

Fast Search Strategies using Optimization for Fractal Image Compression

D.Venkatasekhar*

Associate Professor
Department Of CSE
Annamalai University
Chidambaram.
india

*ramavenkatasekar {at} yahoo.co.in

P.Arana

Professor
Department Of CSE
Annamalai University
Chidambaram
india

B.Parthiban

M.E Student
Department Of CSE
Annamalai University
Chidambaram
india

Abstract—In fractal image compression encoding procedure is time-consuming due to a full search mechanism. In order to speedup the encoder, we adopt particle swarm optimization (PSO) method performed under classification and Dihedral transformation to reduce the encoding time. This PSO based proposed technique is used for the mean square error (MSE) based on the stopping criterion between range block and domain block. Dihedral transformation, only four transformations for each domain block are considered so as to reduce the encoding time. An experimental result on encoding time of the proposed method is faster than that of the conventional methods on good image quality.

Keywords—Fractal image compression, particle swarm optimization, Dihedral transformation, discrete wavelet transform, Encoding time.

I. INTRODUCTION

The idea of the image redundancies can be efficiently exploited by means of block self-affine transformations may call the fractal image compression (FIC). In based on the partitioned iteration function system (PIFS) which utilized the self-similarity on first practical fractal image compression scheme was introduced in 1992 by Jacquin [1]. The fractal transform for image compression was introduced in 1985 by Barnsley and Demko. The very high encoding time is the main disadvantages because of exhaustive search strategy. Therefore, decreasing the encoding time is an interesting research topic for FIC.

One way of decreasing the encoding time is by using stochastic optimization methods using Genetic Algorithm (GA) this recent topics of GA-based methods are proposed to improve the efficiency [2]. The idea of special correlation of an image is used in these methods while the chromosomes in GA consist of all range blocks which leads to high encoding speed[3].

Other researchers focused on improvements by tree structure search methods of the search process and parallel search methods [4, 5] or quad tree partitioning of range blocks to make it faster.

Wavelet transform is used to decompose the original image to various frequency sub bands in which the attributes can be extracted from the wavelet coefficients belonging to

different sub-bands. The distribution of wavelet coefficients can be used in context based multiscale classification of document image [6]. The fast and efficient algorithm [7] was applied to triangular mesh to approximate surface data using wavelet transform coefficients. It directly determined local area complexity in an image and divides square cells depending on complexity. In implemented a hybrid image classification method combining wavelet transform, rough set approach, and artificial neural network. Zou and Li have proposed image classification using wavelet coefficients in low-pass bands [8]. This approach was based on the distribution of histograms of the wavelet coefficients.

In this paper, it use particle swarm optimization method to reduce the search space for FIC. If the two blocks are not of the same type no similarity will not be calculated. The classification method is to partition all of the blocks in domain pool and range pool into three classes according to third level wavelet coefficients. Each range block calculates the similarity measure only with the domain block from the same class. In the meanwhile, we consider the special property of the Dihedral transformation so that only four transformations are required to calculate the similarity. Therefore the encoding time can be further reduced. Experiments are conducted on 1 image using 4 methods including the full search method, discrete wavelet transform (DWT), PSO, and the proposed method. Experimental results show that the proposed method outperform all the other 3 methods. In average, the proposed method is about 178 times faster in comparison to the full search method with only 1.46 dB decay in image quality. Comparing to Wu's schema genetic algorithm (SGA) [2], the proposed method is better than the performance of SGA method. Moreover, the encoding speed of the proposed method is about 174 times faster than that of the full search method, while the penalty of 0.9 dB decays for retrieved image quality.

II. FRACTAL IMAGE COMPRESSION

In local self-similarity property in a nature images. The fundamental idea is coming from the Partitioned Iterated Function System (PIFS). Suppose the original gray level image f is of size $m \times m$. Let the range pool R be defined as the set of all non-overlapping blocks of size $n \times n$ of the image f , which makes up $(m/n)^2$ blocks. For obeying the

Contractive Mapping Fixed-Point the domain block must exceed the range block in length. Let the domain pool D be defined as the set of all possible blocks of size $2n \times 2n$ of the image f , which makes up $(m - 2n + 1)^2$ blocks. For m is 256 and n is 8, the range pool R is composed of $(256/8) \times (256/8) = 1024$ blocks of size 8×8 and the domain pool D is composed of $(256 - 16 + 1) \times (256 - 16 + 1) = 58081$ blocks of size 16×16 . For each range block v from the R , in the fractal affine transformation is constructed by searching all of the domain blocks in the D to find the most similar one and the parameters representing the fractal affine transformation will form the fractal compression code for v .

To execute the similarity measure between range block and domain block, In the size of the domain block must be first sub-sampled to 8×8 such that its size is the same as the range block v . Let u denote a sub-sampled domain block. The similarity of two image blocks u and v of size $n \times n$ is measured by mean square error (MSE) define

$$MSE(u, v) = \frac{1}{n \times n} \sum_{j=0}^{n-1} \sum_{i=0}^{n-1} (u(i, j) - v(i, j))^2. \quad (1)$$

The fractal transformation allows the eight Dihedral transformations in Table 1, $T_k: k = 0, \dots, 7$, of the domain blocks. If the coordinate origin is assumed to locate at the center of the block, the transformations T_1 and T_2 correspond to flip the block along horizontal and vertical line, respectively. T_3 is the flip along both horizontal and vertical lines. T_4, T_5, T_6 and T_7 are the transformations of T_0, T_1, T_2 and T_3 performed by an additional flip along the main diagonal line, respectively.

The fractal coder also allows the adjustment of the contrast scaling p and the brightness offset q on the block u . Thus the similarity is to minimize the quantity $e = \|p u_k + q - v\|$, where $u_k = T_k(u), 0 \leq k \leq 7$, are the 8 orientations of u . By direct computation, p and q can be computed by

$$p = \frac{n^2 \langle u_k, v \rangle - \langle u_k, \vec{I} \rangle \langle v, \vec{I} \rangle}{n^2 \langle u_k, u_k \rangle - \langle u_k, \vec{I} \rangle^2},$$

$$q = \frac{\langle v, \vec{I} \rangle - p \langle u_k, \vec{I} \rangle}{L^2} \quad (2)$$

where L^2 is the number of pixels in the block of the range pool 8×8 vector with all entries being 1, and the Euclidean inner product of two vectors. Finally, the position (i, j) of the domain block (after sub-sampled, it is denoted by u), the contrast scaling p , the brightness offset q , and the orientation k constitute the fractal code of the given range block v . For each range block v , it will make up a fractal code of i, j, k, p and q .

In the decoding process, In first make up the 1024 affine transformations from the compression codes. We choose one arbitrary image as the initial image and then perform the 1024 affine transformations on the image to obtain a new one. The transformation is proceeded recursively.

According to Contractive Mapping Fixed-Point sequence of image will converge. The stopping criterion of the recursion is designed according to user's application and the converged image is the retrieved image of fractal coding.

III. FAST FRACTAL IMAGE ENCODING

A. Particle Swarm Optimization (PSO)

A population-based algorithm is PSO for searching global optimum. To simulate a simplified social behavior is the way of original idea of PSO [9]. Similar to the crossover operation of the GA, in PSO the particles are adjusted toward the best individual experience (PBEST) and the best social or global experience (GBEST). However, PSO is unlike a GA, why because in that each potential solution, particle is "flying" through hyperspace with a velocity, the particles and the swarm have memory for process; in the population of the GA memory does not exist.

Let $x_{j,d}(t)$ and $v_{j,d}(t)$ denote the d^{th} dimensional value of the vector of position and velocity of j^{th} particle in the swarm, respectively, at time t . The PSO model can be expressed as

$$v_{j,d}(t) = v_{j,d}(t-1) + c_1 \cdot \phi_1 \cdot (x_{j,d}^* - x_{j,d}(t-1)) + c_2 \cdot \phi_2 \cdot (x_d^{\#} - x_{j,d}(t-1)), \quad (3)$$

$$x_{j,d}(t) = x_{j,d}(t-1) + v_{j,d}(t), \quad (4)$$

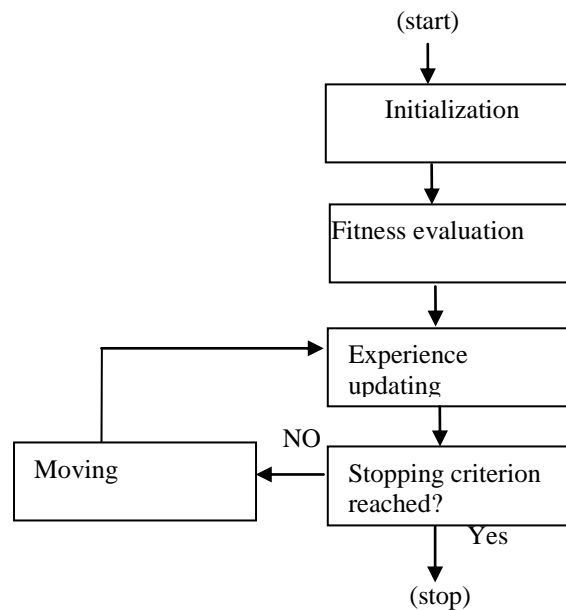


Figure 1. Block diagram of PSO

Where $x_{j,d}^*$ (PBEST) denotes the best position of j^{th} particle up to time $t-1$ and $x_d^\#$ (GBEST) denotes the best position of the whole swarm up to time $t-1$, φ_1 and φ_2 are random numbers, and c_1 and c_2 represent the individuality and sociality coefficients, respectively.

The steps involved here is the population size is first determined, and the velocity and position of each particle are initialized. Each particle moves according to Equations (3) and (4) fitness is then calculated. Meanwhile, the best positions of each swarm and particles are recorded. Finally, as the stopping criterion is satisfied, the best position of the swarm is the final solution. The block diagram of PSO is displayed in Fig 1 and the main steps are given as follows:

- Set the swarm size. Initialize the velocity and the position of each particle randomly.
- For each j , evaluate the fitness value of x_j and update the individual best position $x_{j,d}^*$ if better fitness is found.
- Find the new best position of the whole swarm. Update the swarm best position $x^\#$ if the fitness of the new best position is better than that of the previous swarm.
- If the stopping criterion is satisfied, then stop.

B. Dihedral Transformation

To each range block, we must calculate the similarity measure with all eight transformed blocks of domain block according to the Dihedral transformations in Table 1 to find the best match. The relations of F_{10} and F_{01} after applying Dihedral transformations are show in Table 2, It is clear that all items are $\pm F_{10}$ or $\pm F_{01}$. This fact can be utilized to further reduce the encoding time. For example, the transformation T_1 is to flip the block along the horizontal line. Relations between the coefficients of the T_1 transformed block to the original block f can be easily calculated as

TABLE 1. THE 8 TRANSFORMATIONS IN THE DIHEDRAL GROUP

T_0	T_1	T_2	T_3
$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$
T_4	T_5	T_6	T_7
$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix}$

$$F_{10}(T_1) = \frac{1}{4} \left(\sum_{j=4}^7 \sum_{i=0}^3 f(i, j) + \sum_{j=4}^7 \sum_{i=0}^3 f(i, j) \right) - \frac{1}{4} \left(\sum_{j=4}^7 \sum_{i=4}^3 f(i, j) + \sum_{j=0}^3 \sum_{i=4}^3 f(i, j) \right) = F_{10}(T_0),$$

$$F_{01}(T_1) = \frac{1}{4} \left(\sum_{j=4}^7 \sum_{i=0}^3 f(i, j) + \sum_{j=4}^7 \sum_{i=0}^3 f(i, j) \right) - \frac{1}{4} \left(\sum_{j=0}^3 \sum_{i=0}^3 f(i, j) + \sum_{j=0}^3 \sum_{i=4}^3 f(i, j) \right) = F_{01}(T_0).$$

Following the Dihedral transformations, the coefficient pairs $(|F_{01}|, |F_{10}|)$ and $(|F_{01}|, |F_{10}|)$ are located symmetrically with respect to the ray $\theta = 45^\circ$ in the first quadrant in Fig. 2. With this relation, It separate the unit circle in the first quadrant into two regions Ω_1 and Ω_2 according to the line $\theta = 45^\circ$. For a given range block v , we pick a domain block u . If u and v are located in the same region, say Ω_1 , then we need only four Dihedral transformations, $T_k: k = 0\sim 3$ performed on u because the transformations $T_k: k = 4\sim 7$ will move u another region i.e., Ω_2 . The 4 transforms $T_k: k = 0\sim 3$ or $T_k: k = 4\sim 7$ have the same edge directions since their $|F_{10}|$ and $|F_{01}|$ are the same. Therefore, there are only four MSE computations required. On the other hand, if u and v are located in the different region, then we need only four Dihedral transformations, i.e., $T_k: k = 4\sim 7$. From the argument above, the amount of MSE computations will be reduced two times.

TABLE 2. RELATIONS OF F_{10} AND F_{01} AFTER APPLYING DIHEDRAL TRANSFORMATIONS.

	T_0	T_1	T_2	T_3	T_4	T_5	T_6	T_7
HL3	$+F_{10}$	$+F_{10}$	$-F_{10}$	$-F_{10}$	$+F_{01}$	$+F_{01}$	$-F_{01}$	$-F_{01}$
LH3	$+F_{01}$	$-F_{01}$	$+F_{01}$	$-F_{01}$	$+F_{10}$	$-F_{10}$	$+F_{10}$	$+F_{10}$

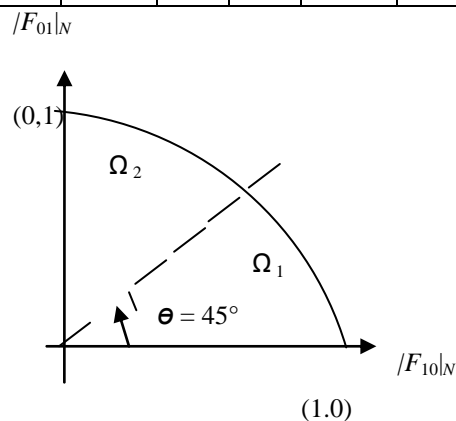


Figure 2. Illustration of the Dihedral property.

IV. PROPOSED METHOD

In the proposed fast fractal encoding using PSO, reduce the encoding time by reducing the searching time to find a best match domain block for the given range block from all domain blocks.

1. Set the swarm size must be proportional to $(N-2L+1)^2/(\text{maximum no. of iterations for PSO})$ and initialize the each particle velocity and the position randomly
2. The fitness value includes finding , MSE between domain block specified by the particles position and given range block
3. Update swarm best position if the fitness of the new best position is better than that of the previous swarm.
4. If swarm best position is not changed for some percentage of maximum iteration for PSO, then stop
5. The best position of the particles is updated using Equation (3) and (4) and go to step 2.

V. EXPERIMENTAL RESULTS

The results have been compared to the full search FIC mentioned in the previous sections in terms of encoding time and PSNR of fast fractal encoding using PSO.

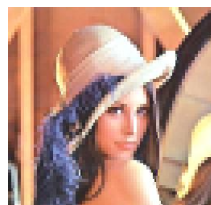
The distortion or error between the original image f and the decoded image g caused by lossy compression process is measured in peak signal to noise ratio (PSNR) defined by

$$\text{PSNR} = 10 \log_{10} \left(\frac{255^2}{\text{MSE}(f, g)} \right)$$

where MSE is defined in Equation (1). The tested image on Lena in which each is a gray scale image of size 256×256 selected from the CVG-UGR image database [10] The related PSO parameters swarm size, number of clusters, and inertial weight, are set as 40, 4 and 0.9, respectively. The velocity in is limited in and the maximal number of iterations is set as 30.



Figures 3.(a) Original image of Lena



3. (b) Full Search



3. (c) Wavelet Classification



3. (d) PSO



3. (e) SGA



3. (f) Proposed method

TABLE 3. SIMULATION RESULTS FOR PSNR(DB) AND COMPARISON.

image	methods				
	Full search	Wavelet Classification	PSO	SGA	Proposed Method
lena	28.910	28.812	27.643	27.338	9.2450

TABLE 4. SIMULATION RESULTS FOR CPU TIME(SEC) AND COMPARISON.

image	methods				
	Full search	Wavelet Classification	PSO	SGA	Proposed Method
lena	1433.281	482.953	41.281	14.610	0.5000

TABLE 5. SIMULATION RESULTS FOR MSE COMPUTATION AND COMPARISON.

image	methods				
	Full search	Wavelet Classification	PSO	SGA	Proposed Method
lena	475,799,552	158,664,808	10,166,272	2,835,497	7.7371

Figs. 3 (a) show the original image lena for decoding, Figs. 3(b) (c) (d) (e) (f) show retrieved images of Full Search, Wavelet Classification, PSO,SGA and proposed methods have PSNR 28.910 dB, 28.812 dB 27.643 dB, 27.338 dB, 9.2450 dB respectively. The results of the proposed method is listed in the tabular column. Compared to the Full search method, the speedup ratio is about three times faster. The detailed results of PSNR, executing time, and the amount of MSE computations and CPU time of the image listed in Tables (3-4-5) respectively. The retrieved image qualities are very close. The CPU time of the proposed method is 0.5000 seconds, which is the least. The speedup ratio with respect to the full search method shown in that is low. Under the condition of similar quality of decoded images, the encoding time of the proposed method reduces about 1.78 times which is better than that of the SGA

method. As an complexity analysis, the amount of MSE computations in the proposed method for lena image is 7.7371 which is 225 times of the amount of the full search method, which is shown in the last column. As demonstrated, the proposed method has better performance than that of other methods.

VI. CONCLUSION

In this paper, particle swarm optimization method is adopted with classification and Dihedral transformation in order to speedup the fractal image encoder. By using particle swarm optimization (PSO) based proposed method for fractal coding can reduce CPU time, PSNR and MSE Values and produces better compression ratio at acceptable quality, when comparing with existing full search, wavelet classification and SGA methods.

REFERENCES

- [1] E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Transactions on Image Processing*, Vol. 1, 1992, pp. 18-30.
- [2] M. S. Wu, J. H. Jeng, and J. G. Hsieh, "Schema genetic algorithm for fractal image compression," *Engineering Applications of Artificial Intelligence*, Vol. 20, 2007, pp. 531-538.
- [3] T. K. Truong, C. M. Kung, J. H. Jeng, and M. L. Hsieh, "Fast fractal image compression using spatial correlation," *Chaos, Solitons and Fractals*, Vol. 22, 2004, pp. 1071-1076.
- [4] C. Hufnagl, A. Uhl, Algorithms for fractal image compression on massively parallel SIMD arrays, *Real-Time Imaging* 6 (2000) 267–281.
- [5] D. Vidya, R. Parthasarathy, T.C. Bina, N.G. Swaroopa, Architecture for fractal image compression, *J. Syst. Archit.* 46 (2000) 1275–1291.
- [6] J. Li and R. M. Gray, "Context-based multiscale classification of document images using wavelet coefficient distributions," *IEEE Transactions on Image Processing*, Vol. 9, 2000, pp. 1604-1616.
- [7] H. J. Yu and J. B. Ra, "Fast triangular mesh approximation of surface data using wavelet coefficients," *The Visual Computer*, Vol. 15, 1999, pp. 1432-2315.
- [8] W. Zou and Y. Li, "Image classification using wavelet coefficients in low-pass bands," in *Proceedings of International Joint Conference on Neural Networks*, 2007, pp. 114-118.
- [9] M.S. Wu, W.C. Teng, J.H. Jeng, J.G. Hsieh, Spatial correlation genetic algorithm for fractal image compression, *Chaos Solitons & Fractals* 28 (2) (2006) 497–510.
- [10] CVG-UG Image Database, 2007, <http://decsai.ugr.es/cvg/dbimagenes/index.php>.