IJETST- Vol.||02||Issue||04||Pages 2172-2177||April||ISSN 2348-9480 2015

Open access Journal International Journal of Emerging Trends in Science and Technology

Fast Fractal Image Compression Using Non Search Methodology

Authors

Prerna Mundada[1], Bhagyashri Patil[2], Rohini Mali[3], Bhagyashri Pawar[4]

[1,2,3,4]Computer, KKWIEER, Nashik, India

Email: *prernamundada.18@gmail.com[1]*, *bhagyapatil20@gmail.com[2]*, *rohini1793@gmail.com[3]* *bhagyashripawar2793@gmail.com[4]*

Abstract

Compression rates of fractal images are improved by applying the loss-less compression technique on the parameters of affine transformations of the fractal compressed image. By keeping PSNRs unchanged, proposed improved fractal image compression techniques give better compression rates. But, the compression time of proposed techniques are significantly increased than its counterparts. In order to solve the problem of long encoding time of fractal image compression, non-search fractal image compression coding is put forward. In this algorithm specific domain block is assigned as a matching block, so search is required and coding is accelerated. Besides, this algorithm adopts the method of range block adaptive decomposition and combination, and can solve problems like part of range blocks incapable of matching and low compression ratio of non-search method Experiments indicate that this algorithm is better than search fractal image compression algorithm and JPEG algorithm.

Keywords— *Compression ratio, Fractal Image Compression Coding, Non-search, Quardtree partition, Range Block Adaptive Decomposition, Range Block, Adaptive Combination*

INTRODUCTION

It is found that the memory space needed for GUI-based application software is increasing at a very high rate. By eliminating the high frequency components of the image, DCT-based JPEG compression techniques [6,7] obtain high compression rates. It is fine at low compression ratios. But, with the improvement of compression rate, images become “blocky” and poor quality images are produced. Sometime very visible artifacts are found, particularly for sharp edges in the images. Again, it is resolution dependent. To magnify the images, pixels replications are done to “zoom-in” on a particular area of images. The resultant images exhibit a certain level of “blockiness”. In order to overcome this problem, the image is stored at different resolutions wasting more memory space. Therefore, this well-established standard JPEG has its limitations. An alternative technique i.e. fractal image compression [1,2,4,5] is proposed. The technique offers good quality images at high compression ratios also and is resolution independent. A way to improve the compression ratio of the same is proposed in this paper. Fractal compressed file contains information about affine transforms. Compression rates of images are improved by applying the loss-less compression techniques on the parameters of the affine transformations of the compressed images. Fractal image compression is an algorithm that uses the self-similarity of image for compression[8]. Self-similarity denotes that however geometric dimension changes, each a small proportion of shape within image resembles the large proportion of shape. The fractal image compression is drawing extensive attention as an image coding algorithm with advantages of new idea, great potential, fast development, large compression ratio and high decompression speed. However, the drawback of fractal coding is high encoding time. The reason is that fractal coding usually requires every range block to search for its matched domain block, and plenty of time is during the process of search [9-10]. The current research on fractal is mainly focusing on reducing the search time and increasing the encoding speed. This paper puts forward a kind of adaptive non-search fractal image compression algorithm, and it does not need to do the searching. On condition that the quality decomposition image is guaranteed, search time is saved and compression ratio is raised.

FRACTAL IMAGE COMPRESSION

To designate images that are similar at different scales like mountains, galaxies, clouds etc. fractal images are used. There are details at every scale in these images. The fractal image compression representation is described mathematically as an iterated function system (IFS)[12]. An IFS is a set of contractive mapping {w 01: R2->R2 | i=1, 2, n} that map the plane R2 to itself in this case. According to above function the IFS defines a map w (A) =U^N i=1 w (A) where A C R2. The two important facts are:

1. If WI in the plane is contractive then we say w is contractive in space of subset of that plane.
2. For given 'w' on an image space, there is a unique image called as attractor & it is denoted by Tw that has properties:

   i) If w is used to Xw, the output is equal to input i.e Xw is considered as fixed point of w. W (Xw) =Xw=w1 (Xw) U w2 (Tw) U…U wn (Xw).

   ii) If given input image is s0, s1=w (so), s2=w (s1) =w (w (s0)) =w02 (s0) & so on are obtained. Here the attractor (Xw) does not depend on choice of s0.

   iii) So Xw is unique.

The above properties are used for compressed image and are known as contractive mapping fixed point theorem. In fractal compression with Partitioned Iterated Function systems(PIFs), the image which is to be compressed is itself a dictionary. In decomposition process the image is reconstructed by successive iteration of PIFs. The compression technique used here, initially divides the image into list of non-overlapping pages RI. By partitioning the image, a domain pool D is also created. These domain may overlap to each other. The partitioning scheme that is used to select the range Ri is Quardtree partitioning [2,3,5]. In Quardtree partitioning technique, a range of image is divided into 4 same size sub ranges, when it is found that range is covered by any domain of the domain pool. This process of partitioning continues recursively which begins from the entire image until the ranges are small enough to be covered within a predefined RMS value. The good suitable w is obtained by the compression technique to find match among domain Dj & range Ri. For this purpose affine transformation are used.

The complete algorithm of fractal image compression with quadtree partitioning scheme is as follows[13]:

1. Input a gray scale image F and choose a tolerance level i.e quality Q.
2. Select minimum range Rmin and then set the initial range R1= I^2 i.e unit square and make it uncovered.
3. Now select the domain pool D such that it must be bigger than the range to keep contractive mapping from domain to range.
4. While (Uncovered ranges Ri exist) , Do

   4.1 Obtain the domain Di and the affine map wi from the domain pool that well cover range Ri i.e that minimizes the D(Fn(Ri *I), wi(F)), where I means interval [0,1] and RMS is root mean square matric Using Euclidean distance formula.

   4.2 Then check IF ((D(F∩(Ri X I), Wi(F)) < Q) OR (SIZE(Ri) <= Rmin)), THEN

   4.2.1 Cover range Ri and then output the wo to the output file.
   Else

   4.2.2 Perform further partitioning as partition range Ri into four equal size of sub-square ranges Ri1,

Ri2, Ri3, Ri4 (Quadtree partition) that are marked as uncovered ranges.
End If
End While;

5 Stop.

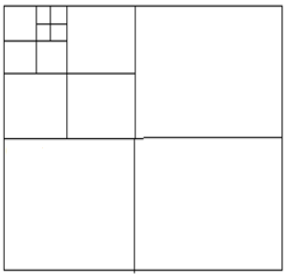

Figure 1: Quardtree partitioning scheme

$$W_i \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_{i,1} & a_{i,2} & 0 \\ a_{i,3} & a_{i,4} & 0 \\ 0 & 0 & C_i \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} d_{i,1} \\ d_{i,2} \\ b_i \end{bmatrix}$$

Figure 2: Affine Transform.

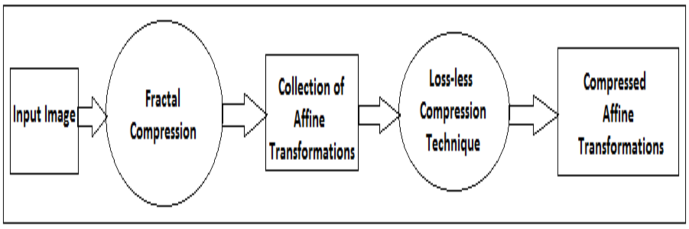

Figure 3: Compression process.

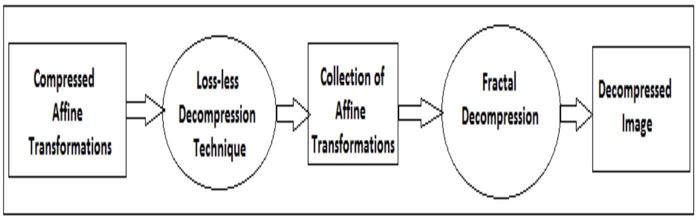

Figure 4: Decompression Process.

NON SEARCH FRACTAL IMAGE COMPRESSION

Unlike fractal image compression, the non search algorithm of image compression does not require searching of optimal domain block D for each range block R. Instead, a special domain block is assigned for each range block.Because of this the search process is eliminated and encoding time is reduced.

Basic Idea:

The basic requirement of non search algorithm[13] is to find position relation between D and R specifying affine transformation $wi : Di \rightarrow Ri$, so that by using coding error of wi on Ri given condition is satisfied and quality of image is guaranteed.

Coding Error:

- As we are assigning the domain block D to a range block R in advance, coding error di is taken into consideration.
- Coding error is the matching error between range block Ri and domain block assigned domain block Di .
- If the value of coding error di is smaller ,larger is the similarity between Ri and Di.
- As Di is assigned in advance [11], a situation may occur that current range block does not match the domain block i.e. Di is much larger. As a result, quality of coding image can not satisfy given requirement and coding even may not be completed. Therefore, a threshold value ε is set.
- When $di > \varepsilon$, it is it is considered that Ri does not matches Di . Ri and Di require to be redivided.
- When deciding the threshold value ε, it cannot be too large, otherwise low quality image is generated.
- It cannot be too small either, otherwise compress ratio of the image is lowered.
- Experiments show that if the threshold value is taken between 80 to 200, an equilibrium point can be reached which results in an image with good quality and compression ratio.
- If high quality image is required, the value of ε is set to 80. If high compression ratio is required, the value of ε is set to 200.

Range Block Adaptive Decomposition and Combination

- In order to guarantee the image quality after compression, range block needs to be divided as small as possible. This is because the matching error of assigned domain block is relatively larger. The larger range block is, the larger error is.

- This paper puts forward a definition of Basic Range Block. It denotes that not only the quality of compressed image can be guaranteed but also range block is not too small. For convenience of calculation, this paper considers basic range block of size B × B.
- Experiments prove that non-search fractal image compression significantly improves speed of image coding but its compression ratio is not large enough. The solution to this is adaptive decomposition and combination.

Adaptive Decomposition

If $di > \varepsilon$ i.e. *Ri* does not matches *Di* , the quality of compressed image is lowered. So it is necessary to divide *Ri* into a combination of multiple small range blocks and make each small range block satisfy the condition $di > \varepsilon$.

Step 1: adaptive decomposition is performed on range block that does not meet the condition of $di > \varepsilon$. Upon discovering one certain range block that meets $di > \varepsilon$, B ×B large range block is automatically divided into two B/2 × B large new range blocks *Rii*, each range block is assigned with new domain block , matching error *dii* of each new range block is calculated, and comparison is made between *Dii* and threshold value ε. If every $dii < \varepsilon$, matching is finished;

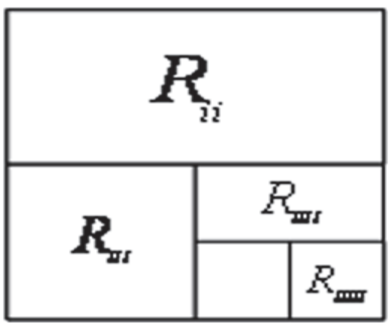

Figure 5: Schematic diagram of range block adaptive decomposition

Step 2: for new range block *Rii* that does not meet the condition of $dii < \varepsilon$, *Rii* is once again divided into two B/2 ×B/2 large new range blocks *Riii*, each new range block is also assigned with new domain block, matching error is calculated. If every $diii < \varepsilon$, matching is finished;

Step 3: for new range block *Riii* that still does not meet the condition of $diii < \varepsilon$, Step 1 and Step 2 will be continually repeated until new range block is 2 × 2 large. At this time, if the condition of smaller than threshold is still not met yet, four 2 × 2 large pixel values are directly transformed.

Adaptive Combination

For solving the problem of low non-search compression ratio, this psper is using B × B large basic range block; the number of each range block is $R(x,y)$, x and y are respectively refer to row coordinates and vertical coordinates of the matrix that range block locates in, *Rx* is defined as new range block formed by multiple $R(x,y)$, initial value of *Rz* is basic range block $R(x,y)$[13]. Then, adaptive combination is performed based on following procedure.

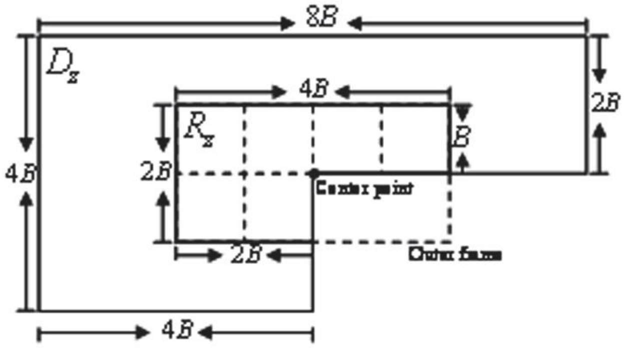

Figure 6: Schematic diagram of the position of irregular range block and domain block

Step 1: range block *Rz* (initial value of *Rz* is basic range block $R(x,y)$) is combined with basic range block $R(x+i,y+j)$ (initial value of i is 0, initial value of j is 1), new range block *Rz* is formed.

Step 2: range block *Rz* is assigned with domain block *Dz*. Here, when *Dz* is in irregular shape, domain block can still be assigned.

Step 3: matching error of range block *Rz* and domain block *Dz* is calculated, comparison is made with threshold value ε. If $Rz < \varepsilon$ is met, then j +1, repeat Step 1. If $Rz < \varepsilon$ is not met, then i +1 and j is reset as 0, repeat Step 1.

Step 4: if *Rz* fails to meet $Rz < \varepsilon$ for successive two times, adaptive combination is terminated. Stipulation is made that combination can only be performed on each basic range block $R(x,y)$ for one time; if combination is performed twice, it is considered that new range block Rz does not meet $Rz<\varepsilon$.

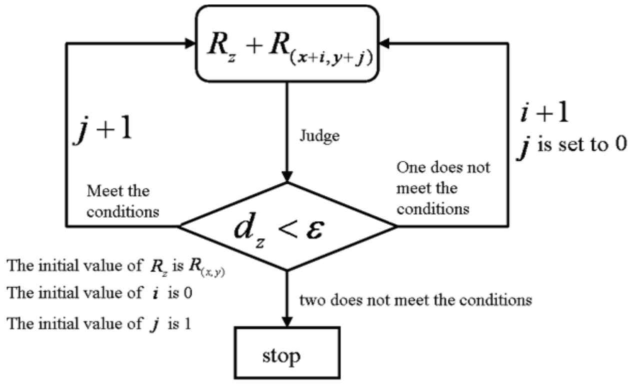

Figure 7: Flow diagram of range block adaptive combination

The process of adaptive combination mentioned above specifies that, basic range block $R(x,y)$ is continually combined with its adjacent basic range block on its right side until the formed new range block Rz is unable to meet the condition; then, range blocks in next line of $R(x,y)$ are respectively combined inside from right side, starting from $R(x+1,y)$, by this analogy, repeat the step until the condition of $Rz<\varepsilon$ is not met for successively two times, then this step is terminated.

OBSERVATIONS

The images used for compression are show in fig.

Images	FIC(msec)	NFIC(msec)	Difference
64 X 64	710	690	20ms
128 X 128	11513	9828	1685msec
256 X 256	20421	18454	1967msec
512 X 512	2665237	2242942	422295msec

We find that as the resolution of image increase the time required for compression also increases. The difference in compression time of both FIC and Non Search FIC increases. So using non search methodology for compression we have successfully reduced encoding time that is required for FIC compression.

CONCLUSION

This paper puts forward a kind of adaptive non-search fractal compression algorithm, whose most distinctive advantage of this algorithm is the non-search matching mode of range block and domain block and the adaptive decomposition and combination mode of range block. As the experiment has proved, by directly assigning a specific domain block to each range block as matching block, the most time-consuming search matching time can be saved and coding time can be shortened. This paper comes up with the definition of basic range block, namely the range block producing highest compression ratio in premise of guaranteeing the quality of compressed image. With basic range block set as basis, adaptive fractal algorithm is adopted on basic range block that fails to satisfy threshold value, so as to improve image quality, while adaptive combination algorithm is adopted on basic range block that satisfies threshold value, so as to improve compression ratio of image. Finally, the experiment indicates that the comprehensive performance of this algorithm is better than the fractal image compression algorithm with search mode and the standard JPEG algorithm as well.

REFERENCES

1. Jean Cardinal, “ Fast Fractal Compression of Greyscale mages”,IEEE transactions on image processing, vol. 10, january 2001.
2. Nandi, U., Mandal, J. K., “Fractal Image Compression using Fast Context Independent HV partitioning Scheme”, 3rd International Symposium on Electronic System Design (ISED-2012), pp. 306 – 308, December, 2012, Kolkata,India.

3. Nandi, U., Mandal, J. K., “Fractal Image Compression with Adaptive Quadtree Partitioning Scheme”, International Conference on Signal, Image processing and Pattern recognition (SIPP-2013), vol. 3, no. 6, pp. 289–296, 2013,Chennai, India.
4. J. Kominek, “Algorithm for fast fractal image compression,” in Proc. IS&T/SPIE 1995 Symp. Electronic Imaging: Science Technology, vol.2419, 1995.
5. Y. Fisher, et al., “Fractal Image Compression: Theory and Application ”,Springer Verlag, New York, 1995.
6. Wallace, Gregory K., : The JPEG Still Picture Compression Standard : Communications of the ACM, Volume 34, No. 4, pp 31-44 (1999).
7. Pennebaker, William B., Joan L. Mitchell, L., : JPEG Still Image Data Compression Standard : New York, Van Nostrand Reinhold (1992).
8. Fisher Y, Fractal Image Compression-Theory and Application,New York:Springer-Verlag,1995.
9. Belloulata K, Konrad J, Fractal image compression with region-based functionality. IEEE Transactions on Image Process, 4(2):351-362,2002.
10. Lai C M, Lam K M, A fast fractal image coding based on kick-out and zero contrast conditions. IEEE Transactions on Image Process, 11(2):1398-1403,2003.
11. Zhao Jian, Lei Lei, Pu Xiao-qin, Fractal theory and its application in signal processing. Bei JingTsinghua university press,57-71.2008.
12. Utpal Nandi, Jyotsna Kumar Mandal, Fractal Image Compression By Using Loss-Less Encoding On The Parameters Of Affine Transforms
13. Shiping LI1, Lei Xia2, Non-search Fractal Image Compression Algorithm Research