FRACTAL IMAGE COMPRESSION WITH ADAPTIVE QUARDTREE PARTITIONING

Utpal Nandi, Jyotsna Kumar Mandal

Dept. Of Computer Sc. & Engg. , University of Kalyani,
Nadia-741235, West Bengal, India
{nandi.3utpal,jkm.cse}@gmail.com

ABSTRACT

The image partitioning scheme in fractal image compression is one of the important aspect for enhancement of performance. In this paper, an adaptive quardtree partitioning scheme is proposed where the entire image is sub-divided recursively into four sub-images. The partitioning points are selected adaptively in a image context-dependent way instead of middle points of the image sides as in quardtree partitioning scheme. Biased successive differences of sum of pixel values of rows of the image are calculated to divide the image row-wise into two sub-images. Then, each sub-image is farther divided column-wise into two parts using biased successive differences of sum of pixel values of columns of the sub-image. Then, a fractal image compression technique is proposed based on the proposed partitioning scheme. The comparison of the compression ratio and PSNR are done between fractal image compression with quardtree and proposed adaptive quardtree partitioning schemes. The comparison of the compression time between the same is also done. The fractal image compression with proposed partitioning scheme offers better compression rates most of the times with comparatively improved PSNR. But, the compression time of the fractal image compression with proposed partitioning scheme is much more than its counterpart.

KEYWORDS

Fractal compression, compression ratio, Quardtree partition, Affine map, Iterated Function Systems (IFS), Partitioned Iterated Function Systems (PIFS).

1. INTRODUCTION

Currently, a lossy image compression technique got its popularity is fractal image compression technique [1, 2, 4, 5, 7, 8, 9, 14, 15]. The technique can offer much better compression ratios than DCT-based JPEG [11, 12, 15] compression technique. It is resolution independent. Therefore, it works well not only for achieving high compression ratios, but also for enlarging a complete image by some scale factor from same compressed image and zooming on a portion of an image. In fractal compression with PIFS, the input image is used as its own dictionary during encoding. The compression technique first partitions the input image into a set of non-overlapping ranges. There are many possible partitioning scheme [14] that can be used to select the range R_i like quardtree, HV and triangular partition. The process of partitioning image repeats recursively starting from the whole image and continuing until the partitions are covered with a specified tolerance or the partitions are small enough. For each range, the compression technique searches

Jan Zizka (Eds) : CCSIT, SIPP, AISC, PDCTA - 2013
pp. 289–296, 2013. © CS & IT-CSCP 2013 DOI : 10.5121/csit.2013.3633

for a part of the input image called a domain, which is similar to the range. The domain must be larger in size than the range to ensure that the mapping from the domain to the range is contractive in the spatial dimensions. To find the similarity between a domain D and a range R, the compression technique finds the best possible mapping T from the domain to the range, so that the image T(D) is as close as possible to the image R. The affine transformations are applied for this purpose. The best affine map from a range to a domain is found. This is done by minimizing the dissimilarity between range and mapped domain as a function of the contrast and brightness. The RMS metric or any other metric can be used to find the dissimilarity between range and mapped domain. The ways of partitioning image in this technique is one of the important factor determining the compression rate, compression time and quality of the image. One way to partition the image is quardtree partitioning scheme [14, 15] where the image is broken up into four equal-sized sub-images. The compression technique repeats recursively starting from the whole image and continuing until the squares are either covered with in some specified RMS tolerance or smaller than the specified minimum range size. But, there are some weakness of the quardtree based partitioning. It makes no attempt to partition the image in a image-context dependent way such that partitions share some self-similar structures. A way to remedy this problem i.e. an adaptive quardtree partitioning scheme is proposed. The variable position of the partition based on context of image makes the proposed partitioning scheme more flexible. The partitioning of the image is done in such a way that they share some self-similar structure. The detail of the proposed partitioning scheme is discussed in section 2. Then, a fractal image compression technique is proposed based on the proposed adaptive quardtree partitioning scheme and termed as Fractal Image Compression with Adaptive Quardtree Partitioning (FICAQP). The proposed compression technique is discussed in section 3. Results are given in section 4 and conclusions are done in section 5.

2. The Proposed Adaptive Quardtree Partitioning Scheme

Let, a RxC range contains the pixel values $P_{i,j}$ for $0 \leq i < R$ and $0 \leq j < C$, where R and C are the number of rows and columns of the range respectively and the top-left point of the rectangular range is (x,y) as shown in Fig. 1. The proposed partitioning scheme is a three step process. In the first step, the range of the image to be divided is partitioned horizontally. In order to partition the range horizontally, pixel value sum $\sum_{j=y}^{y+C-1} \mathrm{P}_{\mathbf{i,j}}$ is calculated for each pixel raw i, $x \leq i \leq x+R-1$. These sums are used to compute absolute value of successive differences between pixel value sum of ith and (i+1)th row i.e. ($\sum_{j=y}^{y+C-1} \mathrm{P}_{\mathbf{i,j}}$ - $\sum_{j=y}^{y+C-1} \mathrm{P}_{\mathbf{i+1,j}}$) for each pixel raw i, $x \leq i \leq x+R-1$. Then, a linear biasing function min(i, R - i - 1) is applied to each of these differences to multiply them by their distance from the nearest side of the rectangular range. This gives the biased horizontal differences between ith and (i+1)th row,

$$h_i = \min(i-x, R-i+x-1)\{|(\textstyle\sum_{j=y}^{y+C-1} p_{\mathbf{i,j}} - \sum_{j=y}^{y+C-1} P_{\mathbf{i+1,j}})|\}, \text{ for } x \leq i \leq x+R-1 \qquad (\mathbf{1})$$

Then, the value of i (say Rp) is found such that h_i is maximal that is mathematically,

$$\begin{gathered}Rp = \text{Maximize } [\min(i-x, R-i+x-1)\{|(\textstyle\sum_{j=y}^{y+C-1} p_{\mathbf{i,j}} - \sum_{j=y}^{y+C-1} P_{\mathbf{i+1,j}})|\}] \qquad (\mathbf{2}) \\ \text{Subject to } x \leq i \leq x+R-1\end{gathered}$$

The partition is made by the row number Rp to produce upper sub-ranges REC1 and lower sub-range REC2 shown in Fig.2. In the second step, the sub-range REC1 is partitioned vertically. In order to partition the sub-range REC1 vertically, sum of pixel values $\sum_{i=x}^{Rp-1} \mathrm{P}_{\mathbf{i,j}}$ is calculated for each column j, y≤j≤y+C-1 of REC1. Similarly, the absolute value of successive differences between sum of pixel values of jth and (j+1)th columns i.e. ($\sum_{i=x}^{Rp-1} \mathrm{P}_{\mathbf{i,j}}$ - $\sum_{i=x}^{Rp-1} \mathrm{P}_{\mathbf{i+1,j}}$) are computed for each pixel column j, y≤j≤y+C-1 and the liner biasing function min(j-y, C-j+y-1) is applied to each of these differences to calculate the biased vertical differences between jth and (j+1)th columns,

$$\text{v1j=min (j-y,C-j+y-1)} \{|(\sum_{i=x}^{Rp-1} \mathrm{P}_{\mathbf{i,j}} - \sum_{i=x}^{Rp-1} \mathrm{P}_{\mathbf{i,j+1}})|\},\ \text{for}\ y \le j \le y+C-1 \qquad (\mathbf{3})$$

Then, the value of j (say Cp1) is found such that v_{1j} is maximal that is mathematically,

$$\text{Cp1= Maximize [min (j-y, C-j+y-1)} \{|(\sum_{i=x}^{Rp-1} \mathrm{P}_{\mathbf{i,j}} - \sum_{i=x}^{Rp-1} \mathrm{P}_{\mathbf{i,j+1}})|\}] \qquad (\mathbf{4})$$
$$\text{Subject to } y \le j \le y+C-1.$$

The partition at vertical position Cp1 of REC1 is made to create two sub-ranges REC1.1 and REC1.2 as shown in Fig.3. Similarly, in the third step, the sub-range REC2 is partitioned vertically. The pixel value sum $\sum_{i=Rp}^{R-1} \mathrm{P}_{\mathbf{i,j}}$ is calculated for each column j, y≤j≤y+C-1 of REC2 that are used to compute absolute value of successive differences($\sum_{i=Rp}^{x+R-1} \mathrm{P}_{\mathbf{i,j}}$ - $\sum_{i=Rp}^{x+R-1} \mathrm{P}_{\mathbf{i,j+1}}$) for each pixel column j, y≤j≤y+C-1.

Similarly, biased vertical differences are calculated as

$$v_{2i}\text{= min (j-y,C-j+y-1)}\{|(\sum_{i=Rp}^{x+R-1} \mathrm{P}_{\mathbf{i,j}} - \sum_{i=Rp}^{x+R-1} \mathrm{P}_{\mathbf{i,j+1}})|\},\ \text{for}\ y \le j \le y+C-1 \qquad (\mathbf{5})$$

The value of j (say Cp2) is found such that v_{2j} is maximal that is mathematically,

$$\text{Cp2= Maximize [min (j-y, C-j+y-1)} \{ |(\sum_{i=Rp}^{x+R-1} \mathrm{P}_{\mathbf{i,j}} - \sum_{i=Rp}^{x+R-1} \mathrm{P}_{\mathbf{i,j+1}})|\}] \qquad (\mathbf{6})$$
$$\text{Subject to } y \le j \le y+C-1$$

The partition at vertical position Cp2 of REC2 is made to create two sub-ranges REC2.1 and REC2.2 as shown in Fig.4. Finally, Rp x Cp1 sub-range REC1.1, Rp x (C-Cp1) sub-range REC1.2, (R-Rp) x Cp2 sub-range REC2.1 and (R-Rp) x (C-Cp2) sub-range REC2.2 are formed with their top-left points (x,y), (x,Cp1), (Rp,y) and (Rp,Cp2) respectively. The algorithm of the proposed adaptive quardtree partitioning scheme is given in section 2.1.

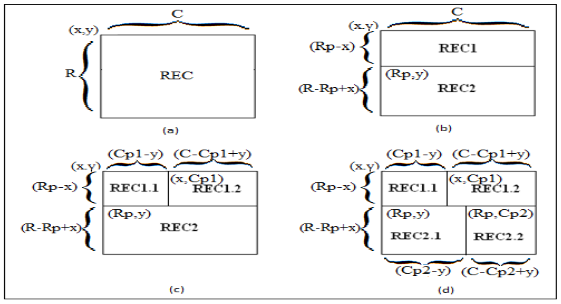

Fig. 1. (a) initial image, (b) after 1st step, (c) after 2nd step, (d) after 3rd step.

2.1 The algorithm of proposed adaptive quardtree partitioning scheme

Let us consider a RXC range, REC with top-left point (x,y) where R and C are the number of rows and columns of the REC respectively. The detail algorithm of proposed adaptive quardtree partitioning scheme is given in Fig. 2.

3. THE PROPOSED FRACTAL IMAGE COMPRESSION WITH ADAPTIVE QUARDTREE PARTITIONING SCHEME

The proposed image compression technique is quite similar to the existing fractal image compression technique except the used proposed partitioning scheme. Initially, a grey scale rectangular image F is taken as input for compression and a tolerance level Q (Quality) is set where encoding with lower value of Q will have better fidelity. The minimum range size R_{min} is also set. A range with size R_{min} is mapped even if there is no such domain that covers the range with the specified tolerance level Q. Initially, the whole image is the initial range R_1 that is marked as uncovered. Then, another parameter i.e. domain density is also set that determines domain step (distance between two consecutive domains) and a domain pool is created where each domain must be larger than the ranges to ensure contractive mapping from domain to range in spatial dimension. The detail algorithm of proposed technique is shown in Fig. 3.

4. RESULTS

The comparison of compression ratios, compression times and PSNRs among fractal image compression technique with quardtree partitioning scheme (FICQP) and the proposed fractal image compression technique with adaptive quardtree partitioning scheme (FICAQP) have been made in table I using five gray scale images with a domain density value of 1 and image quality 1. The graphical representations of the same are shown in Fig. 4 (a), (b) and (c) respectively. The proposed technique offers comparatively better rate of compression than its counterpart for all five images. But, the compression times taken by the proposed technique are much more than existing technique for all five images. The PSNR in proposed technique are improved significantly than the existing technique for all five images. The effect on compression ratio, compression time and PSNR for increasing the quality value Q in both FICQP and FICAQP for a particular domain density (i.e. D=1) and image

```
Step 1: Initially, set Hmax = Vmax1 = Vmax2 = 0;
Step 2:  /*Find the horizontal partitioning row number Rp */
              For i = x to x+R-1, do
              X1=X2=0;
              For j = y to y+C-1, do
                  X1 = X1 + P[i][j];  X2 = X2 + P[i+1][j]; End for;
              H = min (i-x,R-i+x-1) * ABS (X1 – X2);
              If H > Hmax, then  Hmax = H;  Rp = i; End if;  End for;
Step 3: Partition the range REC horizontally into two rectangles REC1 and REC2
        using row number Rp with top-left points (x,y) and (Rp,y) respectively;
Step 4: /*Find the vertical partitioning column number Cp1 of range REC1 */
              For j = y to y+C-1, do
              Y1 = Y2 = Y3 = Y4 = 0;
              For i = x to Rp-1, do
                  Y1 = Y1 + P[i][j];   Y2 = Y2 + P[i][j+1]; End for;
                   V1 = min (j-y,C-j+y-1) * ABS (Y1 – Y2);
              If V1 > Vmax1, then  Vmax1 = V1;  Cp1 = j;  End if; End for;
Step 5: Partition the range REC1 vertically into two rectangles REC1.1 and
        REC1.2 using column number Cp1 with top-left points (x,y) and (x,Cp1)
        respectively.
Step 6: /*Find the vertical partitioning column number Cp2 of range REC2 */
           For j = y to y+C-1, do
              For i = Rp to x+R-1, do
                  Y3 = Y3 + P[i][j];  Y4 = Y4 + P[i+1][j];  End for;
              V2 = min (j-y,C-j+y-1) * ABS (Y3 – Y4);
              If V2 > Vmax2, then  Vmax2 = V2;  Cp2 = j;  End if;  End for;
Step 7: Partition the range REC2 vertically into two rectangles REC2.1 and
        REC2.2 using column number Cp2 with top-left points (Rp,y) and
        (Rp,Cp2) respectively.
Step 8: Return four sub-ranges i.e. REC1.1, REC1.2, REC2.1 and REC2.2 with
        their corresponding top-left points (x,y), (x,Cp1), (Rp,y) and (Rp,Cp2)
        respectively.
Step 9: Stop.
```

Fig. 2. The proposed Adaptive Quardtree Partitioning scheme.

```
Step 1: A gray scale rectangular image F is taken as input and a tolerance level
        quality (Q)  (Encoding with lower Q will have better fidelity) is chosen.
Step 2: Minimum range size $R_{min}$ is selected and the initial range $R_1$ is marked
        as uncovered.
Step 3: Domain density value is selected and a domain pool D is created where
        each domain must be larger than the range.
Step 4: Classify all the domain by the ordering of the image brightness in the
        four quadrants of the domain.
Step 5: While ( There are uncovered ranges $R_i$), do
Step 5.1:  Find the class $C_i$ of range $R_i$ by the ordering of the image brightness
           in the four quadrants of the range.
Step 5.1:     Find the domain Di from domains of similar class $C_i$ of the domain
              pool D and the corresponding affine map Ti that cover range $R_i$
              better than other i.e. that minimize the expression $D_{rms}(F\cap(R_i \text{ X } I),T_i(F))$ ,where I means interval [0,1] and RMS(root mean squre)
              metric

                  $D_{rms}(F1,F2)=\sqrt{\int (F1(x,y) - F2(x,y))dxdy}.$

Step 5.2:     If ( ( $D_{rms}(F\cap(R_i \text{ X } I),T_i(F)) < Q$ ) OR ( SIZE($R_i$) <= $R_{min}$ ) ),then
Step 5.2.1:       Range $R_i$  is marked as covered and the affine transformation $T_i$
                  is written out to the output  file.
              Else
Step 5.2.2:       Partition range Ri into four sub- ranges $R_{i1}$, $R_{i2}$, $R_{i3}$, $R_{i4}$ using
                  proposed Adaptive Quardtree Partitioning scheme.  Produced
                  sub-ranges are marked as uncovered.
              End if;
        End while;
Step 6:  Stop.
```

Fig. 3. Algorithm of fractal image compression using PIFS with Adaptive Quardtree Partitioning scheme.

Table 1. Comparison of compression ratios, times and PSNRs.

File name	FICQP %comp	FICQP Time (m.sec)	FICQP PSNR (dB)	FICAQP %comp	FICAQP Time (m.sec)	FICAQP PSNR (dB)
CHEETAH.GS	82.56	140	27.50	82.76	310	27.93
LISAW.GS	82.59	170	38.91	82.91	360	41.24
ROSE.GS	87.45	130	28.97	87.50	290	29.25
MOUSE.GS	85.87	150	38.03	85.93	340	38.67
CLOWN.GS	82.82	190	29.20	83.13	420	29.72

(MOUSE.GS) have been given in Table II and the graphical representations of the same are shown in Fig. 5 (a), (b) and (c) respectively. The compression ratios increase and the compression times decrease with the increasing value of quality Q in both the techniques. But, the PSNRs are reduced.

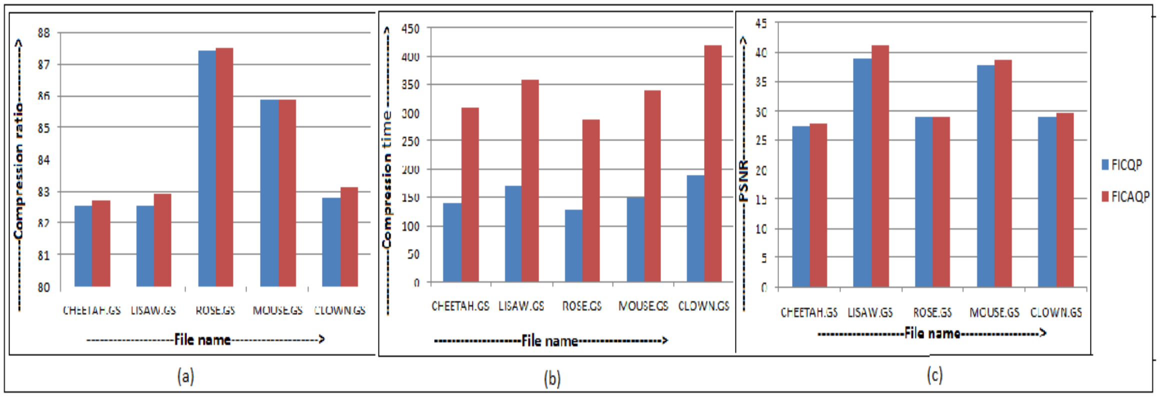

Fig. 4. Comparison of (a) compression ratio, (b) compression time and (c) PSNR .

Table 2. Comparison of compression ratios, times and PSNRs with varying quality values.

Quality Factor(Q)	FICQP %comp	FICQP Time (m.sec)	FICQP PSNR (dB)	FICAQP %comp	FICAQP Time (m.sec)	FICAQP PSNR (dB)
1	85.87	150	38.03	85.93	340	38.67
3	92.06	90	37.66	92.19	200	37.94
5	94.37	50	36.86	94.48	130	36.93
7	96.10	40	35.71	96.36	110	35.85
10	97.45	20	34.08	97.31	60	34.13
12	97.96	20	33.07	97.95	40	33.15
15	98.30	20	32.20	98.44	50	32.21

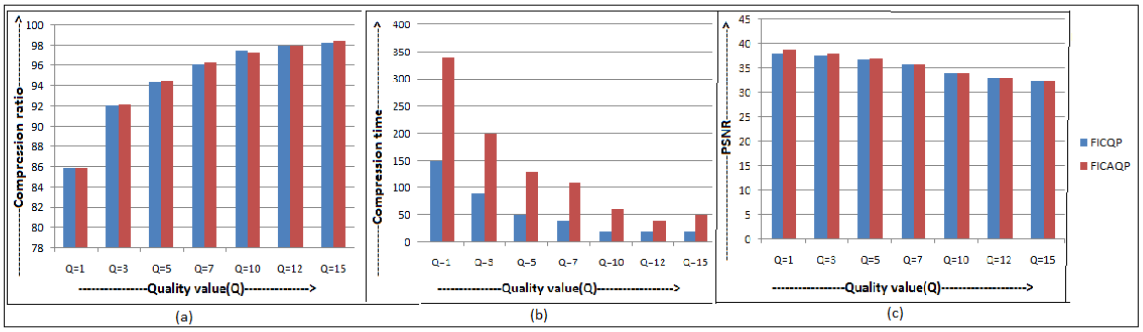

Fig. 5. Quality value vs. (a) compression ratio, (b) Compression time and (c) PSNR .

5. CONCLUSIONS

The proposed adaptive quardtree partitioning scheme sub-divides the entire image into four sub-images and the partitioning points are selected adaptively in a context-dependent way such that some self-similar structure are shared. The proposed compression technique uses the proposed adaptive quardtree partitioning scheme and offers better compression than its counterpart for all most all images. The PSNR of the proposed technique between original image and image after compression decompression cycle are also significantly improved. But, the compression time is much more than the existing fractal image compression technique with quardtree partition. Farther investigation is needed to reduce the compression time of the proposed technique without loss of PSNR. In the proposed technique, ranges and domain are compared only with the same orientation. The compression ratios could be further improved by taking into account possible symmetries and rotations between ranges and domains and by using any loss-less data compression techniques like Huffman or arithmetic encoding on the parameters of affine transformations.

ACKNOWLEDGEMENTS

The authors extend sincere thanks to the department of Computer Science and Engineering and PURSE Scheme of DST, Govt. of India and Academy Of Technology, Hooghly, West Bengal, India for using the infrastructure facilities for developing the technique.

REFERENCES

[1] Fisher, Y.: Fractal Image Compression: Theory and Application, Springer Verlag, New York (1995).
[2] Cardinal, J, : Fast Fractal Compression of Greyscale Images : IEEE transactions on image processing, vol. 10, no. 1 (2001).
[3] Barnsley, M.F. : Fractals Everywhere, Academic Press,2nd edition, San Diego (1993).
[4] Jacquin, A.: Fractal image coding based on a theory of iterated contractive image transformations : Proc. SPIE Visual Communications and Image Processing, pages 227-239 (1990).
[5] Belloulata, K.., Stasinski, R., Konrad, J., : Region based Image compression using fractals and Shape adaptive DCT : in proceeding of IEEE international conference on image processing,1999, icip99, October 24-28, vol.2, ISBN : 0-7803-5467-2, pp. 815-819 (1999).
[6] Mandelbrot, B.B, :The Fractal Geometry of Nature : W.H. Freeman and company, New York (1983).
[7] Hutchinson, J.E., : Fractals and self-similarity : Indiana University Mathematics Journal, Volume 3, Number 5, pages 713-747 (1981).
[8] Kominek, J., : Algorithm for fast fractal image compression : in Proc. IS&T/SPIE 1995 Symp. Electronic Imaging: Science Technology, vol.2419, (1995).
[9] Wohlberg, B. E., G. de Jager,: Fast image domain fractal compression by DCT domain block matching : Electron. Lett., vol. 31, pp. 869–870, (1995).
[10] Lu, N., : Fractal Imaging : New York: Academic, (1997).
[11] Wallace, Gregory K., : The JPEG Still Picture Compression Standard : Communications of the ACM, Volume 34, No. 4, pp 31-44 (1999).
[12] Pennebaker, William B., Joan L. Mitchell, L., : JPEG Still Image Data Compression Standard : New York, Van Nostrand Reinhold (1992).
[13] Salomon , D., : Data Compression, The complete reference : ed. Third , Springer, (2002).
[14] Fisher , Y., : Fractal Image Compression, Theory and application : ed. Second , Springer,(2008).
[15] Nelson , M., : The Data Compression Book : ,ed. Second , India, BPB Publications (2008).