Advanced Image Compression And Decompression Method Using 3-D Matrix Function Parameter

Sukhjinder Singh[1], Deepinder Singh[2]

[1] Research Scholar, Department of Electronics and Communication Engineering, Bhai Gurdas Institute of Engineering & Technology, Sangrur (Punjab), India

[2] Assistant Professor, Department of Electronics and Communication Engineering, Bhai Gurdas Institute of Engineering & Technology, Sangrur (Punjab), India

[1]sukhi91188@gmail.com, [2]wadhwadeepinder@rediffmail.com

Abstract— Recent respective literature shows that wavelet based fractal image compression is a lossy image compression technique. In this paper we have proposed a hybrid technique which is much successful than pure fractal image compression technique. Use of the wavelet accelerates the matching process and reduces the encoding time and up to some extent it tries to overcome one of the major disadvantages of the pure fractal compression technique. Further, we have introduced 3-D matrix function which creates wavelet tree with the help of Haar transform so there is no requirement of an additive constant in fractal-wavelet compression technique as was required in pure fractal technique. The compressed information is stored in the form of simple text file which can be send to the destination point for decompression purpose. For decompression technique we have implemented Inverse wavelet transform and inverse Haar transform for successful recovery of compressed image. We have investigated the performance of our scheme by comparing various compression algorithms by different researchers. The results show that the proposed algorithm for compression saves the compression time and avoids unnecessary calculations.

Keywords— 3-D Matrix Function, Advanced image, Function, Function Parameter.

I. INTRODUCTION

In image processing, a computer takes an input digital image and produces an output digital image. Digital images are numerical representations of images or objects. The image input to the computer is in numerical form and the output image after computer processing is a digital image. Image compression occurs in this component. Image compression addresses the problem of reducing the amount of data required to represent a digital image. Image compression uses various algorithms to remove data from an image. The under lying basis of the reduction is the removal of redundant data. Redundant data is the data which contains no relevant information or simply restate that which is already known. It is said to contain data redundancy. In digital image compression, three basic data redundancies can be identified and exploited:

- Coding redundancy, which is present when less than optimal code words are used.
- Interpixel redundancy, which results from correlations between the pixels of an image.
- Psychovisual redundancy, which is due to data that is ignored by the human visual system.

Data compression is achieved when one or more of these redundancies are reduced or eliminated. The purpose of compressing digital images is to reduce the size of the image to decrease transmission time and reduce storage space requirements.

2. Related Techniques and Proposed Technique

2.1. *Recent idea*. Recent researchers used discrete wavelet transform along with Iterated Function System where the calculation of range blocks and domain blocks was done using calculations having different constants. The respective method named Multiple Description utilized approximation sub band method where the basic range block was selected including the nearby approximate area to reduce the redundancy.

2.2. *Proposed Technique*. This paper presents a new technique to resolve the image into sets of range and domain blocks. The algorithm will compress the respective image by partitioning it into 512 x 512 non overlapping range blocks to extract redundant information and than using the 3-D function to compress the image in short time and to perform inverse wavelet transform in order to regain the original image.

3. 3-D Matrix Function Parameter

The recent Multiple Description algorithm partitions data image into size N x N non-overlapping range blocks Rj of predefine size D x D. After that domain blocks are defined of size 2B x 2B. Each range block is related to domain block by satisfying the following mapping equation:

$$w_j(D) = s_j \times D + o_j$$

Further this mapping function is required for following minimization equation:

$$(D_i, w_j) = \arg\min_{(D_i, w_j)} \{ w_j(D_i) - R_j \}_2$$

where R_j is the range block and D_i is the domain block. After solving the minimization process the required affine parameters can be obtained by:

$$s_j = \frac{(R_j - r_j I, D_i - d_i I)}{\{D_i - d_i I\}^2_2}$$

Where I is the matrix and r_j, d_i are the average pixel value of range and domain block.

The newly proposed technique in this paper defines a new affine parameter called 3-D matrix function parameter which finds domain and range pixels in three dimensional pattern i.e. Horizontal, Vertical and Diagonal.

This newly introduced method can tolerate the loss of lost data with certain amount of introduced redundancy. The modified affine parameter introduced in this paper is as follows:

$$s_{ij} = \frac{\Sigma\, D_{ij} R_{ij}}{\Sigma\, D_{ij}^2}$$

Where, s_{ij} is the approximated pixel value and D_{ij} , R_{ij} are the average pixel intensity for the range and domain blocks. Therefore in fractal coding the parameters basically required to be placed in correct order bit stream are s_{ij} , o_{ij} , position of domain block. The parameter o_{ij} is defined as:

$$o_{ij} = \frac{1}{N^2} \Sigma_{ij} (R_{ij} - s_{ij} D_{ij} – 0)$$

4. Wavelet Based Image Compression Using Fractal Coding

In this paper the given image is processed using wavelet transform and thereafter fractal compression method is applied in order to get image pixels information in rows and columns format to make a matrix. This matrix is in 3 X 3 form. During the compression method it is natural that the pixels obtained at different instants might have similar information i.e. same contrast and texture information and this similar information is called as redundancy which is very much unnecessary in the case of image compression. So, in this paper the proposed algorithm strongly removes this redundant information in order to compress the given image in lesser time and less memory consumption.

4.1 3-Dimensional Scheme For Finding Range Blocks

The basic idea behind this algorithm is to design a system which is capable of compressing the given image by applying discrete wavelet transform and then applying 3-dimensional fractal image compression algorithm in order to detect range and domain blocks at a faster speed as shown in Figure 2. During this process the algorithm searches for non-overlapping information i.e. removes redundant information and finds the difference between range and domain blocks. The main objective of designing this algorithm is to save maximum time during compression process as the traditional fractal image encoding method has a large amount of computing time in which is block is searched, compared and matched with respective domain block in order to restore the given image.

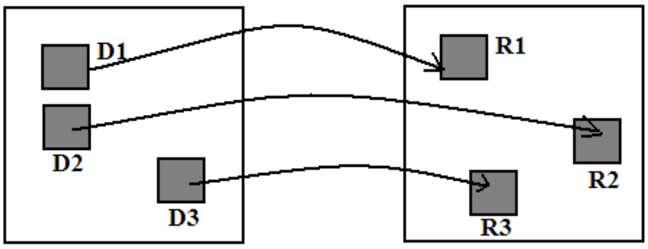


Figure 1: Matching of Range Blocks and Domain Blocks

4.2 WAVELET SUB-TREE

Fractal image compression was one of the earliest compression schemes to take advantage of image redundancy in scale. The theory of iterated function systems motivates a broad class of fractal schemes but does not give much guidance for implementation. Fractal compression schemes do not fit into the standard transform coder paradigm and have proven difficult to analyze. We introduce a wavelet-based framework for analyzing fractal block coders which simplifies these schemes considerably. Using this framework we find that fractal block coders are Haar wavelet sub tree quantization schemes, and we thereby place fractal schemes in the context of conventional transform coders. We show that the central mechanism of fractal schemes is an extrapolation of fine-scale Haar wavelet coefficients from coarse-scale coefficients. We use this insight to derive a wavelet-based analogue of fractal compression, the self-quantization of sub trees (SQS) scheme. We obtain a simple SQS decoder convergence proof and a fast SQS decoding algorithm which simplify and generalize existing fractal compression results. We describe an adaptive SQS compression scheme which outperforms the best fractal schemes in the literature by roughly 1 dB in PSNR across a broad range of compression ratios and which has performance comparable to some of the best conventional wavelet sub tree quantization schemes. The Figure 2 shows the wavelet sub-tree

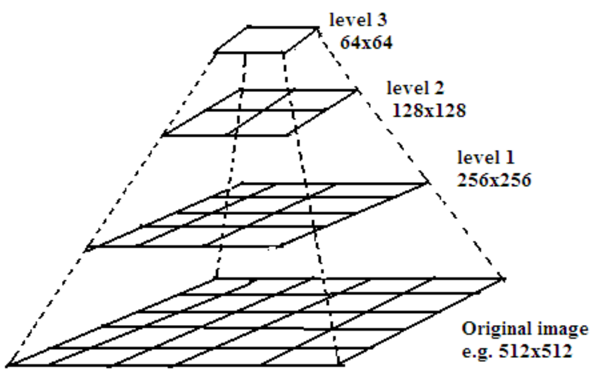


Figure 2: Wavelet Sub-tree

4.3 Discrete Wavelet Transform

Wavelets are functions defined over a finite interval and having an average value of zero. The basic idea of the wavelet transform is to represent any arbitrary function (t) as a superposition of a set of such wavelets or basis functions.

These basis functions or baby wavelets are obtained from a single prototype wavelet called the mother wavelet, by dilations or contractions (scaling) and translations (shifts). The wavelet-based transform uses a I-D sub band decomposition process in which a I-D set of sample is converted into the low-pass sub band (Li) and high-pass sub band (Hi). Where "i" represents level of decomposition. The low-pass sub band represents a down sampled low resolution version of the original image. The high-pass sub band represents residual information of the original image. In 2-D sub band decomposition, the entire process is carried out by executing I-D sub band decomposition twice, first in one direction (horizontal), then in the orthogonal (vertical) direction. For example, the low-pass sub band (Li) resulting from the horizontal direction is further decomposed in the vertical direction, leading to LLi and LHi sub bands. Similarly, the high pass sub band (Hi) is further decomposed into HLi and HHi. After one level of transform, the image can be further decomposed by applying the 2-D sub band decomposition to the existing LLi sub band. This iterative process results in multiple "transform levels". We refer to the sub band LLi as a lowresolution sub band and high-pass sub bands LHi, HLi, HHi as horizontal, vertical, and diagonal sub band respectively since they represent the horizontal, vertical and diagonal residual information of the original image.

$$A = \left[\begin{array}{cc|cc} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ \hline a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{array}\right] = \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right] \text{ or } \left[\begin{array}{c|c} HAH^T & HAG^T \\ \hline GAH^T & GAG^T \end{array}\right]$$

HAH^T gives the average of the elements HAG^T can be viewed as for vertical differences GAH^T can give as for horizontal differences GAG^T can be viewed as differences along the diagonal.

4.4 Affine Transformations

An Affine transformation is an important class of linear 2-D geometric transformations which maps variables by applying a linear combination of translation, rotation, scaling or shearing

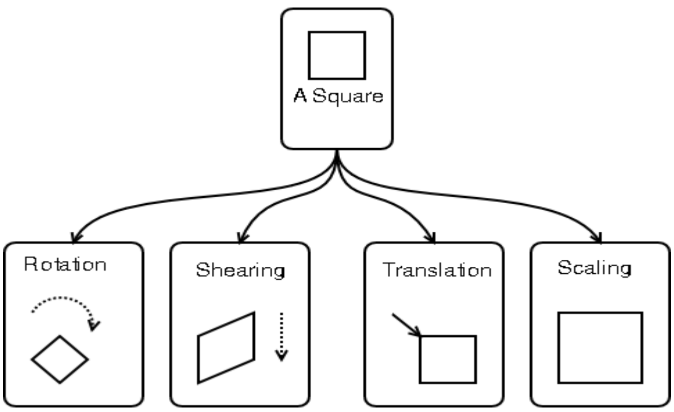


Figure 3: Different operations on an image using Affine Transformation operations

The word affine came from *affine* means "connected with". Mathematically affine transformation consists of a linear transformation followed by a translation:

$$x \rightarrow Ax + b$$

where A is a matrix and b is a vector. Affine transformations preserves:

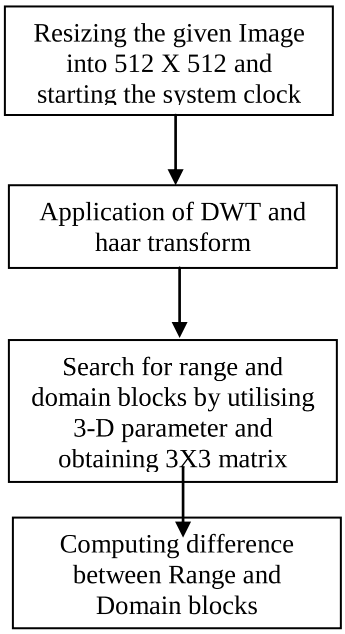
1. The co-linearity relations between point size three points which lie on a line continue to be collinear after the transformation.
2. Ratios of the distances along a line i.e., for distinct points.

5. The proposed 3-Dimensional Compression Algorithm

This section clearly describes the implementation of the newly proposed compression algorithm. The section includes steps to be performed during encoding and decoding process clearly. The encoding process involves discrete wavelet transform, Haar transform, 3-D matrix formation, saving difference between range and domain blocks, saving the obtained data in text file and calculating the compression time. Further the decoding process involves reading the text file, wavelet decomposition of constructed image, applying affine transformations, combination of wavelet tree, PSNR calculation, and Compression ratio calculation and displaying the decompressed image.

5.1 Encoding method:

The encoding method involves applying DWT and haar transform on the given image so as to extract pixel information. Thereafter the proposed 3-D algorithm is performed to extract range and domain blocks along with their differences between them. The last step to be performed is to save the extracted data in a simple text file. This text file can be transmitted further where it can be decompressed and the respective image can be regained in a small span of time. The flow chart as shown in Figure 4.1 shows the whole encoding process.



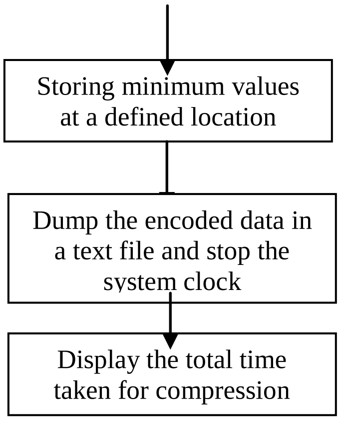
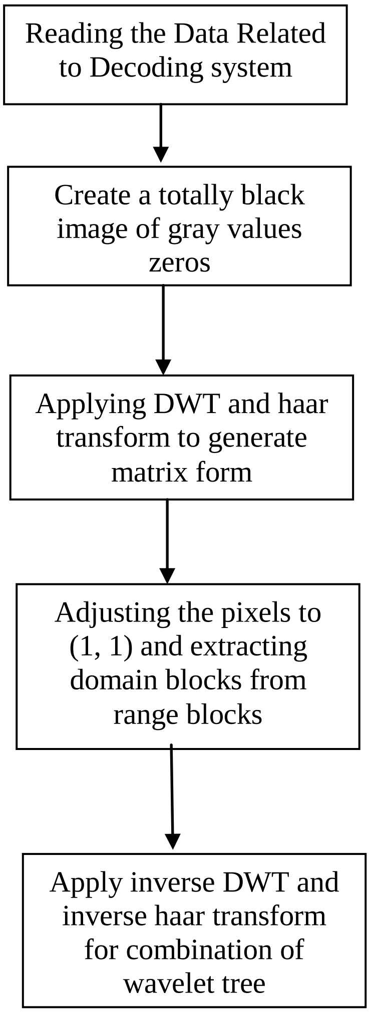


Figure 4.1: Encoding method for fractal compression

5.2 Decoding Method: The decoding method involves reading the encoded text file thoroughly and applying DWT and haar transform to generate matrix form. Thereafter performing affine transformation to adjust the pixel coordinates to (1,1) and to generate range and domain blocks and then utilizing inverse DWT in order to recombine wavelet tree so as to produce the compressed image and calculating PSNR and Compressed ratio calculation. The Figure 4.2 shows the steps performed during decoding process.



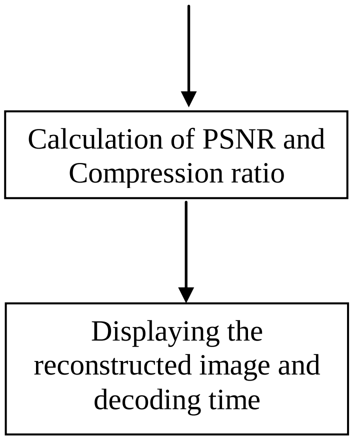


Figure 4.2: Encoding method for fractal decompression

5. Experimental Results and Analysis:

The proposed algorithm was designed and tested by performing the calculations on Matlab. The proposed 3-D parameter was tested on some sample pictures and different values were obtained and these are shown in the excel format in this paper.

After comparing the data received from calculations we concluded that the respective algorithm had compressed the pictures at a faster rate than the previously introduced algorithms studied by us. This was done by comparing the time taken by our algorithm with the previously introduced algorithms.

Figure 5(a): Original Image (lenna.tif)

Figure 5(b): Decoded Image (lenna.tif)

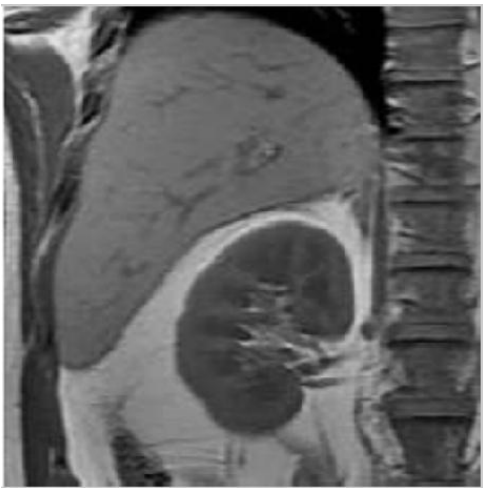


Figure 5(c): Original Image (image1.png)

Figure 5(d): Decoded Image (image1.png)

Figure 5(e): Original Image (photographer.bmp)

Figure 5(f): Decoded Image (photographer.bmp)

Sr .No.	Image Name	PSNR (db)	Compressed Ratio (%)	Compression Time (secs.)	Decompression Time (secs.)	MSE
1	lenna.tif	28.0086	85.59	80.8860	10.53	103.6600
2	image1.png	34.8234	84.5469	80.0120	11.51	21.5841
3	photographer.bmp	28.7311	86.546	81.8840	10.67	87.7744

REFERENCES

[1] Jie Yang., "Multiple Description Wavelet-Base Image Coding Using Iterated Function System," Volume 2013, Article ID 924274, 18 February 2013.

[2] M. I. Khalil , "Image Compression Using New Entropy Coder," International Journal of Computer Theory and Engineering, Vol. 2, No. , February, 2010.

[3] Eslami,R ; Radha,H., "Wavelet-based contourlet transform and its application to image coding," Image Processing, 2004. ICIP '04. 2004 International Conference, Volume:5, 24-27 Oct. 2004.

[4] A. Jacquin, Jan 1992. "Image coding based on a fractal theory of iterated contractive image transformations," IEEE Trans. Image Processing, vol. 1, pp. 18–30.

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

[6] A. Jacquin, 1990. "Fractal image coding based on a theory of iterated contractive image transformations," Proc. SPIE Conf. Visual Communications and Image Processing, pp. 227–239.

[7] D. Saupe, Mar 1995. "Accelerating fractal image compression by multidimensional nearest neighbor search," in Proc. Data Compression Conf., Snowbird, UT , pp. 222–231.

[8] B. Hurtgen and T. Hain, 1994. "On the convergence of fractal transforms," in Proc. ICASSP, vol. 5, pp. 561–564.

[9] J. Kominek, Mar 1995. "Convergence of fractal encoded images," in Proc. Data Compression Conf., Snowbird, UT, pp. 242–251.

[10] G.M.Davis, Feb 1998. "A Wavelet-Based Analysis of Fractal Image Compression" in IEEE Trans. Image Processing, VOL. 7, pp 141-154.

[11]Hannes Hartenstein, et al, Jul 2000. "Region-Based Fractal Image Compression" in IEEE Trans. Image Processing, VOL. 9, pp 1171-1184.

[12] Kamel Belloulata and Janusz Konrad, Apr 2002. "Fractal Image Compression With Region-Based Functionality" in IEEE Trans. Image Processing, VOL. 11, pp 351-362.

[13]Jyh-Horng Jeng, Chun-Chieh Tseng, and Jer-Guang Hsieh, May 2009. "Study on Huber Fractal Image Compression" in IEEE Trans. Image Processing, VOL. 18, pp 995-1003.

[14]Chong Sze Tong and Man Wong, Jun 2002. "Adaptive Approximate Nearest Neighbor Search for Fractal Image Compression" in IEEE Trans. Image Processing, VOL. 11, pp 605-615.

[15]C. M. Kung, W. S. Yang, C. C. Ku, C. Y. Wang,2008. "Fast Fractal Image Compression Base on Block Property" in Proc. International Conference on Advanced Computer Theory and Engineering pp 477-481.

[16] Xiao-Yan Xu, Philip Chen, Juan Dai, 12-15 July 2008. "Hybrid Encoding Analysis of fractal image compression method based on wavelet transform" in Proc. International Conference on Machine Learning and Cybernetics, Kunming.

[17]Venkata Rama Prasad, Vaddellal Ramesh Babu and Inampudi, Feb. 22-24, 2007 "Adaptive Gray Level Difference to Speed Up Fractal Image Compression" IEEE - ICSCN 2007, MIT Campus, Anna University, Chennai, India,pp.253-258.

[18] Nguyen Ky Giang, Dietmar Saupe, May 1997 "Adaptive Post-Processing for fractal image compression" IEEE proc. pp 183-186.

[19] J. Kominek, Mar 1995 "Convergence of fractal encoded images," in Proc. Data Compression Conf., Snowbird, UT, pp. 242–251.