Fast Image Domain Fractal Compression by DCT Domain Block Matching

B E Wohlberg and G de Jager
Digital Image Processing Laboratory,
Electrical Engineering Department,
University of Cape Town,
Private Bag, Rondebosch 7700, SOUTH AFRICA
{brendt, gdj}@dip1.ee.uct.ac.za

June 26, 1995

Abstract

Fractal image compression entails a computationally costly search for matching image blocks. It is proposed that a block be represented by its DCT coefficients, which may be made invariant to many of the usual block transforms. Due to the energy packing properties of the DCT, the efficiency of an existing multidimensional nearest neighbour search is improved.

1 Introduction

Fractal image compression [1] is closely related to certain forms of Vector Quantisation (VQ), with the distinction that a codebook is not required since contractivity of the mapping allows decoding by iterative application to an arbitrary initial image. While fractal compression enjoys certain advantages over other coding methods [2], the encoding process is extremely time consuming. A number of solutions such as block classification [3] have been proposed to the problem of finding a matching domain block for each range block.

Saupe [4] reports on the use of a multi-dimensional nearest-neighbour search algorithm, based on a partition of the search space by a *k-d* tree [5], operating in $O(N \log_2 N)$ time for N domain blocks. A scale and offset invariant representation is found within which to perform the nearest-neighbour search.

2 Discrete Cosine Transform

Promising results have previously been obtained for fractal compression in the Discrete Cosine Transform (DCT) domain [6] [7] but since the inverse DCT is performed for every image block on decoding, the decoding stage is not as rapid as fractal compression in the image domain. We propose that the properties of the DCT be utilised for compression in the image domain, speeding up the encoding process and avoiding the inverse DCT on decoding.

Published as: B. Wohlberg and G. de Jager, "Fast image domain fractal compression by DCT domain block matching," *Electronics Letters*, vol. 31, pp. 869–870, May 1995.

The relevant transform properties of the DCT domain representation are:

$$C_{f(x,y)-o}(u,v) = C_{f(x,y)}(u,v) - oN\delta_0^u\delta_0^v \quad (1)$$

$$C_{sf(x,y)}(u,v) = sC_{f(x,y)}(u,v) \quad (2)$$

$$C_{f(y,x)}(u,v) = C_{f(x,y)}(v,u) \quad (3)$$

$$C_{f(N-1-x,y)}(u,v) = (-1)^u C_{f(x,y)}(u,v) \quad (4)$$

$$C_{f(x,N-1-y)}(u,v) = (-1)^v C_{f(x,y)}(u,v) \quad (5)$$

where $C_{f(x,y)}(u,v)$ is the $N \times N$ (indices range from 0 to $N-1$) DCT [8, pg. 144] of $f(x,y)$ and δ_x^y is the Kronecker delta.

All of these operations commute with each other, for example equations 4 and 5 may be combined to give $C_{f(N-1-x,N-1-y)}(u,v) = (-1)^{(u+v)}C_{f(x,y)}(u,v)$.

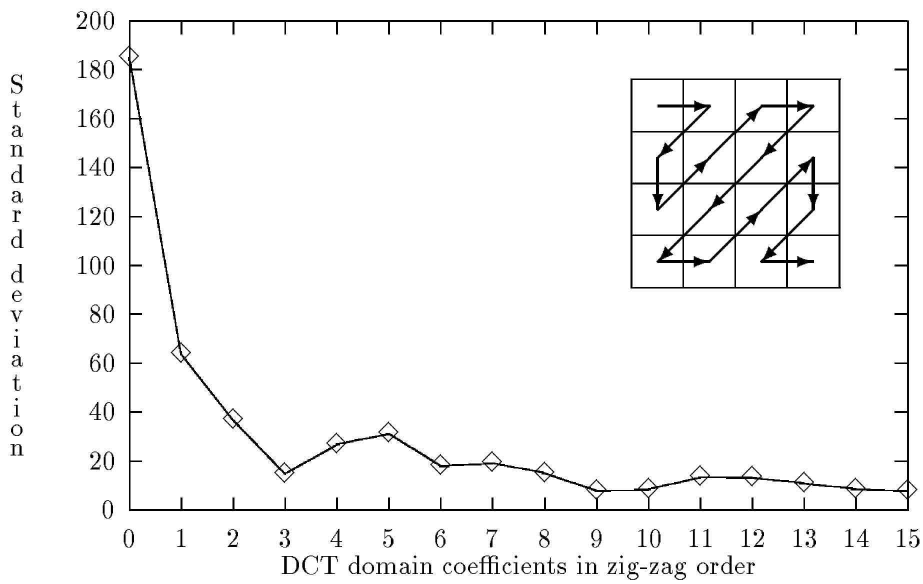

Figure 1: Standard deviations of DCT coefficients for all 8×8 domain blocks (after spatial contraction to 4×4 blocks) in the 256×256 grey scale "Lena" image. Inset illustrates "zig-zag" scan order on a 4×4 block of DCT coefficients.

The main utility of the DCT in this application is its energy packing properties. When the search keys are constructed by taking the DCT coefficients in "zig-zag" scan order (see inset in figure 1), which corresponds to increasing spatial frequency [9, pp. 172-173], there is a rapid decrease in variance from the most to least significant keys (see figure 1), which improves the efficiency of the search algorithm [5, pg. 214].

Other advantages of the DCT representation are:

- Storage requirements for each processed domain block may be reduced by the application of a normalisation matrix [8, pg. 386], after which each component value may be represented by a decreased number of bits.
- Since the RMS distance measure is known to be suboptimal, Human Visual System (HVS) weighting [9, pp. 34-38] may be applied to the DCT coefficients to provide a more accurate measure of perceived distance.

- Since the scale and offset are decoupled by the DCT, contractivity ($|s| < 1$ for scaling factor s) may be rapidly checked without the inner product computations required in the image domain.
- Symmetry operations may be rapidly applied in the DCT domain, requiring only multiplications by ± 1.

3 Canonical representation

A canonical representation with respect to the allowed operations is required to allow application of nearest-neighbour search techniques, translating the problem from a distance minimisation over all domains and operations to a minimisation over domains alone.

Such a representation with respect to scale, offset and transposition may be obtained by considering the DCT of each image block. We achieve offset invariance by setting the DC coefficient to zero (see equation 1). Scale invariance, up to a factor of ± 1, is achieved by normalising the remaining coefficients (see equation 2) by the block magnitude, which is the l^2 norm on the appropriate coefficients. Transposition invariance (see equation 3) is obtained by ensuring that the magnitude of the "lower triangle" is greater than that of the "upper triangle" (excluding the components on the main diagonal). Unfortunately a distance preserving invariant representation with respect to the remaining symmetry operations is not possible within this framework.

4 Implementation

An algorithm closely related to the *k-d* tree method, but utilising binary searches within a lexicographically ordered array of multidimensional keys was used. Canonical representations were computed for each spatially contracted domain vector, followed by normalisation and quantisation [8, pg. 386] to reduce memory requirements before insertion into the array.

The DC coefficient, the scaling factor and a flag indicating whether transposition was applied, were stored with each set of coefficients. A similar procedure was followed for each range block prior to the nearest neighbour search. Once a matching domain block had been identified, the scale and offset for the mapping were

$$s = \frac{S_D}{S_R} \quad o = \frac{O_R - sO_D}{N}$$

respectively, for $N \times N$ range blocks, where S_D, O_D were the scaling factor and DC coefficient for the domain block, and S_R, and O_R were the corresponding range block values.

5 Results

Tests were conducted on 256×256 grey scale "lena" and "bridge" images for fixed range block sizes of 4×4 and 8×8, where domain blocks were spatially contracted by a factor of two on each side. Comparisons of the canonical representation suggested in [4] and the DCT representation reveal an improvement in total computation time by a factor of 1.5 for the larger blocks to 2.0 for the smaller blocks. Comparisons of subjective image quality indicated a slight advantage for the DCT based method, where differences were discernible. Since an optimised DCT algorithm was not utilised, it is felt that these results convey a conservative estimate of the possible performance gains.

6 Comments

There are sound theoretical reasons for expecting improved performance during the domain search by application of the techniques presented here. While empirical evidence has been obtained, further comparisons are required for an accurate assessment of the computational advantage, and further investigation into the use of HVS weighting is necessary. It should be noted that this method is also expected to be applicable to certain forms of VQ coding.

References

[1] Y. Fisher, E. Jacobs, and R. Boss, "Fractal image compression using iterated transforms," in *Image and Text Compression* (J. A. Storer, ed.), ch. 2, pp. 35–61, Norwell, Massachusetts: Kluwer Academic Publishers, 1992.

[2] Y. Fisher, T. P. Shen, and D. Rogovin, "A comparison of fractal methods with DCT (JPEG) and Wavelets (EPIC)," in *SPIE Procedings, Neural and Stochastic Methods in Image and Signal Processing III*, vol. 2304-16, (San Diego, CA), July 28-29 1994.

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

[4] D. Saupe, "Breaking the time complexity of fractal image compression," tech. rep., Institut für Informatik, Universität Freiburg, 1994.

[5] J. H. Friedman, J. L. Bentley, and R. A. Finkel, "An algorithm for finding best matches in logarithmic expected time," *ACM Transactions on Mathematical Software*, vol. 3, pp. 209–226, Sept. 1977.

[6] Y. Zhao and B. Yuan, "Image compression using fractals and discrete cosine transform," *Electronics Letters*, vol. 30, pp. 474–475, Mar. 1994.

[7] K. U. Barthel and T. Voyé, "Adaptive fractal image coding in the frequency domain," in *Proceedings of the International Workshop on Image Processing*, Theory, Methodology, Systems and Applications, (Budapest, Hungary), June 1993.

[8] R. C. Gonzalez and R. E. Woods, *Digital Image Processing.* Reading, Massachusetts: Addison-Wesley, 3rd ed., 1992.

[9] W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standard.* New York: Van Nostrand Reinhold, 1993.