = \Gamma(\Omega)\Psi(\Omega) \qquad (22)$$

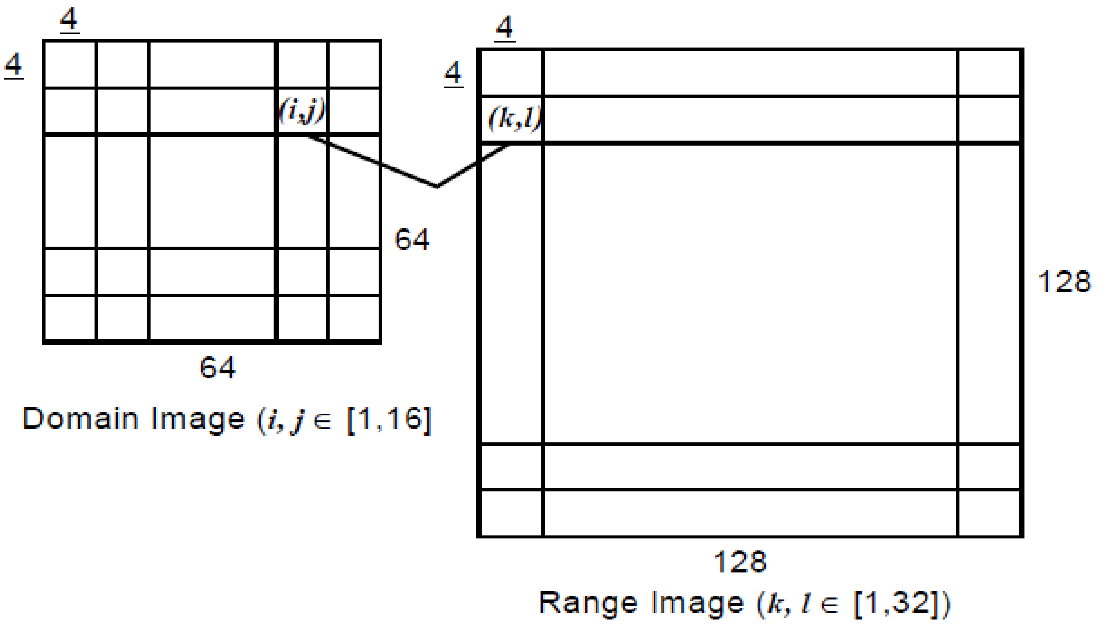

Fig. 4. Partition of Range and Domain

Again more $\Gamma(\Omega)$ represents the down sampling and low pass filtering of an image W to create a domain image e.g. for reducing a 128x128 image to a 64x64 image as it may describe previously. The symbol $\Psi(\Omega)$ represents the ensemble of the transformations defined by proposed mappings from the domain blocks in the domain image to the range blocks in the range image as recorded in the fractal. So Ωn will converge to a good approximation of $\Omega orig$ in less than 5 iterations

6. EXPERIMENTAL RESULT & DISCURSIONS

From the experiment the Lena image (128x128) are decoded the using the set-up described in Figure 5 and 6. This is performed using the 2x2 and 4x4 block size respectively and several different reference images. Table-1 shows the proposed method comparison with Jacquin method decoded images and figure-7 (a) shows the original Lena image, figure-7 (b) shows the Jacquin coding image and figure-7 (c) shows the our proposed method.

Table-1 shows the proposed coding scheme comparison with Jacquin coding scheme

Tested Result	**Jacquin codind scheme**	**Method 1 (Our proposed coding scheme)**	**Method 2 (Our proposed coding scheme)**
Block Size	4x4	2x2	4x4
No Iterations	4	4	4
Size of the code book	18242 bytes	16238 bytes	6013 bytes
Time to encode	10.65 s	09 s	8.43 s
Time to decode	43 s	41s	36s
Peak Error	113	109	99
SNR	36.6 dB	24 dB	22 dB
Bit Rate	10.13 byte/pixel	9.0 byte/pixel	0.351 byte/pixel
Reference Image	square	square	Square

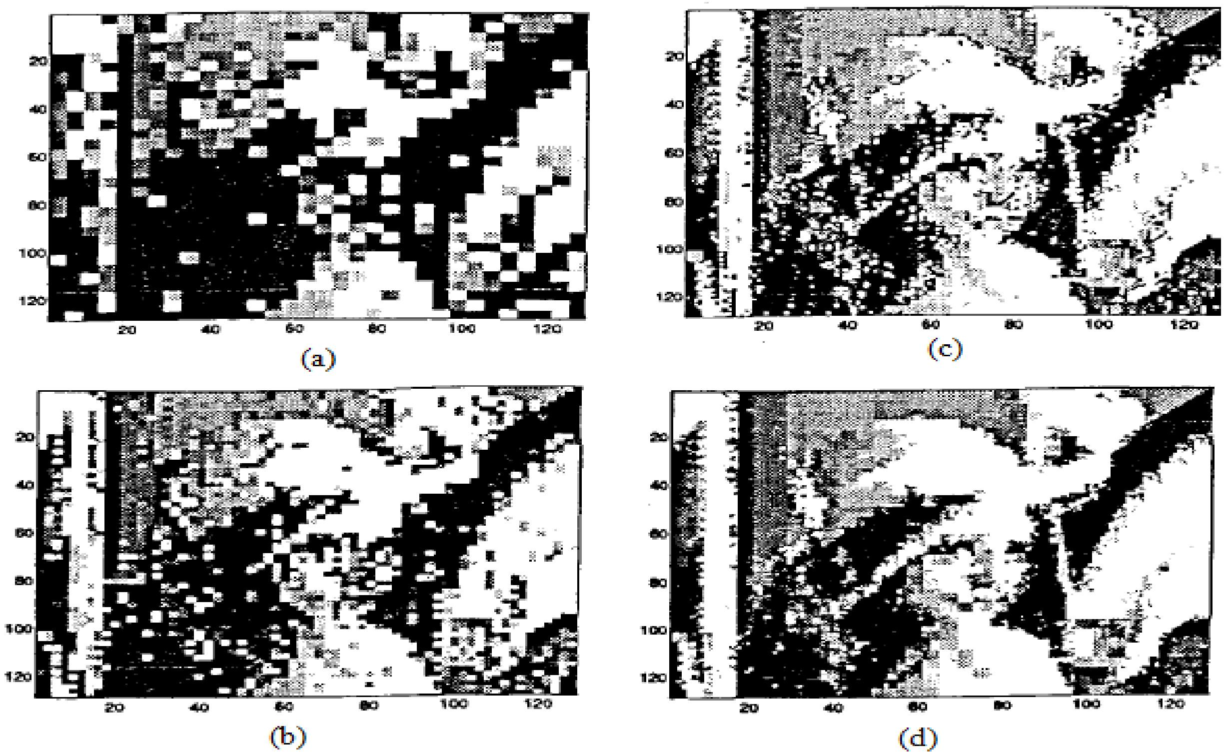

Fig. 5. Decoding iterations with 2x2 decoding using fractal coding

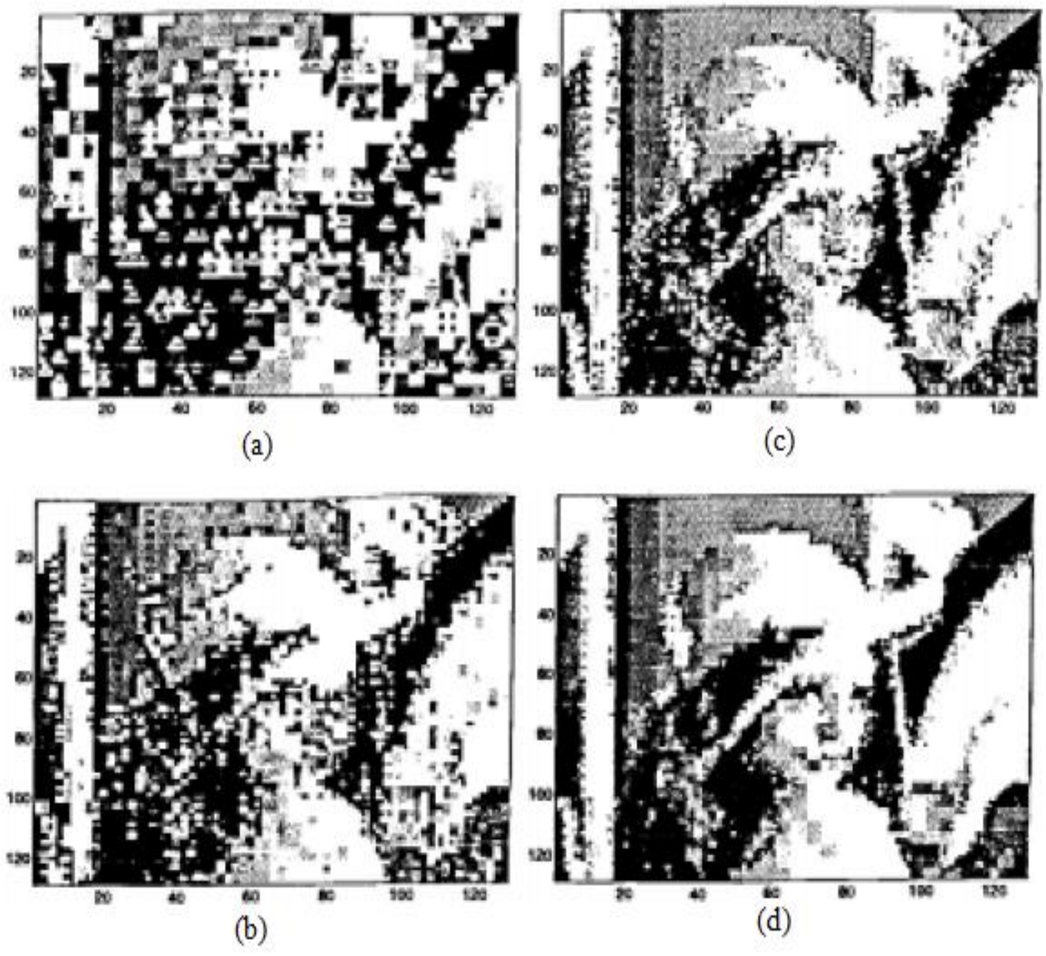

Fig.6. Decoding iterations with 4x4 decoding using square

Fig.7(a) Original Lena image

Fig.7(b) Jacquin method decoded image

Fig.7(c) Our proposed method

7. CONCLUSION

In this paper described the design of digital image coding systems referred to as Fractal Block Coders which are based on a theory of iterated contractive image transformations. The proposed preliminary design issues are to select an adaptive image partition made of non-overlapping range cells. Then the encoding of an original image then consists of capturing the self-transformability of the original image by searching a global transformation pool for a transformation defined block wise-a fractal code-under which the image is approximately invariant. For a specific block-base fractal image coding system was presented as well as encoding and decoding results. Mentioned block-based fractal coding system affects all of the following: visual quality of coded images, compression, encoding complexity and speed, in a complex manner. Moreover decoding complexity remains fairly low and stable for various system designs. This strategy of piecewise self-similarity and its capture through the construction of contractive image transformations which leave original images approximately invariant provides a new scheme for the exploitation of image redundancy for image compression-this property is what makes fractal image coding work can be applied in Medical Imaging, where doctors need to focus on image details, and in Surveillance Systems, when trying to get a clear picture of the intruder or the cause of the alarm. The proposed methods gives a clear advantage over the Discrete Cosine Transform Algorithms such as that used in JPEG or MPEG.

8. ACKNOWLEDGMENTS

We would like to thank Dr. P.K. Bose, Director, NIT, Agartala, India for supporting this research work.

9. REFERENCES

[1] B.B. Mandelbrot, The Fractal Geometry of Nature, Freeman, San Francisco, 1983.

[2] B. Mandelbrot, The Fractal Geometry of Nature. San Francisco, CA: Freeman, 1982.

[3] G. Cherbit, Fractals, Dimensions non Entireness et Applications. Paris, France: Masson, 1987.

[4] K. Falconer, The Geometry of Fractal Sets. London, UK Cambridge Univ. Press, 1985.

[5] Fractal Geometry, Mathematical Foundations and Applications. New York Wiley, 1990. Y. Fisher, E.W. Jacobs, and R. D. Boss,

[6] H-0. Peitgen and P. H. Richter, The Beauty ofFractals. Berlin: Springer, 1986.

[7] J. Gleick, Chaos, Making of a New Science. New York: Vicking, 1987.

[8] H-0. Peitgen, H. Jiirgens, and D. Saupe, Chaos and Fractals. New York Springer-Verlag, 1992.

[9] A. P. Pentland, "Fractal-based descriptions of natural scenes," IEEE Trans. Pattern Anal. Machine Intell., vol. PAMI-6, no. 6, 1984.

[10] 'Fractal surface models for communications about terrain," SPIE Visual Comun. Image Process. 11, vol. 845, 1987.

[11] M. C. Stein, "Fractal image models and object detection," SPIE Visual Commun. Image Process. II, vol. 845, 1987.

[12] M. F. Bamsley and S. Demko, "Iterated function systems and the global construction of fractals," Proc. Roy. Soc. London, vol. A399, pp. 243- 275, 1985.

[13] M. F. Bamsley, V. Ervin, D. Hardin, and J. Lancaster, "Solution of an inverse problem for fractals and other sets," Proc. Nut. Acad. Sci., vol. 83, pp. 1975-1977, 1986.

[14] M. F. Bamsley, Fractals Everywhere. New York Academic Press, 1988.

[15] M. F. Bamsley, J. H. Elton, and D. P. Hardin, "Recurrent iterated function systems," Constructive Approximation. Berlin, Germany: Springer-Verlag, 1989, pp. 3-3 1.

[16] M. F. Bamsley and A. Jacquin, "Application of recurrent iterated function systems to images," Proc. SPIE, vol. 1001, pp. 122-131, 1988.

[17] M. F. Bamsley, A. Jacquin, F. Malassenet, and L. Reuter, "Harnessing chaos for image synthesis," in Proc. SIGGRAPH, '88, 1988, pp. 131-140.

[18] A. Foumier, D. Fussell, and L. Carpenter, "Computer rendering of stochastic models," Commun. ACM, vol. 25, pp. 371-384, 1982.

[19] J. C. Hart, D. J. Sandin, and L. H. Kauffman, "Ray tracing deterministic 3-D fractals," Comput. Graph., vol. 23, no.3, pp. 91-100, 1989.

[20] C. Hart and F. K. Musgrave (Co-chairs), "Fractal modelling in 3D computer graphics and imaging," in SIGGRAPH '91, course notes, 1991.

[21] H-0. Peitgen and D. Saupe, The Science of Fractal Images. New York Springer-Verlag, 1988

[22] L. Hodges-Reuter, "Rendering and magnification of fractals using iterated function systems," Ph.D. dissertation, Georgia Inst. Technol., Atlanta, 1987

[23] F. J. Malassenet, "Texture coding using a pyramid decomposition," in Proc. ICASSP-93, 1993, vol. V, pp. 353-356.

[24] R. Rinaldo and A. Zakhor, "Fractal approximation of images," in Proc. Data Compression Conf.. Mar.-Apr. 1993, p. 451.

[25] Shouji Chen, Liming Zhang: Fractal and image compression. *Shanghai Science And Technology Education Publishing House*(1998)

[26] A. E. Jacquin. Image Coding Based on a Fractal Theory of Iterated Contractive Image Transform. IEEE trans. on Image Processing, vol. 1, no.1, pp. 18-30, 1992.