A Review on Fractal Image Compression

Ankit Garg
Assistant Professor
Amity University, Haryana
Manesar, Gurgaon

Akshat Agrawal
Assistant Professor
Amity University, Haryana
Manesar, Gurgaon

Ashish Negi
Associate Professor
GBP, Engineering college and Technology
Pauri, Garhwal

ABSTRACT

Now a days most of the researchers are doing lots of work in the area of image compression. Fractal image compression requires lots of mathematical computation to compress an image. Fractal image compression is a recent technique based on the representation of an image by a contractive transform, on the space of images, for which the fixed point is close approximation to the original image. Main aim of fractal image compression algorithm is to reduce computation time required to compress image data. Fractal image compression is a lossy compression method for digital images, based on fractals. It is based on affine contractive transforms and utilizes the existence of self-symmetry in the image.This paper presents method for generating fractal images using iterated function system, method to partition image for compressing image using fractal image compression technique and various quality measures in fractal image compression.

Keywords

IFS, CMT,FIC, Fractal, PIFS

1. INTRODUCTION

The concept of fractal was described by IBM mathematician Benoit Mandelbrot. He found that traditional geometry was inadequate to describe the structure of natural objects which are complex such as mountain, cloud, coastlines and tree. The non-Euclidean geometry or fractal geometry deals with irregular and fragmented patterns. Fractals are complex objects which has property of self similarity- A small section of fractal object is similar to whole object, hence fractal are the repetition of the same structural form. There are two main groups of fractals: linear and nonlinear[1]. The latter are typified by the popular Mandelbrot set and Julia sets, which are fractals of the complex plane[1]. Fractal may have condensation sets. Fractal with condensation set are not quite self similar. In general to create any fractal three thing are required: a set of transformations (IFS), a base from which iteration start, and a condensation set(possibly the empty set).

Any image can be stored in the computer as collection of picture elements. Image stored in computer may have some amount of redundancy. An image with high redundancy requires not only more storage space but also requires high bandwidth channels to send that image from one network to another network. To minimize transmission time and storage cost compression of images is required.

M. Barnsley suggests principle of fractal image compression in 1988. In Fractal image compression we apply some mathematical function and affine transformation which is also called contractive affine transformation. Affine transformation applied on fractal image compression is combination of transformations i.e. rotation, scale change, skew or translation. A contractive affine transformation is important because the distance between two points in original image moves closer and closer to a fix point and this fixed point after affine transformation is nothing but an image. The image after applying contractive affine transformation is an approximation of original image. Iterated function system is another way of generating fractals. It is based on taking a point or a figure and substituting it with several other identical ones[2]. Iterated function system is just a collection of contractive affine maps c1, c2 and so on to cn [3].

When we keep applying iterated function system again and again, IFS will converge to some point which is nothing but attractor of IFS. Barsnley has derived a special form of the Contractive Mapping Transform (CMT) applied to IFS’s called the College Theorem. The approach of FIC is based on college theorem. College theorem suggests that the hausdorff distance between two images should be as minimum as possible, means the attractor of IFS should be close approximation of original image. The distance between original image and attractor is known as college error and it should be as small as possible. In partitioned IFS instead of applying affine transformation on whole image, contractive affine transformations are applied on parts of image, and the union of affine transformation is the final image. In Jacquin’s method the image is partitioned in sub images called as ‘Range blocks’ and PIFS are applied on sub-images, rather than the entire image to find out most suitable domain block. Basic steps in fractal image compression are: (1) Partition of image to be encoded to form range blocks (2) Selection of domain pool (3) Form a class of transform applied to the domain block (4) Searching most suitable domain block for formation of particular range block.

The aim of compression algorithm is to increase compression ratio and to minimize compression time. The other consideration is size of compressed file versus quality of decompressed image.

There are two types of compression: (1) 'Lossless' compression (2) Lossy compression. 'Lossy' compression always produce images which are not exact copy original data, It removes redundancy and creates an approximation of the original. Lossy compression are required to apply on those data where reproduction of original data is not required. Fractal image compression is a lossy compression method.

2. TRANSFORMATIONS

2.1 Affine Transformation

Affine transformations are linear transformations. Affine transformation is composition of rotation, translations, dilations and shears. An affine transformation do not preserve angles or length. Two or more successive transformations can be applied on the image with the use of affine transformation.

A transformation ω: $\mathbf{R}^2 \rightarrow \mathbf{R}^2$ of the form

$$w(x,y) = (ax + by + e, cx + dy + f)$$

where *a*, *b*, *c*, *d*, *e*, and *f* are real numbers, is called a (two-dimensional) *affine* transformation. Using equivalent notations:

$$T\binom{x}{y} = s\begin{pmatrix}\cos\theta & -\sin\theta \\ \sin\theta & \cos\theta\end{pmatrix} + \binom{e}{f}\binom{x}{y} \quad ...(1)$$

2.2 Banach's Contraction Mapping

Definition

Generating a sequence of images by applying contractive transformation over a image again and again moves images closer and closer to a fixed point. This fixed point generated by the contraction mapping is nothing but attractor.

A transformation $f : \mathbf{X} \to \mathbf{X}$ on a metric space ($\mathbf{X}$, d) is called *contractive* or a *contraction mapping* if there is a constant $0 \le s < 1$ such that

$$\mathrm{d(f(x), f(y))} \le \mathrm{s.d(x, y)} \forall \mathrm{x, y} \in \mathrm{X} \quad ....(2)$$

Any such number *s* is called a *contractivity factor* for *f* [1].

2.3 The Hausdorff Space

Definition

Let (**X**, *d*) be a complete metric space. Then the *Hausdorff distance* between the points *A* and *B* in H(X) is defined by

$$\mathrm{h(A, B) = d(A, B)} \forall \mathrm{d(B, A)} \quad .....(3)$$

We also call *h* the *Hausdorff metric* on H[1].

2.4 Iterated Function Systems

In the mid 1980's, IFS's become very popular. It was Barnsley and his co-workers at Georgia Institute of Technology who first noticed the potential of IFS for applications in computer graphics. Barnsley and Sloan advertised in popular science magazines the incredible power of IFS for compressing color images at rates of over 10,000 to 1.Following are two algorithm which comes under IFS. Transformation used to generate fractals should be contractive. By applying contractive transformation on an image will create an attractor which is fractal with the same structure at any level of detail. This is the basic property of fractal. Transformation should be contractive otherwise the attractor will spread out to be infinite size. Different attractors can be generated by applying different affine transformation over the same image. Each iteration under iterated function system provides reduced copy of original one. The multiple copies of original image is nothing but the result of affine transformation.

An iterated function system with probabilities consists of an IFS

$$\{X; w_1, w_2, \ldots\ldots, w_n\}$$

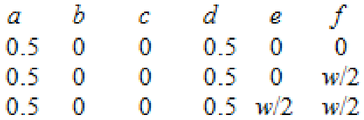
together with an ordered set of numbers $\{p_1, p_2, \ldots, p_N\}$, such that

$$p_1 + p_2 + p_3 + \ldots + p_N = 1 \text{ and } p_i > 0 \quad \text{for } i = 1, 2, \ldots, N.$$

The probability p_i is associated with the transformation w_i. The nomenclature *IFS with probabilities* may be used as an abbreviation[4]. The full notation for such an IFS is

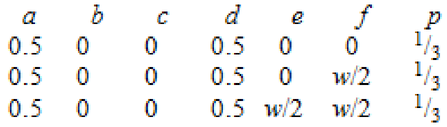
$$X: w_1, w_2, \ldots\ldots\ldots, w_n; p_1, p_2, \ldots\ldots, p_n$$

The probabilities are related to the measure theory of IFS attractors, and play a role in the computation of images of the attractor of an IFS attractor using the random iteration algorithm[1]. The IFS can be written as follows:



a	b	c	d	e	f
0.5	0	0	0.5	0	0
0.5	0	0	0.5	0	w/2
0.5	0	0	0.5	w/2	w/2

Fig1: Cofficient of IFS, where *w* is the width of the initial image.



a	b	c	d	e	f	p
0.5	0	0	0.5	0	0	$^1/_3$
0.5	0	0	0.5	0	w/2	$^1/_3$
0.5	0	0	0.5	w/2	w/2	$^1/_3$

Fig 2: IFS with probability

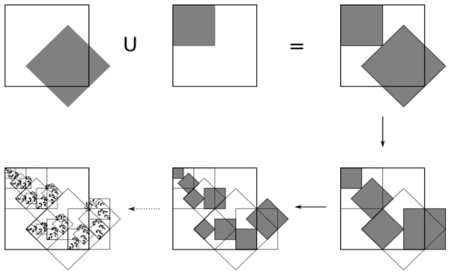


Fig 3: Construction of image using IFS and transformation.

2.4.1 Problem with IFS

The main problem with IFS is to obtain set of affine transformation to represent a given image. In order to fully automate fractal image compression, an algorithm must be used which requires no 'user interference', such as adjusting the coefficients in an IFS[1]. Tree like Objects in nature are fully different, IFS for one object (Tree) can not be applicable on other tree like structure. One theoretical reason why IFSs may never be successfully used in compression is the fact that given the IFS for object A, and the IFS for object B, they cannot be combined to give the IFS for object A ∪ B, or object A ∩ B[1].

3. ALGORITHMS UNDER IFS

3.1 The Deterministic Algorithm

In this method the we start with an image and apply some affine transformation on each subset of this image and try to find out next image which should be complete subset of R2 space where image lie. After applying affine transformation again and again we get a sequence of image which should be converging at some point which will be the limit point. This limit point is nothing but an image. The apply some mapping to get an image from other image, this mapping should be contractive. For using this approach to generate fractal we require heavy amount of memory, because in each iteration we are generating some image and to store image generated by affine transformation requires large amount of memory.

Let {**X** ; *w*1, *w*2, , *w*N} be a hyperbolic IFS. Choose a compact set *A*0 **R**2. Then compute successively *An* = *W* ° *n*(*A*) according to

An+1 = *n* **N** *wj*(*An*) for *n* = 1, 2, .

Thus construct a sequence {*An* : *n* = 0, 1, 2, 3, } H(X). Then by the IFS Theorem the sequence {*An*} converges to the attractor of the IFS in the Hausdorff metric. Fractal can be generated by using Ishikawa iteration. The objective of Ishikawa iteration is to know different types of orbit to generate fractal.

3.2 The Random Iteration Algorithm

The random approach is different from the deterministic approach in that the initial set is a singleton point and at each level of iteration, just one of the defining affine transformations is used to calculate the next level. Which will also be a singolton point? At each level, the affine transformation is randomly selected and applied. Points are plotted, except for the early ones, and are discarded after being used to calculate the next value. The random algorithm avoids the need of a large computer memory, it is best suited for the small computers on which one point at a time can be calculated and display on a screen. On the other hand it takes thousand of dots to produce an image in this way that does not appear too skimpy.

Let {**X** ; *w*1, *w*2, , *w*N ; *p*1, *p*2, , *p*N} be an IFS with probabilities. Choose *x*0, **X** and then choose recursively, independently,

xn {*w*1(*xn*-1), *w*2(*xn*-1), , *wN*(*xn*-1)} for *n* = 1, 2, 3, ,

where the probability of the event *xn* = *wi*(*xn*-1) is *pi*. Thus, construct a sequence {*xn* : *n* = 0, 1, 2, 3, } X [1].

4. THE COLLAGE THEOREM

To find an IFS which results close approximation to original and attractor is called "inverse problem". Genetic algorithm have been proposed for the inverse problem. College theorem does not solve the inverse problem but it provides some better way of approaching it. The Collage Theorem gives the idea is to find an IFS whose attractor is very close approximation to the original image. The approximation or closeness of original image and attractor can be measured by using the Hausdorff metric. This result is the key to finding an IFS that can be used to compress an image.

The collage theorem says that if D(C(T),T) <D then d(T,A) < d/(1-S), where C is an IFS with contractivity S and attractor A and T is any iamge. D(C(T),T) <D says that the distance between to point should be small because all the IFS should be contractive and the d(T,A) < d/(1-S) says the distance between orginal and attractor should be as minimum as possible or the hausdorff distance between orginal image and attractor should ne as minimum as possible. If the value of S close to 1, nothing ensures that this method provides a good approximation. Yet this was the original idea of Barnsley and most of the fractal based algorithm rely on the same approach. Quality of compression is depend upon how well the transformation manage to map the sections of the image in the range block to each domain block.

Continuity Condition in collage theorem state that Small changes in the parameters of an IFS will lead to small changes in the attractor. In other words, as the coefficients of an IFS are altered, the attractor changes smoothly and without any sudden jumps [1].

5. FRACTAL AND CHAOS

The relation between fractal and chaos is very strong. Fractal geometry is used to describe the behavior of chaotic system we usually find in nature.

Chaos theory is closely related to fractal theory. Many events were considered to be chaotic, unpredictable and random. A butterfly flapping its wings in a south American jungle, it is said, can lead to a hurricane in china. This is the signature of chaos theory [5].

Chaos Theory deals with nonlinear things that are effectively impossible to predict or control, like turbulence, weather, the stock market, our brain states, and so on [6]. Chaos is somthing which is very sensitive to initial condition. Chaotic systems are mathematically deterministic but nearly impossible to predict. Long term predictions of any system is not possible if the initial condition are known to an accurate degree. For example the weather is an example of chaotic system. In any chaotic system bifurcation means when attractor splits in two parts due to most complex calculation which is chaotic in nature. System often become chaotic when there is a feedback present. Fractal is a never ending patterns. They are created by repeating a simple process again and again in an ongoing feedback loop.To see just how far a fractal or certain conditions can diverge from its starting point, we can iterate with two initial conditions that differ by just a very small amount [7].

The result of any IFS is fixed point which is called the attractor. As the function iterates it generates some sequence of points which finally converges to the attractor. Basin of attraction is an area which bounds the solution. Points which are not bounded by basin of attraction can approximate to infinity or further from attractor. Factals that bear this particular trait are called escape-time fractals. Points which are outside from basin of attraction are called Repellers. Some of the attractor discovered are most surprising in form. The attractor which arise for chaotic systems have been called Strange attractors . Strange attractor are differ from regular attractors in that it is impossible to tell where they will be. Fractal are related to chaos because they are complex systems that have definite properties. Fig below describe chaotic function after recursively iterate function with some value of C.

5.1 Escape time and Escape time Algorithm

The value is iterated into given mathematical function to produce some list of number is called orbit. If the list of numbers (orbit) begins to take off to infinity, its ecape time is how many iteration it took before it was going to take off infinity.

If the list of orbit does not take off to infinity then it is a prisoner point. Each pixel is either colored black if it is a prisoner, or given a color based on how many iterations it took to escape.

It refers to some region in complex plane and typical color is used to indicate the number of iterations before iterate gets large. Escape time algorithm are employed to visualize the dynamics corresponding to the Julia sets and Mandelbrot sets.

The algorithm corresponding to F(z) or L(z), to create the escape time criteria is as follows:

Select a maximum iteration bound N, and a sense of unbounded M. For all pixels (j, k) corresponding to the points z in a rectangular portion of the complex plane, do the following:

```
start
let $i = 0$
While $z < M$ and $i < N$ do
$z = f(z)$
$i = i+1$
end while
If
$i = N$, mark the pixel $(j, k)$ black,
otherwise,
mark the pixel a hue that
corresponds to $i$.
end
```

6. FRACTAL IMAGE COMPRESSION

Fractal image compression is a lossy compression technique. The main of FIC is to produce image which is approximation of original image. According to Barnsley storing image as a collection of affine transformation may leads to image compression. Storing image as a collection of pixels requires more amount of memory. By storing only coefficient of affine transformation memory reuquirement can be minimize.

Compresssion is more computationally expensive because intensive search is used to find self similar block. At high compression ratio frctal compression may offer superior quality. Standared image compression methods can be evaluated using their compression ratio: the ratio of the memory required to store an uncompressed image and the memory required to store an image in compressed form. The images after FIC become resolution independent. This means when higher magnified images are generated, more details are visible as compared to ordinary zoom which blurs out gradually [8].

The difference betweens between the majority of existing fractal coding schemes may be classified into the following categories.

- partition imposed on the image support by the range blocks;
- composition of the pool of domain blocks;
- class of transforms applied to the domain blocks;
- type of search used in locating suitable domain blocks;
- representation and quantization of the transform Parameters[11].

7. PARTITION ITERATED FUNCTION SYSTEM

According to Barnsley it is possible to start with a fractal and find out IFS which is applied on fractal to reach on an attractor which is itself a fractal. This is known as inverse problem of fractal geometry.

Partition iterated function system is the solution to the inverse problem. PIFS is different from IFS because partition iterated function system apply only on the subset of image rather than entire image.

It may be possible that a natural image may contain self similarity which can not be found in the fractal.

Rather than having the image formed of copies of its whole self it may be formed of copies of properly transformed part of itself. This means the parts of image is only approximation of whole image. The combine mappings constitute a transform on the image as a whole. Collage error (the distance between collage and the image) guaranteeing that the fixed point of that transform is close to the original image. Distance are usually measured by MSE(mean squared error).

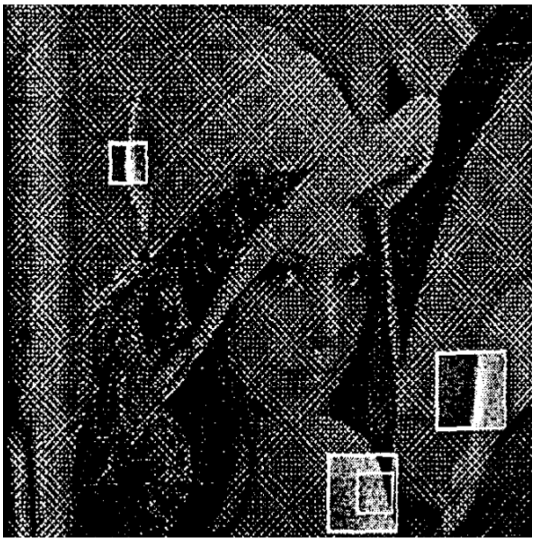


Fig 4: Image having self similar portions

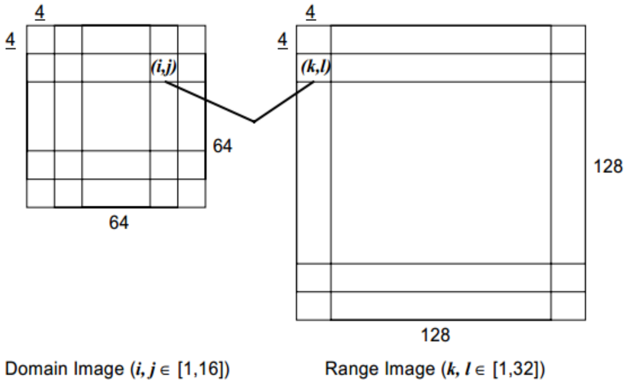


Fig 5: Formation of Range block and domain Block

The distortion measure between range block and domain block can be calculated by following equation. Each distortion is stored and minimum is choose for further processing.

$$d_{l2}\big(\Gamma\big(D_{i,j}\big), R_{k,j}\big) = \sum_{min}((\Gamma\big(D_{i,j}\big) - \big(R_{K,j}\big)m, n)^2 \text{....(4)}$$

Finally best domain block will assign to the range block.

$$\Gamma(D_{i,j})_{best} \Longrightarrow R_{k,j} \qquad \text{.....(5)}$$

the coordinates of the domain block along with its α and to are saved into the file describing the transformation. This is what is called the Fractal Code Book [9].

8. PARTITION SCHEMES

While designing fractal coding scheme concrete decision must be taken about the type of image partition, which is used in the formation of range block. A wide variety of partitions are available.

8.1 Fixed Size Squared Block

This is the simplest method to make image partitions. This type of partition is successful in transform coding of individual image block such as implementation in the JPEG standared.

8.2 Quadtree

This type of partition is based on recursively splitting of selected image quaderants. After partition image will look like a tree structure in which each non-terminal node has four descendants.

8.3 Horizontal-vertical partitioning

In this partition scheme the image is split into two by horizontal or vertical line to form two new rectangle. The partitioning repeats recursively until a covering tolerance is satisfied, as in quadtree scheme.

8.4 Triangular partitioning

Another way of image partitioning is based on triangle. A rectangular image is divided diagonally into two triangles.

Each of these are subdivided into 4 triangles by segmenting the triangle alonf the line that join three partitioning points along the tree sides of the triangle. This is good and flexible as compared to H-V partitioned scheme.

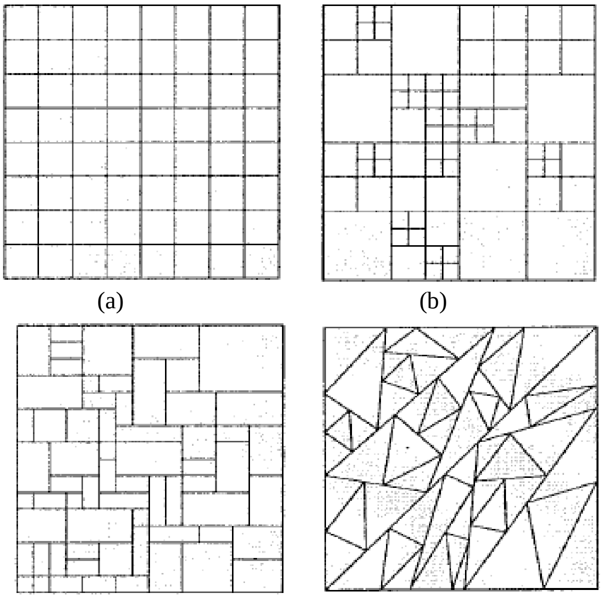


Fig 6: Right-angled range partition schemes. (a) Fixed block size. (b)Quadtree. (c) Horizontal-vertical. (d) Triangular partitioning scheme.

9. QUALITY MEASURES IN FIC

The quality of the compressed images becomes an important factor along with considerable compression ratio determines effective image compression. Compressiing an image reduces the amount of redundant data from the image. The objective of quality measures is to reflect the quality of decoded images approximately.A good objective quality measure should reflect the distortion on the image well due to, for example blurring-noise, compression.

9.1 Mean square Error

MSE or signal to noise ratio (SNR) is a quality measure which is frequently used to measure the deviation between original and coded image. The effectiveness of an image coder is optimize by having the minimum number of MSE at a particular compression.

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(f(i,j) - f'(i,j))^2 \quad .....(6)$$

9.2 Peak Signal – to – Noise-Ratio

Larger SNR and PSNR indicate a smaller difference between the original and reconstructed image. This is the most widely used objective image quality/ distortion measure. The main advantage of this measure is ease of computation but it does not reflect perceptual quality.

An important property of PSNR is that a slight spatial shift of an image can cause a large numerical distortion but no visual distortion and conversely a small average distortion can result in a damaging visual artefact. PSNR can be calculated using.

$$PSNR = 20log_{10}(\frac{N}{RMSE})dB \quad .....(7)$$

9.3 Average Difference

A lower value of Average Difference (AD) gives a “cleaner” image as more noise is reduced and it is computed using eq.(8).

$$AD = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(f(i,j) - f'(i,j) \quad ...(8)$$

9.4 Maximum Difference

Large value of MD means that the image is of poor quality.

$$MD = \max(|f(i,j) - f'(i,j)|) \quad ....(9)$$

9.5 Normalised Correlation

These measures measure the similarity between two images. All the correlation based measures tend to 1, as the difference between two images tend to zero. As difference measure and correlation measures complement each other, minimizing Distance measures are maximizing correlation measure [10].

$$N\mathbf{K} = \frac{\sum_{\mathbf{i=1}}^{\mathbf{M}}\sum_{\mathbf{j=1}}^{\mathbf{N}}[\mathbf{f(i,j),f'(i,j)}]}{\sum_{\mathbf{i=1}}^{\mathbf{M}}\sum_{\mathbf{j=1}}^{\mathbf{N}}\mathbf{f(i.j)^2}} \quad .....(10)$$

9.6 Mean Absolute Error

Large value of MAE means that the image is of poor quality.

$$MAE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(|f(i,j) - f'(i,j)|) \quad .....(11)$$

9.7 Normalised Absolute Error

According to this measure how far is the decompressed image from the original image with the value of zero being the perfect fit. Large value of NAE indicates poor quality of the image.

$$NAE = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}|\{f(i,j),f'(i,j)\}|}{\sum_{i=1}^{M}\sum_{j=1}^{N}|f(i,j)|} \quad ....(12)$$

10. FRACTAL IMAGE COMPRESSION ALGORITHM

1. Divide the image into range block
2. Divide the image into nonoverlapping domain blocks, D_i. The union of the domain blocks must cover the entire image, G, but they can be any size or shape [1].
3. Define a finite set of contractive affine transformations, w_i (which map from a range block R to a domain block D_i).
4. For each domain block
    - For each range block
        - For each transformation
        - Calculate the Hausdorff distance $h(w_i(R \cap G), D_i \cap G)$ (or use another metric)
        - Next transformation
    - Next range block
5. Record the transformed domain block which is found to be best approximation for the current range block is assigned to that range block.
6. Next domain block [1].

11. THE FRACTAL IMAGE COMPRESSION PROCESS

1. Read fractal code file.
2. Create memory buffers for the domain and range screens.
3. Initialize the range screen buffer to an arbitrary initial stage.
4. Repeat
    - For each domain
    - Replace this domain with the appropriate transformed range block
    - Next domain

```
        Copy contents of domain screen to range screen
        Until number of iterations required is attained.
5. Output the final domain screen, to produce a .BMP file with
   the same header, and of the same size as the original [1].
```

11.1 Speed Problem in compression

The time consumed in compression of any fractal image is major drawback of any fractal compression algorithm.

The main important part of fractal image compression algortithm is to map range block with suitable domain block, such that the housdroff difference between the original iamge and its attractor is minimum.

FIC algorithm requires lots of time in searching the when exact match does not found. For a large fractal image compression time can be measured in hours, rather than seconds or even minutes for large images. However, it could be argued that decompression time is of greater importance than compression time.

12. COMPARISION AND RESULT

There are various methods to compress the image i.e. Joint Photographic Experts Group (JPEG), Vector Quantization(VQ),Wavelet Transform and Fractal image compression.

Image compression algorithms based on Wavelet Transform [9], JPEG/DCT[7], Vector Quantization[18], and Fractal[18] methods were tested for 256×256 real image of Lenna and 400×400 fingerprint image. The results of performance are shown in Table 3, 4 and 5.

Table 1 : performance of coding algorithms on 256×256 Images

Algorithm	PSNR values OF figure print image (in dB)	CPU time	
		Encoding	Decoding
Wavelet	34.65	0.34 sec	0.26 sec
JPEG	31.72	0.11 sec	0.11 sec
VQ	29.27	2.45 sec	0.17 sec
FRACTAL	29.03	5.65 hrs	1.34 sec

Table 2: Performance of coding algorithm on 256×256 Fingerprint Image Of 0.5bpp

Algorithm	.5bpp		
	PSNR values Of figure print image (in dB)	Encoding	Decoding
Wavelet	36.70	0.6 sec	0.6 sec
JPEG	34.27	0.2 sec	0.1 sec
VQ	28.25	5.9 sec	0.6 sec
Fractal	27.20	6.2 hrs	3.4 sec

Table 3: Performance of coding algorithm on a 256X256 figureprint image of 0.25 bpp

Algorithm	.25bpp		
	PSNR values OF Leena's image (in dB)	Encoding	Decoding
Wavelet	32.45	0.6 sec	0.4 sec
JPEG	29.62	0.1 sec	0.1 sec
VQ	N/A	N/A	N/A
Fractal	N/A	N/A	N/A

In Table 1, 2 and 3 the performance of algorithms is shown. and comparison of Compression ratio of different algorithms is shown in Table 4 which is given below.

Table 4 : Performance On The Basis Of Compression Ratio Of Different Coding Algorithms

Methods	Compression Ratio
Wavelet	>>33
JPEG	<=50
VQ	<33
Fractal	>=15

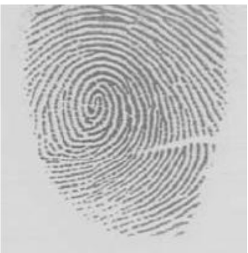


Fig 7: Original 256X256 fingure print image

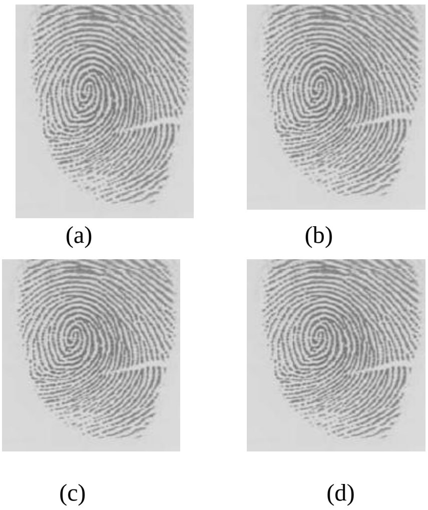


Fig 8: Decoded image of Fingre print by (a) Wavelet, (b) JPEG, (c) VQ,and (d) Fractal algorithms

12.1 Advantages of Fractal Compression

1. Resolution and device independent.
2. Decompression is 6 times faster than JPEG.
3. Superior for high compression ratios

Compression ratios can be improved by taking more time during compression without any increase in decompression time or decrease in image quality.

12.2 Disadvantages of Fractal Compression

1. Compression time is 12 times longer than JPEG.
2. An unproven technology.
3. It is expensive for small user.

12.3 Advantages of JPEG

1. JPEG is good where low compression ratios is required.
2. Because it is an industry standard, it is free.

12.4 Disadvantages of JPEG

1. JPEG can not work on binary images.
2. JPEG can not work well on the images which are non-realistic.
3. Quality of images degrades seriously with high compression ratios.

12.5 Fractal video compression

Fractal techniquies can also be useful for sound and motion videos. There is a fractal compression version of MPEG which can be used to compress real-time video.

13. CONCLUSION

Although fractal image coding is a relatively new technique but it has some advantages and disadvantages over JPEG or VQ. There are many directions in FIC which has not yet investigated. The main advantages of FIC is to provide high compression ratio, the speed of its decoding process and its multi resolution properties. Finally more attention need to be devoted to the encoding process which still suffers from long computation time.

14. ACKNOWLEDGEMENTS

I thank my guide Dr. Asish Negi, Associate professor in Department of Computer Science and Engineering, Govind Ballabh pant Engineering College, Pauri Garhwal for providing me an opportunity and resource to carry out the work.

15. FUTURE WORK

Several new scheme of image compression are being developed day-by-day. New recent filed in Fractal image compression is fractal wavelet transformation. Researches are investigating many new things to make fractal image compression fast. Below are a few areas which could be explored.Many fractal compression algorithm divided an image in rectangular or in squared form to make range block and domain block. Other geometries could be considered. To minimize searching time in compression can be minimize by reducing the number of affine transformation. Number of range block and size of domain pool should also be minimize To minimize compression time eliminate those range block which has been tested earlier in the process if it is unlikely to be a 'best-fit'. The pool of range blocks could be split into different sets; e.g. smooth blocks, edge blocks and midrange.

16. REFERENCES

[1] Fractal Image Compression, Martin V Sewell, Birkbeck College, University of London,1994.

[2] online available]: http://library.thinkquest.org/26242/full/tutorial/ch11.html

[3] [online available]:http://www.iprogrammer.info/babbages-bag/482-fractal-image compression.html?start=1.

[4] Suman k. Mitra, A New Probabilistic Approach for FractaL Based Image Compression, *Fundamenta Informaticae 87 (2008) 417–433.*

[5] [online available]:http://cybermax.tripod.com/Exploring.html

[6] [online available]:http://puttincologneontherickshaw.com/authors-blog/the-chaos-theory/

[7] [online available]:http://people.maths.ox.ac.uk/~hausel/m408k/bowens/

[8] A Study on Fractal Image Compression,By DarshanAlagud under guidance of K.R.Rao. Fall 09,Electrical Engineering Department, University of Texas at Arlington.

[9] An Introduction to Fractal Image Compression, Literature Number: BPRA065, Texas Instruments Europe October 1997

[10] Sumathi POOBAL, G.RAVINDRAN, The Performance Of Fractal Image Compression On Different Imaging Modalities using Objective Quality Measures, International Journal of Engineering Science and Technology (IJEST).

[11] Dietmar Saupe, Raouf Hamzaoui, A review of the fractal image compression literature,Computer graphics, vol-24, nov 4,1904.