Chinese Journal of Electronics
Vol.13, No.3, July 2004

Region-Based Fractal Image Coding with Freely-Shaped Partition*

SUN Yunda, ZHAO Yao and YUAN Baozong

(*Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China*)

Abstract — In Fractal image coding (FIC), a partitioning of the original image into ranges and domains is required, which greatly affects the coding performance. Usually, the more adaptive to the image content the partition is, the higher performance it can achieve. Nowadays, some alleged Region-based fractal coders (RBFC) using split-and-merge strategy can achieve better adaptivity and performance compared with traditional rectangular block partitions. However, the regions are still with linear contour. In this paper, we present a Freely-shaped Region-based fractal coder (FS-RBFC) using a two-step partitioning, i.e. coarse partitioning based on fractal dimension and fine partitioning based on region growth, which brings freely-shaped regions. Our highly image-adaptive scheme can achieve better rate-distortion curve than conventional scheme, even more visually pleasing results at the same performance.

Key words — Fractal image coding, Region-based, Freely-shaped partition.

I. Introduction

In FIC an image is modeled as the unique fixed point of a contractive operator on the space of images. This type of image representation was first introduced by Barnsley and Sloan[1]. Afterwards, Jacquin[2] devised the first practical fractal coder with block-based transformations. Fractal coding has since attracted many interests because it has opened up a refreshing new view to image compression, which leads to visually pleasing results at high compression ratios, and provides resolution independent image description.

As the original image is partitioned into so-called ranges, each range is coded by a reference to some other part of the image and by some transformation parameters. These parameters describe how the referenced image part has to be adjusted with respect to contrast and brightness in order to give a good approximation to the range to be encoded. When fixed length codes are used for these parameters, the size of compressed data is basically proportional to number of ranges in partition. Thus the principal and widely studied problem in fractal coding is how to get an efficient segmentation, with a small number of ranges most similar to other image parts under certain transformations.

So far many fractal coders have been devised, among which Fisher's conventional quadtree[3] is the most well-known and successful approach. As the details are distinguished with smooth parts to some extent using variable range-block size, it can adapt to the original image and improve the decoded image quality. To further exploit the image-adaptivity of fractal coders, some schemes with Horizontal-vertical (HV) partition[4,5], triangular partition[6−9], polygonal partition[10,11], irregular partition[12−17] are proposed successively, taking on higher image-adaptivity and performance than before. However, their contours are still linear, including those with irregular partition namely region-based fractal coders.

In this paper we investigate highly image-adaptive freely-shaped partitions in order to improve the rate-distortion performance of fractal coding. In our proposed region-based fractal coder, the partitions are not derived from the split-and-merge approach as usual, but from a two-step partitioning, coarse partitioning based on fractal dimension and fine partitioning based on region growth. Compared with the quadtree scheme, a higher rate is required for encoding region contour. However, we will show that this expense is greatly reduced due to an elaborate growing process. Better rate-distortion performance, a gain of 1.0–1.5dB for compression ratio larger than 10:1, can be seen in the reported result.

II. Basics of Fractal Image Coding

The fundamental concept of fractal coding is based on the theory of contractive transformation. Let (X, d) denote a complete metric space of digital images, where d is a given distortion measure. An original image X_{org} is one element of the space. The inverse problem of contractive transformation theory (i.e. the fractal coding procedure of X_{org}) is to construct a transformation $f : X \to X$, which satisfies the following conditions:

(a) For any $x, y \in X$, there exists a real number $0 \leq s < 1$

$$d(f(x), f(y)) \leq s \cdot d(x, y) \tag{1}$$

(b) $$f(X_{org}) = X_{org} \tag{2}$$

From condition (a), we know that f is a contractive transformation. The fixed-point theorem ensures that f has a unique fixed point and the fixed point can be found by iteration of f. Condition (b) tells us X_{org} is the fixed point of f. So X_{org} can be reconstructed by applying f on any initial image X_0

*Manuscript Received Oct. 2002; Accepted Oct. 2003. This work is jointly supported by Huo Yingdong Young Teachers Foundation and the National Natural Science Foundation of China (No.69802001 and No.60172062).

iteratively. That is:

$$X_{org} = \lim_{n\to\infty} f^n(X_0) \tag{3}$$

In practical use, it is difficult to construct such f directly and exactly. It is usually approximated by the union of a series of contractive affine transformations:

$$f(X_{org}) = X_{org} = \bigcup_{0\le i<N} R_i \approx \bigcup_{0\le i<N} f_i(D_i) = f'(X_{org}) \tag{4}$$

where non-overlapping R_i named ranges construct a partition of the original image, while so-called domains D_i selected from a large domain pool may overlap each other. $f'(X_{org})$ is so-called collage image. Generally f_i consists of four parameters. They are the scale factor α_i, the brightness offset o_i, the symmetry operation sym_i and the position of domain. The encoding of a range R_i consists of finding a "best pair" (D_i, f_i) such that the range R_i is most similar to the transformed domain $f_i(D_i)$ and the distortion is minimized. That is:

$$(D_i, f_i) = \arg\inf_{D_i, f_i} \|R_i - sym_i[\alpha_i(S\circ D_i) + o_i\cdot I]\| \tag{5}$$

where S denotes the operation of contractive average in the $2k\times 2k$ domain D_i into the same image size as $k\times k$ range R_i, I is the flat block with intensity 1 at every pixel. As is implied in Eq.(4):

$$X'_{org} \approx X_{org} \tag{6}$$

where X'_{org} is the fixed point of f' (i.e. the decoded image). Fortunately, the collage theorem states that:

$$d(X_{org}, f'(X_{org})) \le \varepsilon \Rightarrow d(X_{org}, X'_{org}) \le \varepsilon/(1-s) \tag{7}$$

So we can assure the fidelity of decoded image via minimizing the difference between the original image and its collage. Based on the analysis above, how to partition original image efficiently is of great importance. So far many image-adaptive partitioning methods have been proposed to facilitate finding "best range-domain pairs". In the next section, we will give a simple survey of the principal partitioning methods for fractal compression.

III. Previous Work

For FIC, the uniform partition is the most basic one [see Fig.1(a)], by which we denote a partition consisting of square atomic blocks of size $k\times k$ pixels. After the block size specified, the uniform partition is completely image-independent, so it is rarely used for fractal image coding. In the following we will focus on image-adaptive partitioning schemes.

1. Regularly-shaped category

This category includes those fractal coders with regularly-shaped ranges, that is, the rough shape of ranges can be known beforehand, e.g., squares, rectangles, or triangles, always convex-shaped.

In the classical quadtree-based fractal coders proposed in Refs.[3, 18, 19], original image is partitioned in a top-down fashion. One starts with selecting an initial level in the tree, corresponding to maximum range size, and then ranges are split into their four quadrants when the splitting criterion on collage error or intensity variance is satisfied. The alternative bottom-up fashion[20] begins with a uniform partition using minimum range size, and then proceeds to merge those neighboring blocks for which a more efficient representation is provided by the resulting larger block one level up the quadtree, see Fig.1(b). Fractal coding with Horizontal-vertical (HV) partitions has been presented in Ref.[4, 5], see Fig.1(c). If no domain match with acceptable collage error is found for a given rectangular range, the block is split into two rectangles along the most significant horizontal or vertical edge. A number of different triangular partitions [see Fig.1(d)] have been investigated. A Delaunay triangulation is constructed on an initial set of "seed points" and is adapted to the image by adding extra seed points in ranges of high variance[6,7]. Two substitutes are three-side split[8] and one-side split[9].

A step further in adaptivity is polygonal partition [see Fig.1(e)] that is applied for fractal coding by Reusens[10]. It is similar to HV partition, but it also includes 45° and 135° cutting directions. Davoine[11] also gives an alternative by merging Delaunay triangulation.

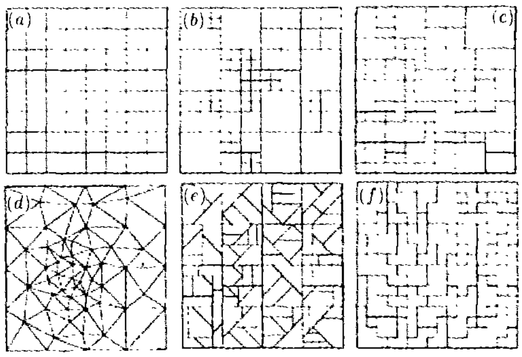


Fig. 1 (a) Uniform partition; (b) Quadtree partition; (c) H-V partition; (d) Triangular partition; (e) Polygonal partition; (f) Irregular partition

2. Irregularly-shaped category

This category consists of so-called region-based fractal coders. For high image-adaptivity and better rate-distortion performance, all published RBFC take the split-and-merge approach, which brings on lots of unpredictable concave-shaped regions, see Fig.1(f).

In The region-based fractal coder using heuristic search of Thomas and Deravi[12], one starts by choosing one of the ranges in uniform partition as a seed, and attempts to search for suitable domain-to-range block transformation. In a recursive manner, the algorithm extends the region (from the seed) in all four principal directions of the newly extended range using the same coefficients as the original seed range transformation. The extension of the domain region is simply under the same direction corresponding to that of range extension to reduce the computational complexity. The region-based fractal image coding with quadtree segmentation is discussed in Ref.[14] by Chang, Shyu and Wang. After the quadtree segmentation, the region-pair with the lowest collage error will be merged in each step. The merging process continues until the error is greater than a selected threshold.

Some other RBFCs are introduced in Refs.[13, 15–17]. Refer to them for the details.

As the results of previous coders have indicated, fractal coding can profit from highly flexible and image-adaptive partitioning schemes. Generally, image-adaptive coders excel image-nonadaptive ones, and more image-adaptive coders achieve better coding performance. However, all published RBFCs have only linear region contours, which greatly limit their adaptivity. The reason lies in two aspects. Firstly, it is impossible to bring truly freely-shaped regions with previous split, merged or combined partitioning schemes. Secondly, the big problem for lack of efficient representation remains unsolved since existing contour coding techniques take on rather low efficiency in the case of freely-shaped regions. In the next section, we'll introduce a novel scheme namely FS-RBFC to overcome such shortcoming. While formed in region growth, freely-shaped ranges can keep more details precisely, leading to better visual effect.

IV. Freely-Shaped Region-Based Fractal Image Coding

In our FS-RBFC the highly image-adaptive partition is derived from two steps, coarse partitioning based on fractal dimension and fine partitioning based on region growth. Fractal dimension is first used to measure image content and produce approximate locations in coarse partitioning; then fine partitioning refines them with growing seeds, leading to final freely-shaped regions. Contour coding is realized by storing only areas of regions, which is the main contribution of our scheme. Meanwhile, region content (transformation parameters) is quantized and stored into the compressed image.

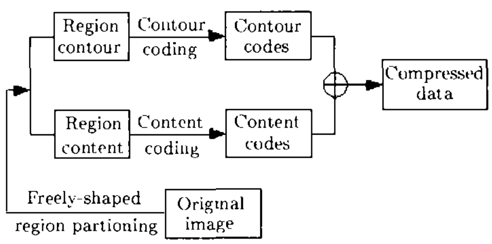


Fig. 2. The framework model of FS-RBFC

In the decoder, freely-shaped region contours are precisely recovered up to known areas, on the basis of which arbitrary image is iterated using transformation parameters to restore region content. Besides, a linear predictor is often applied to a small quantity of remaining pixels, which haven't been grown in original image.

Fig.2 shows a framework model of our coder, in which each part will be explained specifically as follows.

1. Freely-shaped region partitioning

(1) Coarse partitioning based on fractal dimension

As quantitative description of image complexity to some extent, fractal dimension around each seed is computed first in the original image or its contractive image. The coarse partitioning is constructed by seed-pairs with similar dimensions between these two images. That is:

① Distribute range seeds at intervals of RSTEP in original image X_{org} and domain seeds at intervals of DSTEP in contractive image X_{con}, which is acquired by down-2 sampling or 4-pixels averaging. All seeds are simplified into one pixel size, see Fig.3.

② Analyze the growth environment for each seed. That is, to calculate fractal dimension $Dim_R(i)$ for each range seed R_i or calculate $Dim_D(j)$ for each domain seed D_j, within $k \times k$ square around it.

③ Select a certain number (DOM_NUM) of domain seeds D_j for each range seed R_i as its similar region $Rgn(i)$. These seeds should have the most similar growth environment (fractal dimension) to satisfy

$$Rgn(i) = \left\{ \bigcup_j D_j : \min \sum_{j=1}^{\text{DOM\_NUM}} |Dim_R(i) - Dim_D(j)| \right\} \quad (8)$$

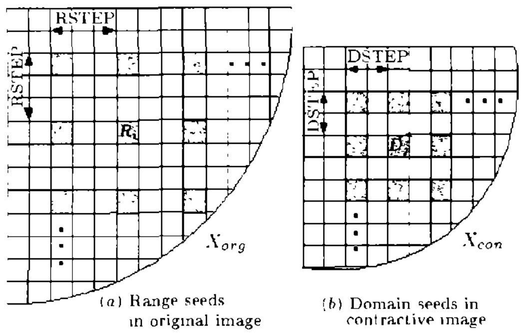


Fig. 3. Sketch map of range seeds and domain seeds

As the distance between range seeds, RSTEP is chosen according to the requirement for bitrate and SNR. While RSTEP increases, SNR as well as bitrate decreases greatly, which deteriorates the quality of decoded image. Therefore it ought to be set between 1 and 10.

As the distance between domain seeds, an increase in DSTEP doesn't affect bitrate and encoding speed obviously, while SNR decreases slightly, so our proposed value is 1.

As the number of domain seeds in every similar region, DOM_NUM is chosen according to the requirement for encoding speed. A smaller DOM_NUM leads to faster coder but lower SNR. Therefore it should be larger than 10.

For our convenience of calculating fractal dimension, the size of square around a seed is integral power of 2. Generally $k = 4$ or 8. We compute box dimension in this scheme, which is widely used in many fields for its simplicity, and defined as:

$$Dim = \lim_{r \to 0} \frac{\log N(r)}{-\log r} \quad (9)$$

Concretely, for a $k \times k$ image block B, b is one of its $r \times r$ sub-blocks with variable size r. We have

$$n(r) = floor\left(\frac{\max P_i - \min P_i}{r}\right) + 1 \quad (10)$$

where P_i denotes gray level of every pixel in b, $n(r)$ is the number of $r \times r \times r$ cubes to contain fluctuant part of b, if gray image is conceived as fractal surface in 3-dimensional space. After all sub-blocks in B are discussed

$$N(r) = \sum n(r) \quad (11)$$

Because box dimension Dim of B satisfies

$$N(r) \cdot r^{Dim} = C, \quad \text{where} \quad C \text{ is a constant} \quad (12)$$

$$\log N(r) = -Dim \cdot \log r + \log C \quad (13)$$

When r varies, $(\log N(r), -\log r)$ pairs are formed successively to approximate a line using linear regression method; thus the slope is box dimension of B.

(2) Fine partitioning based on region growth

The result of coarse partitioning is inaccurate and unsuitable for image compression, so further segmentation is needed. Given a range seed R_i and a domain seed D_j in its similar region, a seed-pair is established for region growth. 8-neighborhood pixels of current contour are added continuously into growing region. Assuming that R is growing range seed in X_{org} and D is growing domain seed in X_{con}. The fine partitioning isn't complete until the collage error rms

$$rms = \|R - (\bar{s} \cdot D + \bar{o} \cdot I)\|^2 \tag{14}$$

is too large or no 8-neighborhood pixels are available, where $\bar{s}, \bar{o}$ are the quantized values of luminance parameters s, o, I is a uniform image block with each pixel having unit intensity. That is:

④ Take out a range seed R_i in original image X_{org} and validate it. If R_i is valid (i.e. it has not been grown during the growth of previous range seeds), go to step ⑤, or discard it and process the next range seed.

⑤ Take out a domain seed D_j from the similar region $Rgn(i)$ of current range seed R_i, and then a seed-pair $R_i \leftrightarrow D_j$ is established to grow jointly. Use a candidate list to store all pixels waiting to be grown, and initialize it as valid 8-neighborhood pixels of R_i.

According to the output of pseudo-random number generator, next pixel to be grown is picked continuously from the candidate list. Add this pixel into R and its valid 8-neighborhood pixels into the candidate list, if only they have not been grown by current range seed. Grow D in contractive image X_{con} at the same time.

In order to assure the cover rate (total area of all regions after encoding vs. the area of original image) and avoid too many remaining pixels, which degrade the decoded image, a minimum cover area MINA is prescribed and reached by R and D. After that the seed-pair won't quit growing until the collage error rms (Eq.(14)) is above pre-selected threshold RMS_TOL or the candidate list is empty. Record rms and all pixels in R.

⑥ If there still exist domain seeds in the similar region of current range seed, repeat step ⑤; otherwise go to step ⑦.

⑦ Select the seed-pair with the largest cover area after region growth, or that with the smallest collage error rms at the same cover area. The corresponding R is a part of final freely-shaped partition.

⑧ If there still exist range seeds in original image, go to step ④. Otherwise, go to contour coding and content coding. when all transformation parameters are quantized and stored, the encoding procedure is finished.

The complete flowchart of our FS-RBFC scheme is given in Fig.4.

The choice of minimum region area MINA has a direct influence on performance and speed. As MINA increases, bitrate decreases out of a quicker encoder, while the quality of decoded image lowers obviously. On the other hand, too small MINA leads to many remaining pixels, which means poor SNR. Consequently, an appropriate MINA between 20 and 50 is proposed.

Once other parameters are fixed, the rate-distortion curve can be drawn with variable threshold RMS_TOL. Larger collage error corresponds to lower bitrate, lower SNR and quicker encoder, which should be also carefully selected.

2. Contour coding

We now further discuss efficient encoding of region contour generated by the freely-shaped region-based fractal coder, an extremely important part in our scheme. So far there exists some mature techniques in published RBFCs, such as:

(1) Track region contour with segmented chain code proposed by Kaneko and Okudaris[21]; (2) Track region contour with derivative chain code given by Dietmar Saupe[15]; (3) Depict region contour with region edge maps via context modeling devised by Tate[22].

Adaptive arithmetic coding or LZW coding is often applied on result symbol string to further decrease the bit rate.

However, in the case of freely-shaped region contour rather than linear one, experiment shows that the aforementioned methods lead to high redundancy and very low compression ratio ($CR < 6$). So an elaborate growing process is designed to make a tradeoff between image-adaptivity and efficiency in our scheme. As implied in previous section, on the one hand, it expands the region to each direction quasi-freely, and only area of grown region (number of pixels) is in need of storage, increasing the compression efficiency greatly at few expense of adaptivity. On the other hand, it is also directive, as more pixels in some direction are grown, more pixels of that direction are in the candidate list, meaning a higher possibility to be picked. Furthermore, the growing process can recur in the decoder until a given region area is reached, restoring the region contour precisely.

To further compact region contour information, the areas of grown regions can be pruned as an arithmetical sequence and compressed with any entropy coder like Huffman-Shannon-Fino's. For a grown region with n pixels, the contour codes are

$$C_1 = floor\left[\frac{n}{MINA}\right], \quad HSF[C_1]bits \tag{15}$$

where $floor[x]$ is the closet integer smaller than x, $HSF[x]$ is the HSF code length of x. Then valid area is $MINA * C_1$ pixels.

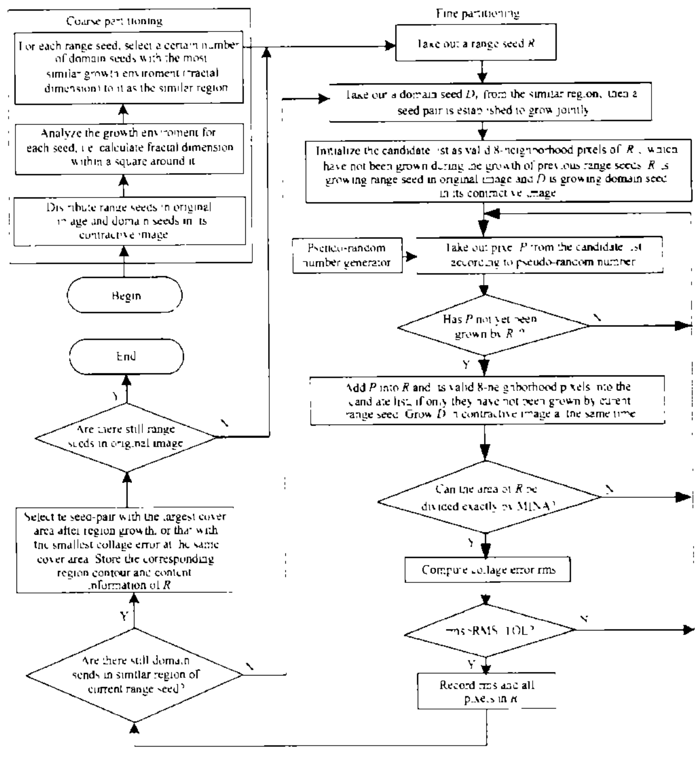


Fig. 4. Flowchart of our FS-RBFC scheme

3. Content coding

Content coding can be deemed as the construction of shape-adaptive fractal transformations based on region partitioning. Similar to most RBFCs, several important parameters including scale factor s, brightness offset o and the position of domain seed in ordinal number seq, computed during the partitioning process, are in need of quantization and storage, except that no symmetry operation is concerned. For an $M \times N$ original image, the content codes of a grown region are

$$C_2 = \bar{s} + \bar{o} + seq,$$
$$S\_BITS + O\_BITS + ceiling\left[\log_2 \frac{M * N}{DSTEP^2}\right] \text{ bits} \tag{16}$$

where S_BITS and O_BITS are the number of bits used to store the quantized values $\bar{s}$, $\bar{o}$ of luminance parameters s, o. $ceiling[x]$ is the closet integer bigger than x.

Except all grown regions, there still exists a small amount of remaining pixels $\tilde{R}$ accounting for less than 5% of the whole image, which need careful attention. Experiments show that they can be omitted at the case of small RSTEP (compact seeds), but filled with linear predictor of non-remaining 4-neighborhood pixels after each iteration in the decoder. Once $RSTEP > 5$, large blocks of remaining pixels often invalidate the predictor, so the mean gray level $MEAN_i$ of remaining pixels in each non-overlapping 8×8 block B_i is stored for better result. That is

$$P_{m,n} = \begin{cases} E[Q_{m,n}], \text{ where } Q_{m,n} \in \{P_{m-1,n}, P_{m,n+1}, P_{m+1,n}, \\ \qquad P_{m,n-1}\} \text{ and } Q_{m,n} \notin \tilde{R}, \quad RSTEP \leq 5 \\ MEAN_i, \text{ where } P_{m,n} \in B_i, \qquad RSTEP > 5 \end{cases} \tag{17}$$

where $E[x]$ is the expectation operator, each remaining pixel $P_{m,n}$ is processed in the raster scanning order.

V. Results

In this section, we test the compression ratios and the fidelity of compressed images using our FS-RBFC scheme and conventional quadtree scheme.

We choose the following parameter setting for our experiments. The interval of range seeds RSTEP and domain seeds DSTEP are 2 and 1; the size of square to compute fractal dimension k is 8; the number of domain seeds in every similar region DOM_NUM is 10; the minimum area after region growth are confined to $MINA = 20$; the default threshold of collage error RMS_TOL is 8.0; the quantization for the luminance parameters is preformed as it is done in conventional quadtree scheme with $S\_BITS = 5$ for the scale factor and $O\_BITS = 7$ for the brightness offset.

Fig.5(a) and 5(b) shows the original Lenna 256×256 image encoded with our FS-RBFC scheme and the corresponding partition. The freely-shaped regions, which are completely different from the ones with linear region contours in published RBFCs, result in less visible blocking artifacts in the decoded image compared to quadtree-based fractal coder. Fig.5(c) and 5(d) show the decoded Lenna using these two methods. At similar reconstruction quality, it can be seen that our FS-RBFC scheme achieves more visually pleasing image than quadtree scheme, say nothing of the higher compression ratios. From Fig.5(b), some characteristics need further attention. In the resulting partition, there are usually larger regions in smooth areas than those in the details, and small regions occur frequently around the contour, because it is difficult to find another region to approximate them where gray values change acutely. Generally speaking, a region is unlikely to traverse several semantic objects, but more than one region are needed to encode an object due to the LIFS theory of FIC. So our freely-shaped RBFC is not designed as a complete content-based coder.

Fig. 5. (a) The original Lenna 256×256 image; (b) Partition corresponding to FS-RBFC (1110 regions); (c) Decoded Lenna using FS-RBFC scheme at $CR = 15.14$ and $PSNR = 29.88$dB; (d) Decoded Lenna using quadtree scheme at $CR = 12.30$ and $PSNR = 29.91$dB

As the threshold of collage error RMS_TOL varies, CR vs. PSNR curves for the Lenna 256×256 image in Fig.6 are obtained with our FS-RBFC scheme and conventional quadtree scheme. The FS-RBFC yields a better rate-distortion performance compared to the quadtree coder, which can achieve coding gains of 1.0–1.5 dB for compression ratios larger than 10:1. It demonstrates that our highly image-adaptive partitioning method leads to significant gains over rigid quadtree-based approach.

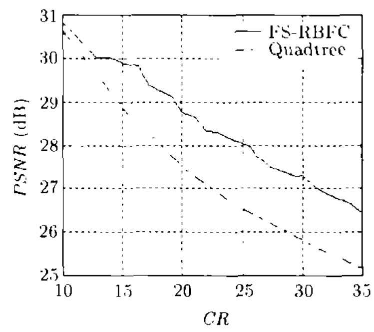


Fig. 6. Comparison of our FS-RBFC scheme and quadtree scheme

Table 1. The coding results of three standard test images

Image	FS-RBFC Region content information Bits	%	Region contour information Bits	%	Cover rate %	PSNR dB	CR	Number of regions	Quadtree CR at similar PSNR	Number of ranges
Lenna	32190	93.0	2434	7.0	96.2	29.88	15.14	1110	12.30	1711
Girl	30653	92.3	2555	7.7	96.7	32.33	15.79	1057	13.30	1585
Boat	47705	94.7	2647	5.3	94.2	28.48	10.41	1645	9.48	2209

Finally, the coding results of three standard test images 256×256 Lenna, girl and boat are analyzed to interpret the advantage of FS-RBFC scheme over quadtree scheme. Refer to Table 1 for statistics. Freely-shaped partitioning decrease the number of regions in the same fidelity, which are mainly region content information. Region contour information, accounting for less than 10% of the total, brings little memory cost in favor of compression efficiency. Furthermore, high cover rates weaken the disturbance of remaining pixels and assure the usability of linear predictor. High adaptivity to image content improves the quality of decoded image; also reduces loss of detail and blocking artifact. So our coder can outperform the conventional one with respect to rate-distortion performance and visual assessment in the nature of things.

VI. Conclusion and Outlook

In this paper, we have presented a highly image-adaptive fractal coder with freely-shaped regions, which is derived from coarse partitioning based on fractal dimension and fine partitioning based on region growth. The experiment results prove that our FS-RBFC is a better solution for fractal image coding than the conventional one. References show it has paralleled or even overtaken many existing fractal coders.

To completely exploit the potential of the FS-RBFC, new anisotropic growing strategies need to be considered to fit the direction of edges, that is, we try to update our region-based fractal coder (RBFC) to content-based fractal coder (CBFC). To excel current RBFCs completely, merging regions can also be applied to our FS-RBFC after the two-step partitioning, following the lead of publications[12–17]. These further improvements are currently being investigated.

Compared with some successful techniques (e.g. DCT, wavelet, Run-Length Coding, Adaptive Arithmetic Coding) which are integrated into still image coding standards JPEG, JBIG, JPEG2000, FIC is still far away from industrial application due to some limitations. Disambiguation of based-on theory and design of real-time algorithm (esp. for RBFC) are current tasks to be accomplished for researchers in this field.

References

[1] M.F. Barnsley and A.D. Sloan, "A better way to compress images", *Byte*, pp.215–223, Jan. 1988.

[2] A.E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations", *IEEE Trans. Image Processing*, Vol.1, pp.18–30, 1992.

[3] Y. Fisher, Ed., *Fractal Image Coding-Theory and Application*, New York: Springer, 1994.

[4] Y. Fisher and S. Menlove, "Fractal encoding with HV partitions", *in Fractal Image Coding-Theory and Application*, Y. Fisher, Ed. New York: Springer, 1994.

[5] D. Saupe, M. Ruhl *et al.*, "Optimal hierarchical partitions for fractal image coding", *in Proc. ICIP-98 IEEE Int. Conf. Image Processing*, Chicago, IL, Oct. 1998.

[6] F. Davoine, M. Antonini *et al.*, "Fractal image coding based on Delaunay triangulation and vector quantization", *IEEE Trans. Image Processing*, Vol.5, pp.338–346, Feb. 1996.

[7] F. Davoine and J.M. Chassery, "Adaptive delaunay triangulation for attractor image coding", *in Proc. 12th Int. Conf. Pattern Recognition*, Jerusalem, Israel, pp.801–803, Oct. 1994.

[8] H.O. Peitgen, H. Jurgens *et al., Chaos and Fractals: New Frontiers of Science*, Berlin, Germany: Springer-Verlag, 1992.

[9] M. Novak, "Attractor coding of images", *in Proc. Int. Picture Coding Symp.*, Lausanne, Switzerland, p.156, Mar. 1993.

[10] E. Reusens, "Partitioning complexity issue for iterated function systems based image coding", *in Proc. VIIth Eur. Signal Processing Conf.*, Vol.1, Edinburgh, U.K., pp.171–174, 1994.

[11] F. Davoine, J. Svensson and J.M. Chassery, "A mixed triangular and quadrilateral partition for fractal image coding", *in Proc. IEEE Int. Conf. Image Processing*, Washington, D.C., Vol.3, pp.284–287, Oct. 1995.

[12] L. Thomas and F. Deravi, "Region-based fractal image coding using heuristic search", *IEEE Trans. Image Processing*, Vol.4, pp.832–838, June 1995.

[13] M. Tanimoto, H. Ohyama and T. Kimoto, "A new fractal image coding scheme employing blocks of variable shapes", *in Proc. IEEE Int. Conf. Image Processing*, Lausanne, Switzerland, Vol.1, pp.137–140, Sept. 1996.

[14] Y.C. Chang, B.K. Shyu and J.S. Wang, "Region-based fractal image coding with quadtree segmentation", *in Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, Munich, Germany, Vol.4, pp.3125–3128, Apr. 1997.

[15] D. Saupe and M. Ruhl, "Evolutionary fractal image coding", *in Proc. IEEE Int. Conf. Image Processing*, Lausanne, Switzerland, Vol.1, pp.129–132, Sept. 1996.

[16] M. Ruhl, H. Hartenstein and D. Saupe, "Adaptive partitionings for fractal image coding", *in Proc. IEEE Int. Conf. Image Processing*, Santa Barbara, CA, Vol.2, pp.310–313, Oct. 1997.

[17] M. Breazu, G. Toderean, "Region-based fractal image coding using deterministic search", *in Proc. IEEE Int. Conf. Image Processing*, Chicago, Vol.1, pp.742–746, Oct. 1998.

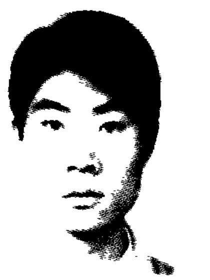
[18] Y. Fisher, E.W. Jacobs and R.D. Boss, "Fractal image coding using iterated transforms", *in Image and Text Compression*, J. Storer, Ed. Norwell, MA, Kluwer, 1992.

[19] D. Saupe and S. Jacob, "Variance-based quadtrees in fractal image coding", *Electron. Lett.*, Vol.33, No.1, pp.46–48, 1997.

[20] D.J. Jackson, W. Mahmoud, W.A. Stapleton and P.T. Gaughan, "Faster fractal image coding using quadtree recomposition", *Image Vis. Comput.*, Vol.15, pp.759–767, Oct. 1997.

[21] T. Kaneko and M. Okudaira, "Encoding of arbitrary curves based on the chain code representation", *IEEE Transactions on Communications*, Vol.COM-33, No.7, July 1985.

[22] S.R. Tate, *Lossless Compression of Region Edge Maps*, Dept. Comput. Sci., Duke Univ., Durham, NC, Tech. Rep. CS-1992-9, 1992.



SUN Yunda was born in Shandong Province, China in 1979. He received his B.E. degree from the School of Mechanics and Electrics, Beijing Jiaotong University in 2000. Since then he has been working towards the Ph.D. degree in signal and information processing in Computer Vision Lab, Institute of Information Science, Beijing Jiaotong University, Beijing. His current research interests include vision surveillance, computer vision and 3D reconstruction. (Email: samy.sun@163.com)

ZHAO Yao was born in Jiangsu Province, in 1967. He received B.E. degree from Fuzhou University in 1989 and M.E. degree from Southeast University in 1992, both in Radio Engineering Department. He received the Ph.D. degree in the Institute of Information Science, Beijing Jiaotong University in 1996. He became a professor in 2001. His research interest includes image coding, fractals, digital watermarking and content-based image retrieval. He has published more than 40 papers in international journals and conferences.

YUAN Baozong was born in Jiangsu Province, in 1932. He received the Ph.D. degree in electrical engineering from Leningrad Institute of Railway Engineering, USSR, in 1960. He has joined the Beijing Jiaotong University since 1953. He was a visiting professor at the University of Pittsburgh, USA, and the University of Wales, UK in 1982, 1983, and 1988 respectively. He is Chairman of Computer Chapter of IEEE Beijing Section, Fellow of British Royal Society, IEE Fellow, Vice Chairman of IEE Beijing Center Development. His research interests include computer vision, virtual reality, image processing, computer graphics, speech signal processing, multimedia information processing and data communication.