A NOVEL SUBTREE PARTITIONING ALGORITHM FOR WAVELET-BASED FRACTAL IMAGE CODING

L. M. Po Y. Zhang** K. W. Cheung* C. H. Cheung**

* CityU Image Processing Lab, Dept. of Electronic Engg., City University of Hong Kong, Hong Kong, China
**Dept. of Planning, Developing & Importing, Guangdong Posts & Telecommunications, Guangzhou, China

ABSTRACT

In this paper, a novel wavelet subtree partitioning algorithm is proposed, which divides a subtree into scalar quantized wavelet coefficients and fractal coded sub-subtree. Based on this new technique, a variable size wavelet subtree fractal coding scheme for still image compression is developed. Experimental results show that the new scheme can achieve nearly optimal partition of wavelet subtree with substantially computational reduction as compared with Davis' scheme.

1. INTRODUCTION

Recently, G. M. Davis [8] and H. Krupnik et. al. [9] independently generalized the fractal coding [4-6] from spatial domain to the wavelet domain [1-3]. Davis coined a new term wavelet subtree for representing the hierarchical data structure of an image decomposed in wavelet pyramid. The wavelet subtree consists of the wavelet coefficients that has the same spatial location but with different resolution and orientation. Fig. 1 is the coefficient structure of 4-level wavelet decomposition for an image. Each subband coefficient at coarser scale is related to a 2×2 coefficients at the next finer scale with the same orientation. Thus, the three coefficients with the same spatial location from the three bandpass subbands at the coarsest scale together with their children and grandchildren, i.e. 2×2, 4×4 and etc. coefficients at successive finer scales, are highly correlated. They can be grouped together to form a wavelet subtree as shown in Fig. 1 with triangle pixels. Such wavelet subtree is denoted as D_p with p = 4 and p is the scale level of the root nodes. The wavelet subtree can have root nodes starting from high frequency subband at finer scale. A wavelet subtree R_q with q = 3 and root nodes at scale level 3 is also depicted in Fig. 1 as square pixels. These wavelet subtrees, R_q and D_p, can form square blocks of corresponding dimension by scanning the coefficients. The scanning order of wavelet coefficients for R_q to construct an 8×8 square block is shown in Fig. 2. As contrasted with spatial domain blockwise fractal coding, R_q and D_p are called range and domain subtrees, respectively.

The main idea of wavelet subtree fractal coding is to approximate each range subtree by a domain subtree through fractal transformation as in spatial domain. Owing to the special structure of wavelet subtree, the affine transformation used for wavelet subtree mapping is different from the spatial domain fractal coding. Let τ denote the affine transformation and it is composed of the following three parts:

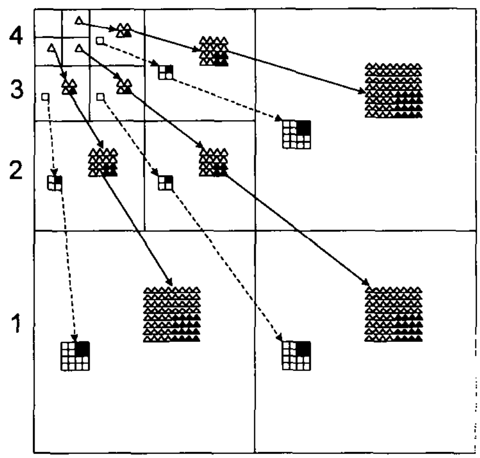

Fig. 1: A four-stage wavelet decomposition: domain subtree and range subtree consists of the triangular pixels and the square pixels. The shaded square/triangular pixels represent the split quadrant range/domain subtree.

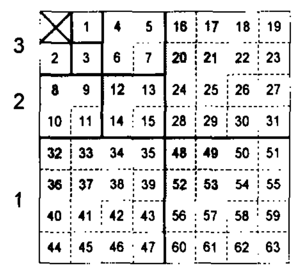

Fig. 2: Scanning order of wavelet coefficients to form a wavelet subtree. The shade area represents one of the four children subtree.

i) *Geometric part* Γ : It truncates all the finest leaf nodes of the domain subtree D_p to match the tree size of the range subtree R_{p-1}, and scale down all coefficients by 1/2.

ii) *Shuffle part* Π : The horizontal, vertical and diagonal reflection of wavelet coefficients are carried out within each subband separately. The rotation operation is performed within each subband and followed by switching the W_j^{LH} coefficients with W_j^{HL} coefficients.

iii) *Massic part* Ω : It operates directly on the coefficients of a wavelet subtree. As the wavelet subtree does not have DC component, the shift factor is unnecessary. Thus Ω is defined as $\Omega(D_p) = \alpha \cdot D_p$, where α is the scaling factor.

The affine transformation τ for wavelet subtrees is the composition of the geometric, shuffle and massic part, which can be expressed as

$$\tau(D_p) = \Omega \circ \Pi \circ \Gamma(D_p) = \alpha \cdot \Pi(\Gamma(D_p)) \tag{1}$$

For a given R_q, the main task of fractal coding is to find the best-matched D_p that can approximate the R_q with the minimum distortion through certain affine transformation. If the mean squared error is used as the distortion measure, the range-domain comparison can be formulated as

$$\varepsilon = \min_{D_p \in S}\left[\left\| R_{p-1} - \tau(D_p) \right\|^2\right] \tag{2}$$

The scaling factor α is obtained when the minimum error occurs $(\partial\varepsilon / \partial\alpha) = 0$. After full searching among the domain pool, the best-matched domain subtree is found and the fractal transformation parameters are recorded as fractal dodes.

Recently, Davis [10] further developed an adaptive wavelet coding algorithm using joint optimized scalar quantization and fractal compression which outperforms the state-of-the-art fractal coding scheme. The blocking effects can be reduced substantially at very low bit-rates. His idea is to partition a wavelet subtree into two parts, separate nodes and several children subtrees. The separate nodes are coarse-scale wavelet coefficients are scalar quantized independently. The children subtrees are quantized by fractal coding. Hence, it is crucial to find the optimal partition of the wavelet subtree for the best coding performance. However, Davis' optimization process is an iterative algorithm carried out on each node of a wavelet subtree from bottom to top. At each iteration, a Lagrangian cost for construction is computed for determining whether the subtree node is a pruned node or a unpruned node with children. At the same time, the partition of the wavelet subtree is modified accordingly. Another step of quantizer optimization for each subband is performed to reduce the total Lagrangian cost of the quantized coefficients and fractal quantized subtrees. This optimization process is repeated for all nodes which have been checked for several cycles. Although the iterative algorithm can obtain the optimal partition of a wavelet subtree, the heavy computation complexity is a burden for practical applications. To tackle this problem, a non-iterative subtree partition algorithm is proposed in this paper. The new subtree partition algorithm can substantially reduce computational complexity as compared with Davis' method while maintaining the coding performance.

2. SUBTREE PARTITIONING

Quadtree partitioning is widely used in spatial domain variable size fractal coding [6] to adaptively separate a range block into quadrant subblocks. Its superior performance leads to the development of adaptive variable size wavelet subtree fractal coding algorithms. In subtree partitioning, if a range subtree R_q cannot find a good-matched domain subtree D_{q+1}, it will be divided into smaller subtrees for separate encoding. A natural extension of quadtree partitioning is to remove the three root nodes from the range subtree and then split the pruned range subtree into four quadrant children subtrees R_{q-1}. Such quad-subtree partitioning is shown in Fig. 1, where the shaded square/triangular pixels represent one of the four children subtrees. These children subtrees will be encoded in the way same as that for their parent. This partition algorithm is very simple in implementation. However, it is not an efficient representation of the tree structure.

In fact, the four children range subtrees R_{q-1} are close to each other in both wavelet and spatial domain. Thus, they are highly correlated and can be combined to form a new range subtree R'_{q-1} (sub-subtree). If R'_{q-1} can find a good-matched domain subtree D'_q, nearly 75% reduction in bits can be achieved as compared with coding four children range subtrees. The proposed variable tree partitioning algorithm for coding a given range subtree R_q can be summarized as follows.

(1) Based on a predefined distortion threshold T_q, check whether the initial range subtree R_q can be fractal encoded by a domain subtree D_{q+1} within the distortion T_q.
(2) If R_q can be fractal coded, the coding is finished and the next range subtree will be processed. Otherwise, the three root nodes are removed from R_q and scalar quantized. The remaining coefficients are combined to form a new range subtree R'_{q-1}.
(3) Based on another distortion threshold T'_{q-1}, the newly formed range subtree R'_{q-1} is encoded as its parent using the corresponding domain subtrees D'_q searching pool. If the distortion between the best-matched D'_q and the R'_{q-1} is below T'_{q-1}, the coding is finished. Otherwise, if the distortion exceeds T'_{q-1}, R'_{q-1} is divided into four children subtrees R_{q-1}.
(4) Each of the four children subtrees R_{q-1} is encoded as their parent. If further splitting is allowed, just let q=q-1 and then go to step 2 to repeat the subtree partitioning.

This subtree partition is repeated until all range subtrees have found their best-matched domain subtrees within the given distortions, or the allowed minimum dimension of range subtree is reached.

3. NEW SUBTREE FRACTAL CODING

Based on the proposed subtree partitioning algorithm, a new variable size wavelet subtree fractal coding scheme is developed. Without loss of generality, the scheme is presented with image of 512×512 size and 6-level wavelet decomposition. The block diagram of the proposed variable size wavelet subtree fractal coding is illustrated in Fig. 3. The first three steps are just the same as the fixed size wavelet subtree fractal coding algorithm: 1) Pyramidal wavelet decomposition, 2) Scalar quantization of the four subbands at the coarsest scale, and 3) Construction of initial range subtrees R_5 and domain subtrees D_6 of size 1023 and 4095, respectively. There are altogether 256 R_5 and 64 D_6. For each R_5, full search is employed to find the best-matched D_6 with minimum approximation distortion ε_5 through fractal transformation. In fixed size wavelet subtree fractal coding, the coding process is finished regardless of the distortion. However, in the proposed variable size wavelet subtree fractal coding, ε_5 is compared with a predefined threshold T_5 to see whether the matching distortion is small enough.

It is obvious that the range subtree can be "fractal coded" if $\varepsilon_5 \leq T_5$. In this case, the bits allocated for R_5 will be the minimum. Note that an additional bit is assigned to indicate that R_5 is encoded by fractal quantization and further splitting is not necessary. On the contrary, if $\varepsilon_5 > T_5$, the proposed subtree partitioning algorithm is applied to divide the R_5 into several parts for separate coding to reduce the overall distortion.

The badly matched R_5 is first divided into two parts, three root nodes and the pruned subtree R'_4 containing 1020 nodes. The three root nodes are scalar quantized independently within their own subbands. The pruned subtree R'_4 differs from R_5 in having 12 root nodes instead of 3 at the highest tree layer. Similarly, the corresponding domain subtrees D'_5 can be obtained by removing the three root nodes from D_6 so that the searching pool still contains 64 disjoint domain subtrees D'_5. As larger searching pool can usually reduce the distortion of the best-matched domain subtree, the domain subtrees D'_5 searching pool is enlarged by overlapping one coefficient with the neighboring domain subtrees in both horizontal and vertical direction within the three subbands at scale level 5. The tree nodes of D'_5 at the successive scale levels will overlap by 2, 4, 8, and 16 coefficients, respectively. That means the searching pool is now formed by 256 D'_5 of size 4092. The effectiveness of using overlapped domain subtrees is confirmed by experimental results.

Same as before, the best-matched domain subtree D'_5 is found and the distortion ε'_4 between the matched subtrees is calculated. If ε'_4 is below a predefined threshold T'_4, R'_4 is encoded successfully and further splitting is not needed. Otherwise, if $\varepsilon'_4 > T'_4$, R'_4 is divided into four children subtrees R_4 of size 255. Again an additional bit is required to indicate whether R'_4 is split or not. The corresponding domain subtrees D_5 of size 1023 with root nodes at scale level 5 are constructed to form the searching pool. This construction is very simple because D_5 are, in fact, the range subtrees R_5. Therefore, there are 256 non-overlapped D_5 for fractal coding. Each of four children range subtree is encoded as their parent nodes. Finally, the best-matched domain subtree D_5, which has the minimum distortion is found.

When the children range subtrees R_4 cannot be encoded within a predefined distortion T_4, the variable tree partition algorithm is further applied to divide them into smaller subtrees. At this time, the pruned children subtrees R'_3 and the corresponding domain subtrees D'_4 are of size 252 and 1020, respectively. We also consider the domain subtrees with one root node overlapping so that the searching pool contains 1024 different domain subtrees. Similarly, each R'_3 will find a matched D'_4 with the minimum distortion ε'_3. If the distortion is below a threshold

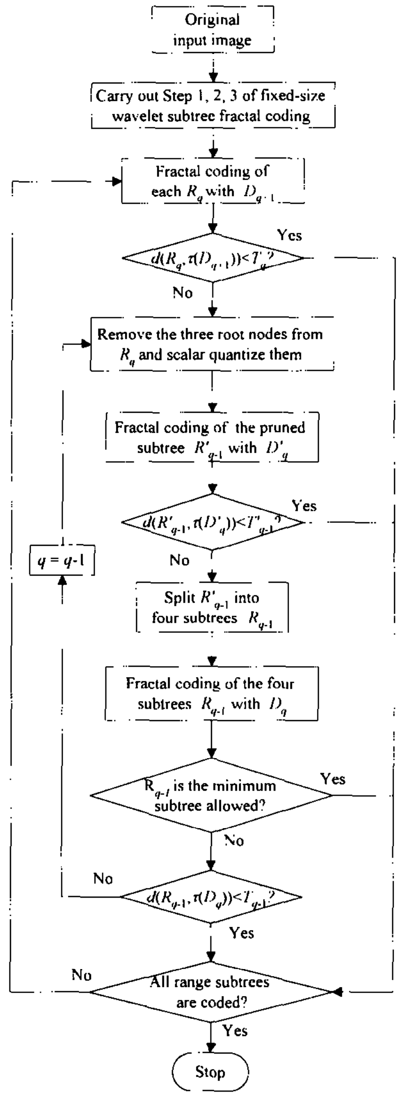

Fig. 3:.The block diagram of the proposed variable size wavelet subtree based fractal coding algorithm.

T'_3, the subtree is coded by fractal quantization. Otherwise, it will be further partitioned into four quadrant subtrees R_3 of size 63. The minimum subtree dimension is set as 63. Thus, each subtree R_3 will be fractal coded regardless of the distortion.

4. EXPERIMENTAL RESUTLS

The 512×512 gray-level *Lenna* with 8 bits per pixel is used for testing our coding scheme. Biorthogonal wavelets, B9/7, is employed. The distortion thresholds to determine the wavelet subtree splitting are adjusted for different size of range subtrees and they are set as T_5 = 7.2, T'_4 = 8.2, T_4 = 10.2 and T'_3 = 11.2. In addition, adaptive arithmetic coding is applied to the scalar quantized coefficients, fractal transformation parameters and tree node symbols to generate the output bit stream. The reconstructed image is illustrated in Fig. 4a with the compression ratio of 64:1. To show the superior coding performance, the result of the proposed algorithm is compared with that of JPEG and Davis' method in terms of PSNR and compression ratio in Table 1. Although JPEG is the standard for still image compression, it is very difficult to obtain an acceptable image quality at very low bits rates. The proposed algorithm can achieve 5.4dB improvement in PSNR as compared to JPEG and the visual quality is also much better than JPEG (see Fig 4b). In addition, the experimental result is also slightly better than that of Davis' algorithm [10] in terms of PSNR value and compression ratio. Thus, the proposed variable tree partition algorithm has obtained nearly optimal partition of wavelet subtrees, and its implementation is simpler due to the relatively lower computation requirement as compared to Davis' method.

5. CONCLUSIONS

A simple wavelet subtree partitioning algorithm for variable size wavelet subtree fractal image coding is proposed. Based on this partitioning algorithm, range subtree is adaptively segmented into various detail regions according to local details. The domain subtrees are constructed with some overlapping area for enlarging the searching pool. Experimental results show that the proposed hybrid image coding algorithm can obtain a much better reconstructed image at very low bit-rates than the JPEG in terms of PSNR as well as subjective quality. In addition, its performance is also slightly better than Davis' iterative optimized partition coding algorithm. The major advantage of the proposed coding scheme is that no computational intensive iteration process is needed as compared to Davis' iterative optimized partition method. The algorithm is carried out from "top-to-bottom" instead of "bottom-to-top", thus, requiring fewer nodes checking to determine the final subtree partition. The heavy Lagrangian sum calculation is also avoided. The simpler encoder structure and lower computation requirement make this proposed coding scheme suitable for practical applications.

	JPEG	Davis [10]	New method
Compression ratio	60.2:1	64.4:1	64.69:1
PSNR	24.24	29.6	29.62

Table 1: Comparisons of coding results for test image *Lenna* in terms of PSNR (dB) and compression ratio.

(a)

(b)

Fig. 4: Coding results for test image *Lenna* of size 512×512. (a) Encoded using biorthogonal wavelets "B97" CR = 64.69:1 with PSNR = 29.62dB, (b) JPEG, CR = 60.2:1 with PSNR = 24.24 dB.

REFERENCES

[1] I. Daubechies, "Orthonormal bases of compactly supported wavelets", *Commun. Pure Appl. Math.*, vol. 41, pp. 909-996, 1988.

[2] J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients", *IEEE Trans. on Signal Processing*, vol. 41, no.12, pp. 3445-3462, Dec. 1993.

[3] M. Antonini, M. Barlaud, P. Mathieu and I. Daubechies "Image coding using wavelet transform", *IEEE Trans. on Image Proc.*, vol. 1, no. 2, pp. 205-220, Apr. 1992.

[4] M. F. Barnsley and L. P. Hurd, *Fractal Image Compression*, AK peters, Wellesley, Massachusetts, 1993.

[5] A. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations", *IEEE Trans. Image Proc.*, vol. 1, no. 1, pp18-30, Jan. 1992.

[6] Y. Fisher, "Fractal image compression with quadtrees", in *Fractal Compression: Theory and Application to Digital Images*. Y. Fisher, Ed. Springer Verlag, New York, 1994.

[7] R. Rinaldo and G. Calvagno, "Image coding by block prediction of multiresolution subimages", *IEEE Trans. on Image Processing*, vol. 4, no.7, pp. 909-920, July 1995.

[8] G. Davis, "Self-quantization of wavelet subtrees", in *Proc. SPIE Wavelet Applications II*, Orlando, vol. 2491, pp.141-152, Apr. 1995.

[9] H. Krupnik, D. Malah, and E. Karnin, "Fractal representation of images via the discrete wavelet transform", in *IEEE 18th Conf. of EE in Israel*, Tel-Aviv, Mar. 1995.

[10] G. Davis, "A wavelet-based analysis of fractal image compression", accepted for publication in *IEEE Trans. on Image Processing, 1997*.