Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 3, March 2014, pg.722 – 726

RESEARCH ARTICLE

Upgrade PIFS to Patterns Recognition

Firas Sabar Miften

Computer Department, College of Education, Thi_Qar University, Iraq

firassebar@gmail.com

Abstract-In this paper, proposed a new method for Patterns Recognition by upgrade PIFS (Partitioned Iterated Function Systems) which used for image compression. PIFS represent affine transformations which when iteratively applied to the range-domain pairs in an arbitrary initial image. In our method the domains construct from many image belong to same class of original image rather than domain construct from one image.

Keyword- pattern; recognition; PIFS (Partitioned Iterated Function Systems); fractal

I. INTRODUCTION

Now, all the research on pattern recognition has been conducted in two steps. The first step, called a feature extraction, is the representation of an image by a discrete number of values called pattern features, seen as elements of a feature vector. These features are typically areas, lengths, shape factors, Fourier descriptors and co-occurrence texture descriptors [2,4,7]. The second step, called a classification, is the allocation of a wear particle represented by a feature vector to a particular class. Other methods used to classifying surfaces are based on fractals, e.g. fractal dimension [3,5] . In this paper, a new pattern recognition method, PIFS has been developed and applied to computer generated images of fractal surfaces. Images of fractal surfaces were used to evaluate the classification accuracy of the method.

II. PARTITIONED ITERATED FUNCTION SYSTEMS (PIFS)

Fisher [6], Suppose we are dealing with a 256 x 256 pixel image in which each pixel can be one of 256 levels of grey (ranging from black to white) . Let R1, R2,..., R1024 be the 8x8 pixel non-overlapping sub-squares of the image, and let D be the collection of all 16 x 16 pixel (overlapping) sub-squares of the image figure 1. The collection D contains 241 • 241 = 58,081 squares. For each R, search through all of D to find a Di € D which minimizes (Equation 1); that is, find the part of the image that most looks like the image above Ri. This domain is said to cover the range. Also, a square in D has 4 times as many pixels as an Ri, so we must either subsample (choose 1 from each 2x2 sub-square of Di) or average the 2 x 2 sub-squares corresponding to each pixel of Ri, when we minimize (Equation 1). Minimizing (Equation 1) means two things. First, it means finding a good choice for Di (that is the part of the image that most looks like the image above Ri,). Second, it means finding good contrast and brightness settings si and oi for wi. For each D € D we can compute si and oi using least squares regression, which also gives a resulting root mean square (RMS) difference. We then pick as Di the D € D with the least RMS difference. A choice of Di, along with a corresponding si and oi, determines a map wi, of the form of (Equation 2). Once we have the collection w1….w1024 we can decode the image by estimating xw. Fig. 1 shows four images: an initial image f0 chosen to show texture; the first iteration W(f0), which shows some of the texture from f0; W°2(f0); and W°lo(f0). The result is surprisingly good, given the naive nature of the encoding algorithm. Figure 1 shows how detail is added at each iteration. The first iteration contains detail at size 8x8, the next at size 4x4, and so on. Jacquin [1] originally encoded images with fewer grey levels using a method similar to this example but with two sizes of ranges. In order to reduce the number of domains searched, he also classified the ranges and domains by their edge (or lack of edge) properties. This is very similar to the scheme used by Boss et al. [1] to encode contours.

$$d_{rms}(f \cap (R_i \times I), w_i(f)) \quad i = 1, \ldots, N. \tag{1}$$

$$w_i \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_i & b_i & 0 \\ c_i & d_i & 0 \\ 0 & 0 & s_i \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \\ o_i \end{bmatrix} \tag{2}$$

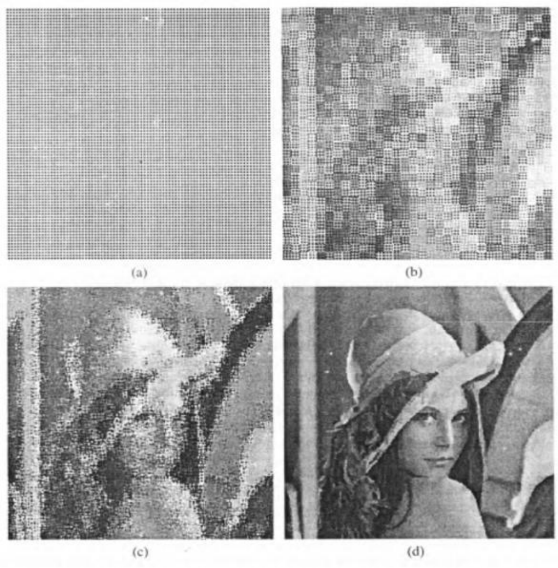

Fig. 1 The initial image (a). and the first (b). second (c). and tenth (d) iteration at the encoding transformations

III. UPGRADE PIFS (UPIFS)

The PIFS can be described as follows. We divide an image into fixed number of non-overlapping range blocks (R). We create a list of a domain blocks (D). The list consists of an overlapping area of the image, larger than the range blocks (usually two times larger) see fig. 1. Next for every range block R we search for the domain block D such that the value ρ (R,F(D)) is the smallest, where ρ is a metric and F is a transformation determined by the position of the R and D blocks.

Now the upgrade on the PIFS is ,as an example we take four images and calculate the energy for all image when, as the image that has the largest energy use it to extract the Range blocks where remained images using to extract Domain blocks therefore, PIFSs constructed from one range and three Domains blocks as in Fig. 3.

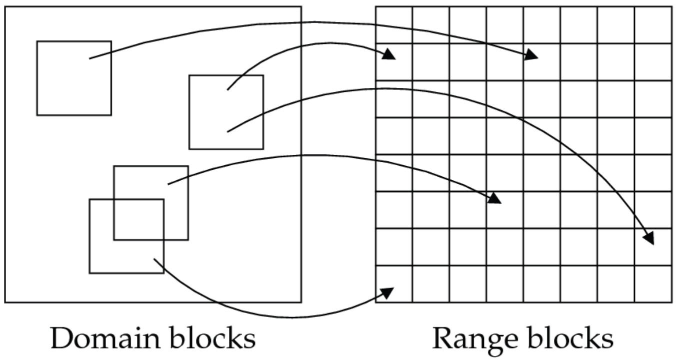

Fig. 2: Domain and Range blocks

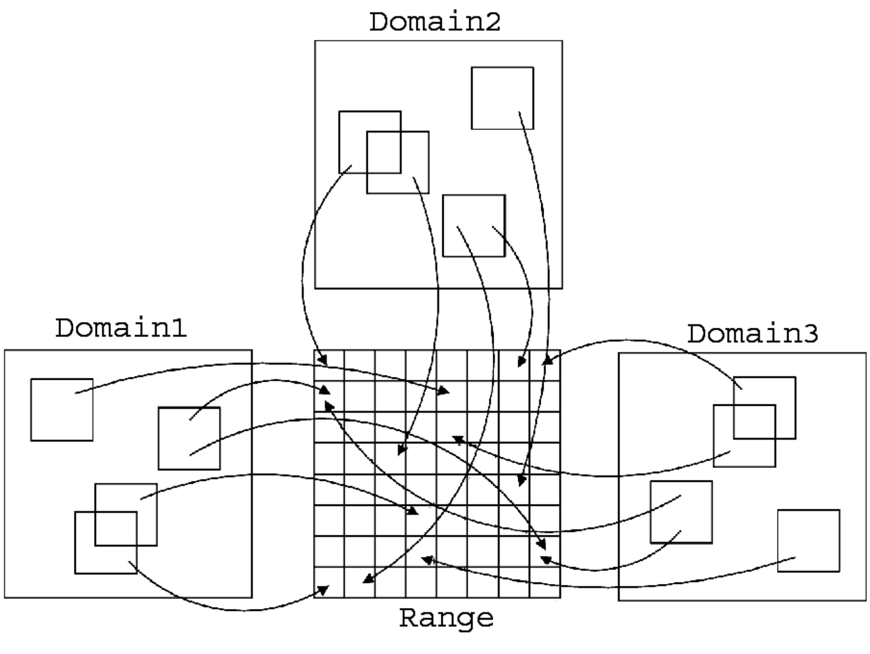

Fig. 3: Show upgrade to PIFS

IV. RESEARCH METHOD

A pattern recognition system based on the PIFS method has been developed for the classification of images. The basic idea behind this method is that UPIFSs constructed for classified images using are decoded using an unclassified image as the initial image. If the unclassified image exhibits a similar morphology to the already classified image, then the decoded image obtained just after one iteration will be 'similar' to the unclassified image. The unclassified image is assigned to the same class as the decoded image. If the decoded image is 'different', then the unclassified image belongs to a different class and a UPIFS constructed for another image is used. This method consists of the following stages:

1. Read classified images.
2. For each class images do the flowing.
    - 2.1. Calculate the Energy for all images.
    - 2.2. Chose the image which has high energy as Range blocks.
    - 2.3. Remained images using to extract Domain blocks.
    - 2.4. UPIFSs constructed from one range and many Domains blocks.
3. Save all UPIFSs in database (one UPIFS for each class).
4. Read unclassified image.
5. For all UPIFSs in database do the flowing.
    - 5.1. Constructed a new image by using an unclassified image as the initial for UPIFS after one iteration.
    - 5.2. Calculate the PSNR between new image and unclassified image.
    - 5.3. Save all calculate the PSNR.
6. The unclassified image is assigned to the class has high PSNR.

V. EXPERIMENTAL RESULTS

Experimental results proof a high level of accuracy. Our proposed method is tested on 84 image for attaining high performance results. We have tested our technique on 60 different images out of which 84 are accurately enhanced giving an accuracy percentage as 71.667%. Table 1 summarizes the results of method.

TABLE 1

Results Accuracy

Proposed algorithm								
Class type	Class1	Class2	Class3	Class4	Class5	Class6	Total	Accuracy%
Class1	6	2	0	0	2	0	10	60
Class2	1	7	0	0	2	0	10	70
Class3	0	0	8	0	1	1	10	80
Class4	0	0	2	7	1	0	10	70
Class5	1	0	0	0	9	0	10	90
Class6	0	0	3	0	1	6	10	60
Total	8	9	13	7	16	7	60	71.667

VI. CONCLUSION

The results of the performed experiments shows that the proposed upgrade of the PIFS give high accuracy . Moreover we note that speed of the recognition process has also grown up. This speed improvement is caused by the fact that in the case of decoding PIFS for all image in class rather than decoding PIFS for each image in class which we are doing the search process are smaller than in the original case.

REFERENCES

1. E.W. Jacobs, R.D. Boss, and Y. Fisher. Fractal-based image compression ii. Technical Report 1362, Naval Ocean Systems Center, San Diego, CA, June 1990.
2. F. Albregtsen, "Statistical Texture Measures Computed from Gray Level Co-occurrence Matrices ", University of Oslo, 1995.
3. J.C. Russ, Fractal Surfaces, Plenum Press, New York, 1994.
4. M. Cheong et al., "An Approach to Texture-Based Image Recognition by Deconstructing Multispectral Co-occurrence Matrices using Tchebichef Orthogonal Polynomials ", IEEE, 2008.
5. T.R. Thomas, B.G. Rosen, N. Amini, Fractal characterisation of the anisotropy of rough surfaces, Wear 232 (1999) 41–50.
6. Y. Fisher Ed. , Fractal Image Compression. Theory and Applica-tion, Springer-Verlag, New York, 1995.
7. Yuan Y. Tang, Yu Tao," New method for feature extraction based on fractal behavior", IEEE, 2001.

