IJCTA, 9(17) 2016, pp. 8773-8784

Fractal Compression of Colored Image Based on Fuzzification in YIQ Color Space and Equilateral Triangle Partition Approach

Shimal Das[*] and Dibyendu Ghoshal[**]

ABSTRACT

The study deals with fractal based compression of colored images in YIQ space. Fuzzification is carried out on the in phase I-component of chroma part and subsequently defuzzication is done after fractal compression is carried out Luma (Y) part and Quadrature (Q) components of chroma part are used in separate channels to augment the compression process. The outcome is obtained by summing the results obtained from three channels separately. For partitioning purposes during iterative processes, equilateral triangle geometry has been followed. The proposed methodology has been utilized on six test images viz. Lena, Tower, Rose, and Boat, Peppers and, Leopard images and it has been found that the proposed method yields better figure of merits compared to earlier reported methods.

Keywords: Fractal compression, YIQ colour space, Affine Transformation, Iterated Function System (IFS)

I. INTRODUCTION

This Fractal geometry has occupied a significant place in the field of segmentation and compression of digital images. Despite a highly lossy compression technique, it can yield infinite compression ratio, at least theoretically [1], as against the reported 1000:1 is a common one [2]. The process of fractal compression depends on partitioning of image plane into various ranges (smaller) and domain (larger) blocks and there has to be a mapping relation between the two blocks. The number of domain blocks should be as low as possible to ensure better compression and faster processing time. During matching process followed by compression, affine transformation has to be done. The collage theorem has to be maintained for carrying out the transformation. The rotation, translation, skewing are essential parts of affine transformation [2-6]. The partitioned area geometry may be rectangular or of any other shape like triangle, etc. Normally rectangular blocks are used for quad-tree approach where the image area is consecutively divided by two to reach the range block and then a matching is done between range and domain blocks.

Colored images are used for image transmission, internetworking, cellular phones, video and computer monitors. These are various color schemes used in different purposes. R.G.B color scheme is widely prevalent in spite of red, green and blue colors are correlated and color and intensity components are intermixed. YIQ color schemes are used in high definition television HDTV [7].

In the present study, in phase component of the color (chroma) component is followed before starting the fractal compression processing. The intensity part and out of phase quadrature (Q) component are normally processed for fractal processing. Equilateral triangle based geometry has been utilized in the present study instead of quad tree approach. It has already been reported [8] that an isosceles triangle based approach yields better compression ratio and higher PSNR for the same image compared to those obtained by using quad tree approach. In the proposed approach, the image plane is sub-divided into equilateral triangle for getting the domain and range blocks.

[*] Department of Computer Science and Engineering, Tripura Institute of Technology, Narsingarh, India, *E-mail: shimalcs.tit@gmail.com*

[**] Department of Electronics and Communication Engineering, National Institute of Technology, Agartala, India, *E-mail: tukumw@gmail.com*

Although a large numbers of studies have been made on fractal geometry based image compression by various researchers [9-21], no study is found to deal with the YIQ color space based images with fuzzification of in phase part of the color component of the image signal. The study has shown that a higher compression ratio, smaller execution time and higher PSNR values on six different text images viz. Lena, Rose, Tower, Boat, Peppers and Leopard.

Images are acquired not only for storage but also for transmission. To achieve smaller transmission time, the number of bits related to the image should become less and it requires image compression. Fractal based image compression yields a higher compression ratio leading to lower number of bits. The present study in YIQ scheme with fuzzification serves that purpose. Regarding the short coming of the present process, fractal based compression is inherently a lossy one and it results in losses of bits which may contain some important and significant area of the image.

The remaining part of the paper has been represented as follows. Section 2 describes mathematical background and some basic terms. Section 3 deals with the proposed algorithms and flowchart of the methodology with the necessary description of the same. Section 4 has shown the implementation, Results and discussions whereas section 5 has stated the conclusion.

II. MATHEMATICAL BACKGROUND AND SOME BASIC TERMS

Fractal geometry based compression involves various properties related to image structure, geometrical transformation and coding methods. All these aspects are described below in a consize manner.

2.1. Self-Similarity

A self- similar item is precisely or roughly similar a piece of itself (i.e. the entire has the same shape as one or a greater amount of the parts). Numerous items in this present reality, for example, coastlines, are factually self- similar: parts of them demonstrate the same measurable properties at numerous scales. Self- similarity is an ordinary property of fractals. Scale invariance is a careful type of self- similarity where at any transformation there is a little bit of the item that is similar the entirety. For example, a side of the Koch snowflake is both symmetrical and scale-invariant; it can be consistently transformed without evolving shape. The non-inconsequential comparability clear in fractals is recognized by their fine structure, or detail on discretionarily little scales. As a counter example, while any segment of a straight line may take after the entire, further detail is not uncovered.

Subsets of fractals when amplified seem comparable or indistinguishable to the first fractal and to different subsets. This property is called self- similarity and it makes fractals autonomous of scale and scaling strategies. In this way there is no trademark size connected with a fractal. A run of the mill image does not contain the kind of self- similarity which is found in fractals. In any case, it contains an alternate kind of self- similarity. The figure 1 indicates districts of Lena that are self- similarity at various scales. A segment of her shoulder covers a littler locale that is practically indistinguishable, and a bit of the impression of the cap in the mirror is like a little piece of her cap.

The distinction here is that the whole picture is not self- similar, but rather parts of the picture are self-similarity with appropriately transformation parts of itself. Most normally happening pictures contain this kind of self- similarity. It is this confined repetition that fractal image compression plans endeavor to wipe out.

2.2. Iterated Function System and Attractor

A contractive mapping dependably unites to an interesting altered point regardless of the shape and size of starting picture, an element of two variables, taken for preparing. On the off chance that such mappings are connected to a picture then the final picture that we get is called Attractor. Like the one of a kind estimation

Figure 1: Demostrates the self similarity in Lena [22]

of a grouping, the state of attractor is free of the state of beginning picture yet is needy upon their position and introduction. It diminishes the underlying picture significantly. It puts the three duplicates of the picture in a triangular design. There is a procurement of giving the yield of one duplicate operation as the contribution for the second duplicate operation i.e. machine can work in an iterative style. Figure 2 demonstrate the contracting transformations by iterated function system.

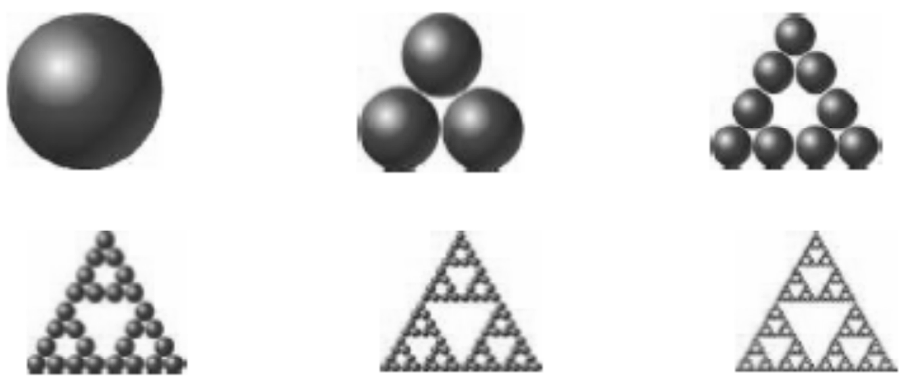

Figure 2: Demonstrates thee contracting transformations by Iterated Function System [22]

Iterated Function System (IFS) is a gathering of contractive mappings .Let IFS be meant by W and the accumulation by $w_1, w_2, w_3, \ldots\ldots\ldots\ldots, w_n$. (1)

$$W(X) = w_1 (X) \cup w_2 (X) \cup w_3(X) \ldots\ldots \cup w_n(X) \quad (2)$$

Here X is the grayness level of the picture in which the change is used. In the event that w's are contractive in plane then W will be contractive in space. Give X_w a chance to be an attractor came about by the rehashed utilizations of W on an underlying image, then in the wake of applying further change it will focalize to X_w as it were.

2.3. Collage Theorem

Image compression based on fractal utilizing the iterative function system (IFS) depends on the collage theorem first introduced by Barnsley et al [3]. This technique, the errors between the recreated image and unique original image is more prominent than the errors between the collage and original picture in light of the fact that the collage theorem does not ensure the previous errors to be littler than the last errors. Utilizing this collage theorem to help you discover an IFS comprising of two affine maps in R^2 whose attractor is near this set:

Consider a grayscale image f, and the contractive transformation W such that

$$d_2\ (f, W(f)) \leq \in$$

$$\text{and } d_2\ (f, f_w) \leq \frac{\varepsilon}{1-\varepsilon} \tag{3}$$

Here ε is the contractivity factor of W, f_w is the fixed point and ε be > 0. It implies that one can start with any image g and iterate W^{on} (g) to get an image that is nearest to f.

$$W^{on}(g) \rightarrow f_w \approx f. \tag{4}$$

Collection Theorem conveys one stage towards to fractal picture encoding. Translating (decoding) comprises of iterating W^{on} any beginning picture g to reform g to recuperate f_w.

2.4. Affine Transformation

The transformations utilized by Barnsley [3] for iterative function systems are called affine transformations. It is a direct mapping technique that preserves focuses, straight lines, and planes, collections of parallel lines stay parallel after this transformation and normally utilized to redress for geometric contortions or distortions that happen with non-perfect camera points. This transformation incorporate interpretation, scaling, homothetic, comparability change, reflection, turn, shear mapping, and syntheses of them in any combination and sequence.

Consider an affine transformation T: $R^2 \rightarrow R^2$ is a transformation of the structure. This transformation maps a plane to itself. The general type of an Affine Transformation is

$$W\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} \text{........} \tag{5}$$

Here a, b, c, d, e, f are indicates scaling, homothetic, comparability change, reflection, shear mapping, and contraction.

Let $\tilde{w}_i(x, y)$ be an affine transformation on $I^2 \rightarrow I^2$, then

$$\tilde{w}_i(x, y) = A_i\left(\frac{x}{y}\right) + b_i \tag{6}$$

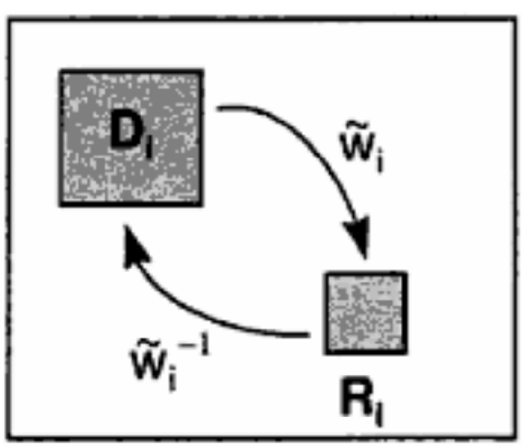

Figure 3: Demonstrates the spatial affine transformation and its reverse

for some 2 × 2 matrix A_i and 2 × 1 vector b_i

Let $D_i \subset I^2$ be few sub domain of the unit square I^2, and let R_i be the range of $\tilde{w}_i$ operating on D_i, that is $\tilde{w}_i$ $(D_i) = R_i$.

one can define w_i: F→F operating on images f(*x*, *y*) by

$W_i(f)(x,y) = s_i\, f(\tilde{w}_i\,(x,y)) + o_i$ provided $\tilde{w}_i$ is invertible and (x, y) € R_i, s_i controls the contrast and o_i controls the brightness. Figure 3 demostrates the affine transformation and its reverse system.

2.5. Fractal Image Coding Method

Method of fractal geometry is invented on the basis of self-similarity [3], [5] of object of any arbitrary shape. The fractal coding is accomplished in digital image at microscopic and block level. Fractal coding is somewhat different from other transformation codes where a single diversion worthy transform maps the image bits to another set of uncorrelated coefficients out of which only dominant codes are retained for subsequent processing. Range blocks have smaller size and at the initial size, domain blocks have double the size of the range block. The searching is made get the best matching between range and domain blocks. This is advanced through performing a size of transformations on the blocks such that a given figure of merits becomes less, such a root means square deviation during iteration. Data compression is obtained by storing only the set of transformations, i.e. fractal code containing the total information needed for reconstruction of an image and thus is done by iterating the set of transformations on an arbitrary starting image.

Decoding is advanced by iterating the set of transformations on an arbitrary starting image and again range blocks are obtained by applying quadtree approach. Here the size of domain block mapping to it is shrunk by two through 2×2 pixel average process. The pixel values of the shrunken domain block are that set into location in the range determined by the orientation information often scaling and off selling are done. After, a number of iterations covering all the range blocks, the reconstructed image normally become nearby identical with the original image.

2.6. The Conversion Formula between YIQ Color Space and R′-G′-B′ (Gamma Correlated RGB Space)

YIQ color scheme is characterized by National Television System Committee (NTSC). In this scheme, Y speaks to the luminance; and I and Q portray the chrominance. The change recipe between YIQ color scheme and R′-G′-B′ (gamma associated RGB space) is given as takes after:

$$Y = 0.299\,R' + 0.587\,G' + 0.114\,B' \quad (7)$$

$$I = 0.596\,R' - 0.274\,G' - 0.322\,B' \quad (8)$$

$$Q = 0.211\,R' - 0.523\,G' + 0.311\,B' \quad (9)$$

And

$$R' = Y' + 0.956\,I' + 0.621 - Q' \quad (10)$$

$$G' = Y' - 0.272\,I' - 0.647 - Q' \quad (11)$$

$$B' = Y' - 1.105\,I' + 1.702 - Q' \quad (12)$$

III. PROPOSED METHOD

YIQ based Fractal image compression utilizing fuzzification with equilateral triangle segmentation methodology are the proposed technique. Studies are separate as

First; the information RGB image is changed over into YIQ scheme.

Second; Fizzification connected on changed over I segment.

Third; Divide I segment (fuzzified), Y and Q segments into non-overlapping l extent range and domain blocks based on equilateral triangle division. The sizes of the domain blocks have at any rate double the extent of the range blocks.

In equilateral triangle division, the image is partition as range and domain blocks based on equilateral triangle. Fig. 4 demonstrates the range blocks, domain block and range and domain block segmentation technique. This division technique is agreeable to drawing nearer the inclining edge and utilizing the self-similarity property of the image which remakes the edge data of original image.

From fig. 4(a), seems the non-overlapping range blocks range blocks $R_1, R_2, \ldots, R_n$ is 4, and $R_1, R_2, \ldots, R_n$ cover the whole image.

Figure 4(b) and 4(c) are the overlapped domain block divisions. As it were, number of pixels between two nearby domain blocks will turn out as L. Number of pixel of domain blocks is eight times more than the range block. The progression length of area square would impact the proficiency and picture nature of fractal coding straightforwardly. The shorter the progression length, the all the more coordinating times between domain and range blocks is required. Be that as it may, over-length step may bring about crisscrossing between blocks prompting the decreasing of the image quality.

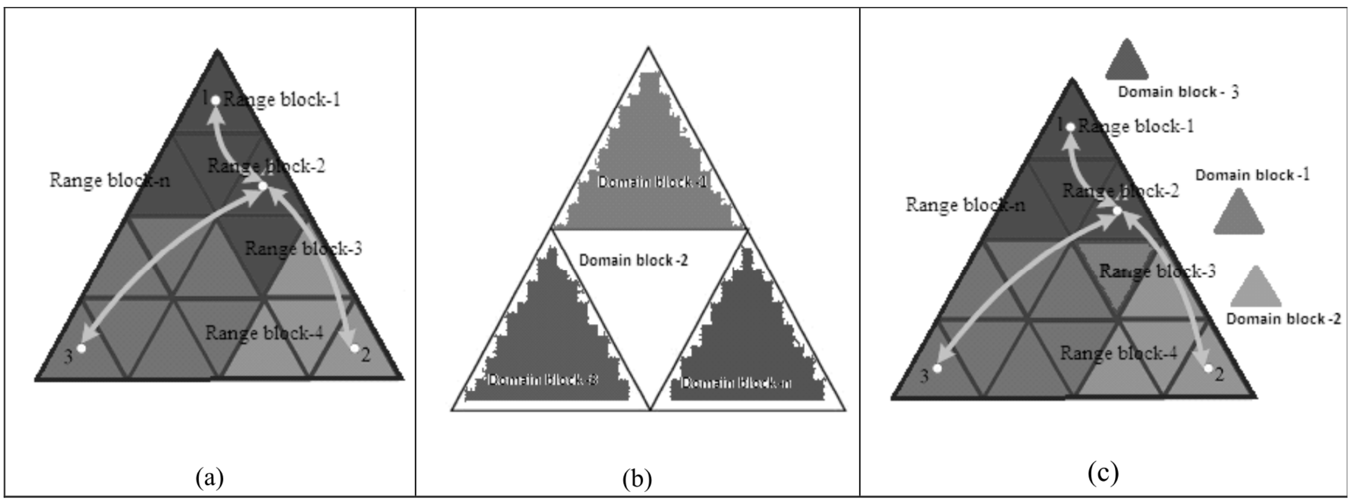

Figure 4: (a) Demonstrates the range block partition, (b) Demonstrates the domain block partition, (c) Partition methods of range blocks and domain blocks [22]

For searching the best matching domain block, each domain block needs to various transformation as isometric transformation, shrink transformation, brightness transformation and brightness excursion as demonstrates in figure 5. When best match domain block of current range block is found after searching and comparing range blocks with transformed domain blocks. Then write the IFS code for corresponding coefficient of affine transformation, which construct the fractal code of current range block.

Fourth; A contractive affine transformation mapping of domain blocks D into the range blocks R, whereas every range block, search a corresponding best matching domain block. If the best match is not found, then repeat again, then write IFS code for corresponding block.

Fifth; Then utilizes a fractal data compression techniques to get the compressed IFS code.

Sixth; Defuzzification of I component (fractal compressed IFS code) and adding of I component (defuzzified data) with compressed Y and Q components to get the compressed image.

Figure 6 demonstrates the entire fractal image compression procedure based on fuzzification in YIQ color space and equilateral triangle partitioning approach.

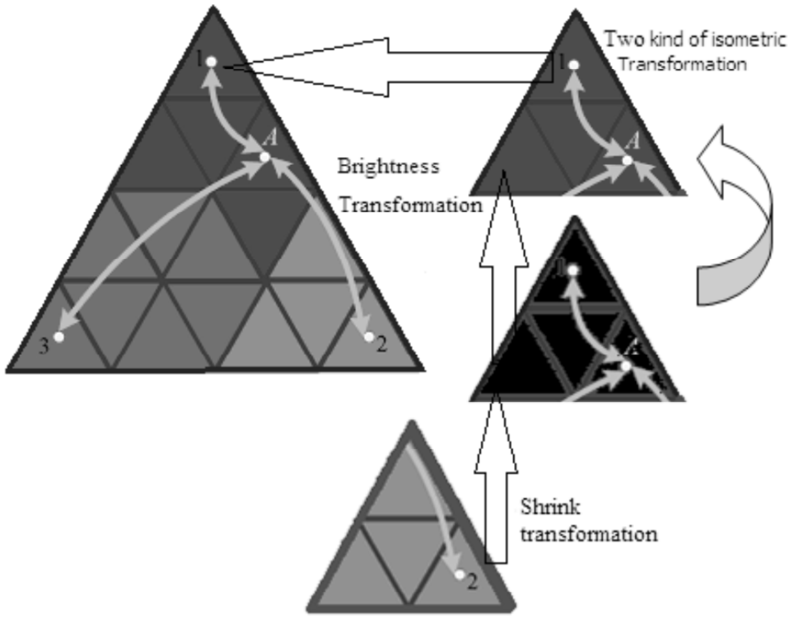

Figure 5: Demonstrates the various transformation of matching procedures [22]

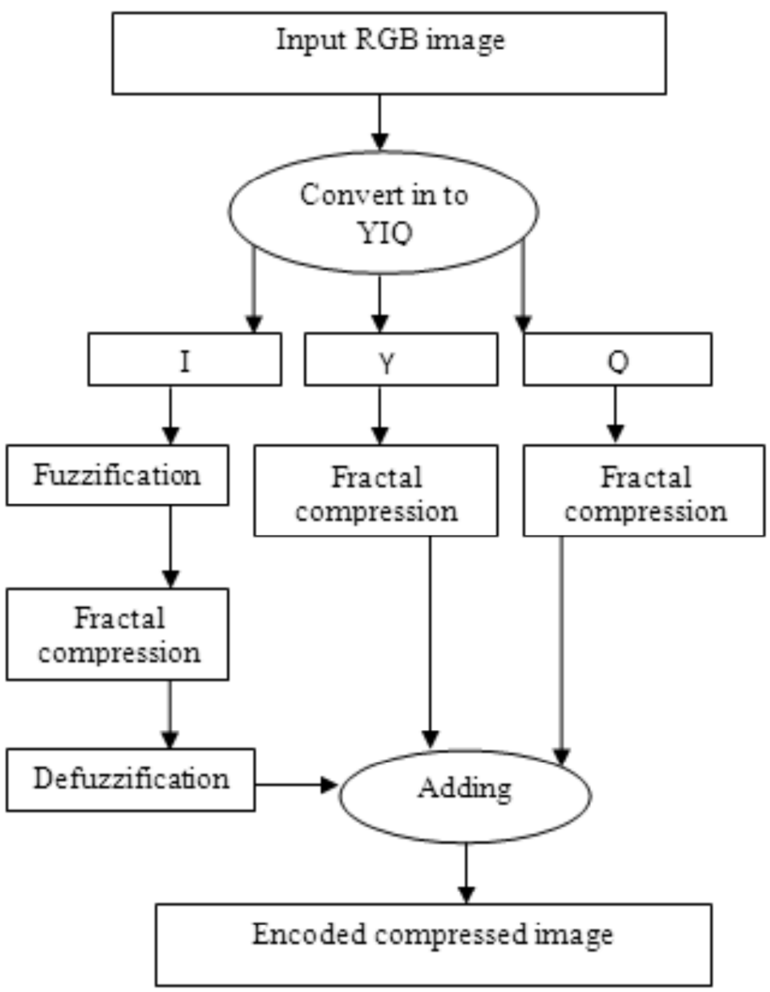

Figure 6: Demonstrates the fractal image compression procedure based on fuzzification in YIQ color space and equilateral triangle partitioning approach

Proposed compression algorithm as below-

Step 1: Input RGB image

Step 2: Convert RGB to Y, I, Q component

Step 3: Fizzify I component

Step 4: Divide I component (fuzzified), Y and Q components into non-overlapping range and domain blocks based on equilateral triangle segmentation. The size of the domain blocks at least twice the size of the range blocks

Step 5: Utilizing a set of affine transformation mapping of domain blocks D into the range blocks R.

Step 6: For every range block, search a corresponding best matching domain block. If the best match is not found, then repeat from Step 4 to Step 6, then write the compressed data in the form of a local IFS code.

Step 7: Then utilized a fractal compression techniques to get the compressed IFS code.

Step 8: Then defuzzification of I component (fractal compressed IFS code)

Step 9: Adding of I component (defuzzified data), Compressed Y component and Compressed Q component and get the compressed image.

IV. IMPLEMENTATION, RESULTS AND DISCUSSIONS

Proposed method is implemented and tested the simulation results for compression of colored image based on fuzzification in YIQ color space and equilateral triangle partition approach. The simulation results are carried out Core I3 processor, 3 GB RAM and Windows 7 operating system.

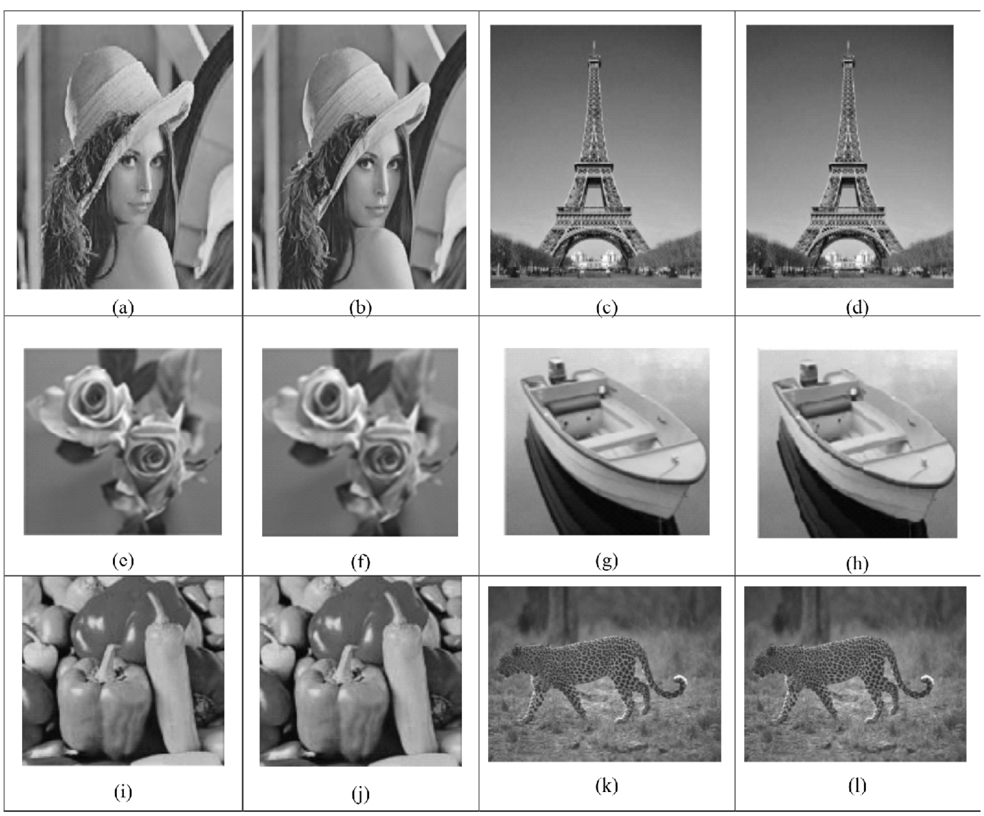

Figure 7: (a), (c), (e), (g), (i) and (k) shows the original images of lena, tower, rose, boat, peppers and, leopard images and (b), (d), (f), (h), (j) and (l) shows the reconstructed images of lena, tower, rose, boat, peppers and, leopard images

In present study six color images are namely lena, tower, rose, boat, peppers and, leopard images of size 256×256×3 are taken. Figure 7 (a), (c), (e), (g), (i) and (k) shows the original images of lena, tower, rose, boat, peppers and, leopard images respectively and (b), (d), (f), (h), (j) and (l) shows the reconstructed images of lena, tower, rose, boat, peppers and, leopard images, respectively; follows the proposed method.

Table 1
Comparison Value of PSNR, Time and Compression Ratio (CR) of RGB, HSI, YUV and Proposed YIQ Colour Space for Various Images

Input colour images		*RGB scheme*			*HSI scheme*			*YUV scheme*			*YIQ scheme*		
		R	*G*	*B*	*H*	*S*	*I*	*Y*	*U*	*V*	*Y*	*I*	*Q*
Lena	PSNR	34.58	34.10	33.26	35.12	34.70	33.42	37.11	38.11	39.01	39.21	39.91	39.97
	TIME	53	53	53	44	42	41	43	41	41	41	37	37
	CR	4.129	4.129	4.129	23.13	24.51	23.72	25.10	25.03	26.03	27.19	29.19	36.10
Rose	PSNR	31.36	32.87	30.74	32.52	33.88	30.92	34.12	34.22	35.02	38.78	39.07	37.73
	TIME	58	53	54	44	42	41	41	40	40	40	37	36
	CR	4.129	4.129	4.129	22.68	22.68	22.68	25.11	27.13	29.10	28.71	28.74	28.43
Tower	PSNR	34.54	34.22	34.31	35.12	34.70	33.54	36.01	36.71	36.71	39.61	36.73	39.73
	TIME	53	53	53	44	42	42	42	41	41	40	38	36
	CR	4.101	4.112	4.123	23.13	24.51	24.62	24.13	26.03	28.01	29.13	29.60	29.93
Boat	PSNR	30.36	30.14	30.33	32.52	33.88	31.32	32.79	35.03	36.03	33.02	36.90	38.78
	TIME	57	54	55	44	42	41	41	40	40	41	38	37
	CR	4.129	4.024	4.121	22.68	23.08	23.48	24.18	24.88	26.08	24.87	27.95	27.98
Peppers	PSNR	34.38	34.52	34.01	35.12	34.74	33.42	35.12	37.14	38.14	37.32	39.85	40.19
	TIME	53	53	53	44	42	41	42	41	41	40	39	37
	CR	4.129	4.001	4.130	23.13	24.51	23.72	24.10	25.12	25.98	27.31	29.51	34.32
Leopard	PSNR	31.16	30.54	31.34	32.52	33.88	30.92	33.52	35.52	35.69	36.64	35.12	34.55
	TIME	57	56	55	44	42	41	43	41	41	40	39	37
	CR	4.129	4.211	4.132	22.68	22.78	24.08	24.68	25.08	26.18	29.08	28.32	27.98

Table 1 demostrate the comparison value of PSNR, time and compression Ratio (CR) of RGB, HSI, YUV and proposed YIQ colour spaces for various images. In present study various colour schemes have been implemented and summarized the experimental results. It has been observed from the table 1 that the value of PSNR,and compression ratio are increased but reduces the execution time in proposed method. It has been found that the proposed method gives better figure of merits compared with the others. Thus the proposed fuzzy logic based YIQ scheme orientated compression may be expected to find an important oriented space in the field of image processing.

Figures 8, 9 and 10 demostrates the comparison value of PSNR (dB), execution time and compression ratio for various colour scheme respectivey. It has been observed that proposed method is better than the other implemented colour schemes.

In the present study, six images are taken for analysis. The followed methodology is found to yield satisfactory and acceptable results for all images. There is no speciality in the images but with self-similarity and they have been selected arbitrarily without any prior testing. Thus it is expected that the proposed method will give rise to satisfactory and meaning full result for other images as well.

It is a normal practice, to consider compression ratio, PSNR and program execution time to evaluate the performance of image compression and the same is found in various journals and reports. Besides,

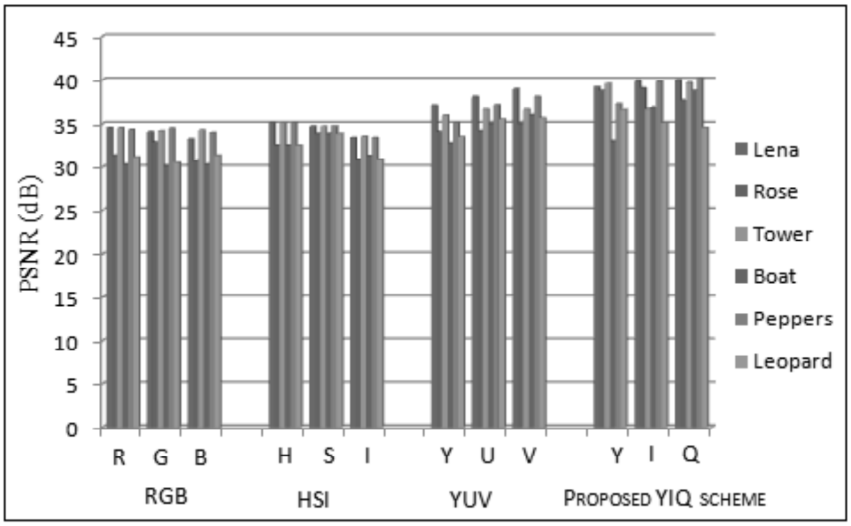

Figure 8: Demonstrate the comparison value of PSNR (dB) for different colour scheme

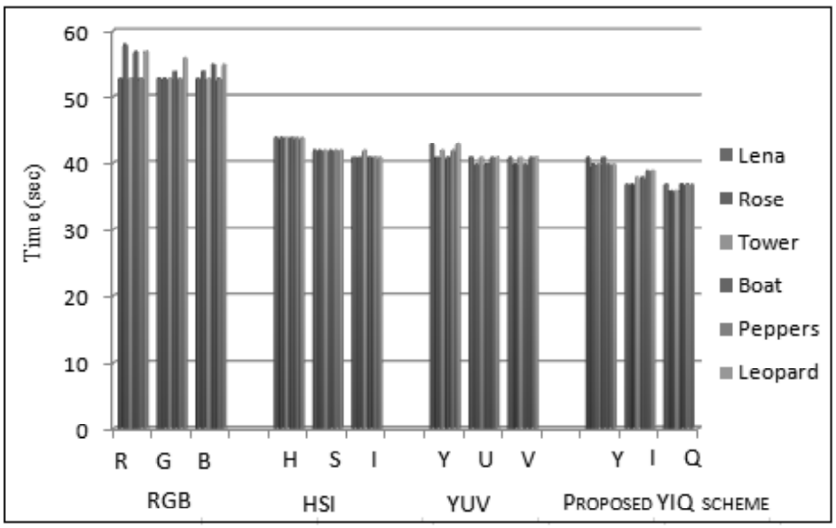

Figure 9: shows the comparison value of execution time (sec) for various colour scheme

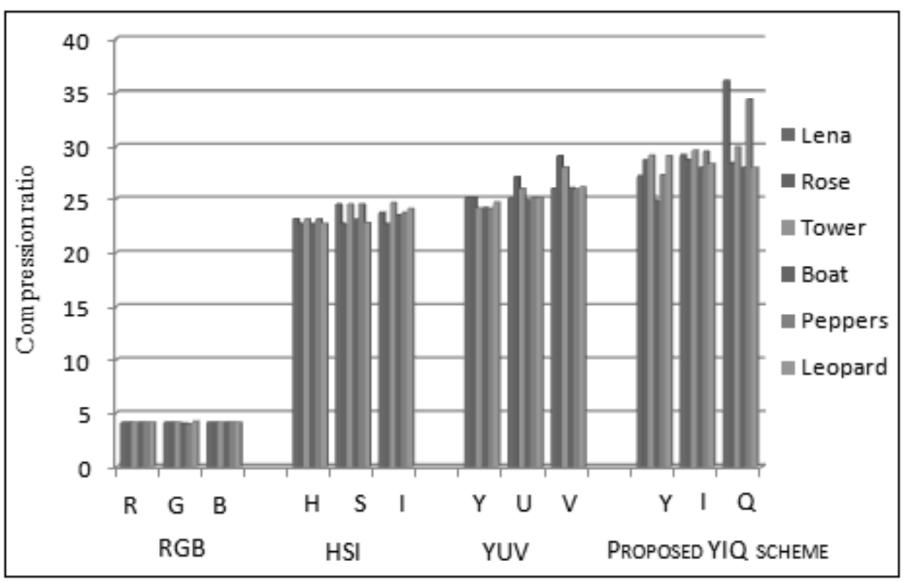

Figure 10: shows the comparison value of compression ratio for various colour scheme

other parameters may be taken as figure of merits. Structural similarity index measure (SSIM), mean standard error (MSE), other statistical parameters like second order and fourth order moments can also be considered. The evalution of the performance through these parameters can well be added for future studies.

V. CONCLUSION

YIQ colour space has been considered to study the fractal compression of six images and this is done by fuzzification of in phase (I) component of chroma part of the colour signal Q (Quadratue) part of the signal is kept unaltered during the processing. The results obtained by the proposed method have been compared with the other state of art of colour image processing like RGB, HSI and LUV schemes. It has been found that the proposed method gives better figure of merits compared with the others. The results have also been compared other recent fractal geometry based image processing studies. Thus the proposed fuzzy logic based YIQ scheme combined compression method may be expected to find an important space in the field of image processing.

REFERENCES

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

[2] A. Jacquin "Fractal image coding: A review," Proc. IEEE, vol. 81, pp. 1451–1465, Oct. 1993.

[3] M. Barnsley, Fractals Everywhere. San Diego, CA: Academic, 1988.

[4] A. Jacquin, "A novel fractal block-coding technique for digital images," in Proc. ICASSP-90, 1990, pp. 2225–2228.

[5] A. E. Jacquin. Image Coding Based on a Fractal Theory of Iterated Contractive Image Transform. IEEE trans. on Image Processing, Vol. 1, pp. 18-30, Jan. 1992.

[6] M. Barnsley and L. P. Hurd, Fractal Image Compression. Wellesley, MA: AK Peters, 1993.

[7] Rafael C. Gonzalez and woods, Digital image processing, 3rd edition, 2012, PHI. Science, 1989.

[8] Zhao Yuli, Zhiliang Zhu, Yu Hai, 'Fractal Color Image Coding Based on Isosceles Triangle Segmentation', *International Workshop on Chaos-Fractal Theory and its Applications, pp. 486-490, 2010*

[9] Y.Fisher, Fractal Image compression [J], Fractals, 2(3): pp.321-329, 1994.

[10] Shimal Das and Dr. Dibyendu Ghoshal, " A Proposed Block-Coding Technique of an Image based on Fractal Compression", International Journal of Computer Applications (0975 – 8887), Volume 57– No.5, pp. 35-42, November 2012

[11] Shimal Das and Dr. Dibyendu Ghoshal, "Colour Image Compression Method Based On Fractal Block Coding Technique", International Journal of Computer, Electrical, Automation, Control and Information Engineering Vol:8, No:11, pp. 1953-1958, 2014

[12] Ruhiat Sultana, Nisar Ahmed and Shaik Mahaboob Basha, "Advanced Fractal Image Coding Based on the Quadtree," Computer Engineering and Intelligent Systems, Vol 2, No.3, pp. 129-136, 2011

[13] Dibyendu Ghoshal and Shimal Das, "A Proposed Hybrid Color Image Compression Based on Fractal Coding with Quadtree and Discrete Cosine Transform", International Journal of Computer, Electrical, Automation, Control and Information Engineering Vol:9, No:11, pp. 2065-2072, 2015

[14] S. Wang and S. L. Tsai, "Automatic image authentication and recovery using fractal code embedding and image inpainting," Pattern Recognit. vol. 41, no. 2, pp.701–712, 2008.

[15] S. G. Lian, "Image authentication based on fractal features," Fractals vol. 16, no. 4, pp. 287–297, 2008.

[16] K. T. Lin and S. L. Yeh, "Encrypting image by assembling the fractal- image addition method and the binary encoding method," Opt. Commun.vol. 285, no. 9, pp. 2335–2342, 2012

[17] X. Tang and C. Qu, "Facial image recognition based on fractal image encoding, Bell Labs Tech. J. , vol. 15, no. 1, pp. 209–214, 2010

[18] Selim, A., Hadhoud, M. and Salem, M.O.," A Comparison Study between Spiral and Traditional Fractal Image Compression," International Conference on Computer Engineering & Systems, pp.39 – 44, 2002.

[19] Gaganpreet Kaur, Hitashi, Gurudev Singh," Performance Evaluation of Image Quality Based on Fractal Image Compression," International Journal of Computers & Technology, Volume 2 No.1, February 2012.

[20] M. F. Barnsley and A. D. Sloan, "A better way to compress images," *BYTE*, pp. 215–223, January 1988.

[21] Chandan Singh Rawat and Sukadev Meher, “A Hybrid Image Compression Scheme using DCT and Fractal Image Compression,” The International Arab Journal of Information Technology, Vol. 10, No. 6, pp. 553-562, November 2013.

[22] Shimal Das and Dr. Dibyendu Ghoshal, “Hybrid Fractal Image Compression Based on Graph Theory and Equilateral Triangle Segmentation”, International Journal of Applied Engineering Research, Volume 11, Number 6, pp 4467-4477, 2016.