Zooming with Implicit Fractals

D. M. Monro and P. D. Wakefield

School of Electronic and Electrical Engineering, University of Bath
Claverton Down, BA2 7AY, England
e-mail: D.M.Monro@bath.ac.uk paulw@ee.bath.ac.uk
Fax: +44 1225 826073
Internet: http://dmsun4.bath.ac.uk/

ABSTRACT

One advantage of fractal image compression schemes is their multiresolution properties. An image can be decoded at higher or lower resolutions than the original, and it is possible to 'zoom-in' on sections of the image. Even so the problem of fractal zooming has received very little attention. In this paper we examine these multi-resolution properties. We study the problem of fractal zooming in general and in particular with a hybrid fractal transform with implicit fractal terms. When decoding at a resolution higher than the original, artefact can be created which affects the visual quality of the zoomed image. We present one solution to the problem of obtaining clear and sharp edges where the original approximation quality was good. Our enhanced implicit fractal transform method compares favourably with fractal zooming using previous standard fractal approximation algorithms.

1. BACKGROUND

Much interest has focused on the Iterated Function System (IFS) as a method of image coding, the theory of which is widely available in the literature. [1-6]. To compress an image, define an IFS to be $W = \{w_k;\ k = 1,..., N\}$, where the w_k are contraction mappings, each defined on a subset A_k of the image support $\boldsymbol{S}$.

The attractor of W is a non-overlapping tiling of the image. A fractal function $f(x, y)$, is then defined which approximates the image brightness $g(x, y)$. An image block taken from the location A_k is referred to as the parent and its mapping $w_k(A_k)$ is referred to as the child. For each tile the brightness function is specified by a recursive mapping v_k such that

$$f(w_k(x, y)) = v_k(x, y, f(x, y)) \text{ for } (x, y) \text{ in } A_k. \qquad (1)$$

In this work we use mappings of the form

$$v_k(x_0 + \delta x, y_0 + \delta y, f) = p_k(x_0 + \delta x,\ y_0 + \delta y) + e_k \tilde{f}_k(\delta x, \delta y) \qquad (2)$$

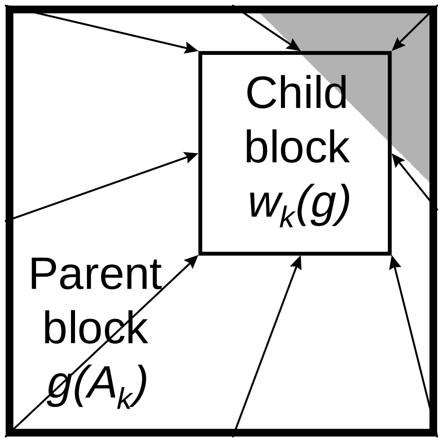


Figure 1. A local fractal transform, with an edge aligned in the parent and child blocks.

where (x_0, y_0) is the bottom left corner of A_k and $p_k(x_0 + \delta x,\ y_0 + \delta y) = \sum_{i=1}^{n} c_i\, b_i(\delta x, \delta y)$, is an approximation by basis functions $\{b_i\}$, e_k is the single fractal coefficient and $\tilde{f}_k$ is the parent block $f_k(\delta x, \delta y) = f(x_0 + \delta x, y_0 + \delta y)$ orthogonalized with respect to the basis. To solve, the known image g is used in place of the unknown fractal function f and the approximation is known to be valid by the Collage Theorem if suitable conditions are satisfied [1].

Usually the tiling of the image is by square or rectangular child blocks, and it is often assumed that p_k is a simple brightness level. Much work has concentrated on reducing the complexity of searching for the best parent to map onto each child [6, 7]. An alternative approach uses more complex basis functions [7, 8] and can restrict or even eliminate searching. With the Bath Fractal Transform (BFT) [9, 10] a pre-determined tiling without searching gives the greatest accuracy at a given compression ratio, when used with a quadratic basis. In combination with the Accurate Fractal Rendering Algorithm (AFRA) [5], the BFT has been used for real time fractal video [11].

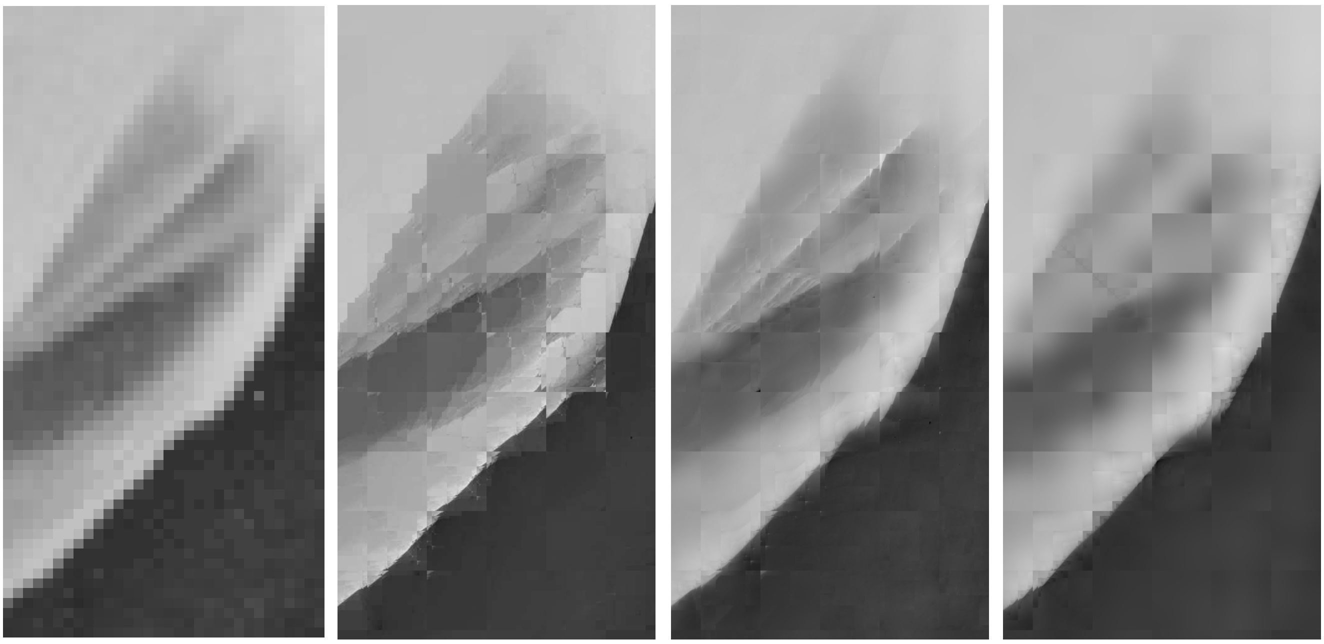
Figure 2. Lena image with zoomed areas marked.

A recent development [12] is the use of an implicit parent block location which the decoder can determine from the basis coefficients. If the existence of an edge is assumed, its location can be calculated from the ratios of the low order DCT coefficients using a pre-generated look-up table. The parent block location is then computed so that the edge is aligned in the shrunken parent and child, as in Figure 1. This implicit parent method has been shown to give a striking improvement in visual fidelity on some images. In [12] the implicit fractal was used in a near-optimal implementation in the rate-distortion sense. Table 1 shows measurements taken on the Lena image. The PSNR is slightly higher than that obtained by Fisher and Menlove at similar compression ratios [13], with much lower coder complexity (at 0.2 bpp, 2.8 sec. on a 200 MHz Pentium compared with 1122.1 sec on a Silicon Graphics IRIS 4D/35.)

2. FRACTAL ZOOMING

The fractal encoding is independent of resolution and a finite resolution decoding is performed with iterates produced at the desired resolution. In this manner the image or part of the image may be decoded to a higher resolution than the original, producing a 'zoomed' image.

To determine the effects of zooming we chose several fractal transforms that would result in a high fidelity approximation to the original. We used fixed size 6x6 child



(a) (b) (c) (d)

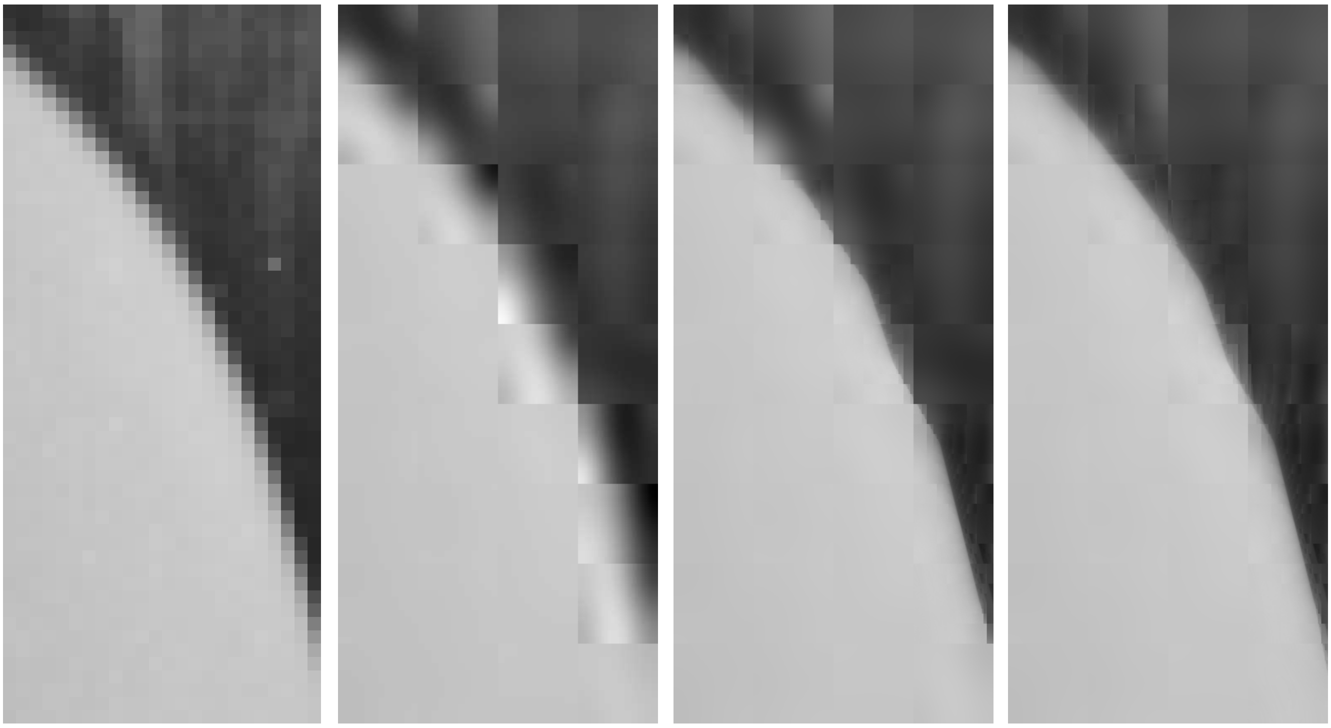
Figure 3. Zooming x16 (x256 pixels) in fractally encoded images with fixed 6x6 child blocks. (a) Original image. enlarged by pixel replication. (b) Zoomed fractal with DC basis and local searching.(c) Zoomed fractal with limited cosine basis and local searching. (d) Zoomed implicit fractal.

Figure 4. Cose up of edge in Figure 3(c), showing misalignment at block boundaries.

bpp	PSNR Basis Approx	PSNR Centred Child	PSNR Implicit fractal	Fractal % of Area	Fractal terms bpp
0.6	35.35	35.29	35.57	10.5	.016
0.5	34.62	34.53	34.80	15.4	.017
0.4	33.68	33.68	33.83	10.1	.010
0.3	32.43	32.42	32.66	13.2	.009
0.2	30.71	30.70	31.07	16.9	.007
0.1	28.13	28.09	28.35	26.3	.005

Table 1. Implicit fractal applied to Lena.

blocks with searching over ± 5 child block widths for the best 12x12 parent. We implemented a standard fractal transform as invented by Jacquin [2] with a simple DC basis, and also a version with a 2D DCT basis limited to the 6 lowest order terms. We compared these with an implicit fractal approximation [12] also using a fixed 6x6 block size partition. Because the distance of the parent from the child is limited in extent, when zooming we need to render only the area of interest extended by 5 child blocks, rather than the



(a) (b) (c) (d)

Figure 5. Fractal zooming x16 and enhanced fractal zooming with implicit fractals, using fixed 6x6 child blocks. (a) Original image with pixel replication. (b) Zoomed basis image. (c) Implicit fractal zoom. (d) Enhanced implicit fractal zoom.

entire image which would be huge. This greatly reduces the computation cost. Figure 2 marks two areas of interest. The results of zooming on the upper one of these by a factor of 16 in each spatial dimenstion (giving 256 times as many pixels) for each transform are displayed in Figure 3.

From these results we make several observations:

1. When edges are well approximated at the original resolution they are sharp and fairly well preserved in the zoomed images

2. The 'traditional' DC basis zoom produces significant distortion, while the more complex basis, as used in the BFT, gives more stable decoding for both the local searching transform and the implicit fractal

3. Edges do not always match well at block boundaries (see figure 4).

3. ENHANCED ZOOMING WITH IMPLICIT FRACTALS

In [12] the implicit fractal transform, used in a optimal quad-tree implementation provided excellent rate/distortion results. It was also shown that the performance of fractal coding schemes is improved by using implicit parent locations because they have zero bit-cost. As a result, we give the implicit fractal transform special consideration for zooming.

In [12] fractal terms were only used in a minority of blocks, in which they improved the PSNR of the compressed image, as in Table 1. When zooming, the non-fractal blocks are less visually satisfactory. For example, in Figure 5 (c), one block near the top left corner does not have a fractal term despite being part of an edge. Elsewhere along Lena's shoulder, 'notches' are created by non-fractal blocks, which are propagated by iteration onto neighbouring blocks.

We improve the implicit fractal zoom by introducing additional fractal terms at the decoding stage. Using the mathematical edge model from the coding stage, we estimate where edges in fractal blocks will intersect non-fractal blocks. If two edges meet the sides of a non-fractal block, we assume that block had an edge in the original which was judged to be too insignificant in the MSE sense to be coded, or which was lost when the image was digitised. We then compute a parent location and fractal coefficient based on the location of the intersecting edges and the coefficients of the fractal blocks. The improvement may be seen in Figure 5 (d).

4. DISCUSSION AND CONCLUSIONS

We have zoomed on images encoded with several different fractal techniques commonly employed for image coding and compression. The zooming process worked as expected and all the fractal zooms we produced looked better than pixel replication from the original resolution. The fractal zoom provides sharp edges in each of the edge blocks which are reasonably well preserved in the normal decoding, although edges between blocks do not always line up. Increasing the number of fractal blocks improved the visual quality of the zoomed implicit fractal transform.

Detail introduced by zooming is not present in the original pictures, so there is no 'original' with which to compare. As a result evaluation must be subjective, based on whether the new detail is visually an acceptable extension of the original. Assuming the original approximation quality is good we conclude that the zoomed image can be visually superior to pixel replication. We believe zooming on edges is a key problem in the development of fractal zooming. We see this as an important area of interest in further work.

5. ACKNOWLEDGEMENTS

P. D. Wakefield is supported by a University of Bath Studentship, with additional support from Network Digital Systems (NDS) Ltd. The authors would like to thank David Bethel for providing rate/distortion results and Gary Dickson for preparation of the drawings.

6. REFERENCES

[1] Barnsley, M. F., 'A better way to compress images', BYTE, Vol. 13, No. 1, pp. 215-224, January 1988.
[2] Jacquin, A. E., 'A novel fractal block-coding technique for digital images', Proc. ICASSP 1990, pp. 2225-2228.
[3] Jacquin, A. E., 'Fractal coding: a review', Proc. IEEE, Vol. 81, No. 10, October 1993, pp. 1451-1465.
[4] Saupe, D. and Hamzaoui, R., 'A review of the fractal image compression literature', Computer Graphics Vol. 28, No. 4, 1994, pp. 268-279.
[5] Monro, D. M., and Dudbridge, F., 'Rendering algorithms for determinisitic fractals', IEEE Computer Graphics, Vol. 15, No. 1, January 1995, pp. 32-41.
[6] Monro, D. M., and Dudbridge, F., 'Fractal approximation of image blocks', Proc. ICASSP 1992, pp. III: 485-488.
[7] Øien, G. E., Lepsøy, S. and Ramstad, T., 'Reducing the complexity of a fractal-based image coder', Proc. Eur. Signal Proc. Conf. (EUSIPCO), pp. 1353-1356, 1992
[8] Saupe, D., 'Accelerating fractal image compression by multi-dimensional nearest neightbour search', Proceedings IEEE Data Compression Conference (DCC95), March 1995.
[9] Monro, D. M., 'A hybrid fractal transform', Proc. ICASSP 1993, pp. V:169-172.
[10] Monro, D. M., and Woolley, S. J., 'Optimum parameters for hybrid fractal image coding', Proc. ICASSP 1995, 4, pp. 2571-2574.
[11] Monro, D. M., and Nicholls, J. A., 'Low bit-rate colour fractal video', Proc. ICIP 95, p.III: 264-267.
[12] Wakefield, P. D., Bethel, D. M., and Monro, D. M., 'Hybrid image compression with implicit fractal terms', Proc. ICASSP 1997, pp. IV-2933-2936.
[13] Fisher, Y. and Menlove, S., 'Fractal encoding with HV partitions', in Fractal Image Compression, Fisher, Y. ed., Springer-Verlag, 1995.