

Neural based domain and range pool partitioning using Fractal Coding for nearly lossless Medical Image Compression

S.BHAVANI¹, K.THANUSHKODI²

Asso.Prof., Department of Electronics and Communication Engineering,¹

Sri Shakthi Institute of Engineering and Technology, Coimbatore,¹

Director,²

Akshaya College of Engineering and Technology, Coimbatore,²

INDIA.

bhavanisns@yahoo.com¹ thanush12@gmail.com²

Abstract: - This work results from a fractal image compression based on iterated transforms and machine learning modeling. In this work an improved quasi-losses fractal coding scheme is addressed to preserve the rich features of the medical image as the domain blocks and to generate the remaining part of the image from it based on fractal transformations. Machine learning based model is used for improving the performance of the fractal coding scheme and also to reduce the encoding computational complexity. The performance of the proposed algorithm is evaluated in terms of compression ratio, PSNR and encoding computation time, with standard fractal coding for MRI image datasets of size 512×512 over various thresholds. The results show the increase in encoding speed, outperforming some of the currently existing methods thereby ensuring the possibility of using fractal based image compression algorithms for medical image compression.

Key-Words: - Image Compression, Iterated transforms, Fractal Image Compression, Medical Image, Fractals

1 Introduction

A fractal is a structure that is made up of similar forms and patterns that occur in many different sizes. The term fractal was first used by Benoit Mandelbrot to describe repeating patterns that he observed occurring in many different structures [15]. These patterns appeared nearly identical in form at any size and occurred naturally in all things. These fractals could be described and mathematically modeled. The interest of applying fractals has increased in recent years. Even though Fractal scheme is promoted by M.Barnsley [1] who found fractal image compression technology, it was first made available to public by E.Jacobs and R.Boss who used regular partitioning of segments and classification of curve of random fractal curve [2].Barnsley et al. were the first to introduce the concept of iterated function systems based fractal image compression [9]. Fractal image coding is described based on theory of Iterated contractive image transformations [4]. A new approach to image compression using iterated transform is presented [5] which have found the basics from the theory of IFS developed by Hutchinson [6] and [7].The problem of finding a suitable IFS code is solved by use of a library of IFS codes and complex moments and by using simulated annealing method for solving nonlinear equations in presented in [8]. Fractal image

compression signal to noise ratio is found to be moderately better for smaller images for a given degree of compression as indicated by Fisher in [3].

Self-similarity or scaling is one of the main properties of fractal geometry. One of the measures of image quality is artifacts. Fractal shows blocking artifacts at higher compression ratio but at low ratio it tends to be localized. Speed up methods in fractal image coding based on feature vector and classification approaches and complexity in fractal image decoding is detailed in [11]. Further speeding up fractal image compression by using 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 is addressed in [12].A fast and efficient hybrid scheme [20] using a wavelet transform improves the image quality in fractal image compression, whereas hybrid coding based on partial mapping where only part of the image is encoded using fractal technique and the remaining part is modeled using other algorithms demonstrates the compatibility of fractal image coding algorithm with other methods[14]. A faster fractal image compression using quad tree recomposition is addressed in [18].The complexity in fractal image decoding is detailed in

[13]. In survey on coding algorithms in medical image compression addressed in [16], it is found that fractal image compression exploits self-similarity among image elements and hence reproduces image elements with high compression rate

For a given degree of image compression we get moderately better signal to noise ratios to get good image quality in retrieved image. Medical Image compression using fractal concept would tend to arrive at higher compression rates and fractal zooming further allows us to increase the size of the image however the loss of information in fractal compression is unacceptable in medical imaging. Lengthy encoding process is another drawback of fractal compression as it leads to increase in computational encoding complexity. This paper addresses to above mentioned issues of fractal image compression.

This paper is organized as follows. Section 2 briefs about the standard Fractal image compression method. Section 3 explains the proposed Fractal coding algorithm-I and proposed Fast Fractal coding algorithm-II based on neural based Machine learning. Section 4 deals with results and discussions. Section 5 derives Conclusion followed by acknowledgements and References.

2 Standard Fractal Image Compression Method

A two dimensional image is represented mathematically as $z = f(x, y)$ where $f(x, y)$ represents the gray level with 0 being black and 1 being white at the point (x, y) in an image. I denote the close Interval $[0, 1]$. On applying transformation 'W', on to the image 'f', we get a transformed Image $W(f)$. W always moves points closer together as it is contractive. Affine transformations are combinations of rotations, scaling and translations of the coordinate axes in n-dimensional space which always map squares to parallelograms. The general form of affine transformation is given by

$$W = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} = \begin{bmatrix} ax + by + e \\ cx + dy + f \end{bmatrix} \quad (1)$$

If the translations (e & f), scaling factors (r & s) and rotations (θ & ϕ) are known in advance, then the coefficients may be calculated. The transformation found suitable for encoding gray scale images thought of as a three dimensional image with coordinates as x & y and intensity as z is given in equation 2 where s_i controls the

contrast and o_i controls the brightness of transformation.

$$w_i \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_i & b_i & 0 \\ c_i & d_i & 0 \\ 0 & 0 & s_i \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \\ o_i \end{bmatrix} \quad (2)$$

2.1 Encoding and Decoding of Images

According to contractive mapping fixed theorem which states that if the transformation is contractive then, when applied repeatedly starting with any initial point, we converge to a unique fixed point. If X is a complex metric space and $W: X \rightarrow X$ is contractive then W has a unique fixed point IWI. In simple, collection of transformation defines an image. The encoding process partitions the image 'f' into pieces to which we apply transform w_i to get back the original image [1]. A portion of the original image we denote by D_i and apply w_i on D_i . The partitioned domain of the original image is represented by v_i where $v_i(D_i) = R_i$ (Range blocks). Hence $UR_i = I^2$ with $R_i \cap R_j = \emptyset$ when $i \neq j$. If 'f' is the image and W is the transformation then the transformed image is given by $f = W(f) = w_1(f) \cup w_2(f) \cup w_3(f) \dots \cup w_n(f)$. The map W is defined as union of $w_i(f)$, where $w_i = D_i \times I$ and we get transformed domain. The transformed domain is compared with the range block and if it matches, it is copied as Range. We find D_i and maps w_i such that when we apply w_i to a part of the image over D_i , some portions are found to be lost in R_i . The problem lies in finding pieces of R_i (corresponding to D_i) in encoding process.

The standard fractal encoding algorithm:

- Load an input image into buffer
- Partition the image into square blocks with non-overlap.
- Choose initially the size of the domain block to be twice the size of the range block.
- Down sample the domain blocks to the size of range blocks and compute the eight possible affine transformations for each block.
- Choose the domain block that resembles the range block with respect to some metric and compute the encoding parameters that satisfy the mapping.
- Save the coefficients which represents fractal element.

The standard fractal decoding algorithm:

- Load the initial image which is to be decoded
- Apply w_i repeatedly until we converge to a fixed point which means for each w_i we find the domain block, rescale to the size of range block
- Multiply the pixel values by s_i then add o_i and compute the pixel values in each R_i , which allows copying the content of the domain blocks to the range blocks
- Take the output of first iteration(Range block) to be the input of the next iteration.
- Repeat doing the same until the desired attractor is reached.

One of the most notable features of fractal image compression is that the decoding process is simple. The decoder proceeds its work in the same way as in the case of the traditional encoder (i.e., fixed block size encoding). The decoder consumes less time for computation compared to that of an encoder. The decoding time generally depends on the number of iterations and here it takes only few iterations ranging from 4-8 to reach the fixed point.

3 Proposed Fractal Image Compression and Fast Fractal Image Compression methods

3.1 Proposed Domain–Range block separation Algorithm

w_i is determined uniquely for a chosen metric. In [4] root mean square error was chosen as the metric. In standard fractal image compression proposed in [3] it uses distance as metric, whereas in [17] it uses entropy as the metric. In our proposed method we have chosen variance as our metric since variance is independent of change of origin but not scale. Standard deviation denoted by σ is the positive square root of arithmetic mean of squares of deviations of the given values from their arithmetic mean. The purpose of squaring deviations overcomes the drawback of ignoring the signs in mean deviation.

In our proposed algorithm the domain and the range blocks are separated based on variance computed of each blocks in the block set. The feature rich blocks are selected as domain blocks and preserved along with transformation coefficients. Image 'f' is partitioned into image B

comprising blocks b_1, b_2, \dots, b_n using quad tree decomposition method. Initially Range and domain block sets are null sets. Using the quad tree decomposition method as proposed in [3], the image is partitioned into large range blocks initially. The best transformation of each block is then found.

If the transformation is discarded using the metric, the range block is divided into 4 quadratic sub blocks and again best transformation is searched for each sub block. This continues until all the blocks are covered. If the subdivision is not done in equal proportions the tree resulting from it may lose the property of symmetry. The minimum and maximum possible values of o_i are restricted corresponding to s_i . Once the choice of R and D has been made, choosing a set $\{R_i\} \in R$ and the corresponding set $\{D_i\} \in D$, for encoding should yield good compression and high picture quality. The encoding time depends on the time taken in finding the domains D_i .

Let B as a set of all the blocks in the image after quad tree decomposition, R be the set of Range blocks and D be the set of Domain blocks which should be separated from the set B

Where $B = \{b_1, b_2, b_3, \dots, b_n\}$,

Let $R = \{ \}$ and $D = \{ \}$

For each block in B

Do

{

If ($S_{b_i} > d_{\min}$)

{

$R \leftarrow R \cup b_i$

}

Else if ($\sigma_{b_i}^2 > \sigma_{\max}^2 \times \tau$ and $\sigma_{b_i}^2 \geq \sigma_{\max}^2$)

{

$D \leftarrow D \cup b_i$

}

Else

{

$R \leftarrow R \cup b_i$

}

}

Where

s_{b_i} is the size of the block b_i
 d_{min} is the minimum Domain Block Size
 $\sigma_{b_i}^2$ is the variance of the block b_i in the set B
 σ_{max}^2 is the maximum variance of the $d_{min} \times d_{min}$ blocks of the image.
 τ is the threshold value which normally lies between 0 & 1 decides the size of the domain pool as well as the features of the blocks in the domain pool.

If τ is 0 then all the blocks of size $d_{min} \times d_{min}$ will be selected as domain blocks. If τ is 1 then all the blocks of size $d_{min} \times d_{min}$ having highest variance only will be selected as domain blocks. Hence it is clear that the compression quality as well as compression time is decided by the value of threshold ' τ '.


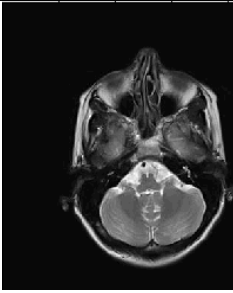
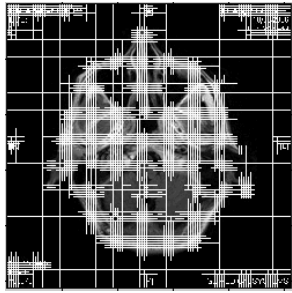
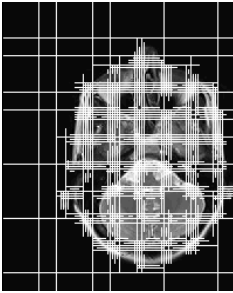
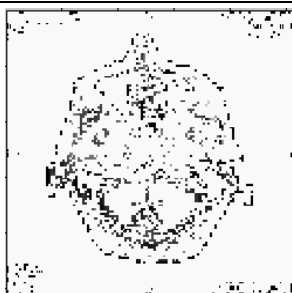

	Sample MRI Image 1	Sample MRI Image 2
Input Image		
The Partitioned Image		
Separated Domain Blocks		

Fig. 1 Feature Rich and separated Domain Blocks

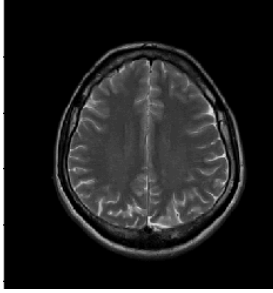
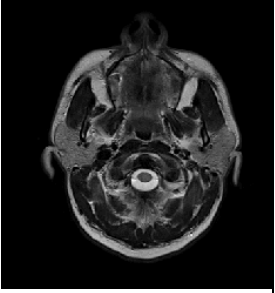


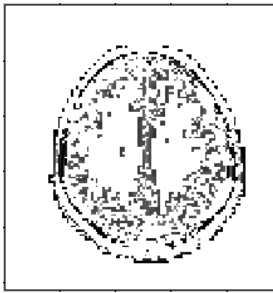
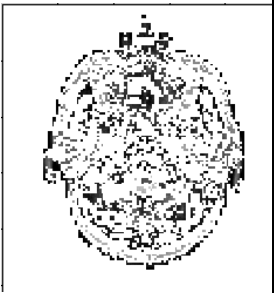
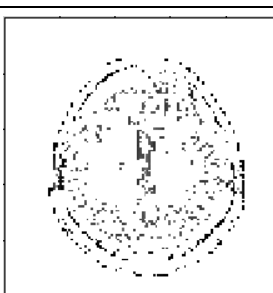
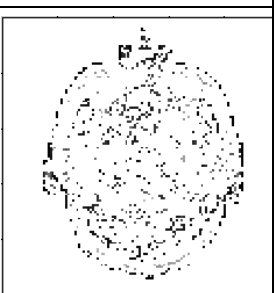
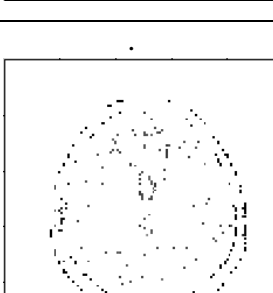

	Sample MRI Image 3	Sample MRI Image 4
Input Image.		
Threshold = 10^{-3}		
Threshold = 10^{-4}		
Threshold = 10^{-5}		
Threshold = 10^{-6}		

Fig. 2 Domain blocks separated for various thresholds

3.1.2 The Proposed Fractal Coding Algorithm-I

The following steps outline the compression process of the proposed compression algorithm:

- Read the Input image I
- Decompose the image I into a number of non overlapping blocks of various sizes using quad tree decomposition.
- Separate all the feature-rich $d \times d$ sized blocks from the decomposed image based on previously mentioned domain-range block separation algorithm, and Mark them as Domain Blocks and assume the remaining as Range Block.
- For each Range Block, find the best matching domain block and record the coefficients of the transformation.
- Compress the Domain blocks using any lossless compression and save them as seed along with the coefficients of the transformation.

Fig. 1 shows the partitioned image and the domain blocks for 2 sample MR Images separated by proposed fractal coding algorithm Fig. 2 shows the domain blocks separated for various thresholds.

3.2 Unsupervised Machine Learning (ML)

In the proposed algorithm-II, we are using a self-organizing neural network based ML technique to group the domain blocks and range blocks for reducing the search space to improve the speed of encoding of the algorithm. Unsupervised learning can be viewed in terms of learning a probabilistic model of the data. Even when the machine is given no supervision or reward, the machine estimates a model that represents the probability distribution for a new input given previous inputs. With a probabilistic model one can also achieve efficient communication and data compression can be achieved.

Self-Organizing Feature Maps or SOMs provides a way of representing multidimensional data in much lower dimensional spaces usually one or two dimensions. This process, of reducing the dimensionality of vectors, leads to data compression technique called as vector quantization. In addition, the Kohonen technique creates a network that stores information which maintains topological relationships within the training set. In addition to clustering, regions of similar properties are usually found adjacent to

each other. An important feature is that SOM's learn to classify the training data without any external supervision and training a SOM however, requires no target vector. Learning in the self-organizing map is to associate different parts of the SOM lattice to respond similar input patterns.

SOM algorithm

- 1) Randomize the map's nodes' weight vectors
- 2) Grab an input vector
- 3) Traverse each node in the map
- 4) Use Euclidean distance formula to find similarity between the input vector and the map's node's weight vector
- 5) Track the node that produces the smallest distance (this node will be called the Best Matching Unit or BMU)
- 6) Update the nodes in the neighborhood of BMU by pulling them closer to the input vector $W_v(t+1) = W_v(t) + \Theta(t)\alpha(t)(D(t) - W_v(t))$

There are two ways to interpret a SOM. Because in the training phase weights of the whole neighborhood are moved in the same direction, similar items tend to excite adjacent neurons. Therefore, SOM forms a semantic map where similar samples are mapped close together and dissimilar apart. The other way to perceive the neuronal weights is to think them as pointers to the input space. They form a discrete approximation of the distribution of training samples. More neurons point to regions with high training sample concentration and fewer where the samples are scarce. With Mat lab's neural network toolbox we can create and use a SOM (neural network) in simple and easy way.

3.2.1 The Proposed Fast Fractal coding Algorithm-II

The following steps outline the compression process of the proposed compression algorithm II. The first three steps are same as previous. In addition to the previous method, a supervised learning algorithm is used to speed up the encoding process.

Compression

- 1) Read the Input image I
- 2) Decompose the image I into a number of non overlapping blocks of various sizes using quad tree decomposition.
- 3) Separate all the feature-rich $d \times d$ sized blocks from the decomposed image based on previously mentioned domain-range block separation algorithm, and Mark them as Domain Blocks and assume the remaining as Range Block.
- 4) Organize two sets of n Groups from the Domain Blocks as well as the Range blocks, based on the features of the blocks using a supervised classification technique.
- 5) For each Range Block, find its group label and find the best matching domain block from the corresponding Domain block the transformation.

De-compression

The following steps outline the decompression process. The decompression can be done by using the fractal IFS code as follows.

- 1) Load the saved coefficients and the Seed Blocks.
- 2) Create memory buffers for the range screens.
- 3) Recreate the feature – rich areas of the range screen directly from the seed blocks (lossless –part)
- 4) Apply the transformation using the seed blocks and recreate the remaining portion of the range screen (lossy-part).
- 5) Reconstruct the rough blocks as well as smooth blocks as it is from IFS code since they are stored without any compression.
- 6) Reconstruct all the remaining blocks from the stored seed blocks with the help of IFS code.

4 Results and Discussions

We have implemented the proposed algorithm using Mat lab and evaluated the performance with respect to PSNR and encoding time for various thresholds as indicated in Tables 1 and 2. From the results obtained it is clear that

as the threshold increases, PSNR increases. Table 3 gives the compression ratio at different thresholds for 4 sample MRI Images. For decrease in threshold it is found that the compression ratio is very much higher and also the encoding time is reduced considerably to 23.96 Sec. To further decrease the encoding time Machine learning based fractal coding has been implemented and encoding is achieved at very less time as against the methods of standard fractal coding and improved fractal coding.

Table 1 PSNR at Different Levels of Compression

Sample MRI Image	PSNR at Different Threshold (db)			
	$\tau = 10^{-3}$	$\tau = 10^{-4}$	$\tau = 10^{-5}$	$\tau = 10^{-6}$
1	25.02	24.10	23.59	22.93
2	34.64	32.48	31.02	29.39
3	34.11	31.88	30.47	27.37
4	32.80	31.03	29.72	27.91
Avg	31.64 25	29.8725	28.7	26.9

Table 2 Time Taken at Different Levels of Compression

Sample MRI Image	Time Taken at Different Threshold (sec)			
	$\tau = 10^{-3}$	$\tau = 10^{-4}$	$\tau = 10^{-5}$	$\tau = 10^{-6}$
1	56.08	55.58	55.32	55.06
2	40.08	35.67	31.31	15.50
3	48.73	45.05	26.09	11.01
4	75.33	66.80	37.17	14.27
Avg	55.05 5	50.77 5	37.4725	23.96

Table 3 Compression Ratio at Different Thresholds

Sample MRI Image	The Compression Ratio at Different Threshold			
	$\tau = 10^{-3}$	$\tau = 10^{-4}$	$\tau = 10^{-5}$	$\tau = 10^{-6}$
1	3.1	3.8	5.8	12.8
2	4.9	7.2	14.7	39.8
3	5.8	9.3	21.2	66.6
4	4.4	7.8	19.6	63.3
Avg	4.55	7.025	15.325	45.625

The performance of the proposed algorithms for various thresholds is plotted in terms of PSNR, encoding time and Compression ratio for 4 sample MRI images in Fig. 3.

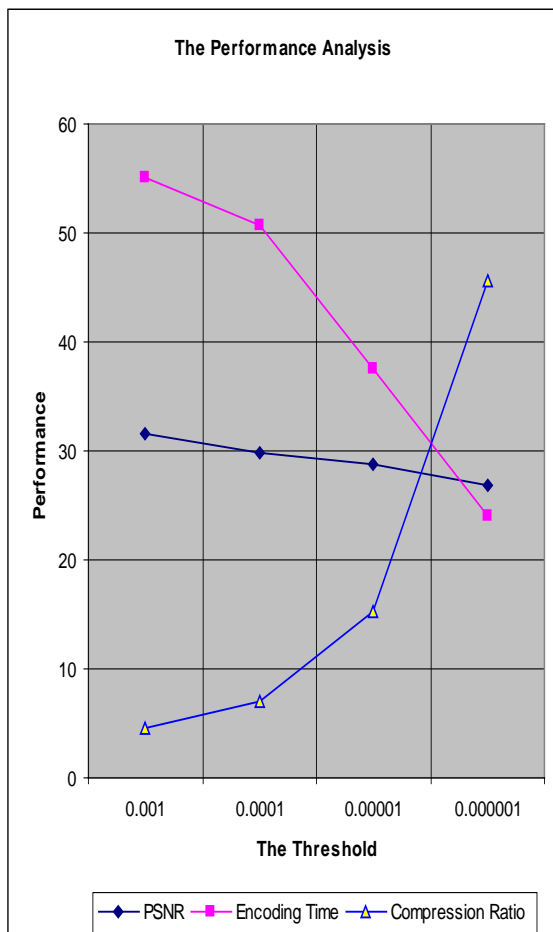


Fig. 3 Performance Analysis of proposed algorithms for 4 sample MR Images at different thresholds

The Comparison of the standard fractal coding, proposed algorithm I (improved fractal coding) and proposed algorithm-II (Fast Fractal coding based on machine learning) is done. We have tested the performance of the above mentioned three algorithms with the Medical image (512x512) shown in Fig 4. (Sample MRI Image 4). Table 4 indicates the performance of all the three algorithms for the Sample MRI Image. From the comparison, it is found that compression time has reduced drastically with improvement in PSNR and compression ratio in proposed algorithm-II.

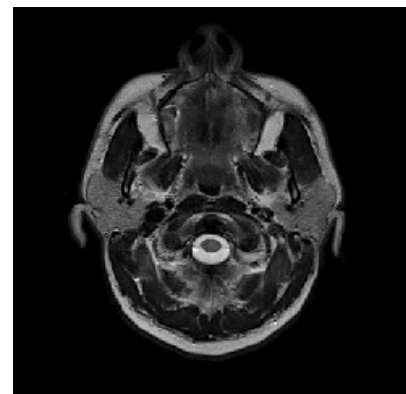


Fig. 4 Sample MRI Image 4

Table 4 PSNR achieved for Different Algorithms

Metric	Standard Fractal Encoding	Proposed Algorithm I ($\tau = 10^{-5}$)	Proposed Algorithm II ($\tau = 10^{-5}$)
PSNR (db)	27.49	29.67	29.72
Compression Time (sec)	1738	459.73	37.17
Compression Ratio	3.20	19.6	19.6

Fig. 5 indicates the comparison of PSNR obtained for all algorithms. Fig.6 and Fig.7 and Fig.8 illustrate the charts plotted with the performance of PSNR, compression time and compression ratio for all the three algorithms.

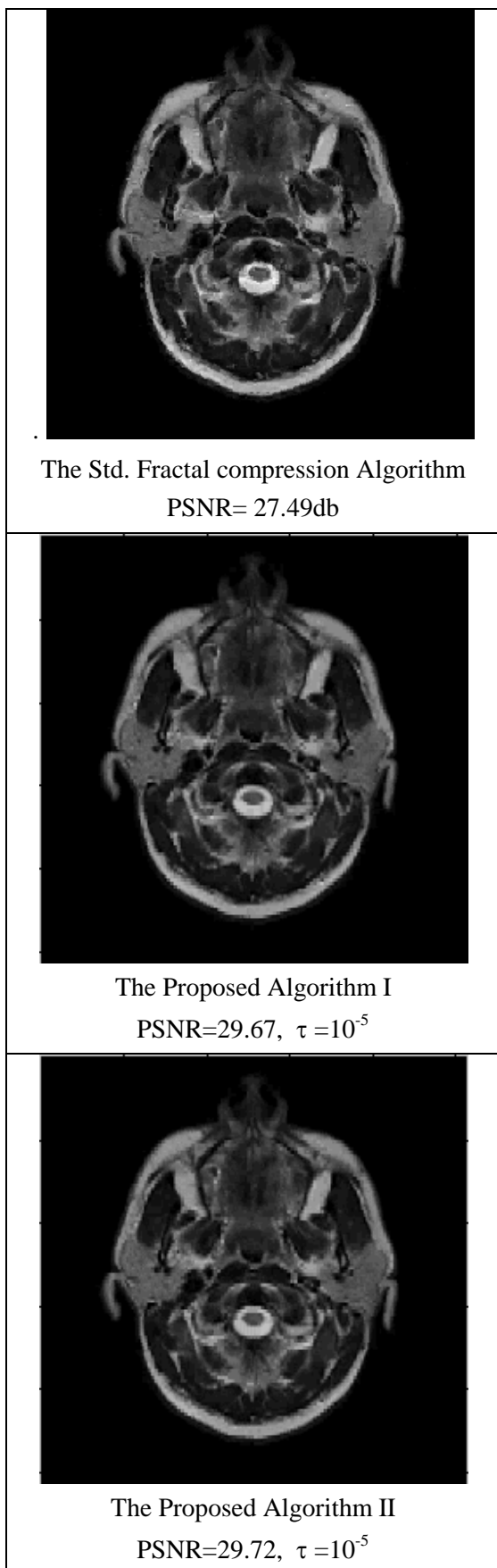


Fig.5 PSNR for all three Algorithms

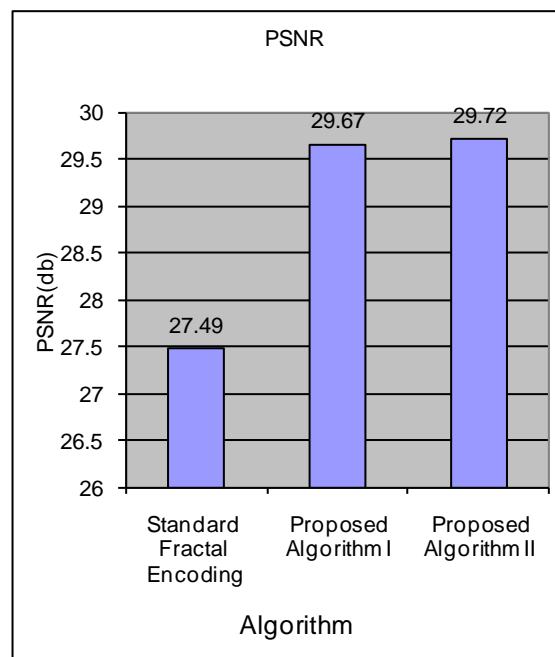


Fig. 6 PSNR Comparison Chart

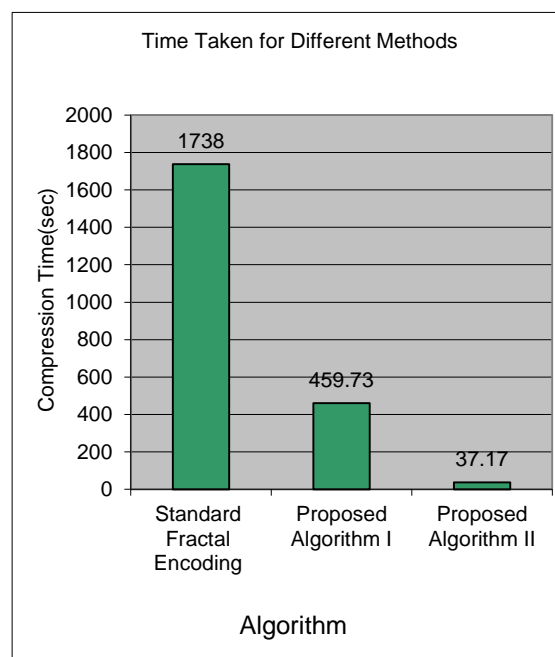


Fig. 7 Compression Time Compression Chart

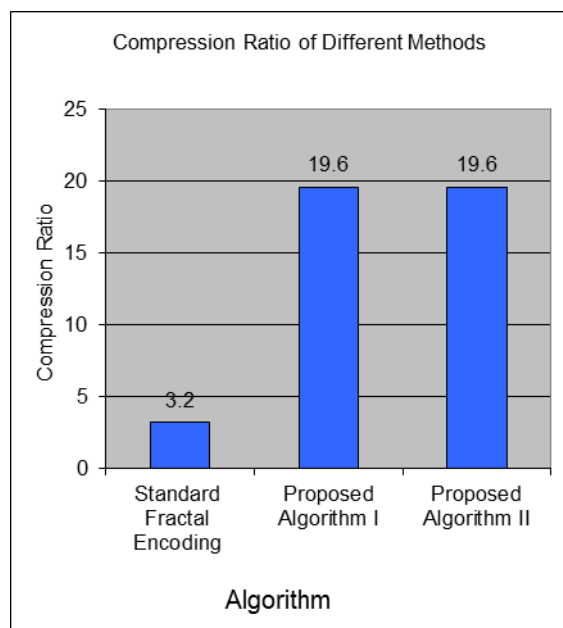


Fig. 8 Compression Ratio Comparison Chart

5 Conclusion

This work addresses to an improved Neural based fractal compression technique which is used to test the possibility of the fractal compression to medical imaging. The images compressed with the proposed fractal compression methods shows promising ways for applying them for medical image compression applications. The two newly proposed methods competes the standard fractal image compression algorithms. Since the proposed algorithm is regenerating feature rich portions of the images without any loss of information at that region, the perceptual quality of the image is found to be very good than that of the standard fractal image compression algorithm. Machine learning based model is used for improving the performance of the fractal coding scheme and also to reduce the encoding time. The performance of the proposed algorithm is evaluated with standard fractal coding in terms of compression ratio, PSNR and encoding computation time for MRI image datasets of size 512×512 for different thresholds. The results show the improvement in encoding speed, outperforming some of the currently existing methods thereby ensuring the suitability of using fractal based image compression algorithms for medical image compression. Hybrid fractal coding is not addressed in our work. Our future work will be based on hybrid coding which allows for region of interest coding and also progressive transmission.

Acknowledgements

We thank those individuals, institutions who have contributed and motivated to come up with this work successfully. Special acknowledgements to Dr. K.Thanushkodi for the insightful comments and suggestions and also gratitude to Dr. S.Venkatakrishnan, Seetharam Hospital for his advice and ACCURA SCANS, Coimbatore for providing the required MRI head scan data sets for analysis

References:

- [1] M.F.Barnsley, Fractals everywhere Academic press. San Diego,(1989)
- [2] E.W.Jacobs,R.D.Boss,Fractal based image Compression.NOSC Technical report,1315, Naval Ocean Systems Centre, San Diego, CA 92152-5000,1989
- [3] Yuval Fisher, Fractal Image Compression.SIGGRAPH '92 COURSE NOTES
- [4] E.W.Jacobs, Y.Fisher, R.D.Boss, Fractal Image Compression A study of iterated transform method, Signal Processing 29, Elsevier, 1992, 251-263.
- [5] A.E.Jacquin, Image coding based on a fractal theory of iterated contractive image transformations, *IEEE Trans. On Image processing*, Vol.1, 1992, 18-30.
- [6] J.E.Hutchinson, Fractals and self-similarity. Indiana University, Mathematics Journal, Vol. 35, No.5,1981
- [7] M.F.Barnsley, A.E.Jacquin, Application of Recurrent IFS to images, *PIE Visual Communication Image processing*, Vol. 1001, 1988, pp.122-131.
- [8] Maaruf Ali, G.Trevor Clarkson, Fractal Image compression. In *proceedings of 1st seminar on information technology and its applications,(ITA '91)*, Mark field Conf.Centre,Leicester,U.K,1991,29.
- [9] M.F.Barnsley, Fractal image compression. Notices of the AMS,1996
- [10] Miroslav Galabov, Fractal Image Compression, *International Conference on Computer systems and Technologies*, CompSysTech, 2003.
- [11] Mario Polvere, Michele Nappi, Speed up in Fractal image coding Comparison of Methods, *IEEE Transactions on Image Processing*, Vol. 9, No. 6, 2000.

- [12] Hanna Mehsen Ali, Najla Abd Hamza Al Mayahee, Amara Abdul Wohid Funjan, Speeding Fractal Medical Image Compression by using Highest Gray Value of Pixels Technique.
- [13] D.Saupe, R.Hamzaoui, Complexity reduction methods for fractal image compression. *IMA Conf Proceedings on Image processing, Mathematical methods and applications*, 1994
- [14] Zhou Wang, David Zhang, Yinglin Yu, Hybrid image coding based on partial mapping, *Signal processing: Image communications* No. 15, ,Elsevier, 2000, pp. 767-779
- [15] B.B.Mandelbrot, The Fractal Geometry of Nature, Freeman, San Francisco, 1983
- [16] S.Bhavani, A Survey on coding algorithms in Medical Image compression, *International Journal of Computer Science & Engineering*, Vol.02, No.05, 2010, pp. 1429-1434.
- [17] Yuzo Iano, Fernando Silvestre da Silva, Ana Lúcia Mendes Cruz, A fast and efficient hybrid fractal wavelet image coder, *IEEE Trans. Image Processing*, Vol.15, No.1, January 2006.
- [18] David.J.Jackson, Wagdy Mahmoud, A.William, Stapleton, T.Patrick, Gaughan, Faster fractal image compression using Quad tree Recomposition. Elsevier, 1995
- [19] M. Hassaballah, M.M.Makky, B.Youssef, Mahdy, A Fast Fractal image compression method based entropy. *Electronic Letters on Computer Vision and Image Analysis*, Vol.5, No.1, 2005, pp. 30-40.

About Authors:



S.Bhavani born in Coimbatore in the year 1968 received B.E degree in Electronics and Communication Engineering from V.L.B Janaki Ammal College of Engineering and Technology, Coimbatore Tamilnadu in the year 1990 and M.E in Applied Electronics from Maharaja Engineering College, Coimbatore, Tamilnadu in the year 2006. Since 1992, she was working in various disciplines in Park group of institutions, Coimbatore. Currently working as

Associate Professor in Sri Shakthi Institute of Engineering and Technology, Coimbatore. She is a research Scholar (Part-time) in the Department of EEE at Anna University, Coimbatore Since 2007. She is also a life Member of ISTE and National Merit scholarship holder.



Dr. K. Thanushkodi, born in Theni District, Tamilnadu State, India in 1948, received the BE in Electrical and Electronics Engineering from Madras University, Chennai. MSc (Engg) from Madras University, Chennai and PhD in Electrical and Electronics Engineering from Bharathiar University, Coimbatore in 1972, 1976 and 1991 respectively. His research interests lie in the area of Computer Modeling and Simulation, Computer Networking and Power System. He has published 26 technical papers in National and International Journal.