ADAPTIVE IMAGE PARTITIONING FOR FRACTAL CODING ACHIEVING DESIGNATED RATES UNDER A COMPLEXITY CONSTRAINT

Reuven Franco and David Malah

Department of Electrical Engineering[1]
Technion – I.I.T, Haifa 32000, Israel
Email: rfranco@tiogatech.com malah@ee.technion.ac.il

ABSTRACT

Fractal image coding is a relatively new technique for compact image representation. The basic coding scheme exploits self-similarities between parts of the image and other parts in it at a different resolution. The various parts are consequences of a partition grid obtained by applying a splitting criterion to the image. In this work, we present an algorithm for adaptive image partitioning, achieving designated rates under a computational complexity constraint. The proposed algorithm results in a reduction of the computational complexity as compared to other known algorithms at the same rate-distortion operating point. Also presented is an efficient procedure for approximating parts in the image by a linear combination of two other parts in it and its combination with the adaptive partitioning algorithm.

1. INTRODUCTION

The first fractal coding algorithm was suggested by Jacquin [1]. According to this algorithm, the image is divided into non-overlapping blocks covering the whole image. These blocks are denoted as *range-blocks*. A second partition of the same image, into larger blocks, is also performed. The collection of all the large blocks, known as *domain-blocks*, constructs a 'self-dictionary' (or codebook) called *domain-pool*. For each range-block, the domain-pool is searched for the best match under a predefined transformation (including spatial contraction) of the domain blocks. The fractal code representing the image is composed of the union of all transformation parameters (including a pointer to the matched domain-block location). Compression is achieved if the amount of information describing these parameters is less than the amount needed to describe the original image (at the cost of some distortion). The task of finding self-similarities (via the matching process) by a *full-search* of the domain pool is of high computational complexity and is considered to be the major drawback of fractal coding. The decoding procedure is typically done by iteratively applying the transformations to any initial image, until convergence is achieved. The decoding stage is less computational demanding than the coding stage. Yet, to save computations, we used a fast hierarchical decoding algorithm introduced in [2] and [4, Ch. 5], and extended in [3] for quadtree partitioning. This type of partitioning is the subject of the next section.

[1] This work was performed at the Signal and Image Processing Lab and was supported by the Ministry of Science – Grant No. 8635196.

2. QUADTREE PARTITIONING CRITERIA

It is well known that coding efficiency can be improved by using adaptive image partitioning methods such as the well-known Quadtree partitioning [3,4]. When a non-uniform partition is used, smooth areas are covered by few large blocks and active areas, containing many details (e.g., texture and edges), are covered by small blocks to better capture the image characteristics. According to the partition criterion presented in [3,4], here denoted as the *threshold-criterion* (T-C), each range-block is divided into four non-overlapping sub-blocks if its collage error (matching distortion) is above a predefined threshold. A good model to determine the threshold is not known since it is image dependent. In addition, the T-C doesn't enable direct control of rate or computational complexity.

In this paper, we aim at finding criteria for adaptive image partitioning that reduce the reconstruction errors and the (computational) complexity for a given rate. Yet, without the need to set an image-dependent parameter - such as a threshold value in the T-C. An optimal, but not practical, solution is to examine all possible quadtree structures, subject to the rate constraint, and to choose the one resulting in minimum distortion. Instead, we adopted a *rate-distortion* based partitioning criterion (denoted R-D), which is described next.

2.1. A rate-distortion based quadtree-partitioning criterion

The adopted R-D criterion gives priority to blocks that their partitioning would result in the highest *reduction in distortion per each bit added* to the representation of their sub-blocks [5]. This sub-optimal solution results in smaller reconstruction errors than obtained by other reported criteria (see Fig. 1). However, the need to find the matching blocks to the sub-blocks, before a splitting decision could be made, leads to a higher computational complexity than needed by the T-C. This disadvantage is avoided by using what we call the *collage-error – computational-complexity* criterion (denoted C-C), presented next.

2.2. A computational-complexity based partitioning criterion

We first propose a quadtree-partitioning criterion that reduces the matching distortion under a complexity constraint. This new criterion, denoted above as C-C, differs from those proposed in other works by taking into account the computational complexity of performing the coding process.

The process begins with a uniform partition of the image into large blocks. For each block, a gain value is computed as the expected reduction in matching distortion (collage-error), when a block is split into sub-blocks, divided by the computational complexity for finding the best matching blocks to its sub-blocks. A list of blocks ordered in descending gain values is produced. At each stage, the block at the top of the list is partitioned. For each new sub-block created, a new gain value is calculated and placed in the proper location in the descending list. The partitioning process continues until a computational complexity constraint is met. Although, this algorithm achieves a lower distortion, for a given complexity, (see below) it doesn't provide yet a direct control of the rate.

2.3. Performance comparison

Fig. 1 presents a comparison of reconstruction errors and computational complexity obtained with the above-mentioned three partitioning criteria (T-C, R-D and C-C) for the 512x512 image "Lena" of 256 gray-levels. It can be seen in Fig. 1(a) that the T-C results in higher reconstruction errors (lower PSNR), as compared to the R-D and C-C partitioning criteria. In addition, it can be seen that the PSNR values obtained with the C-C criterion are very close to those obtained with the R-D criterion.

In terms of the computational complexity required by the different algorithms, Fig. 1(b) shows that C-C is superior to R-D, as does T-C. Yet, C-C achieves a lower distortion than T-C, close to that of R-D - but at a much lower complexity.

The T-C criterion differs from the R-D and C-C criteria by the way the splitting decision is made. When using the T-C, an independent local decision is made for each block. While when using either R-D or C-C, the whole partition structure is actually taken into consideration at each splitting decision. That is, the whole image characteristics are taken into account while allocating coding resources (complexity and bits).

Many fractal-coding schemes use various block classification methods in order to reduce the complexity of the matching process [4]. Since block classification is performed before the coding stage, the above-mentioned criteria can be easily combined with any block classification method.

3. ADAPTIVE PARTITIONING WITH RATE CONTROL

We propose now an adaptive partitioning algorithm, under both complexity and rate constraints, which combines the last two criteria (R-D and C-C) and provides rate control. This algorithm applies two top-down passes.

In the first pass the procedure described for the C-C partitoning criterion is used. This determines a quadtree structure, as depicted in Fig. 2(a), which meets the given complexity constraint. This quadtree structure dictates an upper limit to achievable rates. At the end of this top-down pass, each non-terminal node (corresponding to a block) is assigned a R-D gain value, which is the matching-distortion reduction per each bit added if that block is partitioned into sub-blocks (see section 2.1). The additional computational complexity for obtaining these gain values is neglible since the main computation effort of block matching has already been done.

In the second pass, the R-D gain values, associated with the non-terminal nodes of the given quadtree, are used (as described in section 2.1) to determine a sub-tree (Fig. 2 (b)) that achieves

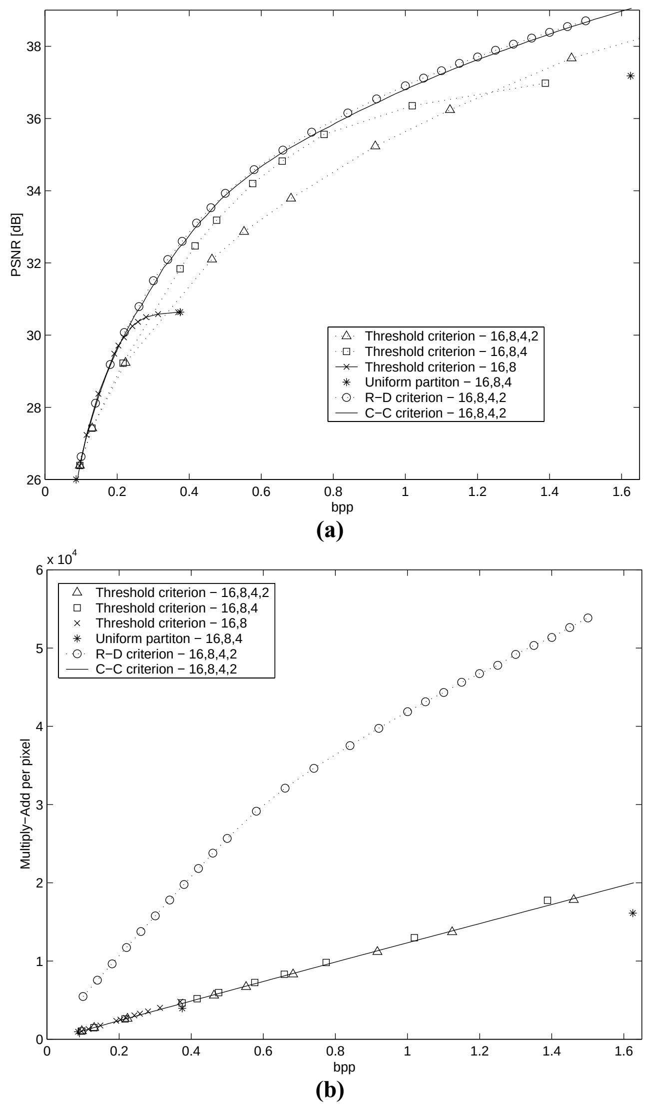

Figure 1: Results for the gray-level, 512x512, image "Lena". The numbers 16,8,4 and 2 indicate the sizes of range blocks.
(a) Reconstruction errors. (b) Computational complexity.

the desired rate. Again, since the gain values are given, the additional computational complexity involved is negligible. If a rate higher than the limit dictated by the first pass is desired, the computational-complexity constraint must be changed or violated (by adding nodes to the tree).

3.1. Experimental results

The above algorithm was applied to the above-mentioned image - "Lena". The range blocks used were of a maximum size of 16x16 and a minimum size of 2x2. The reconstruction errors obtained by the proposed 2-pass algorithm are in between those achieved by applying the R-D and C-C based partitions separately. At the same time, the computational complexity is maintained as low as was obtained by the C-C criterion. Since, to begin with, the C-C criterion resulted in a PSNR value that is very close to that obtained with the R-D criterion (see Fig. 1(a)), there is no point in showing here a graph of the coding results.

The reconstructed images shown in Fig. 3(a-c) were obtained by using the three discussed partition criteria (T-C, R-D and C-C). The coding rate is 0.3bpp with range blocks of sizes 32x32 to 2x2. It is worth mentioning again that the PSNR value of the reconstructed image, obtained by the 2-pass algorithm, is in between the very close PSNR values obtained with R-D and C-C, and hence is not shown.

As seen in Fig. 3 (at proper resolution), the reconstructed image related to the threshold criterion suffers from blockiness (at 0.3bpp), as opposed to the other two. Also, it is hard to make a distinction between the R-D and C-C related images (about 0.18dB difference in PSNR). However, the computational complexity of the R-D partitioning is about four times the complexity of both the C-C and the T-C partitioning approaches.

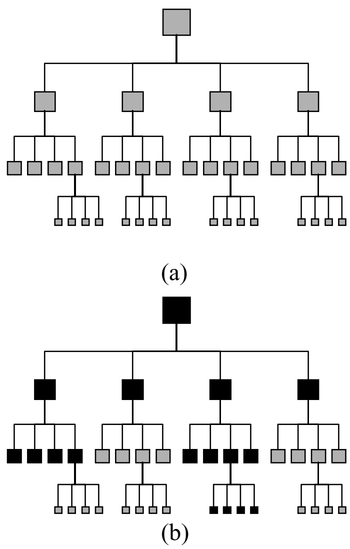

Figure 2: Adaptive image partitioning under complexity and rate constraints. Each node represents a block in the image. (a) A quadtree determined by applying the C-C partitioning criterion - until a complexity constraint is met. (b) The nodes in black represent a sub-quadtree of the above quadtree, obtained by applying the R-D criterion until a designated rate is achieved.

4. DIFFERENTIAL MATCHING PURSUIT

Improved range-block matching is feasible if the number of domain blocks used for the approximation is increased. Considering the tradeoff of increasing the number of domain blocks (enlarged computational load and rate) and the reduction in distortion, we've found that using two domain blocks is usually practical. Such an approximation is given in (1):

$$\min_{S_1,J_1,S_2,J_2} \quad \|E\|_2^2 = \left\|\tilde{R} - S_1\tilde{D}_{J_1} - S_2\tilde{D}_{J_2}\right\|_2^2 \qquad (1)$$

Where, R denotes the range-block to be matched; S_1 and S_2 are scale coefficients; D_{J1} and D_{J2} denote domain-blocks used to match the range-block R, with the indices J_1 and J_2 indicating their location in the domain-pool. The ~ sign indicates the subtraction of the mean value (DC) from the corresponding block. Finally, E denotes the collage error.

An optimal solution can be found by examining all possible domain block pairs, where for each pair the scale coefficients need to be computed for minimizing the collage error. Assuming a uniform partition into domain and range blocks, let N be the number of range-blocks, which is also typically the number of domain-blocks considered. Then, the computational complexity required for such a full search is of $O(N^3)$ blocks queries - as opposed to $O(N^2)$ blocks queries, if a single domain-block is used. Even for modest values of N this approach is not practical.

Mallat [6] suggested a sub-optimal approach called *Matching-Pursuit*, for finding a linear combination of several signals, taken from a large (over-complete) dictionary of signals, in order to represent an input signal. According to this approach, the error derived by projecting the input signal on a selected dictionary signal is repeatedly projected on dictionary signals until a desired linear combination is constructed. After each projection stage, the scale coefficients are either frozen or re-optimized.

We propose a similar approach for estimating a range-block using two domain blocks, but in a way which is better suited to our problem. First, the well-known operation of finding the best match for a range block is performed according to (2). That is, finding a scale coefficient and a domain block such that the collage-error is minimized.

$$\min_{S_1,J_1} \quad \|E_1\|_2^2 = \left\|\tilde{R} - S_1\tilde{D}_{J_1}\right\|_2^2 \qquad (2)$$

Then, given the domain block D_{J1} and the scale coefficient S_1, a better matching is sought by finding D_{J2} and S_2 such that the "updated" collage-error in (3) is minimized.

$$\arg \min_{S_2,J_2|S_1,J_1} \quad \|E_2\|_2^2 = $$
$$= \left\|\tilde{R} - S_1\tilde{D}_{J_1} - S_2\left(\tilde{D}_{J_2} - S_1\tilde{D}_{J_1}\right)\right\|_2^2 =$$
$$= \left\|E_1 - S_2\left(\tilde{D}_{J_2} - S_1\tilde{D}_{J_1}\right)\right\|_2^2 \qquad (3)$$

The use of (3), instead of (1), is motivated by the observation that lower frequencies are dominant in typical images, while the collage error image is characterized by dominant higher frequencies. Thus, trying to directly use domain blocks from the original image, to approximate a collage-error range-block, is less effective than using the difference between two domain blocks (with proper scaling), as done in (3). This is because this difference has much closer spectral characteristic to the collage-error, E_1, as it is also generated by differencing scaled blocks. Another approach would be to enlarge the domain pool by high-pass filtering the blocks in the given pool. This approach entails additional operations and an increase in rate and is not well suited for the fast hierarchical decoding [2] that we used in our work. An additional interesting advantage of using (3) – which we call *Differential Matching Pursuit* (DMP), is that by comparing (3) with (1) we see that after S_2 is computed (for given S_1 and J_1), it actually also modifies the eventual scale factor of D_{J1}, without adding a specific re-optimization stage, thus saving computations.

(a) PSNR 30.39dB (b) PSNR 31.54dB (c) PSNR 31.36dB (d) PSNR 31.85dB

Figure 3. Reconstructed of 0.3bpp coded "Lena" images using range blocks of sizes 32x32 to 2x2. (a) Threshold based partitioning criterion. (b) R-D based partitioning criterion. (c) C-C based partitioning criterion. (d) R-D criterion combined with DMP.

If a full search is applied in each stage of this method, and 2N memory units are used to store certain block inner-products values, to avoid their recalculation [7], then the computational complexity of the DMP is still of $O(N^2)$ block queries.

4.1. R-D partitioning combined with DMP

Finally, we combine the rate-distortion partitioning criterion with the differential matching-pursuit. For this purpose, three gain ratios are defined, each describing the reduction in distortion per each bit added to the representation of a range-block, depending on the type of matching used:

G_{QT} – The reduction in the distortion per bit, if a block - originally matched by a single domain block - is split, and each of its sub-blocks is also matched by a single domain block.

G_{MP} – The above ratio, when the block is matched by a single domain block, but its sub-blocks are each matched by 2 domain blocks, using the DMP.

G_{SMP} – The above ratio, but when the block is matched by 2 domain blocks (DMP), yet each of its sub-blocks is matched by a single domain block only.

It should be noted that the fourth possibility - a block is matched by 2 domain blocks and then each of its sub-blocks is also matched by 2 domain blocks, needs not to be separately included in the following algorithm, which combines the R-D partitioning criterion with the DMP:

1. Start with a uniform partition into large blocks.
2. For each block R, find the gains G_{MP} and G_{QT} and define the current range-block gain to be: $G=\max\{G_{MP}, G_{QT}\}$.
3. Produce a list of blocks ordered in descending gain values.
4. Stop the partitioning process if the rate constraint is met.
5. If the gain G of the block at the top of the list was chosen to be either G_{QT} or G_{SMP} (see step 6) then that block is partitioned, thus replacing the range block by its sub-blocks. Return to step 2.
6. Otherwise, the block is matched by 2 domain blocks ($G= G_{MP}$), set the new G value to be: $G= G_{SMP}$ (split this block next time it is encountered) and return to step 3.

As demonstrated by Fig. 3(d), this algorithm yields a lower reconstruction error than the other algorithms mentioned in the previous sections. Combining the 2-pass algorithm (C-C followed by R-D) with DMP can be done in the same manner.

5. CONCLUSION

New partitioning criteria were presented allowing consideration of the computational complexity during the coding process. An adaptive partitioning algorithm was proposed having several benefits. It results in low reconstruction errors, close to those obtained by applying a *rate-distortion* based criterion, while reducing the computational complexity (by a factor of four in our examples). It also enables direct control of both rate and complexity, as opposed to other reported fractal image coding algorithms. Also, it is simple to implement and can be combined with block classification methods [4] to further reduce the computational complexity. Finally, a new approach for matching-pursuit under rate and/or complexity constraints was presented, yielding improved performance.

6. REFERENCES

[1] A.E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," IEEE Trans. Image Processing, Vol. 1, No 1, pp. 18-30, January 1992.

[2] Z. Baharav, D. Malah and E. Karnin, "Hierarchical interpretation of fractal image coding and its application to fast decoding," International Conference of Digital Signal Processing, Cyprus, pp. 190-195, 1993.

[3] I. Sutskover and D. Malah, "Hierarchical fast decoding of fractal image representation using quadtree partitioning," IPA'99, Image Processing and its Applications, IEE, pp. 581-584, 1999.

[4] Yuval Fisher, "Fractal Image Compression Theory and Application," Springer Verlag, New York, 1995.

[5] I. Messing and D. Malah, "A list directed approach to fractal image coding in the wavelet transform domain," Proc. EUSIPCO, Tampere, Finland, September 2000.

[6] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," IEEE Trans. Signal Processing, Vol. 41, No 12, pp. 3397-3415, Dec. 1993.

[7] R. Franco, "Rate, Distortion, and Complexity Tradeoffs in Fractal Image Coding", M.Sc. Research Thesis (in Hebrew), Technion – Israel Inst. of Technology, Haifa, Israel, Oct. 1999.