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

*Debargha Mukherjee and Sanjit K. Mitra*

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

## ABSTRACT

*The Set Partitioning in Hierarchical Trees (SPIHT) approach for still image compression proposed by Said and Pearlman [8], is one of the most efficient embedded gray image compression schemes till date. The algorithm relies on a very efficient scanning cum bit-allocation scheme for quantizing the coefficients obtained by a wavelet decomposition of an image. In this paper, we adopt this scheme to scan vectors of wavelet coefficients, and use successive refinement VQ techniques with staggered bit-allocation to quantize several wavelet coefficients at once. The new scheme is named VSPIHT (Vector SPIHT). We present some coding results comparing VSPIHT to the scalar counterpart in the mean-squared-error sense. The method readily generalizes to color images and video where the vector-based approach makes more sense. We present the coding results on INTRA frames of QCIF sequences as compared against H.263.*

## 1. INTRODUCTION

Natural images can very well be characterized by a linear combination of energy concentrated in pockets in both space and frequency. The wavelet transform is very well suited for transform coding of such images because it decorrelates the image both in space and frequency, thereby compacting energy into a few low frequency and a few high frequency coefficients. Once a hierarchical wavelet representation of an image is obtained, the coding effort reduces to quantizing the coefficients as efficiently as possible. Recently there has been a plethora of research activity in wavelet based image coding [1-7]. The efficiency of a wavelet-based compression scheme relies on the efficiency of specifying to the decoder which coefficients to quantize before which others, and of the corresponding bit allocation. Shapiro [9] 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 [8] 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 relied on partial magnitude ordering of the wavelet coefficients, followed by progressive refinement, and produced embedded bitstreams. 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. Recently, Xiong *et al* [10] 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 order to code images and video at very low bitrates, in this work, we attempt coding several coefficients at once by vectoring them. We adopt the efficient set-partitioning approach of Said and Pearlman to partially order vectors of wavelet coefficients and refine them successively using multistage or tree-structured VQ [5]. Although use of a successive approximation lattice-based VQ has been proposed earlier on the EZW scheme by E. A. B. da Silva [4], our approach of using trained VQ with vector set-partitioning is more efficient and clearly surpasses their results. In Section 2 we discuss the coding algorithm in detail. In Section 3, we present the implementation details and coding results for gray scale images, and compare them with the scalar SPIHT scheme and other algorithms. Section 4 presents the implementation details and results for INTRA frames of QCIF video sequences, and compares them against DCT-based INTRA frame compression of H.263 [6]. Section 5 concludes the paper.

## 2. VECTOR-BASED SPIHT

### 2.1 Coding Methodology

The SPIHT algorithm [8], while very efficient in transmission of ordering information, essentially involves a scalar quantization operation. As such, the dependencies that exist between the neighboring coefficients are not exploited to the fullest extent. The arithmetic coding enhancement of SPIHT indirectly exploits this redundancy, to gain about 0.3-0.6 dB over the non-arithmetic coded version. An alternative approach to coding images, is to code several neighboring coefficients at once using VQ, rather than perform a scalar quantization of the individual coefficients. The efficient set-partitioning methodology, adapted for vectors, can be used to produce an embedded bit stream. Arithmetic coding over vector-based set partitioning can further improve the coding results. However, in this paper we will concentrate only on the basic non-arithmetic coded vector set partitioning.

In the vector-based approach, we group wavelet transform coefficients in each $H \times V$ window in each band as a single vector

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

of dimension $HV$. The parent child relationship between the vectors in different bands is defined as in [8]. We then use the set-partitioning methodology to achieve a partial ordering of vectors by vector-magnitude. Each new pass yields a new set of vectors which have magnitudes higher than a 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 thick 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 1 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 1).

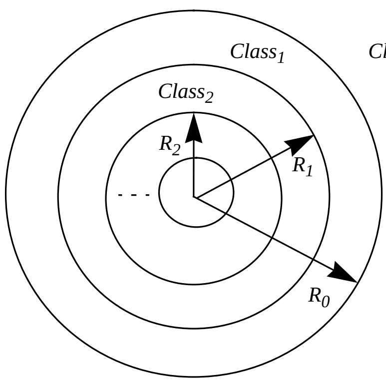


Figure 1. Decreasing Magnitude Thresholds to determine significance of vectors, and the corresponding classes.

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, assuming 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.

Such a vector-based approach has several advantages. First, it allows better exploitation of the spatial redundancies in the wavelet coefficients in the same subband. 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 correlation between the components of a vector is low, there is less to be gained by VQ when compared with 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 Classified Successive Refinement 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. 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 the pass in which the magnitude threshold for significance is $R_i$.

We investigated two standard trained successive refinement schemes - Tree-Structured VQ (TSVQ) and Multistage VQ (MSVQ)[5]. 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. In the next two Sections we present the implementation details, and the coding results of our algorithm.

## 3. GRAY IMAGES

### 3.1 Implementation Details

In our implementation for gray-scale images, we used a 5-stage wavelet decomposition of $512 \times 512$ images using the 9/7 bi-orthogonal wavelets in [2]. Coefficients in each $2 \times 2$ window are vectored to obtain vectors of dimension 4, which 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 $512 \times 512$ 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 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 $2 \times 2$ blocks. For the low-low band, the mean of the wavelet coefficients is finely quantized and transmitted to the decoder before set-partitioning scans commence.

For the MSVQ implementation, the bit allocation chosen is as shown in Table 1. It is evident from the bit allocation that we do not always refine all the significant vectors in all the refinement passes. The reason for choosing such a staggered 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 TSVQ implementation. Note however, that this TSVQ has not yet been designed very rigorously. A training set of 30 images is clearly insufficient for a tree-structured VQ design, where the total number of codevectors required are enormous.

**Table 1. Classes, thresholds and Bit Allocation for MSVQ and TSVQ Implementations for gray scale images.**

| Class | Threshold | MSVQ Bit Allocation | TSVQ Bit Allocation |
|---|---|---|---|
| 0 | 3000 | 9,0,6,0,6,0,6,0,6,0 | 5,3,3,3,3,3,3,3,3,3 |
| 1 | 1500 | 9,0,6,0,6,0,6,0,6 | 5,3,3,3,3,3,3,3,3 |
| 2 | 700 | 9,0,6,0,6,0,6,0 | 5,3,3,3,3,3,3,3 |
| 3 | 350 | 8,0,6,0,6,0,6, | 6,3,3,3,3,3,3 |
| 4 | 225 | 8,0,6,0,6,0 | 6,3,3,3,3,3 |
| 5 | 150 | 8,0,6,0,6 | 6,4,3,3,3 |
| 6 | 100 | 8,0,6,0 | 6,4,4,3 |
| 7 | 64 | 8,2,4 | 6,4,4 |
| 8 | 36 | 7,2 | 6,4 |
| 9 | 18 | 6 | 5 |

### 3.2 Coding Results

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 2), and the Goldhill image (see Figure 3). PSNR comparisons are made with [8] and [9] to show the effectiveness of the VSPIHT algorithm. Our MSVQ-based VSPIHT algorithm easily 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.

## 4. COLOR IMAGES

### 4.1 Implementation Details

The VSPIHT method can be readily generalized to color images, where the vector based approach naturally makes more sense than in gray images. For the ensuing discussion we restrict our attention to color images in the standard 4:2:0 format, where the chrominance components (Cb, Cr) are subsampled by a factor of 2 in the horizontal as well as the vertical direction. Although the color components are supposed to be considerably decorrelated in this representation there exist a significant amount of correlation between the luminance and the chrominance components. A conventional approach would apply the gray-scale algorithm to the three color components separately, and therefore waste bits in representation of the significance information. Moreover, bit-allocation among the luminance and the color components have to be treated separately, which poses difficulties in embedded coding of the color image as a whole. We attempt coding the color image as a whole by adopting the following approach. An equal number of stages of wavelet decomposition are applied to the luminance and the two chrominance components separately. For each $2 \times 2$ window in the luminance wavelet transform domain, the 4 luminance coefficients and the 2 chrominance coefficients (1 for each chrominance component) are grouped to obtain vectors of dimension 6. This 6-dimensional vector, as a single unit, now contains both the luminance as well as the color information at the corresponding space-frequency location in the image. Vector SPIHT now operates as in the gray image case, but with vectors of dimension 6 instead of 4.

In order to reduce the incidence of false colors, we use a weighted VQ, where the two chrominance coefficients in a vector are weighted 1.2 times as compared to the luminance coefficients.

We applied our VSPIHT scheme to code INTRA frames of $176 \times 144$ QCIF sequences. Three stages of decomposition are applied to the Y, Cb and Cr components, with the 9/7 bi-orthogonal wavelets in [2]. Eight classes of multistage codebooks are designed for the following thresholds: 1024, 512, 256, 128, 64, 32, 16, 5. The bit allocation chosen is shown in Table 1. Once again, a staggered bit-allocation is used to improve the efficiency of multistage VQ.

### 4.2 Coding Results

We compared the performance of VSPIHT for INTRA frame coding of standard QCIF sequences, with that obtained by DCT-based compression in the H.263 video coding standard. Figure 4 shows the luminance SNR vs. bitrate for a representative frame 100 of the Mother-Daughter QCIF sequence when compared against the

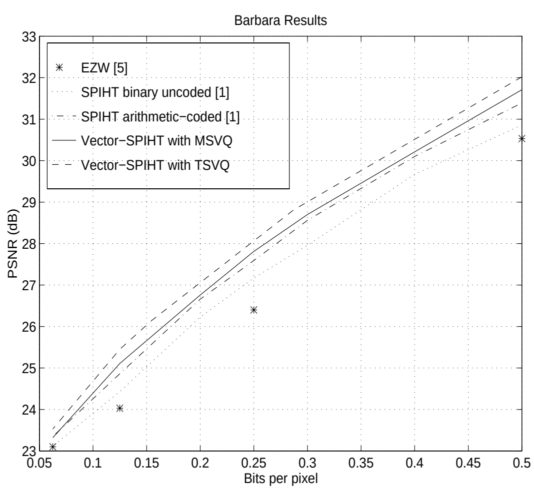


Figure 2. Coding results for the Barbara image.

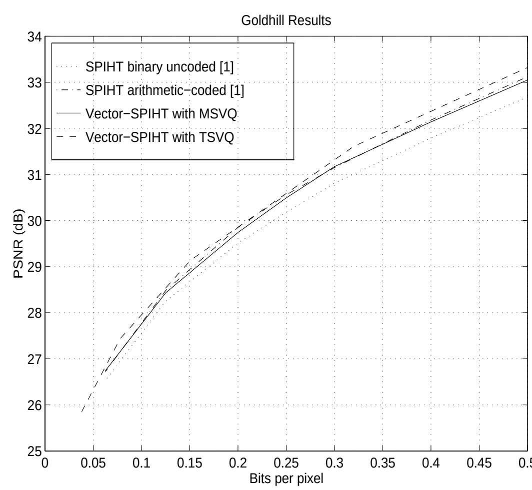


Figure 3. Coding results for the Goldhill image.

TMN5-2.0 implementation of H.263. Figure 5 shows the corresponding results for Frame 100 of the Hall_Monitor QCIF sequence. The gains are as much as 2dB. The chrominance results are similar. Besides obtaining a significantly higher SNR, VSPIHT can accurately control the number of bits spent in coding, unlike the DCT-based scheme.

**Table 2. Classes, thresholds and Bit Allocation for MSVQ Implementation for color images.**

| Class | Threshold | MSVQ Bit Allocation |
|---|---|---|
| 0 | 1024 | 10,0,7,0,7,0,7,0 |
| 1 | 512 | 11,0,8,0,7,0,7 |
| 2 | 256 | 11,0,8,0,7,0 |
| 3 | 128 | 11,0,8,0,7 |
| 4 | 64 | 11,0,8,0 |
| 5 | 32 | 11,0,7 |
| 6 | 16 | 11,0 |
| 7 | 5 | 10 |

## 5. CONCLUSION

We have introduced the VSPIHT for embedded image coding and have demonstrated the effectiveness of classified vector quantization in combination with the very efficient set partitioning scheme, for wavelet image and video compression. Subclassifications within each shell to better exploit the distribution pattern of the vectors within it can also improve on the coding results. For example, for each shell, different VQs can be designed for different subbands. Furthermore, use of arithmetic coding on the vector set-partitioning bits as in [8] will definitely improve the results, especially for images with large uniform regions. However, the gain in doing so will be less than that in the scalar case, because the vector based approach already exploits some of the correlations between adjacent wavelet coefficients.

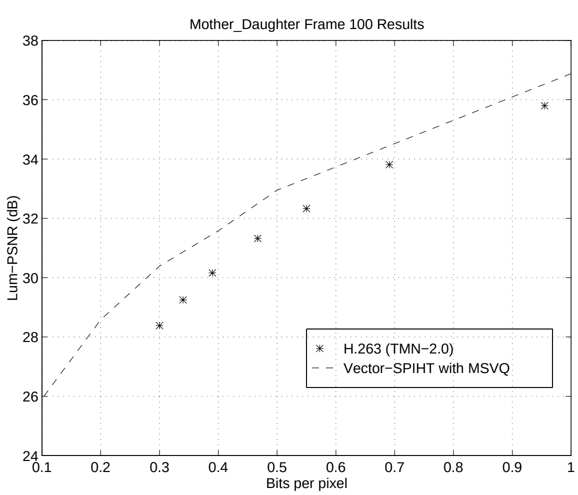


Figure 4. Coding results for Frame 100 of the Mother_Daughter QCIF sequence.

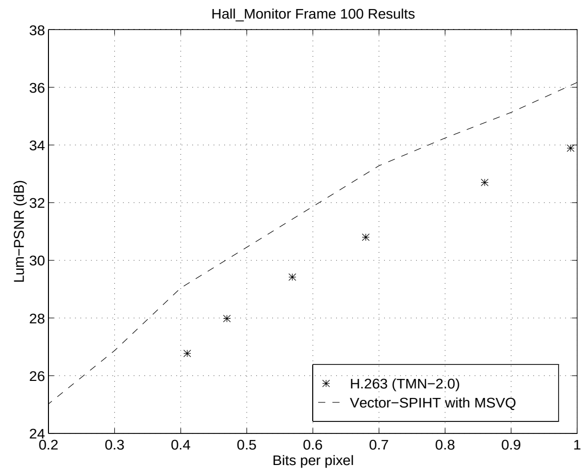


Figure 5. Coding results for Frame 100 of the Hall_Monitor QCIF sequence.

## 6. REFERENCES


[1] 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.

[2] 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.

[3] 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.

[4] 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.

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

[6] ITU-T Recommendation H.263, “Video coding for low bitrate communication”, 1995.

[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] 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.

[9] J. M. Shapiro, “Embedded Image Coding Using Zerotrees of Wavelet Coefficients”, *IEEE Transactions on Signal Processing*, Vol. 41, No. 12, Dec. 1993.

[10] 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.