CONF-9604105--3

LA-UR-96-1014

Title: EVALUATION OF ONBOARD HYPERSPECTRAL -
IMAGE COMPRESSION TECHNIQUES FOR A
PARALLEL PUSH-BROOM SENSOR

VED

1996

JUSTI

Author(s): Scott D. Briles

Submitted to: Algorithms for Multispectral
and Hyperspectral Imagery
SPIE Aerospace/Defense Sensing
and Controls '96
Orlando, FL
April 8-12, 1996

Los Alamos
NATIONAL LABORATORY

Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the University of California for the U.S. Department of Energy under contract W-7405-ENG-36. By acceptance of this article, the publisher recognizes that the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. The Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy.

Form No. 836 R5
ST 2629 10/91

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

MASTER

Evaluation of Onboard Hyperspectral-Image Compression Techniques for a Parallel Push-Broom Sensor

Scott Briles

Los Alamos National Laboratory
Group NIS-3, Mail Stop D440
Los Alamos, NM 87545

ABSTRACT

A single hyperspectral imaging sensor can produce frames with spatially-continuous rows of differing, but adjacent, spectral wavelength. If the frame sample-rate of the sensor is such that subsequent hyperspectral frames are spatially shifted by one row, then the sensor can be thought of as a parallel (in wavelength) push-broom sensor. An examination of data compression techniques for such a sensor is presented. The compression techniques are intended to be implemented onboard a space-based platform and to have implementation speeds that match the data rate of the sensor. Data partitions examined extend from individually operating on a single hyperspectral frame to operating on a data cube comprising the two spatial axes and the spectral axis. Compression algorithms investigated utilize JPEG-based image compression, wavelet-based compression and differential pulse code modulation. Algorithm performance is quantitatively presented in terms of root-mean-squared error and root-mean-squared correlation-coefficient error. Implementation issues are considered in algorithm development.

Keywords: hyperspectral imagery, image compression, wavelets.

1. INTRODUCTION

Hyperspectral imaging sensors employ electro-optics to obtain data which is typically in the visible and near-infrared portions of the spectrum. Hyperspectral data, as well as multispectral data, is three-dimensional data with two dimensions corresponding to spatial position and the third to wavelength. Tens to hundreds of contiguous spectral bands are resolved in hyperspectral sensors to produce a detailed profile of the electromagnetic spectrum. This differs from multispectral imaging sensors that measure relatively fewer bands which are commonly non-contiguous.

Hyperspectral imagery (HSI) can be used in applications that include: global environmental monitoring, mapping, charting, geodesy, and land use planning¹. Readily accessible data can aid in forestry, agriculture, water quality monitoring, and wildlife habitat management. The large number of adjacent spectral bands allows for the measurement of water depth and condition, mapping terrain features, and measuring small differences in crop types. With greater access to larger numbers of data sets and access to near real-time imagery, applications can be extensive including ones as mission specific as fighting the import and production of illegal drugs.

Current advances in camera technology, both in optics and electronics, can produce a sensor with narrow spectral bands and large spatial coverage. Such a sensor would provide an enormous amount of

data with a high degree of spectral isolation. Space-based hyperspectral-imagery systems that implement image compression would provide timely and manageable data.

A satellite-based hyperspectral imaging instrument without data compression would require large storage and/or transmission bandwidths in order to obtain time-continuous data of appreciable duration. In addition to the desire to recover an appreciable time duration of data, the ever increasing demand is to acquire the data in a timely fashion, if not inspection in real time. With both large-bandwidth communications and mass-storage flight hardware being expensive solutions, on-board data compression is a practical solution. Compressed data requires less communications bandwidth to relay the data back to earth and can be stored in-flight with less burden until ground station fly over.

The presented material deals with quantitatively assessing the performance of three compression techniques which can be practically implemented in flight hardware. The formidable amount of data generated by a hyperspectral sensor requires that the data be highly compressed, which in turn means a lossy compression. In addition, the compression technique must insure the integrity of the data and not introduce artifacts that detract heavily from either the spatial or spectral information. HSI is correlated among the two-dimensional spatial and the one-dimensional spectral domains. Exploiting the correlation in all three dimensions would minimize loss but, for some algorithms would require that the flight hardware, although feasible, perhaps be impractical. The algorithmic selection depends on what can be reasonably implemented in flight application-specific hardware. The major constraint of this posed hyperspectral-image-compression problem is that the hyperspectral sensor used is a parallel push-broom sensor.

2. PARALLEL PUSH-BROOM SENSOR

A parallel push-broom sensor generates matrices of pixels at some fixed rate in the same fashion as many of today's common digital imaging sensors. From a single parallel push-broom sensor, it is assumed that the HSI would be presented to the compression hardware in the form of hundreds of pixel rows that are spectrally and spatially adjacent. Thus, a frame from the sensor shows change in the x spatial direction in one axis (across a row) and both y spatial change and λ spectral change in the other axis (down a

Figure 1: Axis relationships for a sensor frame.

column). The satellite direction of travel (DOT) is parallel to the axis that shows both y spatial and λ

spectral change. Rows of the frame showing change in x spatial direction are perpendicular to the satellites DOT. Figure 1 shows the dimensional relationships of a sensor frame. The time-sampling of the frames is

Figure 2: Time progression of sensor frames.

such that the forward movement of the satellite progresses one row of a frame as shown in Figure 2.

Hundreds of temporally adjacent frames would be needed to construct a hyperspectral portrayal of an area of the earth comprised of hundreds of images with each image being of a different observation wavelength, but corresponding to the same area of earth. Thus, frames from the sensor must be dismembered to construct monochromatic images where each image is the result of surveying the same

Figure 3: Image orientations of data cube.

area with different wavelengths. These hundreds of monochromatic images compose a data cube. The data cube can be visualized as varying spatially in the x and y directions and spectrally in the z direction as shown in Figure 3. The term cube is used loosely, since it would most likely be rectangular with the y spatial direction and λ spectral direction being of equal size. Tens of megabytes of temporary, data-processing storage capacity would be needed to collect the raw-pixel data cube required to perform a three-dimensional compression. Such a large capacity would significantly add to the complexity of the hardware.

3. COMPRESSION TECHNIQUES

The compression technique implemented must be designed with consideration for information integrity, hardware complexity, and implementation speed. Correlation is present among all three dimensions of a data cube, and thus compression could exploit the correlation among all three dimension or between any two dimensions. Both implementation speed and hardware complexity have a direct impact on the power and weight needed by the compression electronics. Whether a compression technique compresses in two or three dimensions has an impact on the complexity of hardware with three-dimension compression requiring more hardware complexity. Three compression techniques are examined in this paper. The first two techniques deal with compressing only in the two spatial dimensions and ignoring the spectral direction. The third technique compresses the data taking advantage of the correlation of all three dimensions.

3.1 JPEG compression of octal-manipulated frames

Approaching the compression problem using a standardized algorithm leads to the uses of JPEG compression. The most commonly implemented two-dimensional image-compression techniques are transform coding schemes, i.e., methods in which a transform of the image data is effected followed by compression of the nearly uncorrelated transform coefficients. The most commonly used transform is the discrete cosine transform (DCT) which has found widespread use in image compression and is also part of the widely used JPEG still-image compression standard². Application specific hardware is available that implements JPEG compression.

The JPEG compression standard will block encode an image and then independently transform each block with the block size typically being 8x8 pixels. A simplistic JPEG-compression implementation with parallel push-broom data would be the JPEG compression of frames produced by the sensor. However, in the 8x8 blocks the change in the three dimensions would be compressed together, since y spatial and spectral change occur together in the rows of a sensor frame. This would cause an extensive corruption of the integrity of the data.

Considering eight time-sequential sensor frames, the frames can be manipulated such that eight monochromatic rows are assembled adjacent to each other. The resulting eight new manipulated frames would be comprised of eight-row monochromatic bands with adjacent eight-row bands being spatial adjacent in the y direction. Thus the original frames are dissembled by removing every eighth row from the frame as components for a single new frame. Some additional hardware is needed to construct the manipulated frames. However, since the JPEG algorithm operates on 8x8 blocks, block transforms are only performed on monochromatic blocks, which are most likely more correlated than blocks that are not

monochromatic. This manipulation of frames allows for the new frames to present the spatial data on both directions without discontinuities and to provide spectral information with some integrity.

The eight new octal-manipulated frames would then be compressed using the standard JPEG algorithm. After the 8x8 blocks of a frame has been transformed, scalar quantization of the coefficients is performed followed by zero-run length coding and Huffman encoding. The entire JPEG compression can be implement in as single integrated circuit³. Aside from the interface and control hardware, the only additional hardware required for this compression technique is the hardware required to manipulate the frames.

3.2 Image-by-image discrete wavelet transform (DWT) compression

The JPEG compression standard block encodes an image which produces severe block-boundary artifacts for images with substantial spatial frequencies content at moderate compression levels. Transforming the entire image, instead of just image blocks, would eliminate the block-boundary artifacts. However, the computational burden of two-dimensional DCT is great. To ease computational burden, other transformation techniques that use subband coding and the discrete wave transform (DWT) are investigated. Since subband coding is performed on an entire image, the data cube constructed from sensor frames is the data structure used for compression. For the DWT technique, the monochromatic spatial images of the data cube are individually transformed using subband coding and the DWT. After the transformed images are quantized, zero run-length coding and entropy coding is applied to the transformed images to complete the compression process.

Wavelet transforms have received significant attention recently in the area of image compression⁴. The discrete wavelet transform (DWT) used in subband coding provides a time-efficient method of transforming an entire image. By hierarchically decomposing an input image into a series of successively lower-resolution reference sub-images and their associated detail sub-images, the DWT can regulate the amount of computation needed to transform an image. The wavelet transform is implemented by filtering one-dimensional sequences with related low-pass and high-pass finite-impulse response (FIR) filters and then down sampling the filter outputs by a factor of two. These filtering operations are separable; thus, first the rows are filtered and down sampled and then the columns of the image. The low-pass, low-pass (LL) subimage can then be transformed in a similar fashion to create subimages of subimage. An octave-band decomposition⁵ is performed by repeated transforming of the LL subimages. The number and constellation of the subbands can be tailored for the specific application. Therefore, by using the DWT, a computationally-efficient transform can be designed for a specific input image and the desired results.

The kernels of the low-pass and high-pass FIR filters have been investigated⁶ with a possible kernel option being one that is composed of integer coefficients. This paper utilizes a fifth order low-pass filter and a third-order high-pass filter that have integer coefficients for its filter pair. The reference to integer coefficients disregards the gain of the filters which can be redistributed between the compression and reconstruction operations. Other higher-order filter pairs would produce better results. However, these higher-order filter pairs rely on floating-point operations. The integer-coefficient filter pair was chosen because it presented a more feasible hardware implementation. The z-domain transfer functions of the low-pass, $H_0(z)$, and high-pass, $H_1(z)$ analysis filters are given as

$$H_0(z) = \left(\frac{1}{4\sqrt{2}} \right) (-z^2 + 2z + 6 + 2z^{-1} - z^{-2})$$

$$H_1(z) = \left(\frac{1}{4\sqrt{2}} \right) (-2z^2 + 4z - 2)$$

and the corresponding low-pass $G_0(z)$ and high-pass $G_1(z)$ synthesis filters are

$$G_0(z) = \left(\frac{1}{4\sqrt{2}} \right) (2z + 4 + 2z^{-1})$$

$$G_1(z) = \left(\frac{1}{4\sqrt{2}} \right) (-1z - 2 + 6z^{-1} - 2z^{-2} - z^{-3}).$$

The DWT is implemented by simultaneously high-pass and low-pass filtering an image with the cutoff frequencies at the midpoint of the spectrum. Four sub-images remain after the first filtering. They are the low-pass, low-pass; the low-pass, high-pass; the high-pass, low-pass; and the high-pass, high-pass sub-images. The low-pass, low-pass sub-image can be regarded as the reference sub-image, and the three sub-images with high-pass components can be regarded as associated detail sub-images. Additional filtering can be applied to any of the sub-images in accordance to the importance of the information contained in each. Commonly, the low-pass, low-pass sub-image is iteratively filtered several times to decompose the original image into unequal sub-images. Such convolutional processing could make use of commercially available hardware⁷.

As in all transformed-based compression techniques, the transformed image values are quantized to provide the compression. The level of quantization can and should be different for each component of the transformed image. Thus for the DWT image, each sub-image would be quantized with a different value with more severe quantization being performed on the sub-images at the top of the hierarchy. Quantization is the operation that causes the compression technique to be lossy. The loss is manifested in blurring of edges and to some extent imposing a "ringing" artifact about the edges.

After quantization, many of the coefficients in the detail sub-images will have a value of zero. With so many values in an image being zero after transforming and quantizing, further compression can be obtained by zero run-length coding (ZRLC). Basically this amounts to replacing zeros with the number of zeros that occurred in succession. Entropy coding can then be applied to the zero-run lengths and the remaining non-zero amplitude values. A variable-length entropy coding technique, such as Huffman encoding, uses fewer bits to represent values that occur frequently than values that occur infrequently.

3.3 Hybrid DPCM/DWT compression

For the mage-by-image DWT technique, the monochromatic spatial images of the data cube are individually transformed using subband coding and the DWT. After the transformed images are quantized, differential pulse code modulation (DPCM)⁸ can be applied in the spectral direction to completely transform the data cube. Zero run-length coding and entropy coding is applied to the transformed data cube to complete the compression process.

The above compression algorithm can be referred to as a hybrid DPCM/DWT compression technique. The compression problem is decomposed into spatial compression and spectral compression. A

two-dimensional lossy-compression technique is implemented in the spatial domain using the DWT, and a one-dimensional lossless-compression technique is implemented in the spectral domain using DPCM.

DPCM, the spectral-domain compression technique, capitalizes on the fact that spectrally-adjacent images in the data cube are highly correlated. After transforming and quantizing each image in the data cube, the spectrally-adjacent images are even more so correlated since high-frequency spatial variation have been reduced by quantization. Lossless differential pulse code modulation (DPCM) can be applied in the spectral direction of the data cube to complete compression in all three dimensions. DPCM basically involves successive subtraction of adjacent images. The resulting difference values are highly uncorrelated and are much smaller than the original amplitude values, with many of the difference values being zero. Spectral integrity is maintained since the first image is unaltered and successive transformed images below it in the data cube can be reconstructed without degradation.

4. EXPERIMENT

In order to evaluate the compression techniques, real-world data was compressed and the error associated with the compression calculated. NASA's Jet Propulsion Laboratory provide data from the Airborne Visible Infrared Imaging Spectrometer (AVIRIS) program⁹. The data file is referenced as flight 940407B, run 06, scene 01, and is from a flight over Moffett Field, California in 1994. Imagery data was taken from an altitude of approximately 20 km and is composed of 224 separate images with each image of a different wavelength in the range of 400 to 2500 nanometers. A single image is 614 pixels by 512 pixels with each pixel of an image representing a ground area of approximately 20 square meters.

4.1 Data preparation

In an effort to simulate the data that would be produced by a parallel push-broom sensor, each pixel of the data was linearly converted from a signed sixteen-bit integer to an unsigned eight-bit integer. Of course this conversion resulted in a distortion of the data, causing some images to appear near uniform. Since for a parallel push-broom sensor, the number of image rows must equal the number of spectral wavelength, the images were cropped to a size of a 384 pixels by 224 rows. Figure 4 shows a typical image of the data cube after byte scaling and cropping. The cropping of pixels from 614 to 384 was performed to reduce computational burden. For JPEG compression the data was ordered into octal-

Figure 4: Byte scaled and cropped AVIRIS image at a wavelength of 558.7 nanometers.

manipulated frames, as discussed above. Both DWT-based compression techniques used the three-dimensional data cube.

4.2 Algorithm implementation

JPEG compression was implemented using the function provided with the interactive analysis software IDL¹⁰ on a Unix workstation. The compression ratios for the JPEG compressed images were calculated with the JFIF-header bytes subtracted from the compressed-file size. No attempt was made to customize the quantization tables or the Huffman encoding tables for the JPEG compression. DWT-based compression was implemented in custom C program. The quantization of the wavelet coefficients used a simple scheme based on powers of two. As the hierarchy of subbands descends, the subimages with high-pass components are quantized by factors of 64, 32, and 16. The least low-pass, low-pass subimage was scaled by a factor of 4. These quantization factors were adjusted with a common scaling factor to achieve different compression ratios. Huffman encoding of DWT images used the same encoding as JPEG compression with the modification that zero runs of 32, 64, 128, 256, 512 and 1024 could be encoded with a single Huffman codeword.

4.3 Performance metrics

Compression performance is measured with two metrics in this paper. The first metric is the widely used root-mean-squared error (RMSE)^{11,12}. RMSE does not always correlate well with perceived image quality, but it does provide some measure of relative quality. The second metric is related specifically to multispectral¹³ and hyperspectral imagery and can be referred to as root-mean-squared correlation-coefficient error (RMSCCE). This metric involves first determining the correlation coefficient between each pair of images in the original data cube. Then a similarly correlation-coefficient matrix is computed from the reconstructed data and the square root of the average squared difference between the correlation-coefficient matrices provides the metric value. RMSCCE provides some measure of how well spectral fidelity has maintained through the compression process.

5. RESULTS

Each of the three compression techniques was implemented ten times each with compression varied from approximately 0.15 bits/pixel/band (where band refers to wavelength band) to 0.42 bits/pixel/band. This range of compression values corresponds to compression ratios of approximately 54:1 to 20:1, respectively. A RMSE value computed for all the imagery comprised in a data cube is associated with each of the compression values. Figure 5 shows that over most of the compression range, the hybrid DPCM/DWT technique performed the best and the image-by-image DWT technique performed the worst. The overall poor performance of the image-by-image DWT technique may be improved with adjustments to the quantization scheme used. Even with a simple quantization scheme, at the highest compression level the image-by-image DWT technique appears to perform better than the image-by-image JPEG compression at the extremely high compression ratio.

RMSCCE for the three compression techniques was calculated for a single compression value of approximately 40:1 (0.2 bits/pixel/band). As might be expected, hybrid DPCM/DWT performed the best. However, image-by-image DWT compression performed better than image-by-image JPEG compression even though at this compression ratio JPEG compression has a better RMSE performance.

Figure 5: Comparison for complete data cube of root-mean-squared error versus compression for the three compression techniques

Table 1: Root-mean-square correlation-coefficient errors for a compression of 40:1

Compression Technique	RMSCCE
Image-by-Image JPEG	0.147638
Image-by-Image DWT	0.146221
Hybrid DPCM/DWT	0.142079

6. SUMMARY

This paper presented three data-compression techniques with the intent that they be performed on the hyperspectral data provided by a parallel push-broom sensor. Algorithm selection was governed to some extent by the practicality of implementing the algorithms in hardware onboard a space-based platform. Using real-world data, performance measurements were presented for the three compression techniques. The best performing technique in terms of both overall distortion, (RMSE), and spectral

fidelity, (RMSCCE), was the hybrid DPCM/DWT technique. Image-by-image JPEG compression performed better than image-by-image DWT for all but the most extreme compression ratio. This difference may be overcome with more research into quantization and entropy encoding of the DWT images. For the compression ratio examined, the JPEG-based compression performed poorest in terms of spectral fidelity. The information in this paper provides some benefit in quantitatively examining the tradeoffs in placing a hyperspectral parallel push-broom sensor in space.

7. ACKNOWLEDGMENTS

The author wishes to thank Dr. Jonathan Bradley for his guidance and support. The author would also like to acknowledge the members of the Watertight Hyperspectral Instrument Project for their rapid development efforts which provide the inspiration for this paper. This work was carried out under the auspices of the United States Department of Energy.

8. REFERENCES

1. T. E. Bell, "Remote Sensing," *IEEE Spectrum*, Vol. 32, No. 3, pp. 25 - 27, March 1995.
2. W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Compression Standard*, Van Nostrand Reinhold, New York, NY, 1993.
3. C-Cube Microsystems, *CL550/CL560 JPEG Compression Processor User's Manual*, C-Cube Microsystems, Milpitas, CA, 1993.
4. O. Rioul and M. Vetterli, "Wavelets and Signal Processing," *IEEE Signal Processing Magazine*, pp. 14 - 38, October 1994.
5. M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*, Ch. 7, Prentice Hall, Englewood Cliffs, NJ, 1995.
6. J. D. Villasenor, B. Belzer and J. Liao, "Filter Evaluation and Selection in Wavelet Image Compression," *Proceeding of the IEEE 1994 Data Compression Conference*, Vol. 1068, pp. 351 - 360, IEEE Computer Society, Snowbird, UT, 1994.
7. Graychip Inc., "GC2001 Digital Filter Chip Datasheet Rev. B", Graychip Inc., Palo Alto, CA., June 3, 1995.
8. R. C. Gonzalez and P. Wintz, *Digital Image Processing, Second Edition*, Ch. 6, Addison-Wesley, Reading, Massachusetts, 1987.
9. JPL AVIRIS program world-wide-web page, <ftp://ophelia.jpl.nasa.gov/pub/docs/html/aboutAV.htm>
10. Research Systems Inc., *IDL User's Guide Ver. 4*, Research Systems Inc., Boulder, CO, 1995.
11. M. Rabbani and P. W. Jones, *Digital Image Compression Techniques*, pp. 76 -77, SPIE, Bellingham, Washington, 1991.
12. J. N. Bradley and C. M. Brislawn, "Spectrum Analysis of Multispectral Imagery in Conjunction with Wavelet/KLT Compression," Los Alamos National Laboratory Report LA-UR-93-3959, 1993.
13. J. A. Saghi, A. G. Tesher and J. T. Reagan, "Practical Transform Coding of Multispectral Imagery," *IEEE Signal Processing Magazine*, Vol. 21 No. 1, pp. 32 - 43, January 1995.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.