Compression of Hyperspectral Imagery

Giovanni Motta, Francesco Rizzo, and James A. Storer
Computer Science Department, Brandeis University, Waltham MA 02454
{gim, frariz, storer}@cs.brandeis.edu

Abstract: High dimensional source vectors, such as occur in hyperspectral imagery, are partitioned into a number of subvectors of (possibly) different length and then each subvector is vector quantized (VQ) individually with an appropriate codebook. A locally adaptive partitioning algorithm is introduced that performs comparably in this application to a more expensive globally optimal one that employs dynamic programming. The VQ indices are entropy coded and used to condition the lossless or near-lossless coding of the residual error. Motivated by the need of maintaining uniform quality across all vector components, a Percentage Maximum Absolute Error distortion measure is employed. Experiments on the lossless and near-lossless compression of NASA AVIRIS images are presented. A key advantage of our approach is the use of independent small VQ codebooks that allow fast encoding and decoding.

1. Introduction

Airborne and space-borne remote acquisition of high definition electro-optic images is becoming increasingly used in scientific and military applications to recognize objects and classify materials on the earth's surface. In hyperspectral photography, each pixel records the spectrum (intensity at different wavelengths) of the light reflected by a specified area, with the spectrum decomposed into many adjacent narrow bands. The acquisition of hyperspectral images produces a two dimensional matrix (or "image cube") where each pixel is a vector having one component for each spectral band. For example, with images acquired by the NASA *Airborne Visible/Infrared Imaging Spectrometer* (e.g., Shaw and D. Manolakis [2002]), each pixel contains 224 bands. Although the resolution of AVIRIS imagery is low compared with the millions of bands found in high resolution laboratory spectrometers, the acquisition of these images already produces large amounts of highly correlated data where each image may be over 1/2 Giga-byte. Since hyperspectral imagery may be used in tasks that are sensitive to local error like classification (assignment of labels to pixels) or target detection (identification of a rare instance), the lossy algorithms commonly used for the compression of pictorial images may not be appropriate.

Approaches to the compression of the hyperspectral image cube that have been proposed in literature include ones based on differential prediction via DPCM (Aiazzi, Alparone and Baronti [2001], Roger, Arnold, Cavenor and Richards [1991]), on direct Vector Quantization (VQ) (Manohar and Tilton [2000]), on dimensionality reduction using the Discrete Cosine Transform (Abousleman [1995]), and the Principal Component Transform (Subramanian, Gat, Ratcliff and Eismann [1992]) (where the representation of a spectrum is transformed in order to isolate the smallest number of significant components). In many cases, DPCM-based methods are too simple to exploit properly multidimensional data, VQ design is too computationally intensive to be applied to the full spectrum, and transform based algorithms may be limited by the use of the squared error as distortion measure because the introduction of uncontrolled error in the compressed data may affect the results of classification and recognition algorithms (Aiazzi, Alparone, Barducci, Baronti and Pippi [2001]). PCT-based lossless compressors may not achieve the best results because the transformed signal is harder to entropy code.

Here, we first apply partitioned vector quantization independently to each pixel, where the variable size partitions are chosen adaptively. These VQ indices are entropy coded and used to condition the lossless or near-lossless coding of the residual error. The codebooks, which have negligible sizes, are included as part of the compressed data. In fact, key advantage of our approach is that the use of independent small VQ codebooks allows fast encoding and decoding.

Section 2 presents basic definitions relating to partitioned vector quantization (PVQ). Section 3 presents a locally optimal PVQ codebook design algorithm (LPVQ) that performs comparably for our application to a globally optimal one based on a more expensive dynamic programming algorithm. Section 4 presents the entropy coding that follows quantization. Section 5 presents experimental results for lossless and near lossless compression subject to a number of performance measures, including the Percentage Maximum Absolute Error (PMAE). Section 6 concludes.

2. Definitions

We model a hyperspectral image as discrete time, discrete values, bi-dimensional random source $\bar{\mathbf{I}}(x,y)$ that emits pixels that are D-dimensional vectors $\mathbf{I}(x,y)$. Each vector component $I_i(x,y)$, $0 \le i \le D-1$, is drawn from the alphabet X_i and is distributed according to a space variant probability distribution that may depend on the other components. We assume that the alphabet has the canonical form $X_i = \{0,1,\dots,M_i\}$.

The complexity of building a quantizer for vectors having the high dimensionality encountered in hyperspectral images is known to be computationally prohibitive. A standard alternative, which we review in this section, is *partitioned* VQ, where input vectors are partitioned into a number of consecutive segments (blocks or subvectors), each of them independently quantized (e.g., see the book of Gersho and Gray [1992]). While partitioned VQ leads to a sub-optimal solution in terms of Mean Squared Error (MSE), because it does not exploit correlation among subvectors, the resulting design is practical and coding and decoding present a number of advantages in terms of speed, memory requirements and exploitable parallelism.

We divide the input vectors (the pixels) into N subvectors and quantize each of them with an L-levels exhaustive search VQ. Since the components of $\mathbf{I}(x,y)$ are drawn from different alphabets, their distributions may be significantly different and partitioning the D components uniformly into N blocks may not be optimal. We wish to determine the size of the N sub vectors (of possibly different size) adaptively, while minimizing the quantization error, measured for example in terms of MSE. Once the N codebooks are designed, input vectors are encoded by partitioning them into N subvectors of appropriate length, each of which is quantized independently with the corresponding VQ. The index of the partitioned vector is given by the concatenation of the indices of the N subvectors.

A Partitioned Vector Quantizer (or PVQ) is composed by N independent, L-levels, d_i-dimensional exhaustive search vector quantizers $Q_i = (A_i, F_i, P_i)$, such that $\sum_{1 \le i \le N} d_i = D$ and:

- $A_i = \{\mathbf{c}_1^i, \mathbf{c}_2^i, ..., \mathbf{c}_L^i\}$ is a finite indexed subset of R^{d_i} called codebook. Its elements $\mathbf{c}_j^i$ are the code vectors.

- $P_i = \{S_1^i, S_2^i, ..., S_L^i\}$ is a partition of R^{d_i} and its equivalence classes (or *cells*) S_j^i satisfy: $\bigcup_{j=1}^{L} S_j^i = R^{d_i}$ and $S_h^i \cap S_k^i = \varnothing$ for $h \neq k$.
- $F_i : R^{d_i} \rightarrow A_i$ is a function defining the relation between the codebook and the partition as $F_i(\mathbf{x}) = \mathbf{c}_j^i$ if and only if $\mathbf{x} \in S_j^i$.

  The index j of the centroid $\mathbf{c}_j^i$ is the result of the quantization of the d_i-dimensional subvector $\mathbf{x}$, i.e. the information that is sent to the decoder.

With reference to the previously defined N vector quantizers $Q_i = (A_i, F_i, P_i)$, a Partitioned Vector Quantizer is formally a triple $\overline{\mathbf{Q}} = \left(\overline{\mathbf{A}}, \overline{\mathbf{P}}, \overline{\mathbf{F}}\right)$ where:

- $\overline{\mathbf{A}} = A_1 \times A_2 \times \ldots \times A_N$ is a codebook in R^D;
- $\overline{\mathbf{P}} = P_1 \times P_2 \times \ldots \times P_N$ is a partition of R^D;
- $\overline{\mathbf{F}} : R^D \rightarrow \overline{\mathbf{A}}$ is computed on an input vector $\mathbf{x} \in R^D$ as the concatenation of the independent quantization of the N subvectors of $\mathbf{x}$. Similarly, the index sent to the decoder is obtained as a concatenation of the N indices.

The design of this vector quantizer aims at the joint determination of the $N+1$ partition boundaries $b_0 = 0 \leq b_1 \leq \ldots \leq b_N = D$ and to the design of the N independent vector quantizers having dimension $d_i = b_i - b_{i-1}$, $1 \leq i \leq N$.

Given a source vector $\mathbf{I}(x, y)$, we indicate the i^{th} subvector of boundaries b_{i-1} and $b_i - 1$ with the symbol $\mathbf{I}_{b_{i-1}}^{b_i - 1}$ (for simplicity, the x and y spatial coordinates are omitted when clear from the context). The mean squared quantization error between the vector $\mathbf{I}$ and its quantized representation $\hat{\mathbf{I}}$, is given by

$$\left(\mathbf{I} - \hat{\mathbf{I}}\right)^2 = \sum_{i=1}^{N} \left(\mathbf{I}_{b_{i-1}}^{b_i - 1} - \hat{\mathbf{I}}_{b_{i-1}}^{b_i - 1}\right)^2 = \sum_{i=1}^{N} \left(\mathbf{I}_{b_{i-1}}^{b_i - 1} - \mathbf{c}_{J_i}^i\right)^2 = \sum_{i=1}^{N} \sum_{h=b_{i-1}}^{b_i - 1} \left(I_h - c_{J_i, h-b_{i-1}}^i\right)^2$$

where $\mathbf{c}_{J_i}^i = \left(c_{J_i,0}^i, \ldots, c_{J_i, d_i - 1}^i\right)$ is the centroid of the i^{th} codebook that minimizes the reconstruction error on $\mathbf{I}_{b_{i-1}}^{b_i - 1}$, and:

$$J_i = \underset{1 \leq l \leq L}{\operatorname{argmin}} MSE\left(\mathbf{I}_{b_{i-1}}^{b_i - 1}, c_l^i\right)$$

3. A Locally Optimal PVQ Design

Given the parameters N (the number of partitions) and L (the number of levels per codebook), the partition boundaries achieving minimum distortion can be found by a brute-force approach. First, for every $0 \leq i \leq j < D$ determine the distortion $Dist(i, j)$ that an L-levels vector quantizer achieves on the input subvectors of boundaries i and j. Then, with a dynamic program, traverse the matrix $Dist(i, j)$ in order to find N costs that correspond to the input partition of boundaries $b_0 = 0 \leq b_1 \leq \ldots \leq b_N = D$ and whose sum is minimal. This approach is computationally expensive and, as experimental comparisons indicate, unnecessary for our application. For past work using dynamic programming for PVQ codebook design, see Matsuyama [1987].

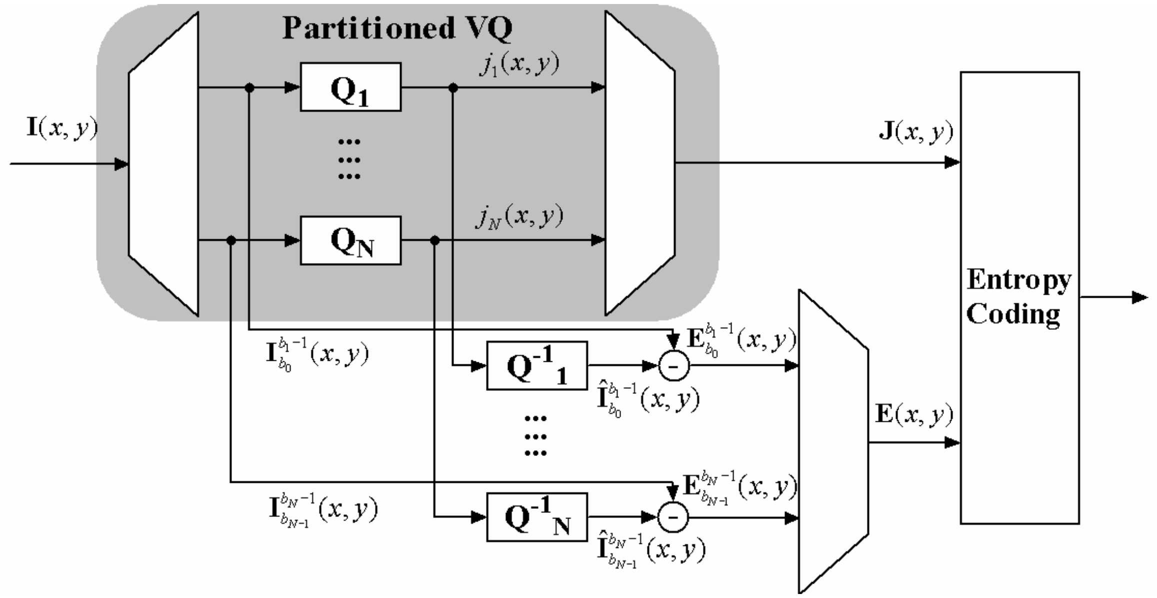
Partitioned VQ
$\mathbf{I}(x,y)$
$\mathbf{Q_1}$
$\mathbf{Q_N}$
$j_1(x,y)$
$j_N(x,y)$
$\mathbf{J}(x,y)$
Entropy Coding
$\mathbf{I}_{b_0}^{b_1-1}(x,y)$
$\mathbf{Q^{-1}_1}$
$\mathbf{E}_{b_0}^{b_1-1}(x,y)$
$\hat{\mathbf{I}}_{b_0}^{b_1-1}(x,y)$
$\mathbf{E}(x,y)$
$\mathbf{I}_{b_{N-1}}^{b_N-1}(x,y)$
$\mathbf{Q^{-1}_N}$
$\mathbf{E}_{b_{N-1}}^{b_N-1}(x,y)$
$\hat{\mathbf{I}}_{b_{N-1}}^{b_N-1}(x,y)$

Figure 1: LPVQ lossless encoder.

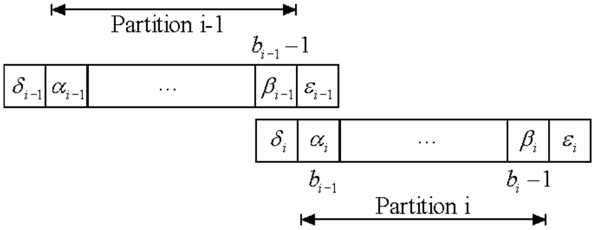

Figure 2: Error contributions for two adjacent partitions.

```
$M_i$ = min( $\beta_{i-1}+\alpha_i$ , $\beta_{i-1}+\varepsilon_{i-1}$ , $\delta_i+\alpha_i$ )
if ( $M_i = \delta_i+\alpha_i$ )
      $b_{i-1} = b_{i-1}-1$
else if ( $M_i = \beta_{i-1}+\varepsilon_{i-1}$ )
            $b_{i-1} = b_{i-1}+1$
```

Figure 3: Partition changes in modified GLA.

Here we propose a locally optimal algorithm for partitioning (LPVQ) that provides an efficient alternative to dynamic programming, while performing comparable in practice for our application of PVQ followed by an entropy coder, as depicted in Figure 1. Our algorithm is based on a variation of the Generalized Lloyd Algorithm (or GLA, Linde, Buzo and Gray [1980]).

Unconstrained vector quantization can be seen as the joint optimization of an encoder (the function $F: R^d \to A$ described before) and a decoder (the determination of the centroids for the equivalence classes of the partition $P = \{S_1, S_2, \ldots, S_L\}$). GLA is an iterative algorithm that, starting from the source sample vectors chooses a set of centroids and optimizes in turns encoder and decoder until the improvements on a predefined distortion measure are negligible. To define our PVQ, the boundaries of the vector partition $b_0 = 0 \le b_1 \le \ldots \le b_N = D$ need to be determined as well. The proposed design follows the same spirit of the GLA. The key observation is that once the partition boundaries are kept fixed, the MSE is minimized independently for each partition by applying the well-known optimality conditions on the centroids and on the cells. Similarly, when the centroids and the cells are held fixed, the (locally optimal) partitions boundaries can be determined in a greedy fashion. The GLA step is independently applied to each partition. The equivalence classes are determined as usual, but as shown in Figure 2, the computation keeps a record of the contribution to the quantization error of the leftmost and rightmost components of each partition:

$$\alpha_i = \sum_{x,y} \left(I_{b_{i-1}}(x,y) - \hat{I}_{b_{i-1}}(x,y) \right)^2 \text{ and } \beta_i = \sum_{x,y} \left(I_{b_i -1}(x,y) - \hat{I}_{b_i -1}(x,y) \right)^2$$

Except for the leftmost and rightmost partition, two extra components are also computed:

$$\delta_i = \sum_{x,y} \left(I_{b_{i-1}-1}(x,y) - \hat{I}_{b_{i-1}-1}(x,y) \right)^2 \text{ and } \varepsilon_i = \sum_{x,y} \left(I_{b_i}(x,y) - \hat{I}_{b_i}(x,y) \right)^2$$

The reconstruction values used in the expressions for δ_i and ε_i are determined by the classification performed on the components $b_{i-1},\ldots,b_i$. The boundary b_{i-1} between the partitions $i-1$ and i is changed according to the criteria shown in Figure 3.

4. Entropy Coding

As was depicted in Figure 1, our LPVQ algorithm follows locally optimal codebook design by lossless and near–lossless coding of the source vectors. We use quantization as a tool to implement dimensionality reduction on the source, where quantization residual is entropy coded conditioned on the subvector indices. After proceeding as described in the previous section to partition the input vector and quantize subvectors to obtain the vector of indices $\mathbf{J}(x,y)$, we compute the quantization error

$$\mathbf{E}(x,y) = \left(\mathbf{E}_{b_0}^{b_1-1}(x,y), \mathbf{E}_{b_1}^{b_2-1}(x,y), \ldots, \mathbf{E}_{b_{N-1}}^{b_N-1}(x,y) \right)$$

where, for each $1 \le i \le N$:

$$\mathbf{E}_{b_{i-1}}^{b_i-1}(x,y) = \mathbf{I}_{b_{i-1}}^{b_i-1}(x,y) - \hat{\mathbf{I}}_{b_{i-1}}^{b_i-1}(x,y)$$

Since the unconstrained quantizers work independently from each other and independently on each source vector, an entropy encoder is used to exploit this residual redundancy. In particular, each VQ index $J_i(x,y)$ is encoded conditioning its probability with respect to a set of causal indices spatially and spectrally adjacent. The components of the residual vector $\mathbf{E}_{b_{i-1}}^{b_i-1}(x,y)$ are entropy coded with their probability conditioned on the VQ index $J_i(x,y)$.

5. Experimental Results

Our LPVQ algorithm has been tested on a set of five AVIRIS images. AVIRIS images are obtained by flying a spectrometer over the target area. They are 614 pixels wide and typically on the order of 2,000 pixels high, depending on how long the instrument is turned on. Each pixel represents the light reflected by a 20m x 20m area (high altitude) or 4m x 4m area (low altitude). The spectral response of the reflected light is decomposed into 224 contiguous bands (or channels), approximately 10nm wide and spanning from visible to near infrared light (400nm to 2500nm). Spectral components are acquired in floating point 12-bit precision and then scaled and packed into signed 16 bit integers. After acquisition, AVIRIS images are processed to correct for various physical effects (flight corrections, time of day, etc.) and stored in "scenes" of 614 by 512 pixels per file (when the image is not a multiple of 512 high, the last scene is 614 wide by whatever the remainder). All files for each of the 5 test images were downloaded from the NASA web site (JPL [2002]) and combined to form the complete images.

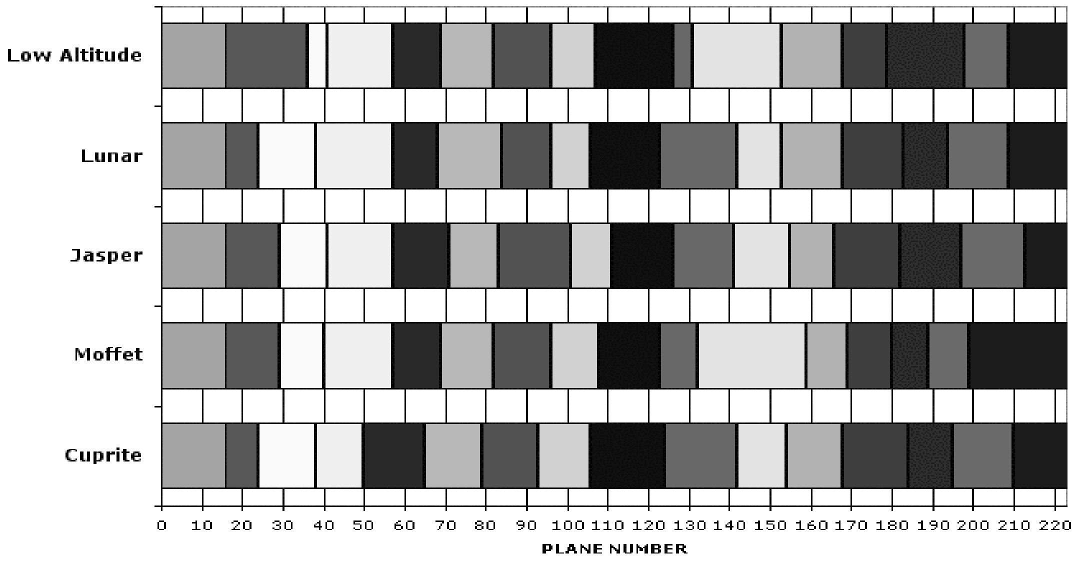

Figure 4: Partition sizes and alignment.

Several experiments have been performed for various numbers of partitions and for different codebook sizes. The results that we describe here were obtained for $N = 16$ partitions and $L = 256$ codebook levels. The choice of the number of levels makes also practical the use of off-the-shelf image compression tools that are fine-tuned for 8 bit data. The LPVQ algorithm trains on each image independently and the codebooks are sent to the decoder as side information. The size of the codebook is negligible with respect the size of the compressed data (256 x 224 x 2 bytes = 112 Kilo-byte) and its cost is included in the reported results.

The partition boundaries for each of the five images are depicted in Figure 4. While similarities exist, the algorithm converges to different optimal boundaries on different input images. This is evidence that LPVQ adapts the partitions to input statistics. Experimentally we have found that adaptation is fairly quick and boundaries converge to their definitive values in less than one hundred iterations.

In the following we analyze the LPVQ performance in terms of Compression Ratio (defined as the size of the original file divided by the size of the compressed one), Signal to Quantization Noise Ratio, Maximum Absolute Error and Percentage Maximum Absolute Error.

The *Signal to Quantization Noise Ratio* (SQNR) is defined here as:

$$SQNR\,(dB) = \frac{10}{D}\sum_{i=0}^{D-1}\log_{10}\left(\frac{\sigma_{I_i}^2}{\sigma_{E_i}^2 + \frac{1}{12}}\right)$$

The correction factor $\frac{1}{12}$ is introduced to take into account the error introduced by the 12 bit analog-to-digital converter used by the AVIRIS spectrometer (Aiazzi, Alparone and Baronti [2001]). This solution also avoids unbounded values in the case of a band perfectly reconstructed.

The *Maximum Absolute Error* (MAE) is defined in terms of the MAE for the i^{th} band as $MAE = \max_i MAE_i$, where MAE_i is:

$$MAE_i = \max_{x,y}\left|\mathbf{I}_i(x,y) - \hat{\mathbf{I}}_i(x,y)\right|$$

AVIRIS	Lossless					Indices Only	
	gzip	bzip2	JPEG-LS	JPEG 2K	LPVQ	CR	SQNR
Cuprite	1.35	2.25	2.09	1.91	3.13	40.44	23.91
Low Altitude	1.38	2.13	2.00	1.80	2.89	39.10	25.48
Lunar Lake	1.36	2.30	2.14	1.96	3.23	47.03	27.15
Moffett Field	1.41	2.10	1.99	1.82	2.94	40.92	25.74
Jasper Ridge	1.39	2.05	1.91	1.78	2.82	35.02	20.37
Average	1.38	2.17	2.03	1.85	3.00	40.50	24.53

Table I: Compression ratio for lossless and lossy mode.

	Constant MAE					Quasi-Constant PMAE					Quasi-Constant SQNR				
Δ	CR	RMSE	SQNR	MAE	PMAE	CR	RMSE	SQNR	MAE	PMAE	CR	RMSE	SQNR	MAE	PMAE
1	4.03	0.82	46.90	1.00	0.19	3.41	0.73	52.15	0.57	0.00	3.45	0.78	51.81	0.62	0.01
2	4.80	1.41	42.49	2.00	0.39	3.97	1.50	47.34	1.54	0.02	3.86	1.56	48.11	1.54	0.02
3	5.49	1.98	39.64	3.00	0.58	4.46	2.27	44.35	2.53	0.03	4.33	2.36	45.06	2.54	0.03
4	6.16	2.52	37.60	4.00	0.77	4.95	3.03	42.04	3.56	0.04	4.75	3.15	42.91	3.53	0.04
5	6.83	3.04	36.05	5.00	0.96	5.36	3.80	40.39	4.54	0.06	5.15	3.95	41.18	4.54	0.06
6	7.50	3.51	34.85	6.00	1.16	5.77	4.56	38.97	5.55	0.07	5.51	4.74	39.85	5.52	0.07
7	8.16	3.97	33.88	7.00	1.35	6.16	5.29	37.83	6.52	0.08	5.88	5.54	38.61	6.55	0.08
8	8.80	4.39	33.08	8.00	1.54	6.55	6.04	36.83	7.53	0.10	6.23	6.31	37.57	7.54	0.10
9	9.42	4.80	32.41	9.00	1.72	6.95	6.76	35.88	8.53	0.11	6.55	7.06	36.73	8.52	0.11
10	0.01	5.19	31.84	9.98	1.85	7.32	7.48	35.14	9.53	0.13	6.90	7.82	35.91	9.53	0.13
11	-	-	-	-	-	7.68	8.15	34.48	10.51	0.14	7.25	8.53	35.19	10.54	0.14
12	-	-	-	-	-	8.07	8.80	33.87	11.53	0.16	7.57	9.21	34.61	11.51	0.16
13	-	-	-	-	-	8.45	9.42	33.32	12.54	0.17	7.94	9.84	34.01	12.51	0.17
14	-	-	-	-	-	8.83	10.01	32.84	13.53	0.19	8.29	10.47	33.50	13.53	0.19
15	-	-	-	-	-	9.17	10.56	32.44	14.51	0.20	8.63	11.03	33.08	14.52	0.20
16	-	-	-	-	-	9.55	11.08	32.05	15.51	0.22	8.97	11.57	32.67	15.53	0.21
17	-	-	-	-	-	9.91	11.58	31.69	16.51	0.23	9.31	12.08	32.32	16.52	0.23
18	-	-	-	-	-	10.32	12.04	31.33	17.53	0.25	9.66	12.55	31.97	17.52	0.24
19	-	-	-	-	-	10.67	12.47	31.04	18.51	0.26	10.00	12.99	31.66	18.52	0.26
20	-	-	-	-	-	11.02	12.89	30.78	19.52	0.28	10.35	13.40	31.35	19.52	0.27

Table II: Average Compression Ratio (CR), Root Mean Squared Error (RMSE), Signal to Quantization Noise Ratio (SQNR), Maximum Absolute Error (MAE) and Percentage Maximum Absolute Error (PMAE) achieved by the near-lossless LPVQ on the test set for different Δ.

The average *Percentage Maximum Absolute Error* (PMAE) for the i^{th} band having canonical alphabet $X_i = \{0,1,\ldots,M_i\}$ is defined as:

$$PMAE\ (\%) = \frac{1}{D}\sum_{i=0}^{D-1}\frac{MAE_i}{M_i}\times 100$$

Table I shows that LPVQ achieves on the five images we have considered an average compression of 3:1, 38.25% better than bzip2 when applied on the plane–interleaved images (worse results are achieved by bzip2 on the original pixel–interleaved image format).

The last column of Table I reports, as a reference, the compression and the SQNR when only the indices are encoded and the quantization error is fully discarded. As we can see from the table, on average we achieve 40.5:1 compression with 24.53dB of SQNR. While extrapolated data suggests that LPVQ outperforms other AVIRIS compression methods, we were unable to perform a detailed comparison because the published studies known to us were either on the old AVIRIS data set (10 bits analog-to-digital converter instead of the newest 12 bit) or reported results for a selected subset of the original 224 bands.

More interesting and practical are the results obtained with the near-lossless settings, shown in Table II. At first, the introduction of a small and constant quantization error across each dimension is considered; that is, before entropy coding, each residual value *x* is quantized by dividing *x* adjusted to the center of the range by the size of the range; i.e., $q(x) = \lfloor (x+\Delta)/(2\Delta+1) \rfloor$. This is the classical approach to the near-lossless compression of image data and results into a constant MAE across all bands. With this setting, it is possible to reach an average compression ratio ranging from 4:1 with the introduction of an error $\Delta = \pm 1$ and a $MAE = 1$ to 10:1 with and error of $\Delta = \pm 10$ and $MAE = 10$. While the performance in this setting seem to be acceptable for most applications and the SQNR is relatively high even at high compression, the analysis of the contribution to the PMAE of the individual bands shows artifacts that might be unacceptable. In particular, while the average PMAE measured across the 224 bands of the AVIRIS cube is low, the percentage error peaks well over 50% on several bands (see Figure 5). Since the PMAE is relevant in predicting the performance of many classification schemes, we have investigated two different approaches aimed at overcoming this problem. In both approaches we select a quantization parameter that is different for each band and it is inversely proportional to the alphabet size (or dynamic). In general, high frequencies, having in AVIRIS images higher dynamic, will be quantized more coarsely than low frequencies. We want this process to be governed by a global integer parameter Δ.

The first method, aiming at a *quasi–constant* PMAE across all bands, introduces on the i^{th} band a distortion Δ_i such that:

$$\frac{1}{D}\sum_{i=0}^{D-1} |\Delta_i| \approx \Delta$$

Since the i^{th} band has alphabet $X_i = \{0,1,\ldots,M_i\}$ we must have:

$$\Delta_i = \pm \left\lfloor \frac{D \cdot \Delta}{\sum_{i=0}^{D-1} M_i} \cdot M_i + \frac{1}{2} \right\rfloor$$

The alternative approach, aims at a quasi–constant SQNR across the bands. If we allow a maximum absolute error $|\Delta_i|$ on the i^{th} band, it is reasonable to assume that the average absolute error on that band will be $\left|\frac{\Delta_i}{2}\right|$. If we indicate with ξ_i the average energy of that band and with Δ the target average maximum absolute error, then the absolute quantization error allowed on each band is obtained by rounding to the nearest integer the solution of this system of equations:

$$\begin{cases} 10\log_{10} \dfrac{\xi_i^2}{|\Delta_i/2|^2} \approx 10\log_{10} \dfrac{\xi_j^2}{|\Delta_j/2|^2} & i, j \in [0,\ldots,D-1],\ i \neq j \\ \dfrac{1}{D}\displaystyle\sum_{i=0}^{D-1} |\Delta_i| \approx \Delta & \end{cases}$$

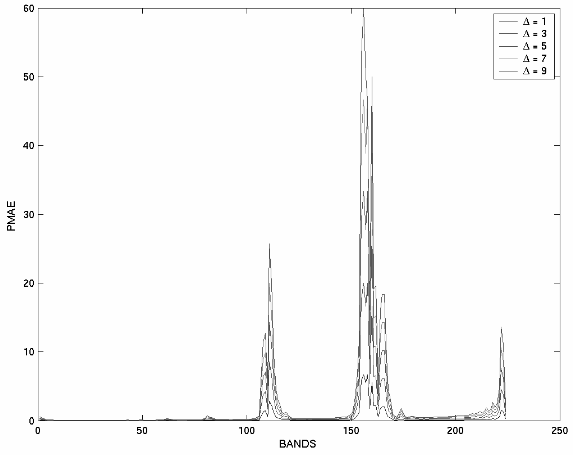

Figure 5: PMAE for near-lossless coding with constant Maximum Absolute Error.

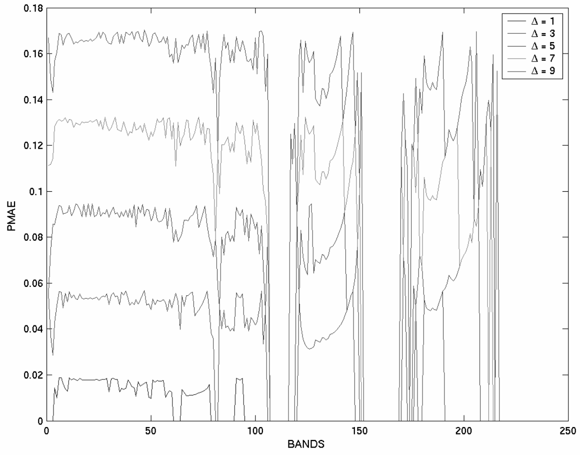

Figure 6: PMAE for near-lossless coding with quasi-constant PMAE.

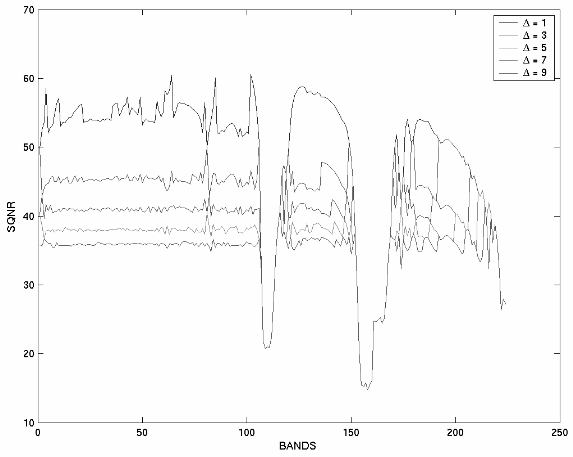

Figure 7: SQNR for near-lossless coding with quasi-constant SQNR.

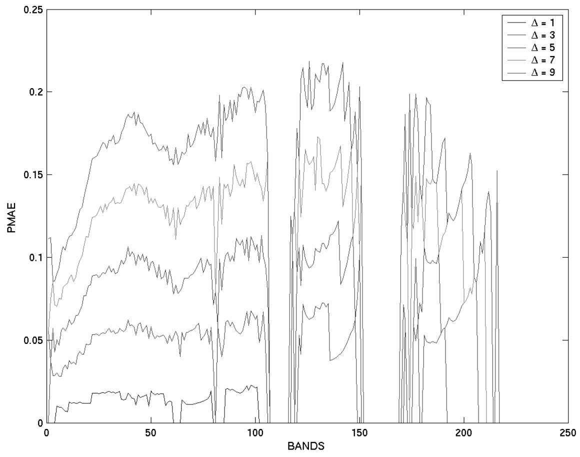

Figure 8: PMAE for near-lossless coding with quasi-constant SQNR.

As can be seen from Table II, the three methods for near-lossless coding of AVIRIS data are equivalent in terms of average SQNR at the same compression. However, the quasi-constant PMAE method is indeed able to stabilize the PMAE across each band (Figure 6). The small variations are due to the lossless compression of some bands and the rounding used in the equations. The average SQNR is not compromised as well. Similar results are observed for the quasi-constant SQNR approach. The SQNR is almost flat (Figure 7), except for those bands that are losslessly encoded and those with small dynamic. The PMAE is also more stable than the constant MAE method (Figure 8).

6. Conclusion

We have presented an extension of the GLA algorithm to the locally optimal design of a partitioned vector quantizer (LPVQ) for the encoding of source vectors drawn from a high dimensional source on R^D. It breaks down the input space into independent subspaces and for each subspace designs a minimal distortion vector quantizer. The partition is adaptively determined while building the quantizers in order to minimize the total distortion. Experimental results on lossless and near-lossless compression of hyperspectral imagery have been presented, and different paradigms of near-lossless compression are compared. Aside from competitive compression and progressive decoding, LPVQ has a natural parallel implementation and it can also be used to implement search, analysis and classification in the compressed data stream. High speed implementation of our approach is made possible by the use of small independent VQ codebooks (of size 256 for the experiments reported), which are included as part of the compressed image (the total size of all codebooks is negligible as compared to the size of the compressed image and our experiments include this cost). Decoding is no more than fast table look-up on these small independent tables. Although encoding requires an initial codebook training, this training may only be necessary periodically (e.g., for successive images of the same location), and not in real time. The encoding itself involves independent searches of these codebooks (which could be done in parallel and with specialized hardware).

References

G. P. Abousleman [1995]. “Compression of Hyperspectral Imagery Using Hybrid DPCM/DCT and Entropy-Constrained Trellis Coded Quantization”, *Proc. Data Compression Conference*, IEEE Computer Society Press.

B. Aiazzi, L. Alparone, A. Barducci, S. Baronti and I. Pippi [2001]. “Information-Theoretic Assessment of Sampled Hyperspectral Imagers”, *IEEE Transactions on Geoscience and Remote Sensing* 39:7.

B. Aiazzi, L. Alparone and S. Baronti [2001], “Near–Lossless Compression of 3–D Optical Data”, *IEEE Transactions on Geoscience and Remote Sensing* 39:11.

A. Gersho and R.M. Gray [1991]. *Vector Quantization and Signal Compression*, Kluwer Academic Press.

JPL [2002]. NASA Web site: http://popo.jpl.nasa.gov/html/aviris.freedata.html

Y. Linde, A. Buzo, and R. Gray [1980]. “An algorithm for vector quantizer design”, *IEEE Transactions on Communications* 28, 84-95.

M. Manohar and J. C. Tilton [2000]. “Browse Level Compression of AVIRIS Data Using Vector Quantization on Massively Parallel Machine,” *Proceedings AVIRIS Airborne Geoscience Workshop*.

Y. Matsuyama [1987]. “Image Compression Via Vector Quantization with Variable Dimension”, *TENCON 87: IEEE Region 10 Conference Computers and Communications Technology Toward 2000*, Aug. 25-28, Seoul, South Korea.

R. E. Roger, J. F. Arnold, M. C. Cavenor and J. A. Richards, [1991]. “Lossless Compression of AVIRIS Data: Comparison of Methods and Instrument Constraints”, Proceedings AVIRIS Airborne Geoscience Workshop.

S. R. Tate [1997], “Band ordering in lossless compression of multispectral images”, *IEEE Transactions on Computers*, April, 477-483.

G. Shaw and D. Manolakis [2002], “Signal Processing for Hyperspectral Image Exploitation”, *IEEE Signal Processing Magazine* 19:1

S. Subramanian, N. Gat, A. Ratcliff and M. Eismann [1992]. “Real-Time Hyperspectral Data Compression Using Principal Component Transform”, Proceedings AVIRIS Airborne Geoscience Workshop.