Region-Based Fractal Image Compression Using Deterministic Search

Macarie Breazu
University "Lucian Blaga" of Sibiu, Romania
4 Zaharia Boiu, Sibiu, 2400, Romania
e-mail: mac@cs.sibiu.ro

Gavril Toderean
Technical University of Cluj Napoca, Romania
26-28 Gh. Baritiu, Cluj Napoca, 3400, Romania
e-mail: toderean@utcluj.ro

Abstract

The paper introduces a new method based on deterministic search to fractal image compression. In order to find a good region-based partitioning, we propose a deterministic search method for finding the blocks to be merged. For each range, a list of best N domains is maintained. When two ranges are to be merged and their common edge disappears, for the new range the best N domains are selected only from the 2×N domain extension of the two ranges. At each step the edge with minimum collage error increase is deterministically selected and the two corresponding ranges are merged. The process starts with atomic blocks as ranges and ends when the desired number of ranges is achieved. In order to reduce the encoding time, a suboptimal initialization method is also considered. Experimental results prove that our method yields a better rate-distortion curve than the classic quad-tree partitioning scheme.

1. Deterministic search method

A major problem that researchers in fractal image compression face, beside reducing the encoding time, is image partitioning. Because of the huge searching space, the optimal partitioning problem for a desired bit-rate, cannot be practically solved. Until now, deterministically *hierarchical partitioning* (quadtree scheme [1], HV partitioning [2], polygonal partitioning [3]) and *split-and-merge* methods ([4], Delaunay triangulations [5][6], quadrilateral [7], heuristic search [8], evolutionary [9]) have emerged as solutions for the problem.

In order to comply with the spatial contraction of the fractal transform we consider the domains twice as large as the corresponding range. The spatial transform applied to a domain for matching the range size is the usual method of shrinking by pixel averaging. We also take in consideration all the 8 isometries (4 rotations and 4 flips) that can be applied to a block, which have the effect of enlarging the domain pool that has to be searched in order to find the best match.

For a range R and a domain D we determinate the scale and offset coefficients ***s*** and ***o*** by minimizing the collage error as a function having ***s*** and ***o*** as parameters. The value obtained for ***s*** is then clamped to the [-1,1] interval in order to assure the contraction in luminance space as well. The collage error becomes, after applying an uniform quantization to the parameters ***s*** and ***o*** and yielding $\bar{s}$ and $\bar{o}$,

$$E(R,D) = \left\| R - (\bar{s}D + \bar{o}\mathbf{1}) \right\|^2 \qquad (1)$$

where $\mathbf{1}$ is a uniform image block with each pixel having unit intensity. The bit stream transmitted to the decoder contains the codebook index of the best corresponding domain and the quantized values $\bar{s}$ and $\bar{o}$. Even if the fractal encoding method previously described is not new, being almost a standard one, we propose a new method for block merging.

Our method starts with an *initialization* phase consisting in:

- splitting the image in small square image blocks called *atomic blocks* of the same size (e.g. 8 by 8 pixels).
- building, for each atomic block, a list of best N corresponding domains, regarding the collage error, by an expensive *full-search* of the domain pool.

The initialization phase is followed by a *merging* phase in which ranges, initially identical to the atomic blocks, are merged in order to find a good partitioning which has less ranges. We consider ranges as frontier-connected sets of atomic blocks, and define "edge" the entire common border of two adjacent ranges. In order to form an edge, the two adjacent ranges must share at least one atomic block border. For each edge we compute the collage error increase if this edge would disappear. When an edge disappears, the two corresponding ranges are removed and replaced by their union as a new range. To

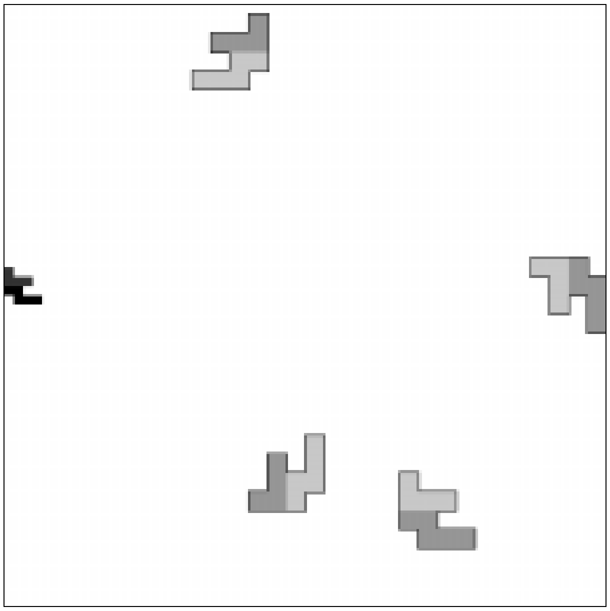

Figure 1. An example of range merging, obtained for N=2 on Lena image when two of the 3182 remaining ranges are merged

determine the edge's collage error increase we need to find the best domain for the union range. To avoid full-search for getting the new best N domains, we follow [9] and restrict the search only to the 2×**N** domains obtained by correspondingly enlarging each of the N domains of both initial ranges, as shown in Figure 1 (*small* figure - the two ranges which merge; *large dark* figures - the corresponding domains from both lists, N domains for each range; *large light* figures - the corresponding extensions of each domain). The best N=2 domains from these 2×N=4 are maintained in list for the merged range.

At each step of the merging process, the collage error increase being already known for all the edges in the partitioning, we can **deterministically** select the edge with the minimum collage error increase. Now the merging phase can be described as follows:

- compute, for each edge, the collage error increase in case of edge disappearance.
- select the edge whose disappearance would give a minimal increase of the collage error.
- merge the two corresponding ranges.
- compute the collage error increase in case of edge disappearance for all the edges of the new range.
- repeat the previous 3 steps until the desired number of ranges or the maximum value for the collage error is achieved.

During the merging phase, at each step, when the number of ranges is decremented, the bit-rate also decreases while the collage error increases, the rate-distortion curve being, approximately, continuously parsed.

Table 1. PSNR (dB) performance of the deterministic search method.

No. of ranges	Compr. ratio	Parameter N 2	5	10	20	50
4000	18.06	31.32	31.32	31.32	31.32	31.32
3500	20.44	31.30	31.31	31.31	31.31	31.31
3000	23.54	31.26	31.27	31.28	31.28	31.28
2500	27.74	31.16	31.19	31.20	31.21	31.22
2000	33.77	30.96	31.01	31.03	31.05	31.06
1500	43.16	30.52	30.59	30.62	30.66	30.67
1000	59.76	29.44	29.59	29.71	29.75	29.79
500	97.16	27.12	27.47	27.70	27.81	27.87

Table 2. Computation times needed to reach different number of ranges in partition

No. of ranges	Init. phase (sec)	Merging phase (sec) N=2	N=5	N=10	N=20	N=50
4000	3778	3	6	10	17	44
3500	3778	16	19	25	40	85
3000	3778	28	34	43	69	140
2500	3778	41	51	67	108	214
2000	3778	54	68	91	150	303
1500	3778	68	87	122	201	423
1000	3778	84	111	165	284	595
500	3778	104	147	231	400	904

An interesting feature of our deterministic merging algorithm is its simplicity, both in description and in parameters: beside the stop-condition it has only one parameter: the number N of the best domains maintained for each range.

2. Results

All the tests hereinafter presented are performed on the standard 512 ×**5**12 grey-scale Lena image. Considering atomic blocks of 8×**8** pixels, produces a 63 ×**6**3×**8** codebook that will be full-searched in the initialization phase. During the quantization step a 5-bit and 7-bit uniform quantization is used for ***s*** and ***o*** respectively. For the partition coding we use the simple method of describing, by using two bits, the status of an atomic bloc: connected or not with its right and lower neighbours.

In a first experiment we compare our deterministic search algorithm to the quad-tree method [10]. For a more correct comparison we use the quad-tree method without any classification (i.e. full-search) and vary the tolerance flag in the [1-20] area. The maximum recursion depth is chosen so that minimum block size is 8×**8** pixels. The

Figure 2. Partitioning with 2000 ranges (N=2, compression ratio 33.77, PSNR=30.96 dB)

Figure 3. Partitioning with 500 ranges (N=2, compression ratio 97.16, PSNR=27.12 dB)

Lena image compressed with our method and the resulting partitioning is shown in Figure 2 and Figure 3 while the rate-distortion curves obtained for both methods are presented in Figure 4. Obviously, the deterministic search method gives always better results, especially at high compression ratios.

We further test the effect of the parameter N on both the image quality (Table 1) and the encoding time (Table 2). The comparison of the ratio of the evolution phase

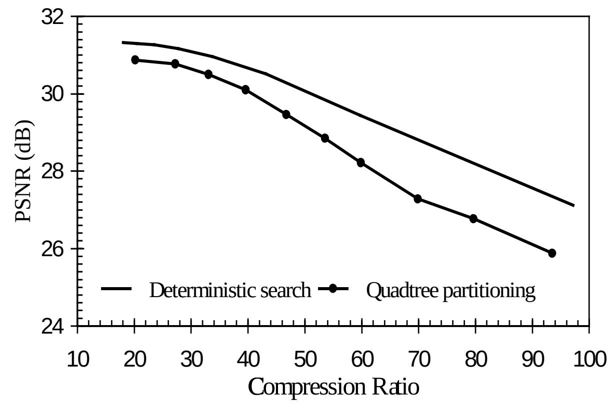

Figure 4. Rate-distortion curves for deterministic search method vs. quad-tree partitioning method (512×512 Lena image, N=2).

time to the initialization phase time (the only thing that can be compared when running on different machines) to the results presented in [9] proves that deterministic search method is a faster way to find a good partitioning of the image.

Studying the influence of the parameter N on the image quality proves us that, regardless of its value in the [2-50] interval, the image quality doesn't significantly decrease with N, even for high compression ratios. Therefore, time consuming values greater than 10 need not to be considered.

All our tests described in this paper are performed by running our test compression program (maybe not best optimized) on an Intel Pentium 166 MMX based PC machine. An advantage of our method, as compared to the quad-tree method, is that the time-expensive initialization phase is the same and needs not to be started over for different desired compression ratios.

Because most of the encoding time is spent on the full-search initialization phase, we study the performance of the deterministic search method with a suboptimal initialization. We choose the suboptimal method of reducing the size of the domain pool (proposed by Saupe in [11]) by keeping in the domain pool only the upper α fraction of domains variance-wise. In Table 3 we present the initialization phase time and the PSNR for the decoded image (having 4096 ranges) as function of parameter α. We notice that, while the image quality slightly decreases, the encoding time decreases linearly with α.

In Table 4 and Table 5 we present the results obtained by the deterministic search method after such a suboptimal initialization phase (corresponding to α=0.25 and α=0.10, respectively).

The effect of parameter α on decoded image quality is presented in Figure 5 (the case α=1.00 coresponds to the optimal full-search initialization phase). In Figure 6 these

Table 3. Suboptimal initialization effect on initialization phase

α	1.00	0.90	0.80	0.70	0.60	0.50	0.45	0.40	0.35	0.30	0.25	0.20	0.15	0.10	0.05
Time (sec)	3778	3426	3069	2706	2339	1964	1776	1583	1392	1200	1003	804	605	406	206
PSNR (dB)	31.32	31.32	31.32	31.32	31.32	31.32	31.30	31.28	31.26	31.23	31.20	31.09	30.96	30.75	30.34

Table 4. PSNR (dB) for deterministic search with suboptimal initialization (α=0.25)

No. of ranges	Compr. ratio	Parameter N				
		2	5	10	20	50
4000	18.06	31.20	31.20	31.20	31.20	31.20
3500	20.44	31.18	31.19	31.19	31.19	31.19
3000	23.54	31.15	31.16	31.16	31.16	31.17
2500	27.74	31.06	31.08	31.09	31.10	31.10
2000	33.77	30.88	30.91	30.94	30.95	30.95
1500	43.16	30.46	30.51	30.55	30.57	30.60
1000	59.76	29.47	29.60	29.64	29.71	29.77
500	97.16	27.21	27.53	27.68	27.78	27.90

Table 5. PSNR (dB) for deterministic search with suboptimal initialization (α=0.10)

No. of ranges	Compr. ratio	Parameter N				
		2	5	10	20	50
4000	18.06	30.75	30.75	30.75	30.75	30.75
3500	20.44	30.75	30.75	30.75	30.75	30.75
3000	23.54	30.72	30.73	30.73	30.73	30.73
2500	27.74	30.65	30.66	30.67	30.67	30.68
2000	33.77	30.50	30.51	30.54	30.55	30.56
1500	43.16	30.14	30.21	30.23	30.25	30.26
1000	59.76	29.29	29.44	29.51	29.54	29.58
500	97.16	27.24	27.55	27.74	27.76	27.86

results are presented (for comparison) together with the quad-tree partitioning method results. Some cases of decompressed Lena image (and corresponding partitionings) are also presented in Figure 7 and Figure 8.

Experimental results prove that image quality descrease due to suboptimal initialization is small and, even with suboptimal initialization, the determinsistic search method gives much better results than the classical quad-tree partitioning method.

3. Conclusion

The results presented in the paper prove that the deterministic search method, despite its simplicity, is, even in the suboptimal initialization case, a good solution for a very important problem in fractal image compression: finding a good image partitioning.

To completely explore the potential of the deterministic search method, future work has to be done regarding

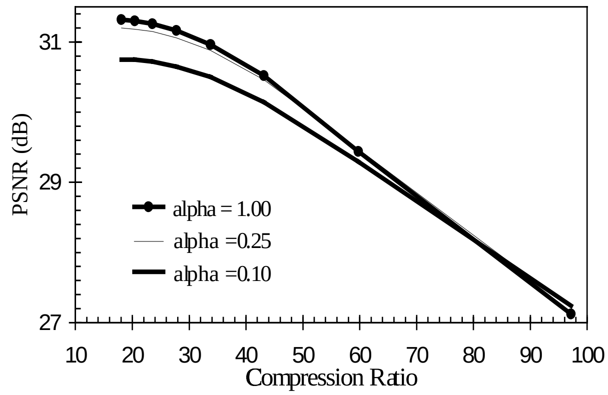

Figure 5. Rate-distortion curves for deterministic search with suboptimal initialization (N=2)

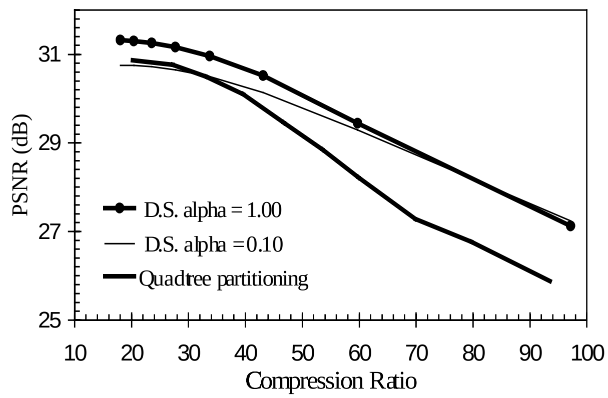

Figure 6. Rate-distortion curves for deterministic search (D.S.) method with suboptimal initialization vs. quad-tree partitioning (N=2)

the following, yet, "open-problems":

- Because most of the encoding time is spent on the full-search initialisation phase, the influence of other speeding up fractal image compression methods, both lossless [12] and suboptimal, needs to be assessed.
- In order to obtain a smoother partitioning of the image it might be promising to choose the atomic block of 4 ×4 pixels. Preliminary work shows that in this case, on the same machine, the full-search phase in the 127 ×127×8 domain pool requires a prohibitive 9h15'10'' computing time. Previous speeding up methods become, once again, very important.
- Classifying the blocks in predefined classes, differently written in the compressed stream, can increase

Figure 7. Decoded 512×512 Lena image with 500 ranges (compr. ratio 97.16), <u>α=0.25</u>, N=2, PSNR=27.21 dB, total encoding time <u>1107</u> sec.

Figure 8. Decoded 512×512 Lena image with 500 ranges (compr. ratio 97.16), <u>α=0.10</u>, N=2, PSNR=27.24 dB, total encoding time <u>510</u> sec.

the compression ratio. For example, classifying the blocks as shade ($\bar{s}$=0) and nonshade blocks ($\bar{s}$!=0) and not transmitting the codebook indices of the shade blocks will increase the 97.16 compression ratio to 97.59 with no decrease in image quality.

- The compression ratio can be further improved by using derivative chain codes [9] to describe the image partitioning.
- Entropy coding and other lossless compression methods can be used, both for range describing bits and for transformation describing bits, in order to reduce the bit stream without decreasing the image quality.
- Finally, the most interesting problem: if taking, at each step, the optimal decision, is the final solution the optimum one, i.e. if the global rather than a local minimum of the collage error is achieved.

4. References

[1] E. Jacobs, Y. Fisher, R.D. Boss, "Image compression: A study of the iterated transform method". *Signal Processing* 29, 1992, pp 251-263.

[2] Y. Fisher, S. Menlove, "Fractal encoding with HV partitions", in [10]

[3] E. Reusens, "Partitioning complexity issue for Iterated Functions Systems based image coding". in *Proc. of VII European Signal Processing Conference*, Vol. 1, Edinburg, U.K., September 1994, pp 171-174

[4] M. Tanimoto, H. Ohyama, S. Katsuyama, T. Kimoto, T. Fujii, "A new fractal image coding employing blocks of variable shapes," in *Proc. of IEEE International Conference on Image Processing* (ICIP'96), Laussane, Sept. 1996

[5] F. Davoine, M. Antonini, J.-M. Chassery, M. Barlaud, "Fractal image compression based on Delaunay triangulation and vector quantization," in *Proc. of IEEE Transactions on Image Processing* Vol. 5, No. 2, February 1996, pp 338-346

[6] F. Davoine, E. Bertin, J.-M. Chassery, "From rigidity to adaptive tessellations for fractal image compression: comparative studies," in *Proc. of IEEE 8th Workshop on Image and Multidimensional Signal Processing*, Cannes, September 1993, pp. 56-57

[7] F. Davoine, J. Svensson, J.-M. Chassery, "A mixed triangular and quadrilateral partition for fractal image coding," in *Proc. of IEEE International Conference on Image Processing (ICIP'95)*, Washington, D.C., 1995

[8] L. Thomas, F. Deravi, "Region-based fractal image compression using heuristic search," in *IEEE Transactions on Image Processing*, Vol. 4, No. 6, June 1995, pp 832-838

[9] D. Saupe, H. Hartenstein, "Evolutionary fractal image compression," in *Proc. of IEEE International Conference on Image Processing* (ICIP'96), Laussane, September 1996

[10] Y. Fisher (editor), *Fractal Image Compression - Theory and application*. New York, NY: Springer-Verlag, 1994.

[11] D. Saupe, "Lean domain pools for fractal image compression", in: *Proceedings from IS&T/SPIE 1996 Symposium on Electronic Imaging: Science & Technology -- Still Image Compression II*, Vol. 2669, Jan. 1996.

[12] D. Saupe, H. Hartenstein, "Lossless acceleration of fractal image compression by fast convolution," in *Proc. of IEEE International Conference on Image Processing* (ICIP'96), Laussane, Sept. 1996