Fractal Image Compression and its Application in Image Processing

Gaurish Joshi

Lecturer, Electronics & Telecommunication Engineering Department, Government Polytechnic College, Sendhwa 451449, India

Abstract: ***Fractal Image Compression scheme has got great importance in last decade because of its application not only in image compression but also in many image processing fields. In this paper Fractal Image Compression scheme is utilized in the sharpening and smoothing of images by using varying affine parameters used in the affine transform. Fractal Image Compression scheme has not become as famous as JPEG because of its large encoding time and complexity present in the system. In this paper two methods RDPS and ERB have been proposed to improve the encoding time of FIC scheme. RDPS mainly focus on reducing the encoding time and ERB focus on increasing compression ratio along with slight improvement in encoding time. Therefore both methods are combined to form new method RDPS-ERB to obtain the best results. It has been shown that compression ratio increased to double that of existing work with very low loss in image quality.***

Keywords: Fractal Image Compression, Affine parameter, MSE,SSIM, RDPS, ERB, Sharpening, Smoothing

1. Introduction

Fractal Image Compression (FIC) is attracted by the researcher because of its advantageous like high compression ratio and resolution independent characteristics[1] , and it is recognized as one of the most three promising next generation image compression technology, besides, fractal compression also can be applied to audio and video compression[2] . For instance, Microsoft Corporation used once fractal compression method to compress a ten hours‟ HD video to a VCD with 700MB capacity[2]. The main advantage of Fractal Image Compression scheme are its ability to provide high compression ratio for large class of images, speed of its decoding process and multi-resolution properties. In fractal compression, an image is encoded by a partitioned iterated function system (PIFS) whose attractor is close to the original image. These parameters and corresponding position information of block pairs are stored instead of the original image to decrease the storage space occupied by the image. Aside from its application as an image compression scheme, FIC has been widely applied in other image processing fields [3], such as image indexing and retrieval, image encryption, image denoising, and some pattern recognition problems such as facial image recognition.

2. Literature Survey

Fractal-based image compression scheme is based on local self-similarity present in the image. The idea of Fractal Image Compression given by Barnsley‟s research for Iterated function system (IFS) [4] andJacquin proposed the image fractal block coding[5].Fractals are the geometrical shapes that are self-similar i.e. these shapes have parts that are similar to the whole image. Jacquin implemented the algorithm automatically using the partitioned iterated function system (PIFS) in 1992. After that the FIC technology has shown rapid development. Fractal Image Compression has become practical since then.Images are encoded using IFS. Basically IFS is just a function and the input of this function is image. IFS encode an image by making the image an unique fixed point of the IFS. IFS is constructed in such a way that image we want to encode will be invariant under IFS mapping and image will be mapped to itself. At the decoding side we start with any initial image and apply the IFS iteratively until the encoding image appears. Therefore it does not matter whichever image we start with the decoding process,we will end in original image.

2.1 Local Iterated Function System

In the iterated function algorithm we try to match the whole image to small part of this image but as we also know that its not possible in most of the natural images. So local iterated function system or partition iterated function system concept was introduced. In this partition iterated function system each part of the image is approximated by applying a contractive affine transformation on another part of the image. In this the image I is partitioned into range blocks Ii where $I = \sum_{i=1}^{n} Ii$. After that each range block is approximated by the transformed version of bigger domain blocks Di [6] .

2.2 Affine Transform

An iterated function system describes an image in FIC. An affine transform is a linear transform of the form [7]:

$$f_i\begin{bmatrix}x\\y\\z\end{bmatrix} = \begin{bmatrix}a_{i1} & a_{i2} & 0\\a_{i3} & a_{i4} & 0\\0 & 0 & c_i\end{bmatrix}\begin{bmatrix}x\\y\\z\end{bmatrix} + \begin{bmatrix}d_{i1}\\d_{i2}\\b_i\end{bmatrix} \qquad (1)$$

In the above equation, f_i is the transform applied on a point (x, y, z) in the ith block of input image. Gray scale images can be represented by such points (x, y, z), in which case z would be the intensity or gray-level at pixel co-ordinates (x, y) in an image. The a_{i1}, a_{i2}, a_{i3}, a_{i4}, d_{i1} and d_{i2} are constants for the ith image block to which the transform is applied. An affine transformation is said to be contractive if distance between the transformed points is less than the original points in the metric space. The affine transform should be contractive in nature so that repeatative application of IFS on any initial image converges to the original image. Therefore the objective of FIC is to find such an IFS that would describe the input image.

2.3 Collage Theorem

To generate the IFS for the attractor of IFS has become clear but inverse problem i.e. to find the IFS code that will approximate the given arbitrary set is rather a difficult task. In the history several efforts have been made to find the mathematical solution of this by using different mathematical tool like Fourier transform [7], wavelet transform [8], other methods. It was earlier also that for given W(Tranformation) the decoding process works based on contraction mapping theorem. Transformation W is applied on any initial image until the transformed image does not change significantly. This convergence to fixed image is guaranteed by contraction mapping theorem since W is contractive in nature. Collage theorem provides solution for inverse problem i.e.to find the contractive transformation for given set f such that its attractor g is closed to f.

Let T be a contraction on complete (E,d) metric space with contractivity factor s and fixed point g. Let f ∈ E Then [4]

$$d(f,g) \leq \frac{1}{1-s} d(f,Wf) \qquad (2)$$

Thus by minimizing the distance between f and Wf (The collage of the image), we hope to minimize the distance between fixed point g and the given image f. Of course, if the value of s is 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.

2.4MSE based Fractal Image Compression scheme

The FIC proceeds as follows. First, the original image is partitioned into two pools of blocks: range pool and domain pool. The range pool is obtained by non-overlapping square range blocks with size B×B of original image, and the domain pool is obtained by square domain blocks with larger size 2B×2B. All domain blocks are then brought down to size B×B by averaging four pixels to one pixel. Range blocks need to be encoded and takes the domain pool as a virtual codebook. The domain pool is called the „„virtual codebook"" because it is only used during encoding but not during the decoding process. To improve the accuracy of decoded images, eight transformations are applied to all domain cells to octuple the number of elements in the domain pool. Subsequently, for an arbitrary block R in the range pool, a block D in the domain pool has to be searched so that one affine transformation sD+o1 exists to minimize the distance with R in some image quality measurements, where s and o are the affine scalar parameters and 1 is a block with size B×B in which all pixels equal to 1. If the image measure is MSE, then [3]

$$MSE(R, sD+o1) = (sYi + o - Xi)^2 \qquad (3)$$

where $m=B^2$, the pixels in block D are Y1,Y2, ….,Ym and the pixels in block R are X1,X2, . . . ,Xm. Finally, the combination of (s, o, index of D in the domain pool) constructs the IFS subsystem of R, and all the subsystems of range cells group into the IFS of the original image. Minimizing MSE in eq. (3), we can obtain the value of parameters s and o using the least-square method as:

$$s_{MSE} = \frac{\sigma_{RD}}{\sigma_D^2} \qquad (4)$$

$$o_{MSE} = \mu_R - s_{MSE}\mu_D \qquad (5)$$

2.5SSIM based Fractal Image Compression scheme

SSIM is an image quality measurement where we used to measure the image quality of test image by comparing it with reference image which is considered to be perfect quality. It models the image quality measurement as a combination of three different factors: coefficient of correlation, luminance distortion, and contrast distortion[15]. Suppose μ_R and μ_D are the mean gray values of image blocks R and D, respectively, σ_R and $\sigma_D$are the standard deviations of R and D, respectively, and σ_{RD} is the covariance between R and D. Then the formula [3] of SSIM between R and D is expressed as follows:

$$SSIM(R, D) = \frac{\sigma_{RD}}{\sigma_R \sigma_D} \frac{2\mu_R \mu_D}{\mu_R^2 + \mu_D^2} \frac{2\sigma_R \sigma_D}{\sigma_R^2 + \sigma_D^2} \qquad (6)$$

SSIM is a more appropriate image quality measurement for the human visual system than MSE. SSIM(R,D) ∈ [−1,1] obtains its biggest value 1 when R is exactly the same as D. The bigger the SSIM value between R and D, the more similar blocks R and D. In FIC, the image distance between sD+o1 and R should be minimized. Therefore, the biggest value of SSIM(R,sD+o1) could be calculated as follows:

$$SSIM(R, sD+o1) = \frac{4s\sigma_{RD}\,\mu_R(s\mu_D + o)}{(\sigma_R^2 + s^2\sigma_D^2)[\mu_R^2 + (s\mu_D + o)^2]} \qquad (7)$$

Parameters s and o in SSIM measurement from Eq.(7) can be calculated using the least-squares method:[9]

$$s_{SSIM} = sgn(\sigma_{RD})\frac{\sigma_R}{\sigma_D} \qquad (8)$$

$$o_{SSIM} = \mu_R - s_{SSIM}\mu_D \qquad (9)$$

2.6 Image contrast

The formula of image contrast proposed by Tamura et al.[10] is used, which is expressed as follows:

$$Fcon = \frac{\sigma}{\alpha_4^{1/4}} \qquad (10)$$

Where σ is the standard of an image and α_4 is the kurtosis defined as:

$$\alpha_4 = \frac{\mu_4}{\sigma^4} \qquad (11)$$

Where μ_4 is fourth moment about mean of the image.

3. Drawback of Previous Work

The main drawback of Fractal Image Compression scheme is its slow speed of encoding and complexity. Many efforts have been made previously by researcher to improve its encoding speed. I have proposed one of method [11] to improve the speed of encoding present in existing work [3] and also to increase its compression ratio with acceptable loss in image quality. The main reason of large encoding time was number of matching operations involved at encoding stage. It can be explained as follows for the previously accepted parameters:

1) Number of range blocks=(image size)/(range block size)=(512x512)/(8x8)=4096
2) Number of domain blocks=(image size)/(domain block size)=(512x512)/(16x16)=1024
3) Number of matching operations performed=4096x1024x8=33554432

Because of these number of matching operations systems becomes very slow.

4. Proposed Work

To reduce the number of matching operations so as to make the system faster, two methods have been used RDPS(reduce domain pool size method and ERB (exclude Range block method).We can reduce the number of matching operations either by reducing the number of domain blocks in domain pool(RDPS) or by reducing the number of range blocks in range pool(ERB). RDPS method focus on reducing the number of matching operations by reducing the domain pool size. In RDPS method size of each domain block is taken 4BX4B instead of 2BX2B which was taken in existing work where B was taken 8 pixels. The size of range block is taken same as earlier i.e.BxB. Now we perform averaging of 4x4 pixels instead of 2x2 pixels to bring down the size of domain block same as range block for comparison purpose. Now

1) number of domain blocks= (image size)/(domain block size)= (512x512)/(32x32)= 256
2) number of range blocks= (image size)/(range block size)= (512x512)/(8x8)= 4096
3) Total number of matching operations= 4096x256x8= 8388608 which are very less compare to 33554432 operations

ERB method focus on increasing the compression ratio and also reducing the encoding time by keeping the reconstructed image quality almost same as of earlier. Therefore to get the best results i.e. to reduce encoding time and to increase compression ratio at the same time we combine both method now we call it as RDPS-ERB method

4.1RDPS-ERB encoding algorithm

Step 1: Start with any initial image of size 512×512 of database and take the range block size of 8×8 and domain pool size of 32×32
Step 2: Store some constant value to variable HP
Step 3: Find the number of blocks of domain pool denoted by blockno and create the cell array D of size (1,blockno^2)
Step 4: Divide the image into 32×32 blocks and store as elements of cell array D
Step 5: Find the number of blocks of range pool denoted by blocknoR and create the cell array R of size(1,blocknoR^2)
Step 6: Contract the size of domain blocks to 8×8 by taking the average of 16 neighbouringpixelsand replace them by one pixel and also createanother cell array E of size(1,blockno^2) which each element is contracted domain block of size 8×8
Step 7: Now 8 transformation are applied to each domain block to octuple the element number in domain pool and to store them, create cell array U of size(1,blockno^2*8)
Step 8: Compute the mean of range pool cell array R and also calculate its variance
Step 9: Compute the mean of domain pool cell array U and also calculate its variance
Step 10: Compute the covariance of cell array R and U
Step 11: For each range block in cell array R find its correlation coefficient with each of domain blocks in domain pool cell array U and creating the array H of size(blocknoR^2,blockno^2*8)
Step 12: Now compare each range block variance with HP value and go to next step if range block standard deviation> HP else go to step 14
Step 13: Now finding the maximum value along each row of array H and also index value of domain cell for which it is maximum
Step 14: Save the mean value of range block
Step 15: Finally calculate the parameters values(s and o) for the index value calculated in the step13 and store these parameter values and corresponding index values, mean values of range blocks in a separate matlab file

4.2RDPS-ERB decoding algorithm

Step 1: Start with any initial image for first iteration with same size as of image used in encoding
Step 2: Take the range block and domain block size same as used in encoding process
Step 3: Apply the same operation as used in decoding on image and create R,D and E cell arrays of same size used in decoding
Step 4: Now load the parameter values in some structure array and index values from the matlab file which is created in the end of encoding process
Step 5: Now read these parameter and index values from structure array and store them in some another matrix of size (blocknoR^2,4)
Step 6: Access each row of matrix obtained in step 5 one by one and store each of 4 elements in 4 separate variables
Step 7: Now generate the same transform copy of each domain block corresponding to maximum correlated index value for each range block and the domain blocks corresponding to 0 index values are replaced by mean values of range blocks saved at the time of encoding
Step 8: Apply affine transform to each domain block obtained in above step and store them in some new 8×8 matrix
Step 9: Combine all the 8×8 matrices obtained above in one cell array of size (blocknoR, blocknoR)
Step 10: Now store the cell array obtained in above step in some matlab file
Step 11: In the next iteration use this matlab file in place of image used in the first iteration
Step 12: After each iteration check whether the encoded image with minimum distortion is obtained or not if not go to step 3.

5. Results

All the results are obtained on Intel(R) Pentium(R) 2.2 GHZ processor with 2 GB RAM.

5.1 Comparison of present work with previous workfor RDPS method

Table 1: Comparison for Sssim parameter value

Work	Encoding time (sec)	PSNR (db)	Ratio of decoded image contrast to original image contrast	Compression ratio
Previous	156.1369	36.5806	1.0026	3.6893
Modified	39.6621	36.1559	1.0022	3.7463

Table 2: Comparison for Smse parameter value

Work	Encoding time (sec)	PSNR (db)	Ratio of decoded image contrast to original image contrast	Compression ratio
Previous	156.1425	37.0017	0.9786	3.6944
Modified	39.7212	36.6118	0.9773	3.7517

Table 3: Comparison for different s parameter values

Work	Parameter values	Encoding time (sec)	Ratio of decoded image contrast to original image contrast
Previous	0.6Sssim	157.3577	0.9145
Modified	0.6Sssim	44.6398	0.9036
Previous	1.4Sssim	158.0925	1.2382
Modified	1.4Sssim	39.6604	1.1410
Previous	1.6Sssim	156.1546	1.4356
Modified	1.6Sssim	39.9074	1.2293
Previous	1.8Sssim	156.1717	1.6916
Modified	1.8Sssim	40.1401	1.3257

Table 1 and Table 2indicates the comparison of modified work with previous work [3] for Sssim parameter value in SSIM scheme and Smse parameter value in MSE scheme respectively. We can see that modified system is more faster than previous system with very low loss in image quality indicated by PSNR value .We can also see that compression is also slightly increased.

From Table 3 we can see that for each parameter value in SSIM scheme encoding time is verymuch reduced at the cost of some less decoded image contrast but it is not disadvantageous because parameter is in our control we can increase or decrease it according to our requirement. The advantage of this RDPS method is that it has made our system much faster compare to original system. In the above results ratio of decoded image contrast to original image contrast is calculate by equation (10).

5.2 Comparison of present work with previous work for RDPS-ERB method

Table 4: Some parameter values for previous work

Encoding time (sec)	Compression ratio	PSNR (db)	Decoding time (sec)
156.1369	3.6893	36.5806	23.1200

Table 5: Same parameter values for present work

HP	Encoding time (sec)	Compression ratio	PSNR (db)	Descoding time (sec)
10	40.0033	3.7412	36.150	21.6183
20	39.2487	4.2289	36.1012	18.3992
30	38.5964	4.9906	36.0583	15.4097
40	38.4517	5.5532	35.9005	13.9815
50	37.3233	6.0450	35.7411	12.7711
60	37.2541	6.5517	35.5694	11.7822
70	36.3794	7.0184	35.4094	11.1246

From Table 4 and Table 5 we can conclude that encoding time,decoding time and compression ratio has been improved. So present system is more faster compare to previous system [3] and has more compression ratio. In Table 5 HP is homogenous parameter value which indicates standard deviation values. We can see that as we goes on increasing HP values encoding time, decoding time goes on decreasing and compression ratio goes on increasing at the acceptable loss in image quality indicated by PSNR values. Compression ratio in Table 5 is obtained by dividing the image size by fractal code file which is obtained at the time of encoding.

5.3Application of s parameter in sharpening and smoothing of image in RDPS method

Original image

Figure 1: Sharpening or smoothing of original image for 0.6 Sssim, 0.8 Sssim, 1.4 Sssim, 1.6Sssim, 1.8Sssim, 2Sssim from top left corner to bottom right respectively

From Figure 1 we can see that we are still getting sharpening and smoothing of images by varying s parameter in this modified RDPS method as we were getting in the previous work [3] but now system is much faster than previous system.

5.4Reconstructed image for different HP values in RDPS-ERBmethod

Previous Method

Figure 2: Reconstructed image of previous method and present work for HP values=10,20,30,40,50,60 shown from top left corner to bottom right corner respectively

From Figure 2 we can see that as we goes on increasing HP values image quality degrades but not much even for approximate double compression ratio compare to previous method [3] which is achieved for HP=60. All the reconstructed images obtained in Figure 2 are for Sssim parameter value in SSIM scheme. Similar results can be obtained for Smse parameter value in MSE scheme.

6. Conclusion

From the above discussion we can conclude that affine parameter of FIC scheme vary with image quality measurement like SSIM and MSE. There is positive correlation between affine parameter s and image contrast i.e. by increasing parameter image enhances and by decreasing parameter smoothing of image occurs. Therefore if we want decoded image to be sharper we will have to keep |S|>|Sssim| and if we want decoded image to be smoother than original image then we will have to keep |S|<|Sssim|. Decoded image has best contrast with original image when |S|=|Sssim|. Present system is made faster by reducing the number of matching operations. Two method RDPS and ERB aresuccessfully implemented for improving the drawback of previous system [3]. RDPS mainly focus on decreasing the encoding time and ERB focus on improving the compression ratio so best results can be obtained by combining the two methods. After employing these two methods high values of parameter s is to be maintained to get the same image contrast as of previous work [3]. The new system becomes more faster and has better compression ratio with acceptable loss in image quality.

7. Future Scope

Although we have utilized Fractal Image Compression scheme in sharpening and smoothing of image but it can be used in some other image processing fields like feature extraction, image watermarking, image denoising, image signature etc. In this paper only gray-scale images are considered but it can also be applied to color images. Mainly there are two methods to improve the encoding time of Fractal Image Compression scheme. The first approach is to reduce the size of domain pool and second approach is based on feature extraction. We have utilized only one feature which is standard deviation in future we can use other features also like skewness, neighbor contrast, beta, maximum gradient and compare which feature extraction technique will be best in terms of encoding time.

References

[1] B.Wohlberg and G.de Jager, A Review of the Fractal Image Coding Literature, IEEE Transactions on Image Processing, 1999, 8 (12):234~245.

[2] Hurd, L.P, Gustavus, M.A,Barnsley, M.F.etc.Fractal video compression. Compcon Spring '92. Thirty Seventh IEEE Computer Society International Conference, Digest of Papers[3] Ilsoo Yun; and Byungkyu (Brian) Park et al "Stochastic Optimization for Coordinated Actuated Traffic Signal Systems. "ASCE vol.138, No.7 (2012).

[3] JianjiWangn; Nanning Zheng; Yuehu Liu; Gang Zhou "Parameter analysis of fractal image compression and itsapplications in image sharpening and smoothing"Signal Processing: Image Communication 28 (2013) 681–687

[4] M.F.Barnsley, Fractals Everywhere, San Diego, CA: Academic press inc., 1988.

[5] A.E. Jacquin, Image coding based on a fractal theory of iterated contractive image transformations, IEEE Transaction on Image Processing ,1992,1(1):18~30.

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

[7] M.Nelson and J.L.Gailly, The Data Compression Book, Second edition, New York, M&T books, 1996.

[8] G.C.Freeland T.S. Durrani, "IFS fractals and the wavelet transform, " in the proceeding of IEEE ICASSP-90,Albuquerque,N.M.,apr 3-6,1990,pp 2345-2348.

[9] D. Brunet, E.R. Vrscay, Z. Wang, Structural similarity-based affine approximation and self-similarity of images revisited, in: Proceedings of the ICIAR, PartII, Lecture Notes in Computer Science, vol. 6754, 2011, pp. 264–275.

[10] H. Tamura, S. Mori, T. Yamawaki, Texture features corresponding to visual perception, IEEE Transactions on Systems, Man and Cybernetics 8 (6) (1978) 460–473.
[11] TahaM.Hasan, Xiangqian Wu, An adaptive algorithm for improving the fractal image compression ,Journal of multimedia, vol.6, No.6, December 2011 .

Author Profile

Gaurish Joshicompleted B.Tech. from Z.H.C.E.T.(AMU)Aligarhin year 2011 & M.Tech (Digital System)from MNNIT, Allahabad in year 2014& working as a Lecturer in Government. Polytechnic College, Sendhwa.