Journal for Geometry and Graphics
Volume 3 (1999), No. 1, 57–65

Fractal Image Coding by Multi-Scale Selection Based on Block Complexity

Aura Conci, Felipe R. Aquino

Computer Institute, Applied Computing and Automation, Federal Fluminense University
r. Passo da Patria 156, CEP 24210-240 Niteroi-RJ, Brazil
email: aconci@caa.uff.br

Abstract. Since the conception of automatic fractal image compression, the research on this topic has grown rapidly. This work is intended to provide a new vision on this automatic process by introducing the idea of multi-scale domain-pool classification based on the complexity of the image to be compressed. A preprocessing analysis of this image identifies the complexity of each image block computing its local fractal dimension. The performance of this proposition, evaluated by means of fidelity versus encoding time and amount of compression, is compared with two well-known image compression methods.

Key Words: image coding system, fractal compression, image compression, PIFS codes, fractal dimension

MSC 1994: 68U10

1. Introduction

With the development of multimedia systems, the growth of the image databases and the intense use of nets, the necessity of efficient storage of images has increased [10]. There are several forms to compress images and new researches have appeared in this direction. Fractal compression can be considered as one of these new forms [1]. A more detailed description of the encoding method, the notation used and a description of basic implementations can be found in [7]. Several researchers have taken up the challenge to improve the basic automated algorithm for compression [11]. Some of the main subjects addressed are compression ratio, time reduction and image fidelity [3]. We show in this work that the image fidelity can be improved considering the contractive factor and the local image complexity. Depending on the block complexity, four contractive factors have been used on the domain blocks in each image compression (improving the compression quality). Fractal dimensions (FD) have been proposed to characterize roughness in images. The FD could be used to obtain shape information and to distinguish between smooth (small FD) and sharp (large FD) regions [4]. In other words, preprocessing local FD of the image makes it possible to generate separate

ISSN 1433-8157/$ 2.50

sets of pools depending on the image complexity. To derive the contractive transformation of the Partitioned Iterated Function System (PIFS) codes just the relation between elements on pools of the same FD range are considered (improving compression time).

2. Development

As a model for the space of grayscale images, we choose a space E of functions $f : X \to G$, where the set X is taken as the set of spatial coordinates of the image while G represents the set of intensity values of the image. A metric d is used such that (E, d) is a complete metric space. The fractal coding of an image $f(N \times N)$, can be seen as the optimization problem: find a contractive mapping T on (E, d) whose fixed point $f = T(f)$ exists and is unique (by the contractive mapping fixed point theorem). Decoding consists of iterating the mapping T from any initial image until the iterates converge to an approximation of the original image. The encoding process consists of the construction of the operator T, which will be defined by matching the better couple of sets, formed from the original image. Let R_i and D_j be two subsets on f (the first called range is formed from partitioning $n \times n$ non-overlapping regions on X and the second called domain is formed also by subset of X but which may overlap). Let $t_i : D_j \to R_i$ be a *contraction* defined by matching the better $t_i(D_j)$ for each R_i, which means that t_i is chosen such that the distance $d(R_i, t_i(D_j))$ is as small as possible. The operator T is given by

$$T_f = \bigcup_{i=1}^{(N/n)^2} t_i(f)$$

considering the contractive factor s of the Collage theorem [1] and fixed point A (attractor). Then

$$d(f, A) \leq \frac{d(f, T_f)}{1 - s}.$$

How far from f will the attractor A of the PIFS be? By this theorem an upper bound of the distance between the original image f and its reconstruction A is obtained as a function of the contractive factor s of T. Several papers have popularized the scheme where a digital image is partitioned into square range blocks (say $n \times n$ pixels) and larger square domain blocks, frequently twice the size of the range blocks [6]. This scheme makes the compression process simple, but observe that the contractive factor of T is fixed to $s = \frac{1}{2}$. So by the Collage theorem

$$d(f, A) \leq 2d(f, T_f).$$

If the contraction of T is reduced to $s = 1/3$, $1/4$, $1/5$ or $1/6$, then by the theorem the upper bound to $d(f, A)$ will decrease to

$$d(f,A) \leq \frac{3}{2}d(f,T_f),\ d(f,A) \leq \frac{4}{3}d(f,T_f),\ d(f,A) \leq \frac{5}{4}d(f,T_f),\ \text{or}\ d(f,A) \leq \frac{6}{5}d(f,T_f),$$

respectively, and it is expected that the attractor A should look (each time more) quite like f, with these new set of contractions. This is the main idea of our work: the utilization of a variable contractive factor, based on image complexity values. Let D be a collection of subsets of D_j from which the better matching are chosen. D consisted of $m \times m$ pixels. The number of elements on D determine how much computation time the encoding takes for finding $t_i(D_j)$ for each R_i. For encoding, the mapping t_i must be specified and the better domain squares D_j must be chosen from a set of potential candidates. The choice of the domain pool as $3n \times 3n$,

$4n \times 4n$, $5n \times 5n$ or $6n \times 6n$ for each $n \times n$ range block, reduce the computation required for a brute force search [2] minimizing image quality reduction.

We use local FD (represented by DF) to subdivide the blocks into classes of complexity (see Figures 1 and 2). Some methods on DF estimation do not give satisfactory results in all range of DF for images (from 2 to 3). For fast DF evaluation a simple and efficient algorithm has been used [4]. We use four levels of complexity, based on the DF of each image part:

$$2 \leq DF < 2.25, \quad 2.25 \leq DF < 2.5, \quad 2.5 \leq DF < 2.75, \quad \text{and } 2.75 \leq DF \leq 3.$$

For the mapping t_i to be specified, domain squares D_j must be chosen from a set of potential candidates, *i.e.* a set with the same DF of the range. All images have been treated as ideal fractal models. In the course of this analysis, the corresponding blocks present semi-fractal behavior. However, we are not addressing to the problem of finding the real fractal dimension of the image blocks, it is only a way for representing its local complexity [9].

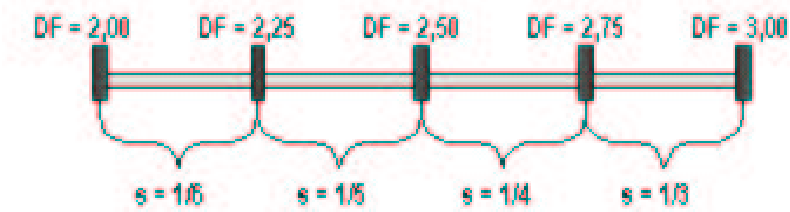

Figure 1: Relating the domain pool and DF with the contractive factor.

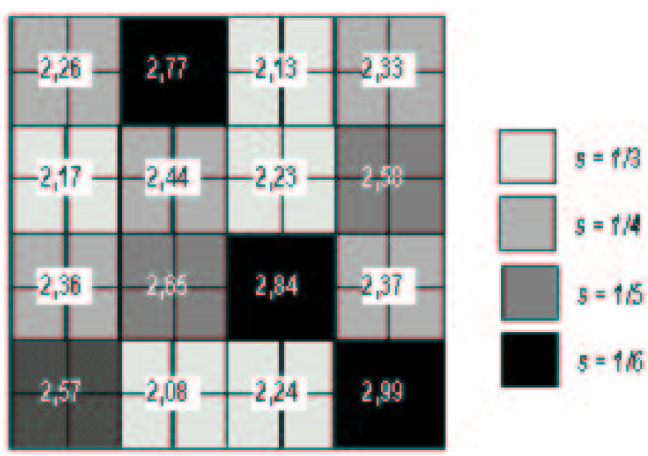

Figure 2: Example of fractal dimension blocks division.

3. Performance comparison

The performance of this proposition, evaluated by means of fidelity versus encoding time, is compared with the brute force algorithm [2], [8] and the adaptive quadtree method [5]. On Fisher's program [5] the defaults flags have been used: 7 bits for the offset factor and 5 bits for the scaling factor, and 4×4 range blocks (no quadtree scheme). The efficiency of this image compression approach is illustrated here by considering both image fidelity and time reduction. Fig. 3 compares the image quality on reconstruction of six images: *Lena, LAX,* *Cameraman, Columbia, Goldhill* and *Couple* using the proposed approach. Similar results for the other two methods can be seen on Figures 4 and 5. All images have 128×128 pixels and 256 gray level. For quality verification these figures show also the result of subtraction operation between the original image f and its reconstruction A. Since the performance of this and other systems has to be compared, Tables 1 – 3 present parameters of the encoding process: time spent on encoding, the root mean square error e_{rms}, the signal to noise ratio SNR, the peak signal to noise ratio $PSNR$, and the compression ratio on bits per pixel of the image. These are defined as:

$$
\begin{aligned}
e(x,y) &:= f(x,y) - A(x,y) \\
e^2_{rms} &:= \textstyle\sum_{x=i}^{N} \sum_{y=i}^{N} e(x,y)^2 \\
SNR_{rms} &:= \textstyle\sum_{x=i}^{N} \sum_{y=i}^{N} A(x,y)^2 \Big/ e^2_{rms} \\
PSNR &:= 20 \log_{10} \frac{2^p - 1}{e_{rms}}
\end{aligned}
$$

where p is the number of bits per pixel of the image.

Image name	time (s)	e_{rms}	SNR_{rms}	$PSNR(dB)$
Lena	47.3460	9.7622	10.4823	8.3398
LAX	62.5650	17.7365	4.9023	23.1535
Cameraman	56.9930	14.3160	8.0053	25.0144
Columbia	50.7170	15.6185	5.6044	24.2580
Goldhill	47.7850	8.7668	10.6057	29.2740
Couple	57.2040	13.5676	8.4255	25.4808
Average	52.9969	13.1421	7.7337	25.9413

Table 1: Performance of the proposed method (local fractal dimension with blocks of size 4×4 and $s = \frac{1}{3}, \frac{1}{4}, \frac{1}{5}$ and $\frac{1}{6}$) on the set of test images. Each image is represented by a file with 25600 *bits*. The compression ration for all images is 1.5625 *bpp* (bits per pixel).

Image name	time (s)	e_{rms}	SNR_{rms}	$PSNR(dB)$
Lena	469.4040	7.61672	13.3478	30.4954
LAX	469.4110	17.4734	4.9517	23.2832
Cameraman	469.3550	14.0104	8.1885	25.2018
Columbia	469.3850	16.3936	5.3475	23.8373
Goldhill	469.3750	6.7977	113.4355	31.4836
Couple	469.3130	13.6817	8.3402	25.4080
Average	469.3098	12.1756	8.69527	26.7874

Table 2: Performance of BARNSLEY's program [2] for the same images. Each image is represented by a file with 57376 *bits*. The compression ratio for all images is 3.5020 *bpp*.

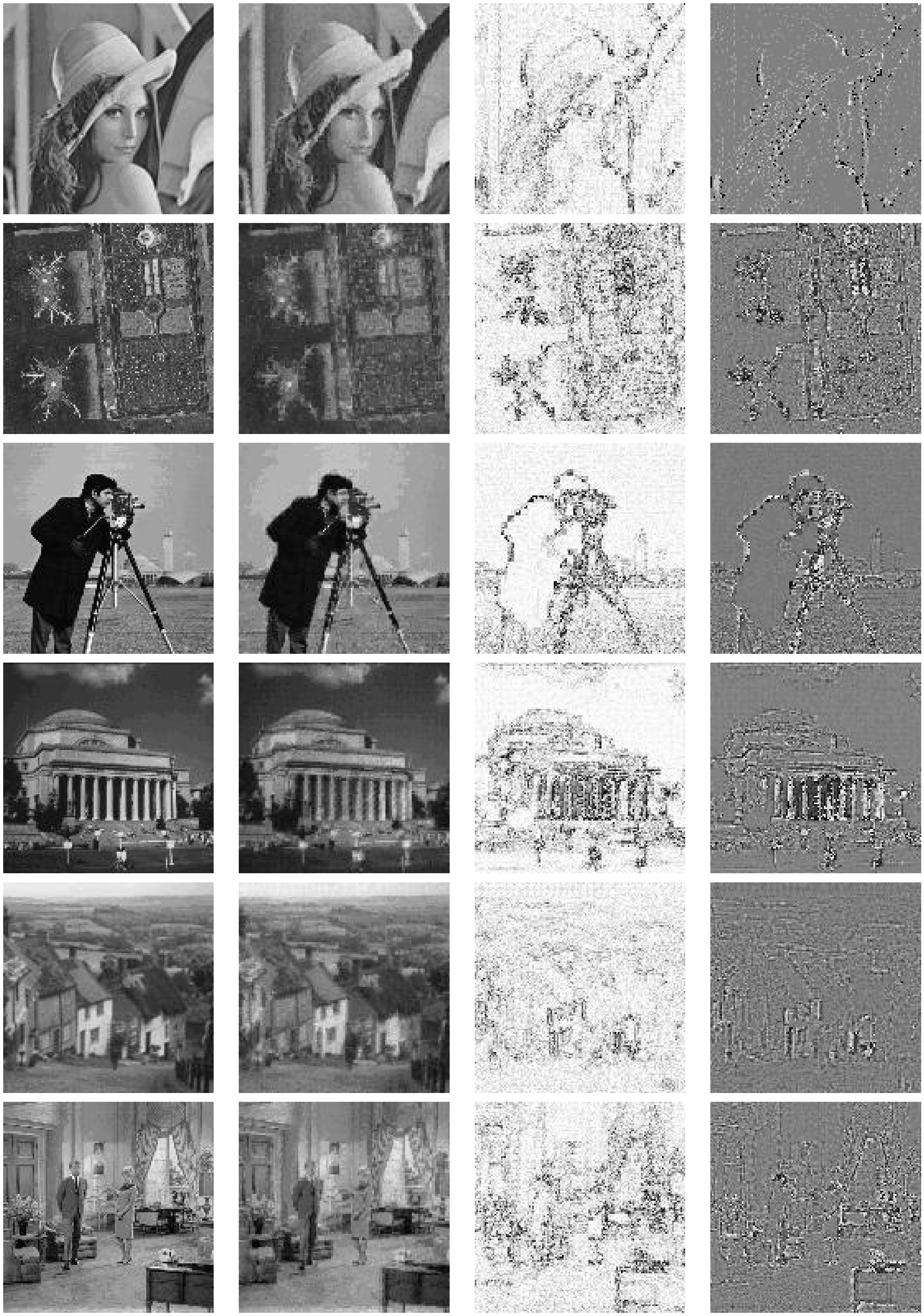

Figure 3: Six test images used in the experiments (first column); reconstructed images using local fractal dimension with range blocks of size 4×4 pixels, $s = \frac{1}{3}$, $\frac{1}{4}$, $\frac{1}{5}$ and $\frac{1}{6}$ (second column); difference image for quality verification: $5.\|e(x,y)\|$ (third column); and $\|5.\frac{e(x,y)}{2} + 127\|$ (fourth column).

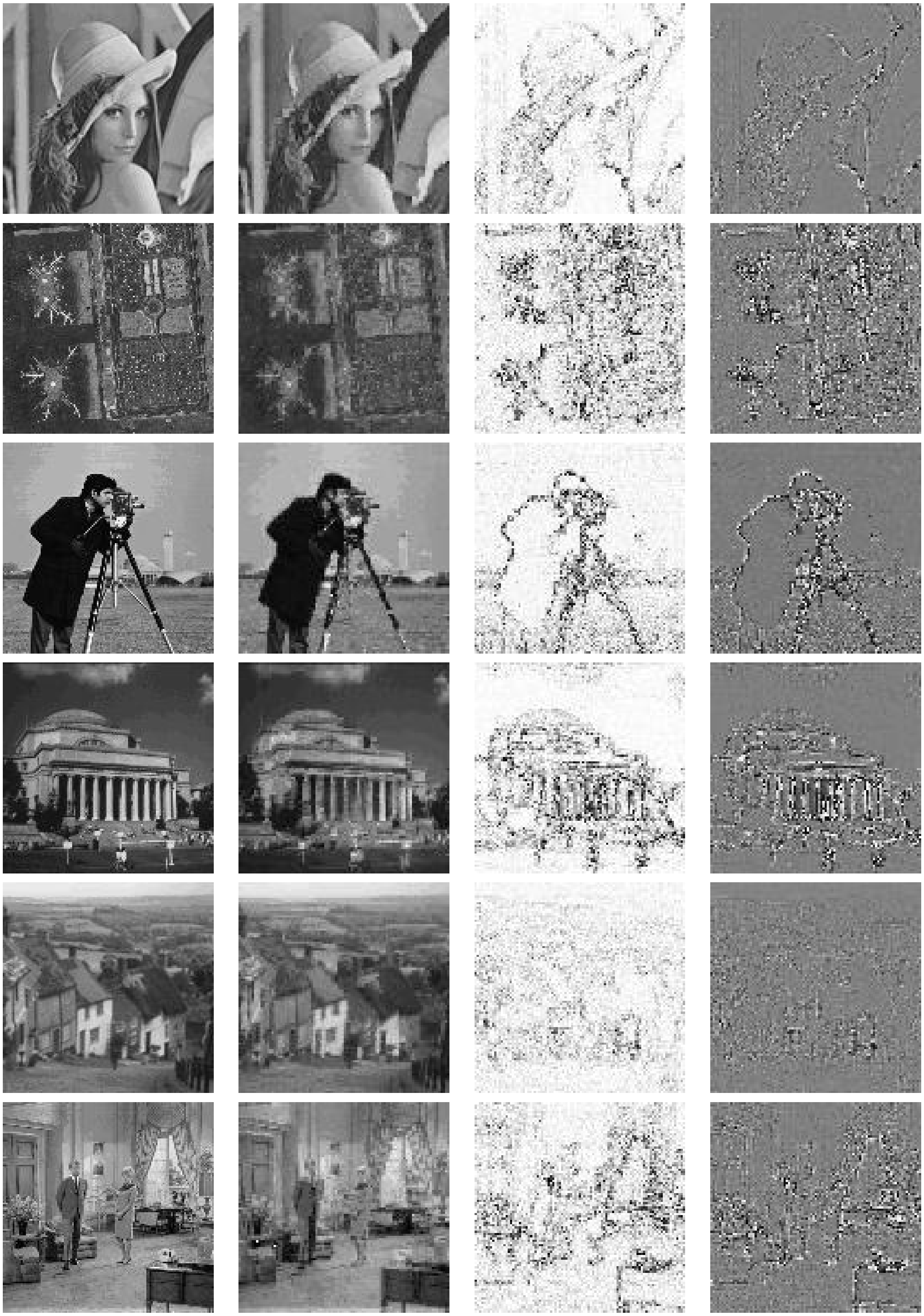

Figure 4: Six test images used in the experiments (first column); reconstructed images using Barnsley's program (second column); difference image for quality verification: $5.\|e(x,y)\|$ (third column); and $\|5.\frac{e(x,y)}{2}+127\|$ (fourth column).

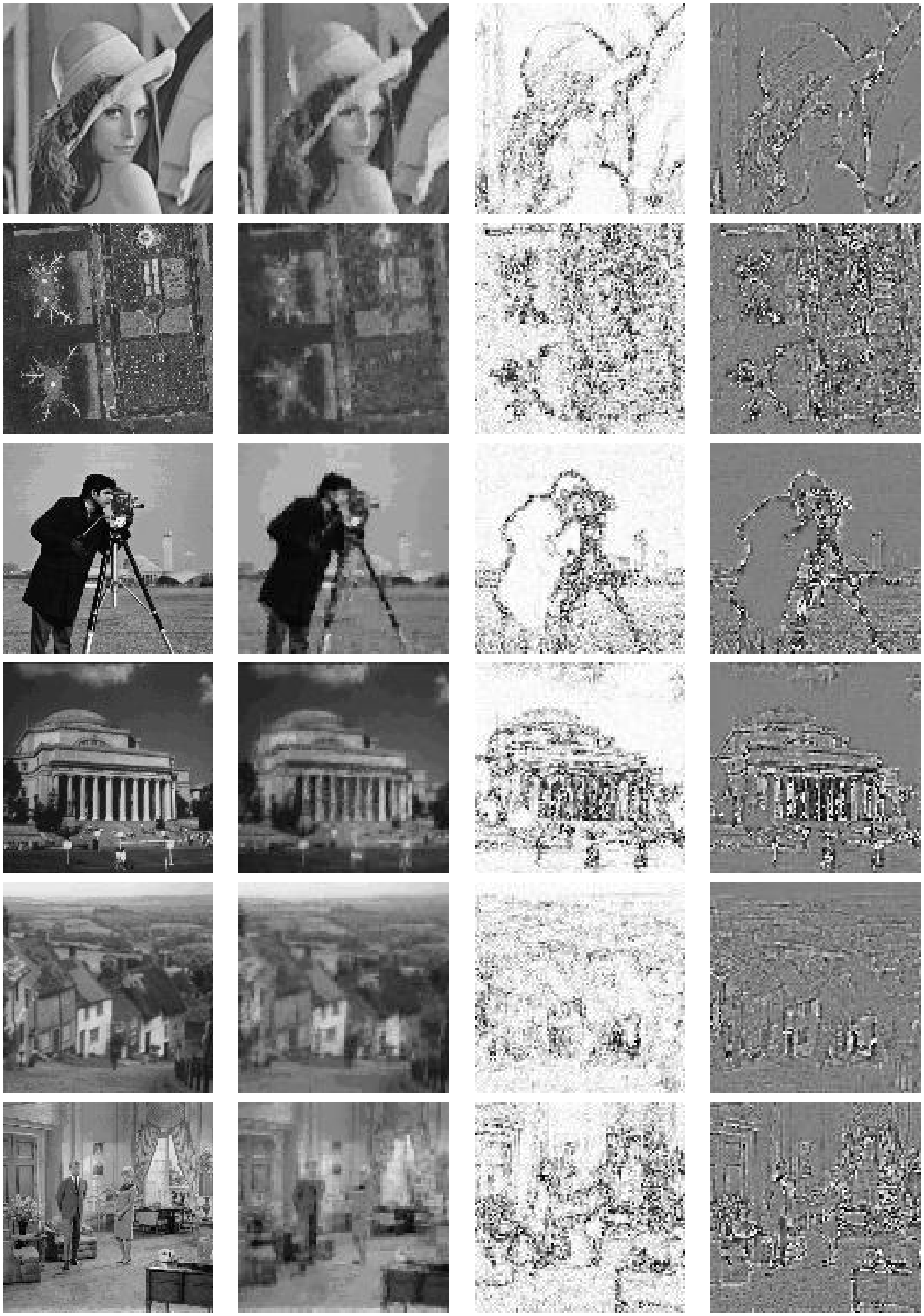

Figure 5: Six test images used in the experiments (first column); reconstructed images using FISHER's program (second column); difference image for quality verification: $5.\|e(x,y)\|$ (third column); and $\|5.\frac{e(x,y)}{2} + 127\|$ (fourth column).

Image name	time (s)	e_{rms}	SNR_{rms}	$PSNR(dB)$	Compression
Lena	0.2730	10.6345	9.7362	27.5964	1.4282
LAX	0.1920	20.8534	4.1191	21.7473	1.4380
Cameraman	0.2120	18.0882	6.6247	22.9829	1.4058
Columbia	0.2610	17.9260	4.9174	23.0611	1.4180
Goldhill	0.2620	9.8373	9.4294	28.2733	1.4360
Couple	0.2480	17.3362	6.6913	23.3517	1.4375
Average	0.2510	15.6659	6.6457	24.4710	1.4268

Table 3: Performance of Fisher's program [5] for the same images.

4. Conclusions

In order to compare the relative merits of each work it is necessary to be able to decide what means to say that one approach is better than the other. It is clear that the best results would have the minimal e_{rms} and that from two given encodes the one with larger *SNR* or *PSNR* looks better. Consequently, the images on Fig. 4, encoded by Barnsley's (http://links.uwaterloo.ca:/pub/Fractals) program [2] might be lightly better than that on the second column of Fig. 3, and the proposed approach produces better results than Fisher's program [5] (Fig. 5, with parameters mentioned in Section 3). However Fisher's program (http://inls.ucsd.edu/y/Frac) presents minimal encoding time and maximal compression ratio.

Acknowledgment

This work has been supported in part by the project FINEP-RECOPE SAGE #0626/96. The authors acknowledge CNPq (*Ref.* 302649/87-5), FAPERJ and CAPES for their support.

References

[1] M.F. Barnsley: *Fractals Everywhere*. 2nd ed., Academic Press, New York 1993.

[2] M.F. Barnsley, L. Hurd: *Fractal Image Compression*. AK Peters, Wellesley 1993.

[3] Chwen-Jye Sze, Hong-Yuan Mark Liao, Kuo-Chin Fan, Ming-Yang Chern, Chen-Kou Tsaoy: *Fractal image coding system based on an adaptive side-coupling quadtree structures*. Image and Vision Computing **14**, 401–415 (1996).

[4] A. Conci, C.F.J. Campos: *An Efficient Box-Counting Fractal Dimension Approach for Experimental Image Variation Characterization*. Proceedings of IWSIP'96 — 3rd International Workshop in Signal and Image Processing, Elsevier Science, Manchester, UK, 665–668 (1996).

[5] Y. Fisher: *Fractal Compression: Theory and Application to Digital Images*. Springer Verlag, New York 1994.

[6] J.C. Hart: *Fractal Image Compression and Recurrent Iterated Function Systems*. IEEE Computer Graphics and Applications **16**, no. 4, 25–40 (1996).

[7] A. Jacquin: *Image Coding Based on a Fractal Theory of Iterated Contractive Image Transformations*. IEEE Transactions on Image Processing **1**, 18–30 (1992).

[8] J. Kominek: *Advances in Fractal Compression for Multimedia Applications*. [ftp://links.uwaterloo.ca:/pub/Fractals/Papers/Waterloo/kominek95c.ps.gz] University of Waterloo, Internal report CS95-28 (1995).

[9] S. Morgan, A. Bouridane: *Application of Shape Recognition to Fractal Based Image Compression*. Proceedings of IWSIP'96 — 3rd International Workshop in Signal and Image Processing, Elsevier Science, Manchester, UK, 23–26 (1996).

[10] M. Nappi, G. Polese, G. Tortora: *FIRST: Fractal Indexing and Retrieval SysTem for Image Databases*. Image and Vision Computing **16**, 1019–1031 (1998).

[11] D. Saupe, R. Hamzaoui: *A Review of Fractal Image Compression Literature*. Computer Graphics **28**, 268–275 (1994).

Received August 14, 1998