VECTOR SET PARTITIONING WITH CLASSIFIED SUCCESSIVE REFINEMENT VQ FOR EMBEDDED WAVELET IMAGE CODING*

Debargha Mukherjee and Sanjit K. Mitra

Image Processing Laboratory
Department of Electrical & Computer Engineering, University of California, Santa Barbara, CA 93106.
Email: (debu,mitra)@iplab.ece.ucsb.edu

ABSTRACT

Set Partitioning in Hierarchical Trees (SPIHT), proposed by Said and Pearlman [1], is generally regarded as a very efficient wavelet-based still image compression scheme. The algorithm uses an efficient, joint scanning and bit-allocation mechanism for quantizing the scalar wavelet coefficients, and produces a perfectly embedded bitstream. This work extends set partitioning to scan vectors of wavelet coefficients, and use successive refinement VQ techniques such as multistage and tree-structured VQ, to quantize several wavelet coefficients at once. The new scheme is named VSPIHT (Vector SPIHT). Coding results are presented to demonstrate that the vector-based approach (without arithmetic coding) surpasses the scalar counterpart (also without arithmetic coding), in the mean-squared-error sense, for most images at low bitrates. The superiority of the vector-based approach is more pronounced for images that are generally regarded as difficult to code (such as Barbara) because of a large amount of detail.

1. INTRODUCTION

Recently, the wavelet transform has been shown to be very promising for efficient compression of natural images. The wavelet transform has the ability to decorrelate an image both in space and frequency, thereby distributing energy compactly into a few low frequency and a few high frequency coefficients. Once a hierarchical wavelet representation of an image is obtained, the wavelet coefficients are quantized as efficiently as possible. The efficiency of an image compression scheme depends both on the wavelet filters chosen, as well as on the coefficient quantization scheme employed. There has been plenty of research in recent years on both these aspects of wavelet based image coding [1-8]. This work focuses on the quantization issue, and not on the issue of designing efficient wavelet filters. Given a particular set of wavelet filters to use, the efficiency of a coefficient quantization scheme relies on the efficiency of specifying to the decoder which coefficients to quantize before which others, and of the corresponding bit allocation. Shapiro [5] introduced the Embedded Zerotree Wavelet (EZW) scheme where the *zerotree* enables efficient prediction of significance information of the wavelet coefficients. Following this work, Said and Pearlman [1] developed an alternate scheme, called set partitioning in hierarchical trees (SPIHT), which, though based on the same basic concepts, was far more effective in transmission of significance information to the decoder. Both the schemes used efficient scans to partially order the wavelet coefficients by magnitude, followed by progressive refinement. The transmission of ordering information is achieved by a subset partitioning approach that is duplicated at the decoder. The refinement is based on ordered bit plane transmission of the magnitudes of the coefficients previously ascertained as significant. The bitstream generated is perfectly embedded. Recently, Xiong *et al* [8] has developed a very efficient space-frequency quantization scheme that uses a rate-distortion criterion to jointly optimize zerotree quantization and scalar frequency quantization.

In this work, we attempt coding several coefficients at once by vectoring them. Earlier, da Silva et al. [6] grouped wavelet coefficients into vectors in the EZW [5] framework. The vectors were quantized with a gain-shape type successive approximation vector quantization, with the shape codebooks derived from multidimensional lattices. The embedding property was sacrificed in favor of increased coding efficiency achieved by separate joint arithmetic coding of significance information and VQ indices. In this work, we adopt the set-partitioning approach of Said and Pearlman to partially order vectors of wavelet coefficients by their vector magnitudes, and refine them successively using multistage or tree-structured VQ [9]. Trained VQ is used in order to exploit effectively the dependencies between the neighboring wavelet coefficients within the same subband. The set-partitioning framework preserves the embedding property, thereby making the scheme applicable to requirements like progressive transmission. In Section 2 the coding algorithm is explained in detail. In Section 3, we present the implementation details and our coding results, and compare them with the scalar SPIHT scheme and other algorithms. Section 4 concludes the paper.

2. VECTOR-BASED SPIHT

2.1 Coding Scheme

The SPIHT algorithm [1], although very efficient in transmission of ordering information, essentially involves a scalar quantization operation. As such, the residual correlation that exists between the neighboring coefficients in the same subband, especially in the lower subbands, are not exploited directly. The residual correlation is exploited indirectly in the arithmetic coded enhancement of SPIHT leading to a gain of about 0.3-0.6 dB over the non-arithmetic coded version. An alternative approach to quantizing wavelet coefficients, is to code several neighboring coefficients at once using Vector Quantization [9], rather than perform a scalar quantization of the individual coefficients. The efficient set-partitioning methodology, adapted for vectors, can be

* This work was supported by ONR grant N00014-95-1-1214.

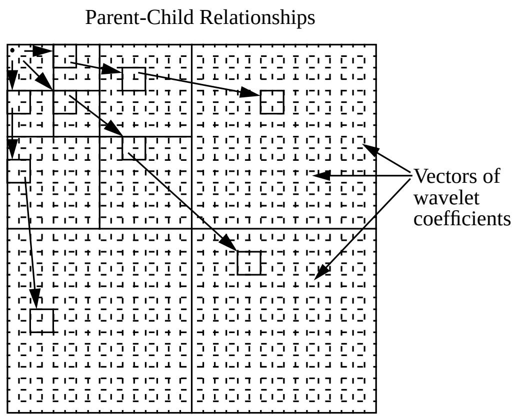

Figure 1. Parent-child relationships between vectors

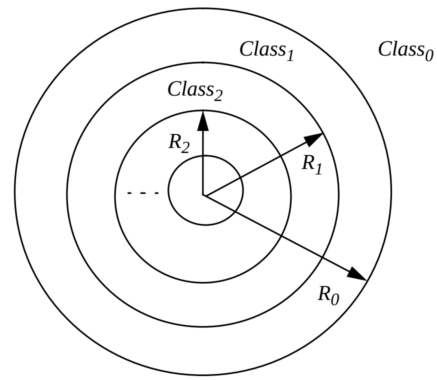

Figure 2. Decreasing magnitude thresholds to determine significance of vectors, and the corresponding classes.

used to produce an embedded bit stream. In this paper, we focus only on the basic non-arithmetic coded vector set partitioning, though we realize that the additional overheads of arithmetic coding on the vector significance bits will further improve the coding results.

In the vector-based approach, wavelet transform coefficients in each $H \times V$ window in each subband are grouped as a single vector of dimension HV. The parent child relationship between the vectors in different subbands is defined as for scalars in [1], and shown in Figure 1. Here each small square represents a HV-dimensional vector of wavelet coefficients. The set-partitioning methodology, with three ordered lists, is then used to achieve a partial ordering of vectors by vector-magnitude. Vector set-partitioning operates in multiple passes, where each pass is associated with a vector magnitude threshold. Each new pass yields a new set of vectors which have magnitudes higher than the threshold associated with the pass. The threshold progressively decreases from one pass to the next. In other words, each pass ascertains as significant the set of vectors that lie within a HV-dimensional shell, bounded on the inside by a hypersphere of radius equal to the current threshold, and on the outside by a hypersphere of radius equal to the previous threshold. The only exception is the first pass, which considers as significant all vectors of magnitude larger than R_0. The R_i's in Figure 2 correspond to the decreasing magnitude thresholds for determining significance of vectors. The progressive refinement of vectors already decided as significant in previous passes is achieved by classified successive refinement VQ schemes such as multistage or tree-structured VQ, where the class is determined by the pass in which a vector becomes significant (see Figure 2).

Note that the use of the L_2-norm (magnitude) in determining significance of a vector in a pass is justified for orthogonal wavelets, because it follows from Parseval's relationship that the squared magnitude error in quantization of the vectors contribute additively to the reconstruction mean-squared-error. That is, a higher magnitude vector when transmitted losslessly, will reduce the reconstruction mean-squared error more than a lower magnitude coefficient, and therefore should be quantized before the other. For bi-orthogonal wavelets, this is not strictly true. However, under the assumption that bi-orthogonal wavelets are approximately orthogonal, the L_2-norm will still be the best criterion to use to determine the significance of a vector in a pass.

Such a vector-based approach has several advantages. First, it allows better exploitation of the spatial redundancies in the wavelet coefficients. For example, vectors in the lower bands (high coefficients) usually show a strong correlation and a larger spread along the $\{1, 1, \ldots\}$ axis, producing a roughly elliptical distribution with major axis along the same direction. Vector Quantization is better suited to exploit this correlation than scalar schemes. Second, since the number of elements to code are reduced by a factor equal to the dimensionality of a vector, less bits are expended in transmitting the ordering information by set-partitioning. A drawback however, is that for images in which the spatial correlation between the components of a vector is less, there is less to be gained by VQ as opposed to scalar quantization. In fact, with VQ, too many bits may be unnecessarily spent in quantizing vectors which have only one or two significant coefficients. For such images, the vector-based approach cannot be expected to be very effective.

2.2 Successive Refinement Classified VQ

The vectors decided as significant in a pass, are roughly quantized in the same pass, and are successively refined in the subsequent passes. The pass in which a vector becomes significant also classifies the vector, and determines the particular successive refinement VQ to use to quantize it. It is necessary to have as many VQs as are classes, to exploit the class distribution pattern effectively. Therefore, if N passes are used in all, N successive refinement VQs need to be designed, one for each class and its associated magnitude threshold. Note that for each class, the codevectors of the corresponding VQ span the shell between two hyperspheres, except for the first class whose codevectors span the outside of a hypersphere. In Figure 1, $Class_i$ refers to the class associated with a pass in which the magnitude threshold for significance is R_i.

We investigated two standard successive refinement schemes - Tree-Structured VQ (TSVQ) and Multistage VQ (MSVQ) [9]. While the most efficient scheme for successive refinement of vectors is Tree-Structured VQ, the storage requirements are usually very large. Multistage VQ strikes a good compromise between storage complexity and efficiency. Note that the larger VQ codebooks required in VSPIHT prevents it from generating a bitstream embedded to the level of a single bit, as in the scalar case [1][5]. However, there are hardly any applications that may require such fine granularity of embedding. In the next Section we present the implementation details, and the coding results of our algorithm.

3. IMPLEMENTATION AND RESULTS

In our implementation, we used a 5-stage wavelet decomposition of 512×512 images using the 9/7 bi-orthogonal wavelets in [4]. Coefficients in each 2×2 window are grouped to obtain vectors of dimension 4. This is just the right size to use before the VQ complexity becomes prohibitive. We designed 10 VQs for a maximum of 10 corresponding passes with the following thresholds: 3000, 1500, 700, 350, 225, 250, 150, 100, 64, 36, 18. A set of 30 images of size 512×512 are used as the training set to design the VQs. Each original sample vector is used to generate 2 training samples by taking its negative vector as well. For sparse high threshold classes, the components of a vector and its negative are further permuted to produce 24 sample vectors each. Such a permutation is justified by the isomorphism of 2×2 blocks.

For the Multistage VQ implementation, the bit allocation chosen is as shown in Table 1. It is evident from the bit allocation that not all the significant vectors are refined in all the refinement passes. The vectors determined as significant in previous passes are alternately refined in the refinement passes. The reason for choosing such a staggered bit allocation as opposed to a uniform bit-allocation is that a single stage VQ is more efficient than a two stage VQ using the same number of bits.

Table 1 also shows the bit allocation for the Tree-structured VQ implementation. In this case, the first few stages in each class VQ are tree-structured, whereafter the VQ switches to multistage. Note that a full tree-structured VQ design is impractical, because of the enormous number of codevectors required.

Table 1. Bit Allocation for MSVQ and TSVQ Implementations for various classes

Class	MSVQ Bit Allocation	TSVQ Bit Allocation
0	9,0,6,0,6,0,6,0,6,0	5,3,3,3,3,3,3,3,3,3
1	9,0,6,0,6,0,6,0,6	5,3,3,3,3,3,3,3,3
2	9,0,6,0,6,0,6,0	5,3,3,3,3,3,3,3
3	8,0,6,0,6,0,6,	6,3,3,3,3,3,3
4	8,0,6,0,6,0	6,3,3,3,3,3
5	8,0,6,0,6	6,4,3,3,3
6	8,0,6,0	6,4,4,3
7	8,2,4	6,4,4
8	7,2	6,4
9	6	5

We present the coding results upto 0.5 bits/pixel for two images with varying levels of coding difficulty. They are the Barbara image (see Figure 3), and the Goldhill image (see Figure 4). PSNR comparisons are made with results obtained in [1] and [5]

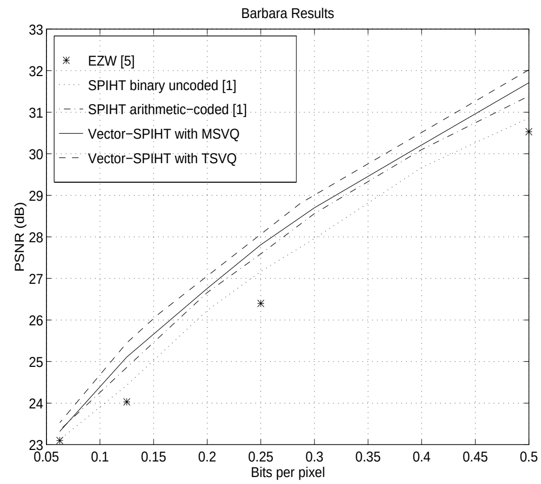

Figure 3. Coding results for the Barbara image.

to show the effectiveness of the VSPIHT algorithm. Our MSVQ-based VSPIHT algorithm surpasses the binary uncoded version of SPIHT for both images, and the arithmetic coded SPIHT, for the Barbara image. The TSVQ-based algorithm surpasses both. It is to be noted, however, that the TSVQ results and bit-allocations are still preliminary. A more rigorous TSVQ design is currently in progress.

Figure 5 presents the original 8 bits/pixel 512×512 Barbara image and four successively decreasing bit rate images at 0.5, 0.4, 0.3, 0.2, and 0.1 bits/pixel respectively, each coded by the MSVQ version of VSPIHT. The PSNR results are also provided here.

4. CONCLUSION AND FUTURE DIRECTIONS

We have introduced the VSPIHT algorithm for embedded image coding and have demonstrated the effectiveness of classified vector quantization in combination with the efficient set partitioning scheme, for wavelet based image compression. A more rigorous codebook design and bit allocation procedure will improve on these preliminary results. It will be worthwhile to investigate various ways of sub-classifying the vectors within the

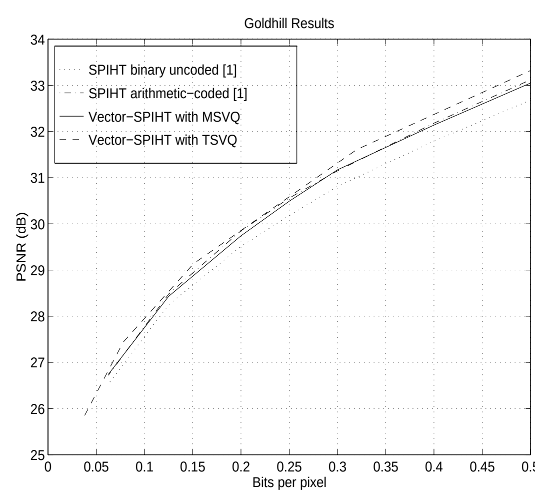

Figure 4. Coding results for the Goldhill image.

(a) Original Barbara at 8 bits/pixel

(b) VSPIHT coded Barbara at 0.5 bits/pixel (PSNR 31.70 dB)

(c) VSPIHT coded Barbara at 0.4 bits/pixel (PSNR 30.21 dB)

(d) VSPIHT coded Barbara at 0.3 bits/pixel (PSNR 28.73 dB)

(e) VSPIHT coded Barbara at 0.2 bits/pixel (PSNR 26.76 dB)

(f) VSPIHT coded Barbara at 0.1 bits/pixel (PSNR 24.36 dB)

Figure 5. The original Barbara image and 5 decreasing bitrate VSPIHT coded images using Multistage VQ at 0.5, 0.4, 0.3, 0.2, and 0.1 bits/pixel respectively, with the corresponding PSNR values.

principal classes obtained by magnitude ordering, and specializing successive approximation VQs for each subclass. Introduction of arithmetic coding in the set-partitioning bits as in [1] will also improve on these results. However, the gain by arithmetic coding in the vector case will be less than that in the scalar case, because the vector based approach already exploits a lot of the correlations in the wavelet coefficients.

The new method can be readily generalized to color images or to multispectral images in general, where the vector based approach naturally makes more sense than in gray images. By defining the vectors to take components evenly from all the spectral components, a jointly embedded bit stream can be obtained. Investigations on such color and multispectral image compression schemes based on VSPIHT are currently in progress.

5. REFERENCES

[1] A. Said and W. A. Pearlman, "A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 6, No. 3, June 1996.

[2] E. H. Adelson, E. Simoncelli, and R. Hingorani, "Orthogonal Pyramid Transforms for Image Coding", *Proceedings SPIE, Visual Communication and Image Processing II*, Cambridge, MA, Vol. 845, pp. 50-58, Oct 1987.

[3] R. A. DeVore, B. Jawerth and B. J. Lucier, "Image Compression through Wavelet Transform Coding," *IEEE TRansactions on Information Theory*, Vol. 38, pp. 719-746, March 1992.

[4] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image Coding using Wavelet Transform," *IEEE Transactions on Image Processing*, Vol. 1, pp. 205-220, April 1992.

[5] J. M. Shapiro, "Embedded Image Coding using Zerotrees of Wavelet Coefficients," *IEEE Transactions on Signal Processing*, Vol. 41, No. 12, Dec. 1993.

[6] E. A. B. da Silva, D. G. Sampson, M. Ghanbari, "A Successive Approximation Vector Quantizer for Wavelet Transform Image Coding," *IEEE Transactions on Image Processing*, vol. 5, No. 2, Feb. 1996.

[7] R. L. Joshi, V. J. Crump, and T. R. Fischer, "Image Subband Coding using Arithmetic and Trellis Coded Quantization," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 5. pp. 515-523, Dec. 1995.

[8] Z. Xiong, K. Ramachandran, and M. T. Orchard, "Space-Frequency Quantization for Wavelet Image Coding," *IEEE Transactions on Image Processing*, vol. 6, No. 5, May 1997.

[9] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Boston, MA: Kluwer, 1992.