Fractal image coding using rate-distortion optimized matching pursuit

Mohammad Gharavi-Alkhansari
Thomas S. Huang

Beckman Institute and Coordinated Science Laboratory
University of Illinois at Urbana-Champaign, Urbana, Illinois, U.S.A. 61801

ABSTRACT

Matching pursuit is a general and flexible method for solving an optimization problem that is of interest in signal analysis, coding, control theory and statistics. In this paper, principles of rate-distortion optimal coding are used in combination with matching pursuit algorithm to obtain an enhanced fractal image coding method. The advantages of using such a method over traditional fractal image coders are described and compression results are presented.

keywords: fractal, image coding, matching pursuit, rate-distortion optimal coding.

1 INTRODUCTION

In recent years, fractal image coders have shown promising results in compression of still images and image sequences. These methods have especially shown good coding performances when combined with other compression methods.[1,2] The combination of fractal techniques and more traditional coding methods has resulted in some generalized methods that take advantage of different types of redundancies in images. However, for optimal performance, these generalized methods require more powerful optimization algorithms than are usually required in other image coding methods.

On the other hand, matching pursuit is a new, general and flexible method for solving an optimization problem that is of interest in signal analysis, coding, control theory and statistics. It is a greedy method for obtaining a near optimal solution to a seemingly untractable optimization problem.

In this paper, we will see how the problem that matching pursuit addresses is the same problem that needs to be solved in generalized coding methods like those that have fractal coding methods as special cases. We will then apply an enhanced version of matching pursuit for solving this problem.

In this section, we give a review of the basic concepts of fractal image coding and the matching pursuit methods. In Section 2, we show why matching pursuit fits well to solving the problem that is proposed by generalized coding methods and how it may be implemented. Experimental results are brought in Section 3.

Finally, conclusions are drawn in Section 4.

1.1 Fractal image coding

Fractal image coding is a coding method that takes advantage of redundancies in an image at different scales. The first automatic fractal image compression algorithm was proposed by Jacquin.[3] In this method, an image is first partitioned into nonoverlapping blocks called range blocks. Each range block is then approximated by a linear combination of a fixed constant (dc) block and a transformed version of a single larger block, called a domain block, taken from the same image. The transformation is a contraction, usually formed by a combination of lowpass filtering, subsampling, and rotation of blocks. The selection is usually made by making a dictionary of transformed domain blocks (which we call adaptive blocks), and for each range block selecting the dictionary block that gives the best match to the non-dc component of the range block.

This method was later extended so that it approximated each range block with a linear combination of a fixed number of fixed blocks, and one single block from the dictionary.[4,5] In a generalized coding framework, this method was further generalized[6] to approximate each range block by a linear combination of any arbitrary number of fixed and adaptive blocks. This is done by including the fixed blocks in the dictionary and allowing the coding process to automatically select as many members of the dictionary, and any combination of fixed and adaptive blocks as needed. In the context of fractal coding, the dictionary is usually redundant and much larger than the dimension of the vector being coded (the range block). This brings up the problem of optimal selection of dictionary blocks.

1.2 Matching pursuit

Approximation of members of a vector space by a linear combination of a small number of members of a possibly large set (dictionary) of redundant vectors in that space has been of interest in different areas of science. More specifically, one may be interested in finding the smallest number of vectors in the dictionary whose linear combination approximates the given vector within a given error threshold. In the general case, this is a rather difficult optimization problem. A similar and equally difficult problem is that, if a positive integer M is given, find the M vectors in the dictionary whose linear combination can best approximate a given vector in the vector space.

These problems are difficult combinatorial optimization problems. In fact it has recently been proven that, in the general case, finding the optimal solution is NP-hard.[7,8] However, an efficient suboptimal greedy solution to this problem has been discovered by different researchers in different contexts but with basically the same underlying mathematics.

In statistics, this greedy algorithm was found and named *projection pursuit*.[9] It was used for computation of conditional expectation of random variables. In control theory, such method was developed for non-linear system identification.[10] In the context of time-frequency decomposition, it was named *matching pursuit*[11] and was used in signal analysis for extraction of patterns from noisy signals. In the context of image coding, it was developed for a generalized image coding method unifying transform coding, vector quantization, and fractal coding.[6] In this paper, we refer to this method as matching pursuit.

Matching pursuit has recently been adopted for video coding. Vetterli and Kalker used a rate-distortion optimized version of matching pursuit for motion compensated video coding.[12] Neff and Zakhor used matching pursuit for coding motion residual images of video sequences.[13,14]

The essence of matching pursuit is that, for a given vector $\vec{v}$ to be approximated, first choose the vector from the dictionary which has the strongest correlation coefficient (highest absolute value) with $\vec{v}$. Then, remove any component of its form from $\vec{v}$, i.e., orthogonalize $\vec{v}$ with respect to the selected dictionary vector, and obtain the residual of $\vec{v}$. The selected dictionary vector is in fact the one that results in the residual of $\vec{v}$ with the smallest energy. Repeat this process for the residual of $\vec{v}$ with the rest of dictionary vectors until the residual becomes smaller than a threshold or until no other dictionary vector has significant correlation with the residual.

1.2.1 Orthogonal matching pursuit

In matching pursuit, after a vector in the dictionary is selected, one may remove any component of its form not only from $\vec{v}$, but also from all other dictionary vectors before repeating the process. This version of the method is called *orthogonal matching pursuit* and is computationally more expensive than the nonorthogonal version, but typically gives significantly better results in the context of coding. However, for example, if all the dictionary vectors are orthogonal, the results for both the orthogonal and the non-orthogonal matching pursuit are the same.

1.2.2 Rate-distortion optimized matching pursuit

The standard matching pursuit, or its orthogonal version, tries to find the smallest number of vectors in the dictionary that can approximate a given vector within a given error threshold. In the context of coding, after the selection process is done, the coefficients of the these dictionary vectors need to be quantized, and also entropy coded along with their indices and the number of selections made. Different dictionary vectors have different costs in terms of bit rate, depending on how frequently they have been used. Hence, a better performance is expected if instead of selecting the smallest number of blocks from the dictionary, one selects blocks that need the shortest code collectively.

More specifically, we would like to approximate each range block $\vec{v}_j, j = 0, \ldots, J$ by vectors selected from a dictionary of vectors. After this selection is made, each of the coefficients of the selected dictionary blocks are quantized and the indices and the quantized coefficients are entropy coded. The goal is to do this using the shortest possible code.

Regarding the selection process in the matching pursuit, we note that for each vector $\vec{v}_j$,

1. Selection of each dictionary vector is based on how much it reduces the energy (distortion) of $\vec{v}_j$.
2. The selection process stops when the distortion of the $\vec{v}_j$ goes below a threshold.

Therefore, the *selection criterion* and the *stopping criterion* are both based on distortion of the residual of $\vec{v}_j$.

Selection Criterion: Due to the entropy coding stage, the number of bits required to encode $\vec{v}_j$ is not exactly proportional to the number of vectors used for coding it. In other words, the number of bits required for representing an index or a quantized coefficient depends on the the frequency of selection of the dictionary vector or the quantized coefficient. Therefore the best dictionary vector to be selected at each stage is not the one which gives the greatest reduction in distortion, but the one which gives the greatest $-\delta D_j/\delta R_j$ where δD_j is the change in the energy of the residual (distortion) of $\vec{v}_j$, and δR_j is the number of bits spent on coding the index and the coefficient of the selected vector. So, rate-distortion optimized matching pursuit uses this improved selection criterion.

Stopping Criterion: Using the selection process described above, the encoder tries to encode all the range blocks in the image with approximately the same distortion energy ε^2, using minimum number of dictionary vectors or bits. However, this is not optimal because by doing this, some $\vec{v}_j$'s select many dictionary vectors, each of them contributing slightly to reduction of the error energy. Many of the bits used for coding these vectors may be used to make larger reduction in the error energy of the whole image if they are used for coding other vectors.

Let's denote the total distortion and rate of the whole image by D and R respectively, i.e.,

$$D = \sum_{j=1}^{J} D_j, \text{ and } R = \sum_{j=1}^{J} R_j.$$

For a given R, we would like to minimize D. Given the above assumptions, the coding problem is that of minimizing $D = \sum_{j=1}^{J} D_j(R_j)$ subject to the constraint $R = \sum_{j=1}^{J} R_j$. This constrained minimization problem may be solved using Lagrange multipliers, giving the solution

$$\frac{\delta D_1}{\delta R_1} = \frac{\delta D_2}{\delta R_2} = \cdots = \frac{\delta D_J}{\delta R_J}.$$

This result suggests that an optimal solution is obtained when $\delta D_j / \delta R_j$, rather than D_j, is kept the same for all $\vec{v}_j$s. This modifies the stopping criterion for the selection process. Therefore, in rate-distortion optimized matching pursuit, the selection process stops when the algebraic value of $-\delta D_j / \delta R_j$ becomes smaller than a threshold.

2 THE PROPOSED APPROACH

In the generalized fractal coders,[6,1] the range blocks in an image are approximated by a linear combination of a set of fixed blocks and one or more adaptive blocks made by applying some simple transformations on larger blocks taken from the same image. The set of fixed and adaptive blocks construct a dictionary for each range block. This dictionary is typically overcomplete and the problem of making the near optimum selection among the dictionary blocks is an important part of these fractal coders.

In this paper, we propose using rate-distortion optimized matching pursuit for solving the optimization problem of such coders. In this section we describe a generalized image coder using fractal techniques that uses rate-distortion optimized matching pursuit for its selection process.

The image is first partitioned into range blocks $\vec{v}_j$, $j = 1, 2, \ldots, J$. Each range blocks is treated as an N-dimensional vector, where N is the number of pixels in the range block. For each range block, a dictionary of blocks U_j is made. This dictionary is made up of two disjoint subsets F and A_j. F contains fixed dictionary blocks $\vec{f}_i$, $i = 1, 2, \ldots, N_F$, and is independent of range block. A_j contains adaptive dictionary blocks $\vec{a}_{j,i}$, $i = 1, 2, \ldots, N_{A_j}$, and is made by applying some transformations on some blocks taken from the same image, which are typically located near the range block. For simplicity, we may denote the members of U_j by $\vec{u}_{j,i}$, $i = 1, 2, \ldots, N_{U_j}$, i.e.,

$$U_j = \{\vec{u}_{j,1}, \vec{u}_{j,2}, \ldots, \vec{u}_{j,N_{U_j}}\} = \{\vec{f}_{j,1}, \vec{f}_{j,2}, \ldots, \vec{f}_{j,N_F}, \vec{a}_{j,1}, \vec{a}_{j,2}, \ldots, \vec{a}_{j,N_{A_j}}\},$$
$$N_{U_j} = N_F + N_{A_j}.$$

N_{U_j} may be less than, equal to, or greater than N, although it is typically larger. Members of A_j may be generated from different sources and are all adaptive, i.e., in contrast to members of F which are independent of j, they may be different from one $\vec{v}_j$ to another. Two sources of A_j that are used for the experimental results of this paper are *higher-scale* and *same-scale* adaptive dictionary blocks.

Higher-scale dictionary blocks are taken from the neighborhood of the range block in the filtered, subsampled image. These blocks are introduced to let the encoder take advantage of the inter-scale redundancies in the image and make the fractal element of the coding. Same-scale dictionary blocks are taken from the causal neighborhood of the range block of the original image. These blocks are introduced to let the coder take advantage of the intra-scale redundancies in the image.[6]

A threshold r (independent of j) is set for $-\delta D_j/\delta R_j$ for the whole image. The rate-distortion optimized matching pursuit as described in Section 1.2.2 is used for selecting a series of blocks from the dictionary. For each $\vec{v}_j$, the selection process of this algorithm stops when $-\delta D_j/\delta R_j$ becomes less than r. The matching pursuit algorithm in fact detects the structure of the range block in terms of the members of the dictionary.

The coefficients of the orthogonalized selected dictionary blocks are quantized linearly and are entropy coded using a semi-adaptive entropy coder. The count of selections made for each index of the dictionary is individually monitored. The count and the index of the selected dictionary blocks are also entropy coded. The entropy coding of indices is based on the conditional probability of its selection as the k-th selected block.

The image is coded this way a few times iteratively, and each time, the statistics of the selected blocks of the previous time are used as the initial statistics for the entropy coding, until the statistics converge.

The resulting fractal coding method has the following advantages over most fractal coding methods:

- It typically gives better compression performance due to a better and less constrained method of selection of blocks from the dictionary.
- Because the dictionary provides an (over)complete set of vectors for approximation of each range block, this method provides a method for arbitrarily high PSNR, and even lossless image coding.
- The method performs relatively well even when the image blocks do not have a strong dc component, e.g., in residual images or some non-photographic 2D signals.
- The compression time is only moderately increased with the increase of PSNR of the decoded image.

3 EXPERIMENTAL RESULTS

In this section we present the coding results of the method discussed in the previous section. The rate-distortion optimized orthogonal matching pursuit was used for the selection of dictionary vectors. The 512×512, 8-bit per pixel, gray-scale Lena image was encoded using 8×8 range blocks. For each range block in the image, a different dictionary of blocks was made. The 64 DCT basis blocks were used as the fixed dictionary blocks. 128 causal same-scale, and 64 higher-scale blocks were used for the adaptive part of the dictionary. The higher-scale dictionary blocks were obtained from lowpass filtering followed by subsampling of 16×16 domain blocks[1]. No rotation, or reflection of domain blocks were used. It was found that the dc block of the DCT fixed basis blocks are automatically selected in nearly all codings of natural images. For better performance, the coefficient of the dc block was first predicted from the dc coefficients of the previous neighboring range blocks and then the difference between the predicted coefficient and the actual one was coded.

Figure 1(a) shows the original 512 Lena image. Figures 1(b), 1(c), and 1(d) show the decoded images at different bit rates.

[1]For more information on the details of constructing the dictionary, see.[6]

(a)

(b)

(c)

(d)

Figure 1: (a) Original 512×512, 8 bits per pixel image, (b) Decoded image at 0.43 bpp 34.5 dB PSNR, (c) Decoded image at 0.22 bpp 31.2 dB PSNR, (d) Decoded image at 0.15 bpp 29.2 dB PSNR.

4 CONCLUSIONS

In this paper, principles of rate-distortion optimal coding were used in combination with matching pursuit algorithm to obtain an enhanced fractal image coding method. The coding performance of this method is among the best performances of published fractal image coding methods and is close to some of the state-of-the-art non-fractal image coding methods like wavelet-based zerotree method.[15] In contrast to most published fractal image coders, the proposed method can approximate an image with arbitrarily small distortion and the compression time is only moderately increased with the increase of PSNR of the decoded image.

5 ACKNOWLEDGEMENTS

This work is supported in part by Joint Services Electronics Program Grant ONR n00014-96-1-0129 and in part by a grant from Mitsubishi Electric.

6 REFERENCES

[1] K. U. Barthel, J. Schüttemeyer, T. Voyé, and P. Noll, "A new image coding technique unifying fractal and transform coding," in *Proceedings of IEEE International Conference on Image Processing*, vol. 3, (Austin, Texas), pp. 112–116, Nov. 13–16, 1994.

[2] R. Rinaldo and G. Calvagno, "An image coding scheme using block prediction of the pyramid subband decomposition," in *Proceedings of IEEE International Conference on Image Processing*, (Austin, Texas), Nov. 13–16, 1994.

[3] A. E. Jacquin, "A novel fractal block-coding technique for digital images," in *Proceedings of IEEE ICASSP-90*, pp. 2225–2228, Apr. 3–6, 1990.

[4] G. E. Øien, S. Lepsøy, and T. A. Ramstad, "An inner product space approach to image coding by contractive transformations," in *Proceedings of IEEE ICASSP-91*, pp. 2773–2776, May 14–17, 1991.

[5] D. M. Monro, "A hybrid fractal transform," in *Proceedings of IEEE ICASSP-93*, vol. V, (Minneapolis, Minnesota), pp. 169–172, Apr. 27–30, 1993.

[6] M. Gharavi-Alkhansari and T. S. Huang, "Fractal-based techniques for a generalized image coding method," in *Proceedings of IEEE International Conference on Image Processing*, vol. 3, (Austin, Texas), pp. 122–126, Nov. 13–16, 1994.

[7] G. Davis, *Adaptive Nonlinear Approximations.* PhD thesis, New York University, Sept. 1994.

[8] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive nonlinear approxiamtions." submitted to Journal of Constructive Approximation, 1995.

[9] J. H. Friedman and W. Stuetzle, "Projection pursuit regression," *Journal of the American Statistical Association*, vol. 76, pp. 817–823, Dec. 1981.

[10] S. Chen, S. A. Billings, and W. Luo, "Orthogonal least squares methods and their application to non-linear system identification," *International Journal of Control*, vol. 50, no. 5, pp. 1873–1896, 1989.

[11] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, pp. 3397–3415, Dec. 1993.

[12] M. Vetterli and T. Kalker, "Matching pursuit for compression and application to motion compensated video coding," in *Proceedings of IEEE International Conference on Image Processing*, vol. 1, (Austin, Texas), pp. 725–729, Nov. 13–16, 1994.

[13] R. Neff and A. Zakhor, "Very low bit rate video coding using matching pursuits," in *Proceedings of the SPIE, Visual Communications and Image Processing I*, vol. 2308, (Chicago, Illinois), pp. 47–60, Sept. 25–28, 1994.

[14] R. Neff and A. Zakhor, "Matching pursuit video coding at very low bit rates," in *DCC'95: Data Compression Conference*, (Snowbird, Utah), Mar. 28–30, 1995.

[15] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Transactions on Signal Processing*, vol. 41, pp. 3445–3462, Dec. 1993.