Wavelet Compression Techniques for Hyperspectral Data

Bruce Evans, Brian Ringer and Mathew Yeates
Data Technologies Division
TRW Systems Integration Group
One Space Park, Bldg. R2 / 2170
Redondo Beach, CA 90278
cuber@spf.trw.com

Hyperspectral sensors are electro-optic sensors which typically operate in visible and near infrared bands. Their characteristic property is the ability to resolve a relatively large number (i.e., tens to hundreds) of contiguous spectral bands to produce a detailed profile of the electromagnetic spectrum. In contrast, multispectral sensors measure relatively few non-contiguous spectral bands. Like multispectral sensors, hyperspectral sensors are often also imaging sensors, measuring spectra over an array of spatial resolution cells. The data produced may thus be viewed as a three dimensional array of samples in which two dimensions correspond to spatial position and the third to wavelength.

Because they multiply the already large storage/transmission bandwidth requirements of conventional digital images, hyperspectral sensors generate formidable torrents of data. Their fine spectral resolution typically results in high redundancy in the spectral dimension, so that hyperspectral data sets are excellent candidates for compression. Although there have been a number of studies of compression algorithms for multispectral data [1,2,3,4], we are not aware of any published results for hyperspectral data.

In this paper we compare three algorithms for hyperspectral data compression. They were selected as representatives of three major approaches for extending conventional lossy image compression techniques to hyperspectral data. The simplest approach treats the data as an ensemble of images and compresses each image independently, ignoring the correlation between spectral bands. The second approach transforms the data to decorrelate the spectral bands, and then compresses the transformed data as a set of independent images. The third approach directly generalizes two-dimensional transform coding by applying a three-dimensional transform as part of the usual transform-quantize-entropy code procedure. The algorithms studied all use the discrete wavelet transform. In the first two cases, a wavelet transform coder (using the algorithm described in [5]) was used for the two-dimensional compression. The third case used a three dimensional extension of this same algorithm.

These algorithms were tested on several data sets obtained from the TRW imaging spectrometer (TRWIS). This sensor provides measurements from 90 uniform width spectral bands which cover a wavelength range from approximately 400 nm to 800 nm, and is mounted in a helicopter or small plane. Spectra are obtained simultaneously from a linear array of 256 spatial resolution cells. Platform motion is utilized to scan this array, thus obtaining spatial samples in a second spatial dimension. A typical TRWIS data set consists of a 90x256x450 array of one byte samples.

Although signal to noise ratio (SNR) and related mean square distortion metrics are convenient and widely used, their relevance to practical utility or perceptual quality is uncertain. This is of particular concern with respect to hyperspectral data, since the art of interpreting and utilizing this data is still developing. To supplement SNR measurements for the different algorithms, we also applied example pixel classification and image segmentation algorithms to the reconstructed data sets in order to assess the impact of compression losses on automatic data exploitation. These applications include pixel classification using a k-means algorithm and region based spectral image segmentation.

Our results showed substantial differences in the performance of the three algorithms. The spectral decorrelation algorithm produces the best results, but also requires the most computational effort. The three dimensional wavelet algorithm's performance came in second, but well ahead of the band independent algorithm. These results clearly demonstrate the importance of exploiting the spectral redundancy. Spectral decorrelation performs best because the transform is optimally matched to the data, whereas the wavelet transform is suboptimal but computationally

more efficient. Interestingly, individual spectral bands displayed as images often look better in the reconstructed data than the original image, particularly for the spectral decorrelation algorithm. This is because the compression process in effect filters out sensor noise from the original signal.

Acknowledgments. This research was supported by TRW's Ballistic Missile Division through the coordination of Lou Cassel and Joan Lurie. Hyperspectral data and computational facilities were provided by TRW's Remote Sensing Center, directed by Stokes Fishburne.

Band-independent Wavelet Compression. This algorithm was primarily studied as a reference point for measuring the gains due to inter band processing. One advantage is that individual bands can be reconstructed without having to decompress the entire data cube. This is useful if one knows in advance that only a few spectral bands will be reconstructed from the compressed data, but not specifically which bands.

The performance of this algorithm of course depends entirely on the algorithm used to compress the individual bands. We selected the wavelet transform coding algorithm in [5] because our previous studies had shown its performance to be superior to DCT and DPCM algorithms and comparable to other wavelet algorithms. This algorithm first computes the discrete wavelet transform of the image using the Mallat [9] recursion and a Daubechies 4-tap wavelet kernel [8]. The transform is then partitioned into a collection of rectangular blocks, and quantizer bit rates are optimally assigned to each block using the algorithm described in [6,7]. The quantized coefficients are Huffman coded, and the side data consisting of the bit rate allocations for each block is losslessly compressed using the UNIX "compress" utility.

Three Dimensional Wavelet Transform Compression. This algorithm is a straightforward generalization of the two dimensional algorithm described above. All the components of the two dimensional algorithm have obvious three dimensional analogs; the major difficulty is the more complex bookkeeping required to manage three dimensional data. Our implementation emphasized simplicity and flexibility over efficiency, relying instead on a powerful workstation (a Sun SPARC 10), plenty of memory, and patience. However, hyperspectral data sets are generally large (around 36MB in our examples) and the despite the algorithm's moderate complexity, processing can be time consuming. We expect that optimizing the implementation, particularly by improving memory management, would speed computation significantly, even on fast machines with large memories.

The three dimensional wavelet transform is constructed as a separable extension of the two dimensional transform, much as the two dimensional transform can be constructed by applying one dimensional wavelet filter banks over each dimension. Each stage of the separable three dimensional transform applies one dimensional filter banks successively across the two spatial dimensions and the spectral dimension. This decomposes the data into seven highpass channels and one lowpass channel. The seven highpass channels contain oriented edge information (in the two spatial directions, the spectral direction and the four diagonal combinations of these directions). Each channel contains one eighth of the original number of samples. Applying this operation recursively to the lowpass channel produces a series of nested octant decompositions.

We quantize the transform coefficients by partitioning each channel at each scale into three-dimensional sub-blocks. Within a sub-block, coefficients are quantized with the same number of bits per sample. Because large magnitude high pass coefficients tend to be sparsely distributed, many blocks can be quantized at low bit rates while introducing little distortion as a result. The actual bit allocation is determined using the algorithm described in [6,7]. This algorithm assumes that the mean square quantizer distortion is an exponential function of the bit rate times the sample variance of the data. It produces a bit allocation which minimizes the mean square quantization error subject to a constraint on the maximum average bit rate.

As in the two dimensional algorithm the quantized coefficients are Huffman coded. One difference is that three dimensional case uses a Lloyd-Max quantizer which is optimized for-each data set, and Huffman codes are determined based on the actual sample distributions for each bit rate. The two dimensional algorithm uses a fixed uniform quantizer and fixed Huffman codes (both optimized for Laplacian statistics). For large hyperspectral data sets, the additional side data needed to transmit the quantizer coefficients and Huffman code tables is relatively insignificant.

The side data also contains the quantizer bit rate allocations, and is compressed using the UNIX "compress" utility.

Band Decorrelation Wavelet Compression. The compression algorithm consists of the following steps. First we organize the data as a collection of spectral vectors $D=\{\mathbf{d}_{k,l}\}$, where a spectral vector $\mathbf{d}_{k,l}$ consists of all spectral samples corresponding to spatial resolution cell (k,l). The spectral vectors lie in an n-dimensional Euclidean space, where n is the number of spectral bands. To each vector in D we then apply an affine transformation $T:\mathbf{d}_{k,l}\mapsto T(\mathbf{d}_{k,l})\equiv\mathbf{c}_{k,l}$, where $\mathbf{c}_{k,l}$ has dimension m < n, to produce the transformed data set $C=\{\mathbf{c}_{k,l}\}$. This data set is then compressed on a band by band basis using two dimensional wavelet coding as described above, with one key difference. The band independent algorithm compresses each spectral band to the same bit rate, but the band decorrelation algorithm varies the bit rate from plane to plane (subject to an upper bound constraint on the average bit rate). This is done because the transformation T concentrates most of the energy in a few spectral bands, so that allocating higher bit rates to these bands (and correspondingly lower rates to lower energy bands) significantly reduces distortion. The bit allocation is determined by the optimal algorithm described in [6,7]. This algorithm minimizes distortion assuming that the band compression algorithm has an exponential bit rate vs. mean square distortion curve with amplitude proportional to the sum squared in-band energy, and assigns bit rates to bands in proportion to their log-sum-square energy.

To reconstruct the data, C is reconstructed from the wavelet encoding for each band, and then the pseudo-inverse transformation $T^{\mp}:\mathbf{c}_{k,l}\mapsto T^{\mp}(\mathbf{c}_{k,l})\equiv\hat{\mathbf{d}}_{k,l}$ is applied to reconstruct the original data. Note that distortion is introduced both from the lossy wavelet coding and because the transform T generally has no true inverse. However, the pseudo-inverse transform spreads reconstruction errors in C over many spectral bands, making them much less perceptible. Furthermore, the decorrelation transform is structured to minimize the loss of information due to its singularity.

Although we use the well known discrete Karhuenen-Loeve transformation (or principle components analysis) for spectral decorrelation, we feel it is worthwhile to outline a derivation of this transform from a physical and geometric approach that may be less familiar than the statistical approach. This approach shows that the transform is optimal in a sense that does not depend on statistical assumptions that may be hard to justify in practice. It also provides insight into the effectiveness of this transform for compression.

We assume that the spectra in any given data set are primarily linear combinations of spectra corresponding to the various materials constituting the scene. Generally, the number different materials is much less than the number of spectral bands. We therefore expect most of the spectral vectors to lie in, or close to, a linear subspace whose dimension is much lower than the dimension of the spectral vectors. If we could find the basis vectors for this space, than we could produce a lower dimensional approximation by projecting the original spectral vectors onto this space.

Stated more precisely, given a collection $D=\{\mathbf{d}_1,\mathbf{d}_2,\cdots,\mathbf{d}_p\}$ of data vectors in a n-dimensional linear space L, we wish to find a set of m orthonormal n-vectors (with m < n), spanning a subspace S of L, such that the sum of the squared distances between each data vector in L and its orthogonal projection onto S is minimized. If we define the sample autocovariance matrix $\mathbf{R}=\sum_{k=1}^{p}\mathbf{d}_k\mathbf{d}_k^T$ it can be shown that the required basis vectors are the unit eigenvectors $\{\mathbf{e}_1,\mathbf{e}_2,\cdots,\mathbf{e}_m\}$ corresponding to the m largest eigenvalues of $\mathbf{R}$. Note that the coordinates in S of the projection any $\mathbf{d}$ in L onto S are simply its inner products with the basis vectors, $(\mathbf{e}_1^T\mathbf{d},\mathbf{e}_2^T\mathbf{d},\cdots,\mathbf{e}_m^T\mathbf{d})$. Furthermore, any vector $\mathbf{c}$ in S with coordinates $(c_1,\cdots,c_m)$ can be

represented in L as a linear combination of the basis vectors $\mathbf{c} = \sum_{k=1}^{m} c_k \mathbf{e}_k$. We thus have the transform T:L $\rightarrow$ S represented by the matrix **T** whose rows are the (transposed) basis vectors of S, i.e. $T(\mathbf{d}) = \mathbf{T}\mathbf{d}$. Furthermore, this transformation has the pseudo-inverse $T^{+}:S \rightarrow L$ with $T^{+}(\mathbf{c}) = \mathbf{T}^{T}\mathbf{c}$.

Note that in our algorithm, **T** is determined specifically for each data set, based on the sample autocovariance **R**. Some spectral decorrelation algorithms, such as [1] use a fixed **T** derived from statistical model that is independent of the actual data. Although this saves computation, it sacrifices the optimality of the transform. Computing **T** might appear burdensome, but for hyperspectral data the effort required to *apply* **T** is typically many times the effort of the eigensystem solution needed to *find* **T**. A more serious objection may be that **T** must be sent as side data in order to decompress the data.

As a corollary of the construction of **T**, it turns out that the eigenvalues of **R** corresponding to basis vectors in S equal the sum of squares of the coefficients in the corresponding "spectral" band of the transformed data set C. The fact is quite useful because these sum squared band energies are the statistics required to allocate average quantizer bit rates to each band. This means that these bit allocations be determined before the spectral decorrelation transform is actually applied. As a result, rows corresponding to zero or near zero bit rates can simply be dropped from **T**, significantly reducing the number of operations required to compute the transform.

Experimental results. We present results for two data sets produced by the TRWIS sensor. These data sets each contain 90 uniformly spaced and contiguous spectral bands, spanning a wavelength range of 400 to 800 nm. Within each spectral band, there are 450 raster lines with 236 samples per line, with eight bit deep samples. They have been calibrated to compensate for variations in illumination intensity with bandwidth, so that the samples actually represent estimated percent reflectance. Consequently, one expects sample values between zero and 100, but because the calibration is with respect to a diffuse white reference reflector, specular reflections can produce values above 100. Figures 1 and 2 show images from one spectral band in each of these data sets. The first data set ("houses") shows a residential area with houses and vegetation. The second data set ("tents") is an aerial view of tents and military vehicles on a sandy background.

Figure 3 shows plots of peak-signal to noise ratio (PSNR) as a function of compression ratio for each data set and each algorithm. We define PSNR as the square of maximum sample value in the original data set divided by the mean squared error between the original and reconstructed images. The vertical scale in the figure shows PSNR in decibel units. The horizontal scale shows the ratio of the original file size to the compressed file size. For every algorithm, the "tents" PSNR is higher than the corresponding PSNR for the "houses", which reflects the greater compressibility of this image. Other than this uniform vertical shift, the results for the two data sets are quite similar. Substantial differences between the algorithms are evident. The PSNR for the 3-D wavelet transform is two to three dB higher than the band independent algorithm, and in turn the spectral decorrelation PSNR exceeds the 3-D wavelet transform by about four dB.

Comparisons of spectral band images clearly reflect the differences in the rate-distortion curves. Figure 4 shows images of the same spectral band from original and compressed/reconstructed versions of the "houses" data set. The band independent algorithm was used for the top row of images, the 3-D wavelet algorithm for the middle row, and the band decorrelation algorithm for the bottom row. Within each row, the leftmost image is the original data, and the three remaining images correspond to increasing compression ratios from left to right. The spectral decorrelation images are clearly much less distorted than the others. When viewed on a high quality display, distinguishing the reconstructed spectral decorrelation image from the original requires close observation, even at the highest compression ratio. In the case of the 3-D wavelet transform, many of the fine, high contrast details are preserved fairly well, but

there is a noticeable loss of texture and detail in low contrast regions. At the highest compression, these losses are quite obvious. The quality of the best band independent reconstruction appears to be about equivalent to the worst 3-D reconstruction. At the highest compression, all detail is lost, although high contrast edges are fairly well preserved.

Examining the spectral decorrelation images reveals some interesting effects. Although distortion is almost imperceptible, at the highest compression ratio there are a few regions in which there are systematic shifts in the gray levels at which certain features in original data are reproduced. (E.g., the small, crescent shaped dark area immediately below the house at the center left of the image and a curved, dark area contained within a bright, semi-elliptical area at the center of the right edge). These areas apparently contain materials whose reflection spectra are outside the subspace spanned by the spectral decorrelation basis. Since the basis is selected to optimize a mean squared criterion, small or infrequently occurring spectra tend to be poorly represented. As a consequence, in applications where one wishes to detect spectral shapes that are sparsely represented in the original image (such as finding a few camouflaged tents in a forest), spectral decorrelation may perform poorly despite producing excellent mean square error based figures of merit, such as PSNR. In contrast, the band independent and 3-D wavelet algorithms appear to be free of such systematic gray level shifts.

This illustrates the point that it is difficult to assess reconstruction quality without considering how the data is to be used. In dealing with ordinary two dimensional images, it is often assumed implicitly that using the data means that a human being looks at it. With hyperspectral data, it is much more likely that human visual processing will be augmented or supplanted by automated processing. One might even go so far as to view hyperspectral data simply as an ensemble of one dimensional spectral signals, so that the concept of an "image" is irrelevant. In order to compare the different algorithms from this standpoint, we applied two spectrally based automatic processing algorithms to the reconstructed data. Although these algorithms may have limited practical utility by themselves, they are potential elements of more practical processing systems, and serve as useful illustrations.

The first algorithm classifies spatial resolution cells either as "object" (i.e., tent or house) or background cells based on the shape of their spectral profiles through the use of a simple Bayesian classifier as described in [10]. This approach was chosen for its simplicity and ease of interpretation. Although other, more powerful classifiers exist, we wanted to avoid clouding the compression evaluation with questions about the classifier. Also, this classifier is well known and was easily implemented through the use of the Khoros Image Processing system [11].

The classifier was designed in several steps. First, the image was preprocessed so that each spectra had unit energy. This was done so that the classifier made its decisions based on the shape of the spectra rather than on the overall intensity and would work equally well under different scene illuminations. Second, the image was clustered by the k-means clustering algorithm which is essentially the Lloyd-Max vector quantizer. Clustering is performed by first starting with an initial set of cluster centers and, at each iteration, assigning each data point to the nearest cluster center and then recomputing the cluster centers. Both the number of clusters and the initial set were chosen by hand so that representative samples from each class were included. Third, the clusters were assigned to classes by visually inspecting the image. The result of the classifier design was, for each data set, a set of clusters for each class and statistics (mean and covariance) for each cluster. Pixel by pixel classification is performed by finding the Mahalanobis distance to each cluster center (using the cluster mean and covariance) and finding the minimum. The class containing this cluster as a member is the class assignment for the data point.

We applied the classifier to the reconstructed data sets, and collected statistics on spatial cells that were classified differently in the original and reconstructed data sets. Figure 5 shows the percentages of "object" pixels in the original data misclassified in the reconstructed data as a function of compression ratio for each compression algorithm (the lines marked with o's). The same trends seen in the PSNR measurements are evident in this table: spectral decorrelation classified the most accurately, followed by 3-D wavelets, then the band-independent algorithm. The differences between algorithms are dramatic. The 3-D wavelets algorithm missclassifies about half as frequently as the band independent algorithm at similar compression ratios, and the worst

case spectral decorrelation algorithm performance is better than the best case for the other algorithms. Figure 6 shows maps of cell classifications for the original data sets and reconstructed data sets at the highest compression rate for each algorithm. Figure 7 shows corresponding maps of misclassified cells. All of these maps are at the lowest compression ratio tested for each algorithm.These results show that the classification algorithm is more sensitive to distortion that visual comparisons. At these relatively low compression rates, spectral band image distortion is not readily visible. Nonetheless, it induces significant classification errors.

The second example algorithm segments a complete hyperspectral data set into spatial regions such that cells within a region have similar spectral profiles. The segmentation process compares the spectral profile of the data at each spatial location to its neighbors; thus both the spatial and spectral properties of the data are important. Segmentations of the original cube and the compressed and uncompressed version are compared, both by visual inspection, and through a measure of differences between the edge maps. This measure combines discrepancies of two types: those where a pixel was marked as an edge in the original and not in the compressed/uncompressed data, and those where a pixel was marked as an edge in the compressed/uncompressed and not in the original. The two types of "errors" were combined to give a final measure of edge detection errors, expressed as a percentage of pixels across the entire image. While this error measure is simple, it is sufficient to provide a measure of the amount of distortion in the spatial and spectral properties of the cube.

The result of applying the spectral segmented to the "tents" data cube is shown in Figure 8. The boundaries of each region of the image are marked in dark. The resulting segmentations of applying the same algorithm to the compressed/uncompressed data sets using the three compression algorithms with three different compression ratios each are also shown in Figure 9. Quantitative measures of the edge errors for each of the three approaches (at various compression ratios) are shown for three different data sets in Figure 5 (the line labeled with x's). For all three cases, the spectral decorrelation algorithm produced segmentations closest to the original data, followed by the three dimensional transform approach.

References

[1] T. Markos and J. Reif, Multispectral Image Compression Algorithms, Proceedings of 1993 Data Compression Conference, Mar. 1993, pp. 391-400.

[2] J.N. Bradley and C.M. Brislawn, Applications of Wavelet-Based Compression to Multidimensional Earth Science Data, Proceedings of 1993 Data Compression Conference, Mar. 1993, pp. 496.

[3] B.R. Epstein, R. Hingorani, J.M. Shapiro and M. Czigler, Multispectral KLT-Wavelet Data Compression for Landsat Thematic Mapper Images, Proceedings of 1992 Data Compression Conference, Mar. 1992, pp. 200-208.

[4] R.L. Baker and Y.T. Tse, Compression of high spectral resolution imagery, Proceedings SPIE: Applications of Digital Image Processing, Aug. 1988, pp. 255-264.

[5] B. W. Evans, Coding Quantizer Parameters for Wavelet Image Compression, TRW IOC G685.6-0301, Jun. 1992.

[6] B. W. Evans and R. L. Renner, Optimal Bit Rate Allocation for Wavelet Transform Coding, Proceedings of 1993 Data Compression Conference, Mar. 1993, pp. 473.

[7] B.W. Evans, Optimal Quantizer Bit Allocation for Wavelet Image Compression, TRW IOC G685.6.0294, Feb. 1992.

[8] I. Daubechies, The wavelet transform, time-frequency localization and signal analysis, IEEE Trans. Information Theory, vol. 36, Sept. 1990, pp. 961-1005.

[9] S. Mallat, A theory for multiresolution signal decomposition: The wavelet representation, IEEE Trans. Acoustics, Speech and Signal Processing, Dec. 1990, pp. 2091-2110.

[10] R. Duda and P. Hart, Pattern Classification and Scene Analysis, Wiley and Sons, N.Y.,1973.

[11] Rasure, Williams, Agiro and Sauer, "A Visual Language and Software Development Environment for Image Processing", International Journal of Imaging Systems and Technology, Vol. 2, pp 183-199, 1990.

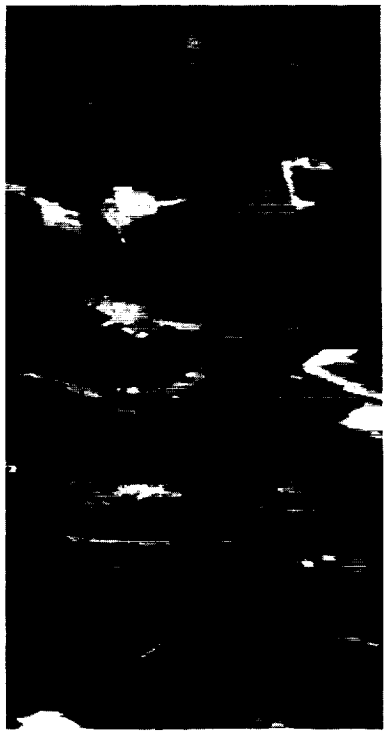

Figure 1. Single spectral band image from "houses" data set.

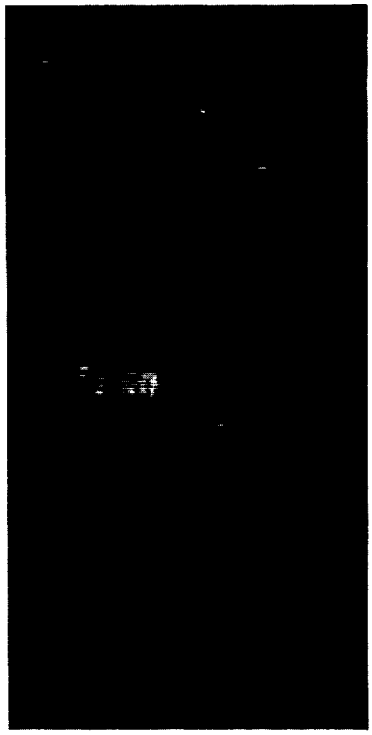

Figure 2. Single spectral band image from "tents" data set.

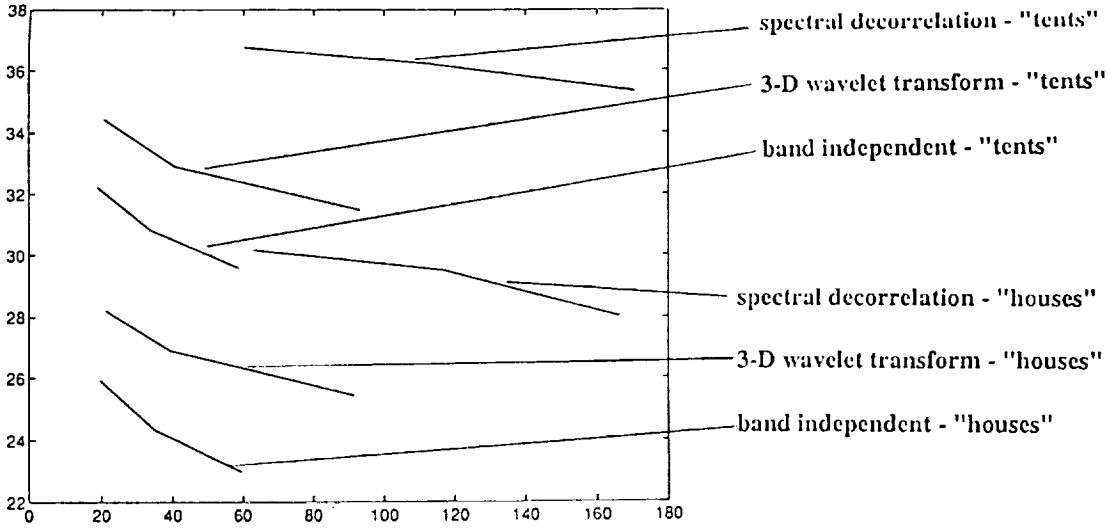

Figure 3. Peak signal to noise ratio vs. compression ratio.

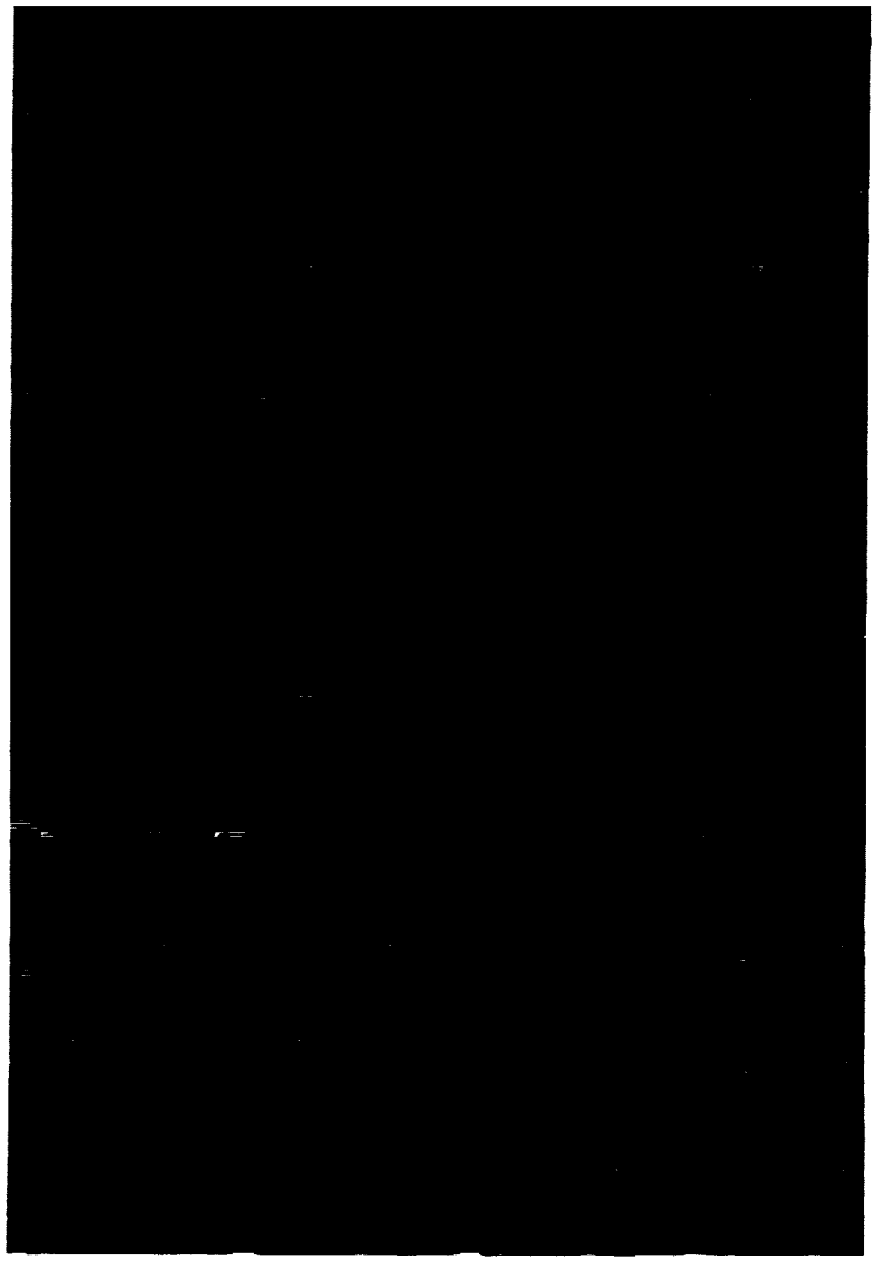

Figure 4. Examples of reconstructed images. Top row: band independent compression, from left to right: original image, 19:1, 34:1 and 59:1 compression. Middle row: 3-D wavelet compression, from left to right: original, 21:1, 41:1 and 92:1 compression. Bottom row: spectral decorrelation compression, left to right: original, 60:1, 112:1, 171:1 compression.

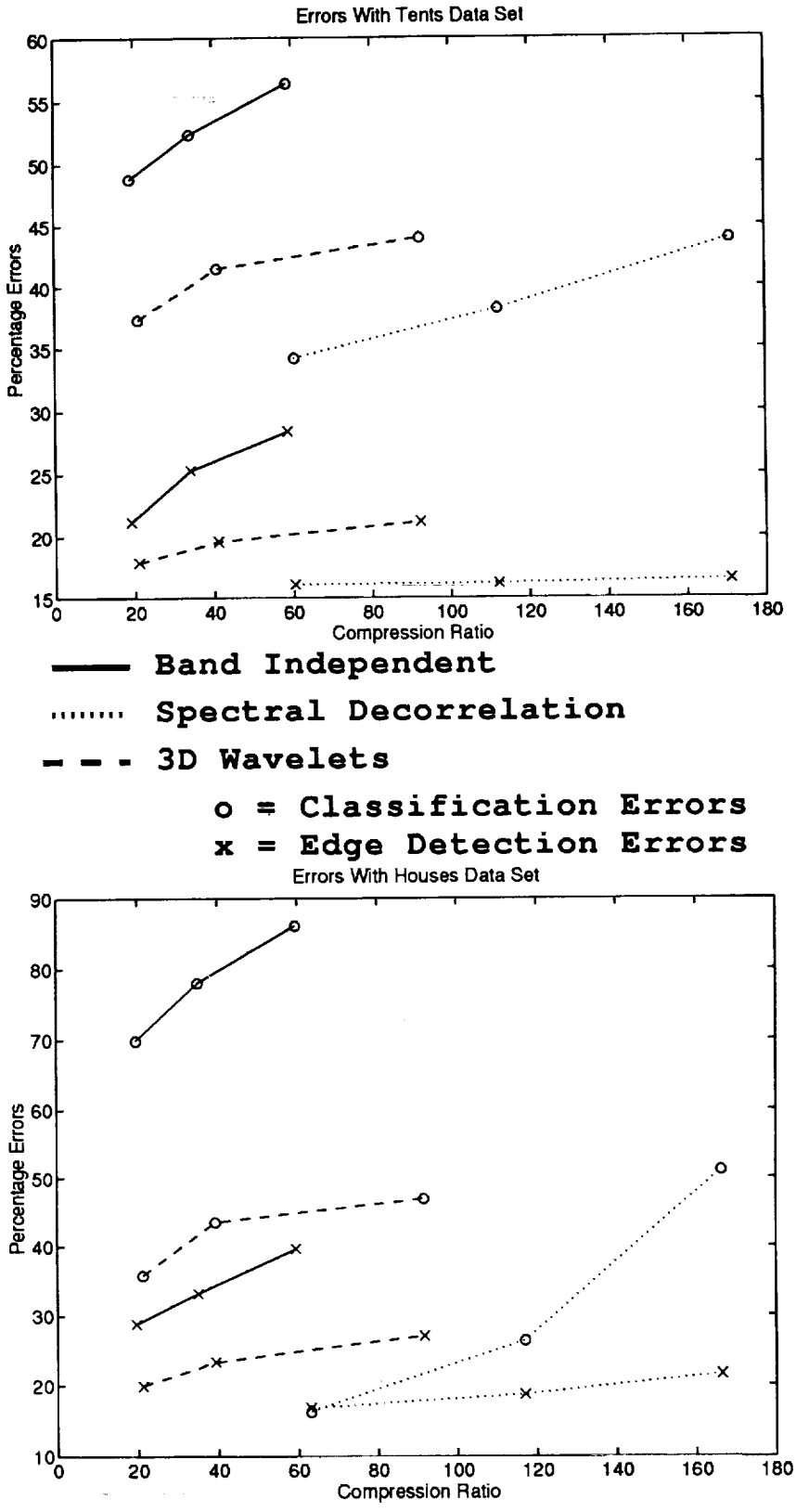

Figure 5. Percent of cells misclassified vs. compression ratio. Top graph: "tents" data set; bottom graph: "houses" data set.

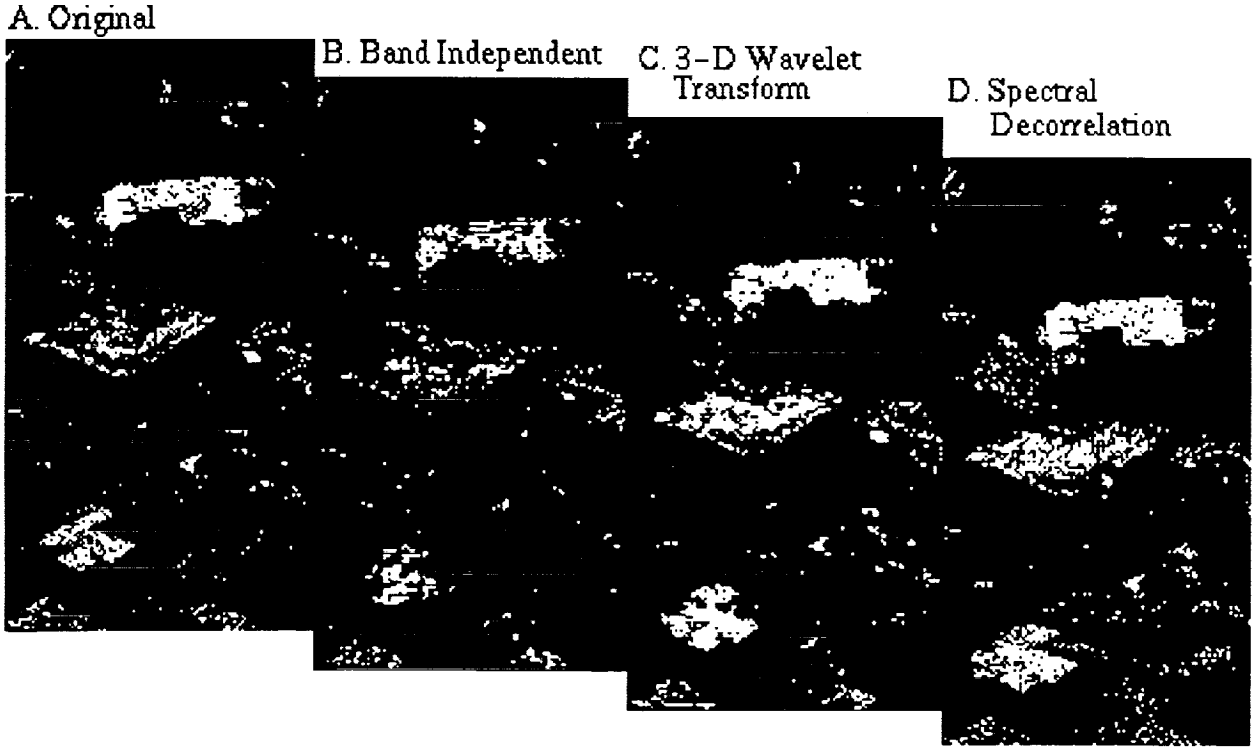

Figure 6. Cell classification maps. "Object" cells shown in white.

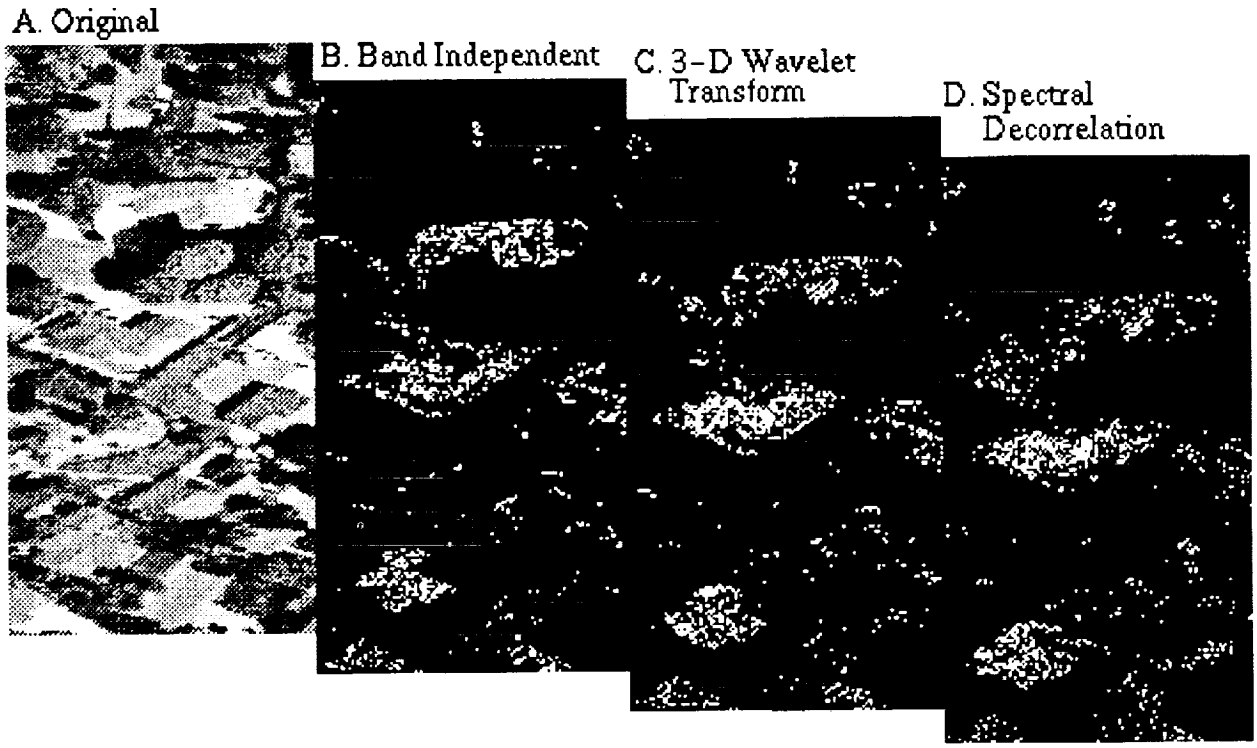

Figure 7. Misclassified cell maps. Incorrectly classified cells shown in white.

Figure 8. Region boundaries for original "tents" data set.

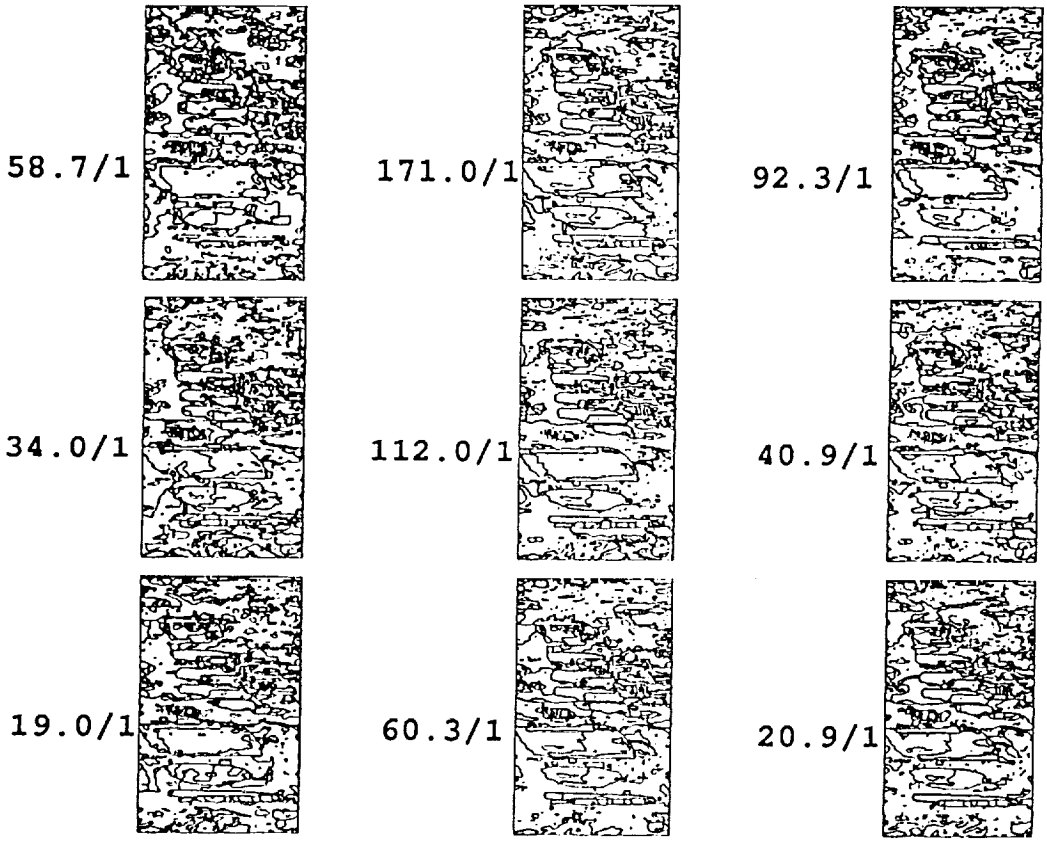

Figure 9. Region boundaries for reconstructed data sets. Left column: band independent algorithm. Middle column: spectral decorrelation algorithm. Right column: 3-D wavelet transform algortithms. Compression ratios shown to left of each image.