Hybrid Fractal Image Compression Using Quadtree Decomposition with Huffman Coding

Shweta Pandey[1], Megha Seth[2]

[1]MTech Scholar, Department of Computer Science and Engineering, Rungta College of Engineering & Technology, Kohka, Bhilai, CSVTU University, Chhattisgarh, India

[2]Reader, Department of Information Technology, Rungta College of Engineering & Technology, Kohka, Bhilai, CSVTU University, Chhattisgarh, India,

Abstract: ***Fractal Image Compression is an approach for better image compression. The objective of this approached method is to introduce simplify and to provide better hybrid color image compression, which consist of Quadtree Decomposition and Huffman Coding. In this propose a method Fractal image compression is done by quadtree decomposition. In Quadtree method the block size varies according to the features of the image and separated into different blocks. After Fractal Image, Huffman Coding is applied in the compressed image that's how its performance will get improved. Different quantitative measures can be found by passing images of different format and dimensions, and on the basis of those quantitative measures which image format is better for compression.***

Keywords: Lossy data compression, Lossless data compression, Fractal image compression, Quadtree image compression, Huffman coding.

1. Introduction

In the past few years, the demand and development of multimedia products and storage capacity of storage device have increased at a tremendous rate. Due to which compression has become a key technique to reduce the size of number of bits as much as possible [1]. An inverse process known as decompression (decoding) can be applied to the compressed data to get the reconstructed image.

Image compression means reducing the size in bytes of a graphics file without hampering the quality of the images which reduce both spatial and spectral redundancy in the image data in order to store or transmit in an efficient form [2]. The process of Image compression is two types which are lossy and lossless compression method. Lossless compression is referred for archival purposes and often for medical imaging, technical drawings, and clip art. Lossy compression methods, especially when used at low bit rates, generates compression artifacts. The suitable methods for natural images such as cloud, tree, and mountain are named as a lossy method where imperceptible loss of fidelity is acceptable, to gain a substantial reduction in bit rate [4]. Most of the methods are used can be classified under the lossy compression. This means that the reconstructed picture is an approximation of the real picture [5].

Now a day, a new compression method is widely used for compressing image i.e. fractal image compression. It is useful in different application areas and research fields to compress the image. In this paper, the hybrid method of Fractal Image Compression is concentrated, which is an efficient method for lossy image compression that works on self-similarity property in various fractions of images. The main challenge today is FIC taking long encoding time and affecting the image quality, therefore, the recent research focused on achieving better PSNR value, higher compression ratio and tries to reduce long encoding time without hampering the quality of the image.

This methodology is related to a hybrid technology known as Hybrid Fractal Image Compression using Quadtree Decomposition with Huffman Coding. The Fractal image compression method is best suited for textures and natural images. The main idea is to decompose the image into segments by using standard image processing techniques [6]. The Quadtree decomposition is the use of fractal image compression. A quadtree decomposition method is used for the reduction of the search space and Huffman coding is used for improving the compression quality. Since fractal image compression takes more time for encoding the image therefore we tried to implement the above algorithm in order to get better encoding time, better PSNR value, higher compression ratio, and good quality of image. Also, we find some parameters through this methodology according to different dimensions by passing different format of images to it. After this compare all parameters according to their dimensions and image format, and then it is found the best format of the image is suited for this hybrid technology.

2. Fractal Image Compression

In our proposed methodology, Image Compression technique uses hybrid method. The Fractal Image Compression is suitable techniques for image compression. Benoit Mandelbrot first introduces the idea about fractal geometry in 1973; the fractal geometry has found self-similarity feature in an image. The idea of the self-similarity can be efficiently exploited by means of block self-affine transformations may call the fractal image compression (FIC). The fractal compression technique is based on the facts that are some images; parts of the images that possess some similarity with other parts of the same image. Amaud Jacquin and Michal Barnsley introduced an automatic fractal encoding system in 1989 [7, 8]. In 1990, Jaquin was first suggested block forming technique for Fractal image compression method.

In fractal image compression main limitation is high encoding time because of exhaustive search technique. Therefore, decreasing the encoding time is an interesting research topic for FIC. Through this proposed hybrid method, also trying to decrease the encoding time. Fractal compression is very efficient because of its high Compression ratio. In this fractal image compression decoding stage takes less time for decompression the image but encoding stage takes more time to compress the image. Fractal image compression is also called as fractal image programming where compressed images are represented as contractive transforms [9]. A contractive mapping is a mapping of the source image through a series of transformations such as scaling, translation, rotation. The mappings are contractive because when the transformation is applied, the points on the plane are brought closer together. Fractal compression is a lossy compression method used in digital images, based on fractals. The method is based on the fact that parts of an image often resemble with other parts of the same image. Fractal algorithms convert these parts into mathematical data known as "fractal codes" which are used to recreate the encoded image [10].

In fractal compression firstly Image is divided into a number of square blocks called range, later the image is divided into bigger square blocks, called domain blocks, which are usually four times larger than the range block [12]. After that, the domain blocks are searched for the best match for every range block. For every range block the number of the appropriate domain and relevant information needed to retrieve that range are stored. The fractal affine transformation is constructed by searching all of the domain block to find the most similar one and the parameters representing the fractal affine transformation will form the fractal compression code. Hence the compression is achieved in place of storing a range block only the parameters are stored. The decoder performs a number of iterative operations in order to reconstruct the original image.

3. RGB Color Model

Previously, RGB color model had been used in visual display devices such as computer monitor, CRT monitor. Complete 24-bit used in the RGB color model each color consist of 8-bits. The RGB color space is denoted by three dimensions, single axis for each of the colors. A huge range of colors can be duly represented with the RGB model - it can be seen that red and blue gives magenta, while blue and green gives cyan, etc. Within the cube, greyscale shades are also represented, and it runs diagonally between the black and white corners of the RGB cube, where each color consists of an equal amount of red, green and blue.

4. Quadtree Partitioning

The IFS method, although offering the potential of efficient compression ratios and relatively rapid decode, is degraded by the difficulty of addressing the Inverse Problem and searching the mappings required to show a target image. These failings define the necessity for image compression algorithm to be identifiable - although it is quite easy to accept that it is difficult to known the compression ratio that will be gained by an algorithm for an image, it is must that we maintain a fine image quality level and that the compression can deliver this in a reasonable amount of computing time [11].

Quadtree-based Fractal image compression means for non deterministic nature of IFS compression techniques, though as a result the compression ratios achieved. However, quadtree compression can be used to any kind of image and implementations deliver better compression ratios. Basic quadtree techniques divide the target image into 4 squares, known as Domains and also divide the image into 16 squares, known as Ranges. The algorithm then attempts to cover every range with a domain, using a contractive mapping, where the fitness of the range/domain map is maximized. If a range is failing to find a match, the process is repeated after partitioning that particular range block into four quadrants [12]. Quadtree compressed images also estimate other fractal properties, specifically the ability to convert a stored image at a higher resolution than the original and have the rendering algorithm to 'scale' the image. While scaling technique doesn't provide extra data that is true to the original image's scene. As this stage is responsible for the removal of spatial redundancy, the performance of the quadtree image compression element of the system under evaluation is critical [13]. In order to examine images in a mathematical setting an appropriate space must be chosen. Once the space in which images live is properly defined we can apply the contraction mapping theorem for suitably defined functions to see when points of this space converge.

5. Huffman Coding Technique

Huffman code is a technique for compressing data. Huffman's greedy algorithm converts each character as a binary string in an optimal way. Huffman coding is a form of statistical coding, which is used to reduce the amount of bits required to represent a string of symbols. The algorithm accomplishes its goals by allowing symbols to vary in their length [14]. It allows variable length symbols is to be assigned for longer codes which appear less frequently and shorter codes are assigned for most frequently used symbols. Code word lengths are no longer fixed like ASCII.

5.1 Huffman Encoding

The Huffman code procedure is based on the two rules [14]. First, more frequently occurred symbols will have shorter code words than symbol that occur less frequently. The two symbols that occur less frequently will have the same length. Second, The Huffman code is designed by merging the lowest probable symbols and this process is repeated until only two probabilities of two compound symbols are left and thus a code tree is generated and Huffman codes are obtained from labeling of the code tree. So, the Huffman encoding algorithm starts by constructing a list of all the alphabet symbols in descending order of their probabilities. Huffman's procedure creates the optimal code for a set of symbols and probabilities subject to the constraint that the symbols be coded one at a time [15].

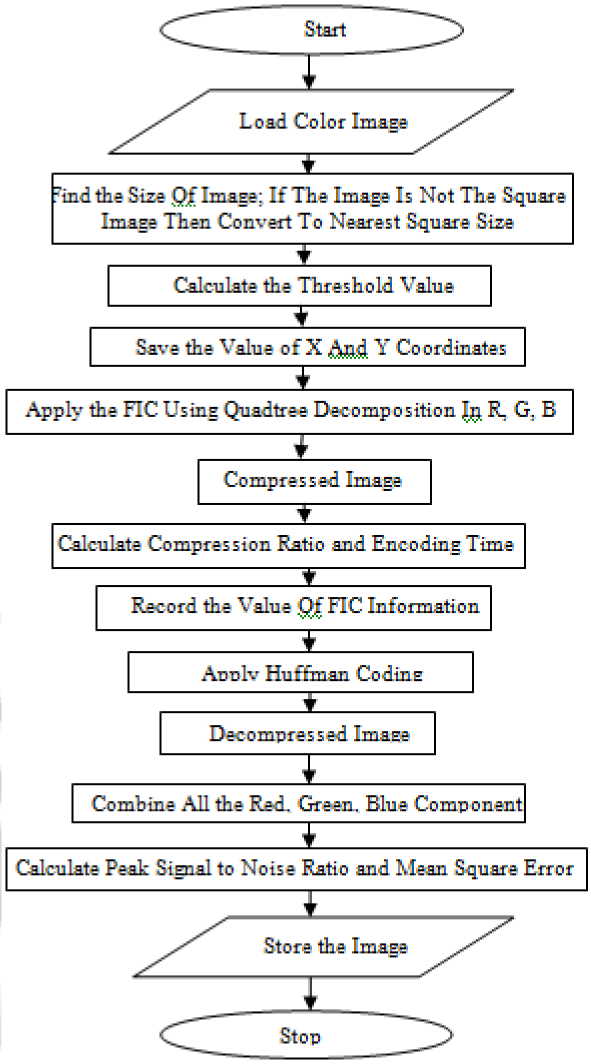
5.2 Huffman Decoding

Decoding of encoded image is simple. Start from the root and read the first bit of compressed file in the left to right manner [16]. When the decoder reaches at a leaf, it finds there the original, uncompressed image, symbol and that code is reflected by the decoder. The process initiated again at the root with the next bit,then it read and can decode the rest of its input. It is basically based on the symbols, i.e. the probability of the symbols. The probabilities must be written, as sub information, on the output [15]. The block code is a unique code itself. The block code refers to that each source symbol is mapped into a sequence which is fixed, order of code symbols. It is instantaneous, for the reason that each code word in the series of code symbols can be decoded without taking reference of the succeeding symbols. It is decodable uniquely, because any series of code symbols can be decoded in only one way [17].

6. The Proposed Algorithm

The algorithm steps are as follows.
1. Read the Input Color image.
2. Find the Size of Image; if the image is not the square image then convert to nearest square size.
3. Calculate the threshold value.
4. Apply the Fractal image compression using quadtree decomposition.
5. Record the fractal coding information then apply Huffman coding.
6. For the encoding image applying Huffman decoding to reconstruct the image.
7. Calculate the compression ratio, PSNR, MSE and Total Time for compression.

Flow diagram shows the proposed method



7. Result and Discussion

In this paper, we are working on color images; we are applying different format and dimensions of images to our method. So, using proposed methodology we tried to achieve less encoding time, high compression ratio, better PSNR value, and less MSE value in different format and dimensions of images.

Peak Signal-to-Noise Ratio (PSNR):
PSNR is used to measure the quality of reconstruction of lossy compression. It is the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation [10], as in (1). Where R is maximum fluctuation of input image data type.

$$PSNR = 10\log_{10}\left(\frac{R^2}{MSE}\right) \quad (1)$$

Compression Ratio (CR):
Data compression ratio is defined as the ratio between the uncompressed size and compressed size.
Cr= Uncompressed image size/Compressed image size.

Encoding Time:
Encoding is the process of putting a Sequence of characters into a special format for transmission or storage purposes.

Mean Square Error (MSE):
The MSE is the cumulative squared error between the compressed and the original image as in (2).

$$MSE = \frac{1}{MN}\sum_{y=1}^{M}\sum_{x=1}^{N}[I(X,Y) - I'(X,Y)]^2 \qquad (2)$$

Where I(x, y) is the original image, I'(x, y) is the approximated version (which is actually the decompressed image) and M, N are the dimensions of the images.

The Table shows the different format of images according to parameter and also compares which format and dimensions of images are best for our method.

Table 1: Encoding time (sec)

Dimensions	*Lena (JPEG)*	*Rafting (png)*	*Img (bitmap)*	*Autumn (tif)*
2 and 64	222.859	165.276	386.335	237.906
4 and 16	87.524	65.009	127.153	90.238
2 and 4	313.155	297.941	477.737	347.606
2 and 16	223.339	173.954	382.799	245.561
4 and 64	86.612	60.7923	126.341	85.165

Table 2: Compression Ratio

Dimensions	*Lena (JPEG)*	*Rafting (png)*	*Img (bitmap)*	*Autumn (tif)*
2 and 64	33.109	17.552	3.227	12.341
4 and 16	27.561	38.058	23.678	29.261
2 and 4	9.006	10.056	6.690	8.566
2 and 16	12.607	17.150	8.202	12.174
4 and 64	27.693	39.927	23.849	30.188

Table 3: Peak Signal to Noise Ratio(dB)

Dimensions	*Lena (JPEG)*	*Rafting (png)*	*Img (bitmap)*	*Autumn (tif)*
2 and 64	12.641	32.94	26.60	32.41
4 and 16	29.927	30.4413	22.441	29.024
2 and 4	34.023	33.8592	26.637	32.901
2 and 16	33.141	33.0571	26.606	32.511
4 and 64	29.912	30.3777	22.440	22.440

Table 4: Mean Square Error

Dimensions	*Lena (JPEG)*	*Rafting (png)*	*Img (bitmap)*	*Autumn (tif)*
2 and 64	0.32	0.33	1.42	0.37
4 and 16	0.66	0.59	3.70	0.81
2 and 4	0.26	0.27	1.41	0.33
2 and 16	0.32	0.33	1.42	0.36
4 and 64	0.66	0.60	3.70	0.82

From the above table, the best results obtain for the type of image-rafting.png and Lena.jpeg image, which posses the best value among all the rest. Below Fig(1) show the original image and Fig(2,3,4,5,6). Shows that the different dimension and formats of images.

Figure 1: Original image

Figure 2: Compressed Images: Dimension (2 and 4)

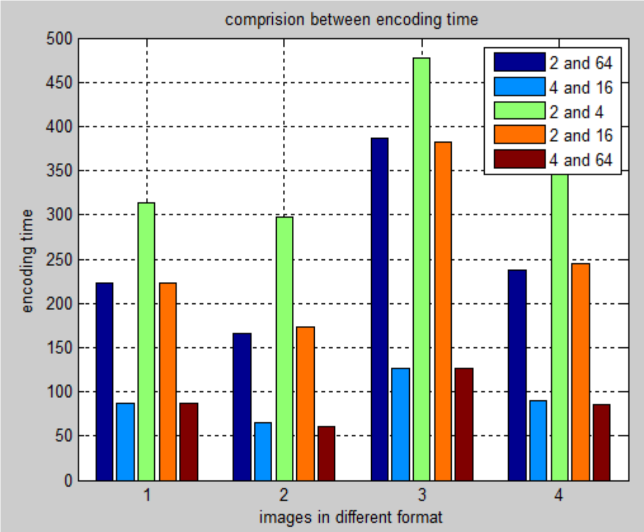
Figure 3: Compressed Images: Dimension (2 and 16)

Figure 4: Compressed Images: Dimension (4 and 16)

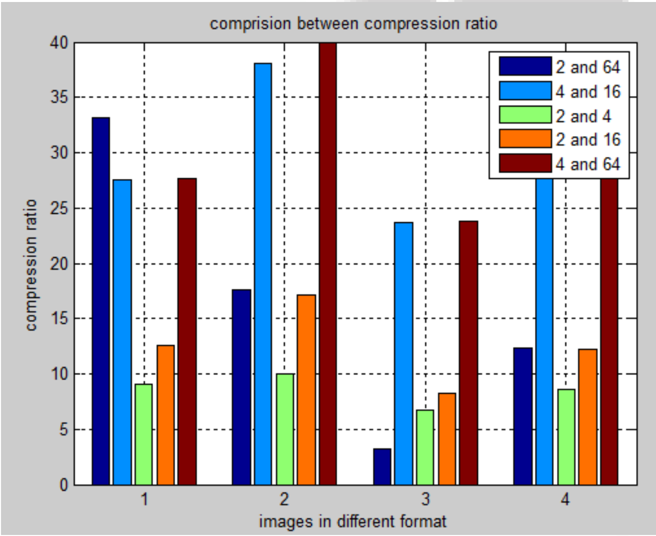
Figure 5: Compressed Images: Dimension (4 and 64)

Figure 6: Compressed Images: Dimension (2 and 64)

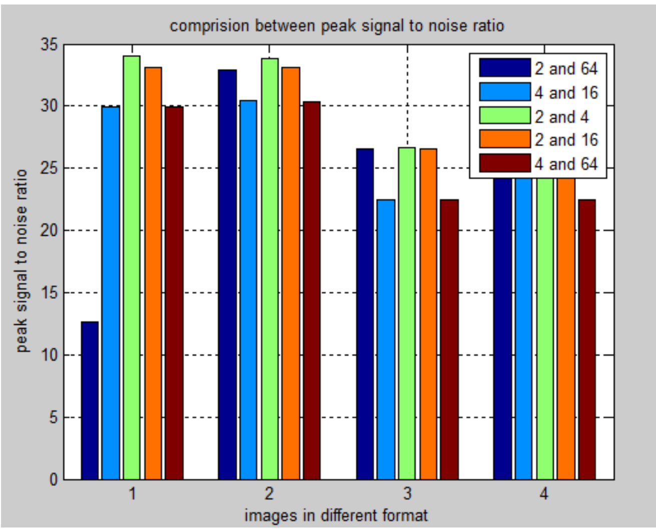
According to Compressed Image result graph variation is as shown below Graph (7, 8, 9, and 10)



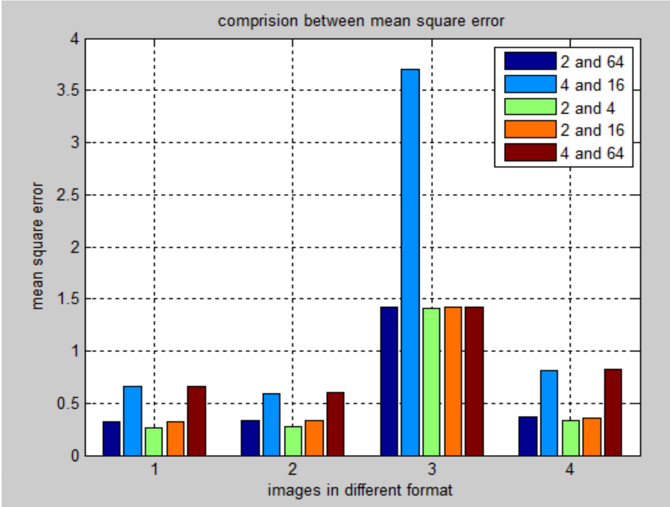
Graph 7: Comparison between encoding time according to dimensions and different format of images



Graph 8: Comparison between Compression ratio according to dimensions and different format of images



Graph 9: Comparison between PSNR according to dimensions and different format of images.



Graph 10: Comparison between MSE according to dimensions and different format of images

Comparative analysis process evaluates the performance of each dimension and acquires which format of image provides the highest performance in fractal image compression, in a graph horizontal axis represent the "Image Format" and vertical axis represent the parameters such as encoding time, compression ratio, PSNR, MSE (shown Graph 7, 8, 9, 10). Above graph 7 shows, for all dimensions Png and jpeg image format take less encoding time from the other image formats. Graph 8 shows, for all dimensions Png and jpeg image format get higher compression ratio from the other image formats. Graph 9 shows, for all dimensions Png and jpeg image format get better PSNR value from the other image formats. Graph 10 shows, for all dimensions Png and jpeg image format get lower MSE from the other image formats, but it also shows that MSE for bitmap image format is greater than one as per the result this method is not suitable for bitmap image format.

8. Conclusion and Scope

In fractal image compression the block size play a very important role. The quality of image and time is depended on the block size according to their dimensions and threshold value. Images of different dimensions and formats are used in hybrid technique and from the above experiment, it is obtained that this method is optimum for png image and jpeg image. The above table shows the various calculated data from different images with the help of Quadtree and Huffman coding, and this gives the best result according to the compression between different images chooses. Also find out the quantitative measures, that are used for comparison of quality of images are PSNR value, MSE, Encoding time. As per analyzed and calculated data we obtained that the png and jpeg image work well with this proposed method but this method is not suitable for bitmap image format. So, this method is done for color images that take more encoding time but achieve high compression ratio and better PSNR value.

The future Scope of this methodology is that the time will be reduce by some advance technique (neural network, artificial intelligence, fuzzy logic) and the threshold value can be calculated automatically by the different thresholding method (HSV and wavelet transform).

References

[1] Sonal, Dinesh Kumar, "A study of various image compression techniques," Guru Jhambheswar University of Science and Technology, Hisar.
[2] Chuanwei s, quanbin, jingao, "The study of digital image compression based on wavelets," 2010.
[3] Ying chen, pengwei hao, "integer reversible transformation to make jpeg lossless," Peking university, Beijing, 2004.
[4] J. Ziv, A. Lempel, "Universal Algorithm for Sequential Data Compression", IEEE Transactions on Information Theory, Vol. 23, No. 3, pp. 337-343, 1977.
[5] Chakrapani, Soundara Rajan, "Genetic Algorithm Applied to Fractal Image Compression," ARPN Journal of Engineering and Applied Sciences, Vol. 4, No. 1, pp. 53-58, 2009.
[6] Dr. K. Kuppusamy, R.Ilackiya, "Fractal Image Compression & Algorithmic Techniques," International Journal of Computer & Organization Trends, Vol. 3 Issue4 – May 2013.
[7] Selim, A., M. Hadhoud, M.O. Salem, "A Comparison Study between Spiral and Traditional Fractal Image Compression," 2002.
[8] Mallikarjuna Swamy, M.L., K. Nagamani, "Fractal image Compression Technique Using Fixed Level of Scaling," 2012.
[9] Kumaravel a., "an application of non-uniform cellular automata for efficient cryptography," pp 1200-1205, 2013.
[10] Aoued BOUKELIF, "Optimization of Fractal Image Compression Based on Genetic Algorithms," SETIT, 2009.
[11] G.J. Sullivan R.L. Baker, "Efficient quadtree coding of images and video," IEEE Trans. Image Proc., vol. 3, no. 3, pp. 327-331, May, 1994.
[12] Veenadevi. S.V., A. G. Ananth, "Fractal image compression using Quadtree decomposition and Huffman coding," Bangalore, SIPIJ, Vol.3, No.2, April 2012.
[13] James Halliwell, "An Investigation into Quadtree Fractal Image and Video Compression," 2006.
[14] Mridul Kumar Mathur, Seema Loonker, Dr. Dheeraj Saxena, "Lossless huffman coding technique for image compression and reconstruction using binary trees," IJCTA,Vol 3 (1), pp. 76-79, 2012.
[15] Parvinder Kaur, "Compression Using Fractional Fourier Transform".
[16] jagadish h. pujar, lohit m. kadlaskar, "A new lossless method of image compression and decompression using huffman coding techniques"
[17] Manoj aggrawal, Ajai narayan, "Efficient huffman decoding".

Author Profile

Shweta Pandey received the B.E. degree in Computer Science and Engineering from Chhattisgarh Swami Vivekanand Technical University, Bhilai, in 2012 and currently pursuing M. Tech. in Computer Technology from Rungta college of Engineering and technology Kohka Bhilai, under Chhattisgarh Swami Vivekanand Technical University, Bhilai Chhattisgarh India. Her research area of interests includes Digital Image Processing.

Ms. Megha Seth received her M. Tech degree in Computer Science and Engineering in 2012. Currently working as a Reader in the Department of Information Technology at Rungta College of Engg and Technology (RCET), Kohka Bhilai, Chhattisgarh, India.