Nationaal Lucht- en Ruimtevaartlaboratorium
National Aerospace Laboratory NLR

NLR-TP-2000-605

Spectral PPCA transform and spatial wavelets using lifting technique for data compression of digital hyperspectral images

W. Verhoef

Nationaal Lucht- en Ruimtevaartlaboratorium
National Aerospace Laboratory NLR

NLR-TP-2001-605

Spectral PPCA transform and spatial wavelets using lifting technique for data compression of digital hyperspectral images

W. Verhoef

This report is based on a presentation held at the the Europto Remote Sensing Conference in Barcelona on 29 September 2000.

Division:	Flight
Issued:	20 November 2001
Classification of title:	Unclassified

Contents

1 Table
7 Figures

(13 pages in total)

Spectral PPCA transform and spatial wavelets using lifting technique for data compression of digital hyperspectral images

Wout Verhoef

National Aerospace Laboratory NLR
P.O. Box 153, 8300 AD Emmeloord, The Netherlands
e-mail: verhoef@nlr.nl

ABSTRACT

Lifting has been recognised as an effective numerical technique to realise linear transformations to digital data in integer-to-integer form, which guarantees perfect reversibility. When applied to decorrelate digital hyperspectral images in the spectral and spatial domains, lifting can be applied to accomplish lossless data compression.

Spectral pairwise principal component analysis (PPCA) and spatial wavelet transforms have been combined to demonstrate data compression of digital hyperspectral images acquired by the AVIRIS instrument, and in both transforms lifting has been applied to realise an efficient algorithm, suitable for on-board implementation in a spaceborne imaging spectrometer.

The cascaded spectral PPCA algorithm produces a large number of noisy images, which subsequently are compressed using a general purpose Lempel-Ziv coder. The resulting signal images are spatially decorrelated using a wavelet transform, and an embedded zerotree encoder (EZT) is applied to achieve data compression for these. Uniform linear quantisation of the spectrally and spatially decorrelated data is applied to allow for quasi-lossless compression, in which case a higher compression ratio is obtained. The overall compression factors obtained for 16-bit AVIRIS data from two scenes vary from about two for lossless compression to four for quasi-lossless compression with an rms error of 2% of the input standard deviation.

Keywords: Data compression, hyperspectral images, lifting, wavelets, principal components, reversible

1. INTRODUCTION

In the framework of the ISAC (Image Scalable Adaptive Compression) ESA project, under ESTEC contract no. 11964/NL/FM/(SC) and in collaboration with IMEC, Leuven, Belgium, software developments and integration have taken place in order to achieve 3D data compression of hyperspectral image data, such as will possibly be acquired by the PRISM instrument on board one of ESA's future Earth Explorer Missions, the Land Surface Processes and Interactions Mission (LSPIM).

In 3D image data compression, redundancies in the normal two image dimensions (i.e. rows and columns) are exploited along with redundancy in the third dimension in order to accomplish data compression. The third dimension might be time, in which case the correlation between successive image frames would be used, or, as in the ISAC study, the spectral dimension, in which case the correlation between images from different parts of the spectrum is exploited.

Imaging spectroradiometers such as PRISM are able to record images of the earth from space in many spectral wavelength bands simultaneously, and the images in adjacent spectral channels are usually highly correlated, so there is a certain degree of redundancy in the spectral dimension. The PPCA (Pairwise Principal Components Analysis) algorithm has been specially developed at NLR to exploit this spectral redundancy. The associated data transformation is integer-to-integer and reversible, due to the use of so-called lifting techniques [1, 2, 3]. The result of PPCA applied to hyperspectral image data is still a series of digital images, but the correlation amongst them has been greatly reduced and many transformed images have a low entropy and are noisy. Noisy images result mainly from residual differences between the correlated images from adjacent spectral bands. These can be compressed directly by means of a relatively simple method, such as quantisation followed by Huffman coding. Another part of the images produced by PPCA still contains meaningful spatial information. These images are reversibly transformed by means of a wavelet transform in the spatial domain with the FlexWave software developed by

IMEC. In the FlexWave software the wavelet transform is also implemented in integer-to-integer form by means of lifting in the spatial domain, so absolute reversibility is guaranteed. In this case a more complex quantisation and coding method is used, namely a modified embedded zero-tree encoding [4] (EZT), which is specially adapted to wavelet-transformed images. After the quantisation and coding step the images are transformed in reverse direction in order to evaluate the distortions in the reconstructed images. As all transformations are essentially reversible, the only place where real distortions are introduced is in the quantisation part.

For the integrated software package a graphical user interface (GUI) has been developed in the TCL/TK language. This language allows scripting of the various processing blocks (executable programs), as well as graphical control of the software by the user in a window environment, under Unix as well as Windows 95/98/NT.

The paper gives an overview of the project, with a brief introduction on the lifting technique, the application of lifting in the spectral PPCA algorithm and in the spatial wavelet transform and the quantisation and coding of the decorrelated data. Finally, the data compression results obtained for AVIRIS imaging spectrometer data under lossless and quasi-lossless conditions are discussed.

2. LIFTING

Lifting is a numerical technique to approximate linear transformations involving real numbers and applied to integer data by integer-to-integer (int-to-int) equivalents without losing information. Sweldens and others [1, 2, 3] have demonstrated the use of lifting for the efficient implementation of int-to-int wavelet transforms for data compression of digital images. The advantages of lifting compared to the convolution approach for the implementation of biorthogonal wavelet filter pairs can be summarised as follows:

- In the limit, the number of required floating point operations can be reduced to 50%
- Less memory is required, as in-place calculation is possible
- The inverse is very simple, as it only involves reversing the order of calculation
- Int-to-int transform guarantees absolute reversibility and therefore lossless compression of digital image data
- Parallelism. All local operations within one elementary lifting step can be executed in parallel

An elementary lifting step consists of the following operation , applied to the integers *i* and *j*, and giving as output the integer result *k*:

$$k = i + \mathrm{nint}(aj) \quad ,$$

where *a* is a real number, and nint is the nearest integer function. Some of the information in *i* and *j* is preserved in *k*, but one more operation is necessary in order to preserve all information. This operation might be

$$l = j + \mathrm{nint}(bk) \quad ,$$

where *l* is a new integer and *b* is a real number. In the example above *i* and *j* have been transformed into the new integer variables *k* and *l*, which are integer approximations of linear transformations applied to the couple *i* and *j*. The above sequence of two lifting steps is absolutely reversible, so if one starts with knowledge of *k*, *l*, *a* and *b*, then *i* and *j* can be exactly reconstituted by performing the above steps in reversed order and changing knowns and unknowns. For the above example this comes down to calculation of

$$j = l - \mathrm{nint}(bk) \quad \text{, followed by}$$

$$i = k - \mathrm{nint}(aj) \quad .$$

Now, regardless of the values of the real numbers a and b, the original values of i and j have been recovered, because the floating-point operations in the forward path are exactly repeated on the way back.

Elementary lifting steps can be represented by means of matrices like

$$\begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix} \text{and} \begin{pmatrix} 1 & 0 \\ b & 1 \end{pmatrix} ,$$

and the above example might be written as

$$\begin{pmatrix} k \\ l \end{pmatrix} = \text{nint}\begin{pmatrix} 1 & 0 \\ b & 1 \end{pmatrix} \text{nint}\begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix} \begin{pmatrix} i \\ j \end{pmatrix} .$$

By combining the above types of lifting steps alternately in longer sequences, one can create arbitrary complex int-to-int approximations to linear transformations of image data in the spatial as well as the spectral domain. Examples of both are discussed in the following sections.

3. SPECTRAL PPCA TRANSFORM

The optimum way to remove the correlation in the spectral domain is to apply the Karhunen-Loève transform (KL), also called principal components analysis. This transform diagonalises the covariance matrix and preserves the total variance, so in principle no information is lost, while the spectral correlation is totally removed. However, for on-board implementation the KL-transform is less suitable, as collecting the required statistics and computation of the eigenvectors of the covariance matrix are much too time-consuming.

Hyperspectral earth observation data are usually featured by very high correlations between neighbouring spectral channels in most parts of the spectrum. This high spectral correlation could be reduced substantially if one would replace each pair of spectral channels by their sum (or their average) and difference. However, this is only effective if both channels have about the same variance, and this is mostly not the case with hyperspectral data, because detector response and solar illumination exhibit rather strong spectral variations. Therefore, applying the KL-transform to pairs of spectral channels is a better solution. In that case the transform adapts automatically to the non-uniform spectral variance, so that the decorrelation of neighbouring spectral channels is always optimum. Another advantage of this approach is that, compared to the full KL-transform, much fewer statistics are needed (only all variances and half of the covariances between neighbours) and computation of the transform parameters is simple. This allows quick adaptation to changing scene statistics and on-board implementation would be feasible. A disadvantage is that only the correlation between neighbours is removed. However, by applying a cascaded series of transforms, where the so-called pairwise principal components analysis (PPCA) is applied to the “sum”components of the previous step (Fig. 1), almost all correlation can be removed. In the implemented version a moving window of statistics from eight scanlines is used, for which the means and spectral covariances are updated for each new scanline.

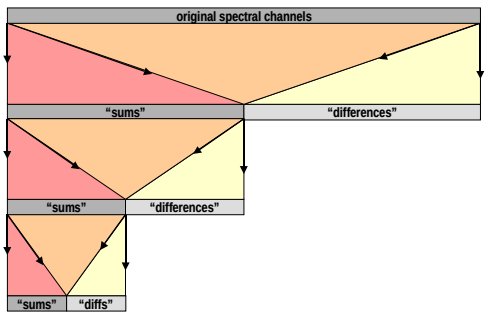

Fig. 1 Cascaded PPCA implementation applied to hyperspectral data

The KL and PPCA transforms are real transforms, and when applied to integer input data, the output consists of real numbers. For data compression this forms a complication, as the representation of real numbers requires relatively many bits, and the inverse transform might not be perfectly exact, as with floating point operations always some loss of numerical precision is involved. Therefore, in the project the PPCA transforms have been implemented using the lifting technique, which makes it possible to execute them as reversible int-to-int transforms. Essentially, the principal components transform applied to a pair of variables consists of a pure rotation about an angle φ in 2D feature space, and in matrix form this can be expressed by

$$\mathbf{R} = \begin{pmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{pmatrix} .$$

A sequence of two lifting steps can only produce matrices like $\begin{pmatrix} 1 & a \\ b & 1+ab \end{pmatrix}$ or $\begin{pmatrix} 1+ab & a \\ b & 1 \end{pmatrix}$, so with two lifting steps the best one can get is a scaled rotation, i.e. a rotation followed by scaling of the output variables. Here it is important to compensate the expansion of one of the variables by the contraction of the other. Otherwise either information might be lost or "holes" in the bivariate histograms might be created, both of which situations are undesirable. This leaves the two options

$$\begin{pmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{pmatrix} = \begin{pmatrix} 1/\cos\varphi & 0 \\ 0 & \cos\varphi \end{pmatrix} \begin{pmatrix} \cos^2\varphi & \sin\varphi\cos\varphi \\ -\tan\varphi & 1 \end{pmatrix} = \begin{pmatrix} \cos\varphi & 0 \\ 0 & 1/\cos\varphi \end{pmatrix} \begin{pmatrix} 1 & \tan\varphi \\ -\sin\varphi\cos\varphi & \cos^2\varphi \end{pmatrix} ,$$

and in both cases the second matrix can be implemented by means of two lifting steps by a proper choice of *a* and *b*. The result is that one can write (with the abbreviations $t = \tan\varphi$ and $s = \sin\varphi\cos\varphi$):

$$\begin{pmatrix} 1 & s \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -t & 1 \end{pmatrix} = \begin{pmatrix} \cos\varphi & 0 \\ 0 & 1/\cos\varphi \end{pmatrix} \begin{pmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{pmatrix} = \mathbf{DR} \quad \text{, and}$$

$$\begin{pmatrix} 1 & 0 \\ -s & 1 \end{pmatrix} \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1/\cos\varphi & 0 \\ 0 & \cos\varphi \end{pmatrix} \begin{pmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{pmatrix} = \mathbf{UR} \quad .$$

The first is a rotation followed by a scale change that reduces the variance of the first principal component and enforces the second. In the second transform the rotation is followed by a scale change which enforces the first principal component and reduces the second. The latter gives a strong upswinging of variances, and certainly in a cascaded implementation this is undesirable, as this might soon lead to word overflow. The former leaves the variance of the first component on a more or less constant level, and therefore this version is preferred. Should one really require an orthonormal transform (i.e. a pure rotation without scaling), then this is still possible using lifting, as one can write

$$\mathbf{R}^2 = \mathbf{RDUR} = (\mathbf{D}^{-1}\mathbf{R}^{-1})^{-1}\mathbf{UR} = (\mathbf{UR}^{-1})^{-1}\mathbf{UR} \quad .$$

As $(\mathbf{UR}^{-1})^{-1} = \left[\begin{pmatrix} 1 & 0 \\ s & 1 \end{pmatrix} \begin{pmatrix} 1 & -t \\ 0 & 1 \end{pmatrix} \right]^{-1} = \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -s & 1 \end{pmatrix}$, one obtains

$$\mathbf{R}^2 = \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -s & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -s & 1 \end{pmatrix} \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -2s & 1 \end{pmatrix} \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} .$$

So a pure rotation can be written as two rotations about half the angle, and this combination in turn can be implemented by means of three lifting steps. However, in the ISAC project preference was given to the first variant, **DR**, as it involves only two lifting steps per PPCA transform and for practical implementation it was considered important to avoid possible word overflow problems.

4. SPATIAL WAVELET TRANSFORM

In the FlexWave software package developed by IMEC, two wavelet filter pairs are implemented, and in both cases lifting is used in order to realise an efficient int-to-int wavelet transform. Fig. 2 shows the two filter pairs, how they are applied to image data and the resultant equivalent convolution filters.

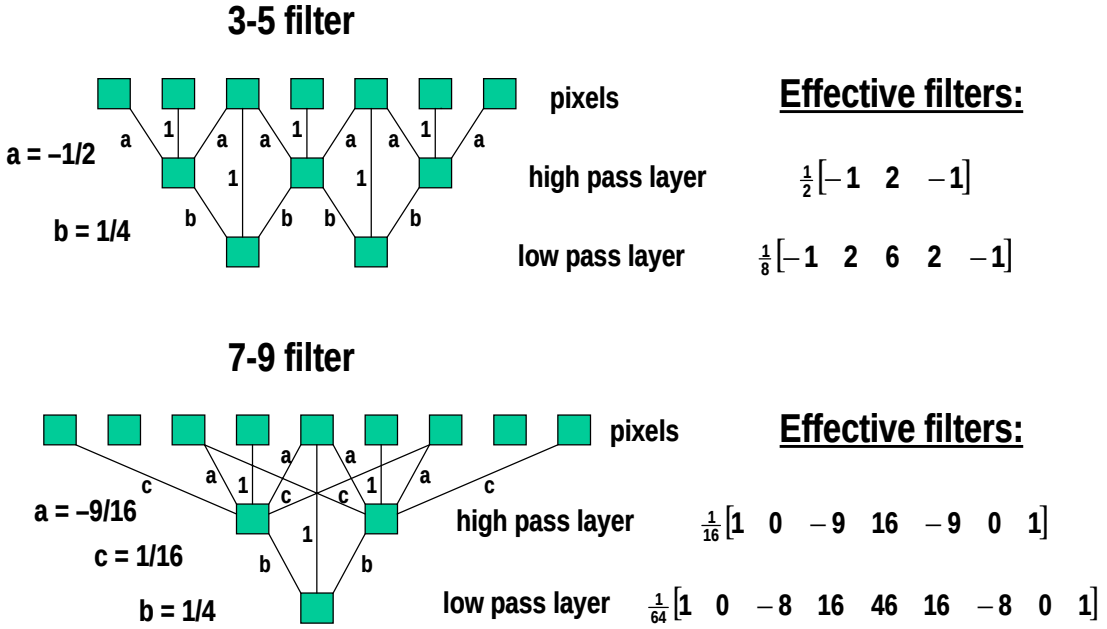

Fig. 2 FlexWave's wavelet filter pairs using lifting technique

This implementation of wavelet filter requires only integer multiplication and bit shifting, so a very efficient processing is possible when applied to digital image data. The absolute reversibility obtained by means of lifting can be demonstrated for the 3-5 filter pair as follows. The forward transformation can be written by means of two lifting steps as

$$h(i) = e(i) + \text{nint}\{-\tfrac{1}{2}[o(i-1) + o(i+1)]\} \quad \text{, and}$$
$$l(j) = o(j) + \text{nint}\{\tfrac{1}{4}[h(j-1) + h(j+1)]\} \quad ,$$

where i and j are locations in the array in original co-ordinates, o and e designate the digital grey values of odd and even pixels, and h and l the high-pass and low-pass outputs, respectively. The first operation produces the high-pass output and should be carried out for all pixels before the low-pass result can be produced. It is possible to overwrite the even pixels with the high-pass output and the odd pixels with the low-pass output (in-place calculation) in order to save memory, but for demonstration purposes it is more clear with separate input and output layers. The inverse transformation is found by simply reversing the above calculation steps, so one obtains

$$o(j) = l(j) - \text{nint}\{\tfrac{1}{4}[h(j-1) + h(j+1)]\} \quad \text{, and}$$
$$e(i) = h(i) - \text{nint}\{-\tfrac{1}{2}[o(i-1) + o(i+1)]\} \quad .$$

As usual with wavelet transformations applied to images, the high and low-pass filters are applied in horizontal and vertical direction, leading to the images LL, LH, HL and HH, at half the original sampling density. Together these four subband images take up the same area as the original image. Repeated application of the wavelet transform to the previous LL image leads to the cascaded algorithm which eventually has a very small LL low-frequency image and for the remainder only high-frequency subbands, mainly edges and local details of small magnitude.

5. DATA FLOW

As mentioned in the introduction, spectral decorrelation by means of PPCA, spatial decorrelation by means of the wavelet transform and quantisation and encoding of the decorrelated data, plus the inverses of these operations, have all been integrated in a single software package. Also the user interface has been incorporated by means of the scripting language TCL/TK. Fig. 3 gives an overview of the various processing steps applied to a hyperspectral input image.

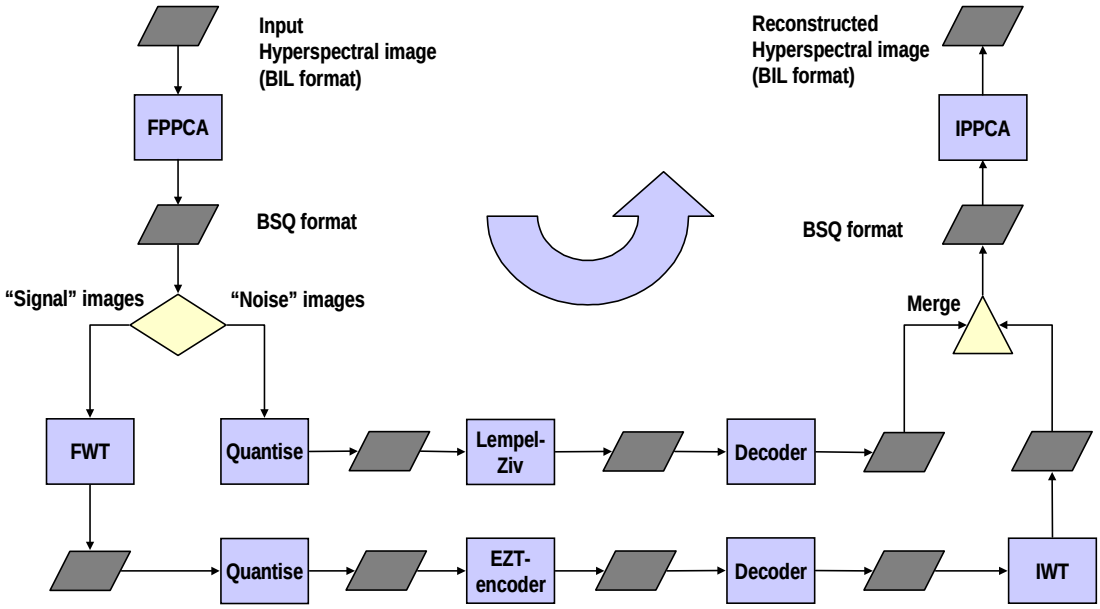

Fig. 3 Data flow in integrated software package

First, the hyperspectral data are decorrelated by means of the forward PPCA transform. During this operation the format is also changed from band-interleaved by line (BIL) to band-sequential (BSQ), which gives separate images for each transformed spectral component. Next, the images are divided in "signal" images and "noise" images, according to a criterion set by the user. For this the spatial correlation between neighbouring pixels is used, as noisy images are featured by a low spatial correlation. The user can specify the number of signal images to retain, or he/she can specify an absolute threshold in terms of the spatial correlation squared. The noise images are linearly quantised and compressed using the standard Unix command "compress", which applies a Lempel-Ziv coder. The signal images are processed using the FlexWave software modules, which include the forward wavelet transform and a quantisation specially adapted to wavelet-transformed images, essentially consisting of the deletion of least significant bitplanes. After quantisation of the wavelet-transformed images an Embedded Zero Tree [4] (EZT) encoder is applied, which exploits the spatial properties of this type of images. Both quantisation and encoding methods lead to binary output files, and the size of these files determines the achieved compression with respect to the input data.

The whole chain of inverse processing steps is also implemented in the package. These steps are the decoding and dequantisation, the inverse wavelet transform, merging of signal and noise images, and finally the inverse PPCA transform. The output consists of the reconstructed hyperspectral image, plus a report of the achieved compression factors and reconstruction errors, for individual spectral channels as well as lumped together for the whole data set. In the case of lossless compression it could be verified whether or not the reconstructed images were really error-free.

6. QUANTISATION STRATEGY

The spectral channels in hyperspectral images vary strongly in dynamic range and signal/noise characteristics. Especially data calibrated in radiance units give high values in the visible part of the spectrum and small values in the ultraviolet and

mid-infrared, just because the solar intensity is smaller in those regions. A uniform quantisation in this case would be inappropriate, because in the weaker parts of the spectrum relatively much information would be lost if this strategy would be followed. Therefore it has been decided to take the quantisation of the spectral information relatively to the radiometric dynamics in the respective channel. As a measure of this the standard deviation is taken. This type of quantisation introduces a constant *relative* quantisation error, and this also means that it better complies with the notion of radiometric noise, which often is defined relatively to the signal as well. Another disadvantage of a uniform quantisation in absolute sense is that after the spectral decorrelation using the cascaded PPCA algorithm the "difference" components have a very weak signal compared to the "sum" components, and in that case uniform quantisation might result in loss of subtle spectral details. However, in order to avoid that strong noisy channels are quantised with too much precision, the quantisation interval is widened according to the noisiness of the image, as measured by the spatial correlation.

For wavelet-transformed channels it is assumed that these are not noisy, and the quantisation interval is derived directly from the standard deviation of the image before the wavelet transform, but adjusted according the normalisation factors in the respective subband images. So in principle the quantisation in the wavelet domain is uniform with respect to spatial frequency, except that the DC image - which is the LL image after a number of resolution steps - is not quantised at all. FlexWave provides for JPEG compression as to this part of the information, but in the implementation at NLR it was bypassed, as the LL image is small anyway.

7. RESULTS FOR AVIRIS HYPERSPECTRAL IMAGES

The evaluation carried out was aimed at determining the achievable compression factors under lossless and quasi-lossless conditions. In addition, the relative importance of certain software components and parameter settings was briefly investigated. Execution speed was considered less important in this stage, but some indications could nevertheless be obtained. For the evaluation two hyperspectral data sets were used, free-of-charge AVIRIS hyperspectral images from Jasper Ridge, California and Cuprite, Nevada, both acquired in 1994. The Jasper Ridge image (Fig. 4) contains a mixture of vegetation (forest, natural vegetation) and urban features. Cuprite (Fig. 6) is a famous geological site without vegetation but a wealth of surface minerals and soil types. Both images have been radiometrically calibrated by JPL, and the digital brightness numbers are 16-bit signed integers, expressing the spectral radiance times 500. The 224 spectral bands of each image are co-registered and each measures 512 rows by 614 columns. Before the application of the data compression software, the number of columns has been reduced to 512 in order to avoid possible problems with image sizes that are no power of two. Fig. 5 shows the standard deviations in all spectral bands of the Jasper Ridge image before and after application of spectral decorrelation by means of PPCA. This clearly demonstrates the strong reduction of variances achieved by the spectral decorrelation. Also the shape of the solar spectral irradiance curve is clearly reflected in the standard deviation spectrum. Similarly, for the Cuprite area this is shown in Fig. 7. For Cuprite the standard deviations are smaller than for Jasper Ridge, and because this area contains no vegetation, spectral correlations are in general very high, giving very small variances of the PPCA transformed data.

Table 1 summarises the compression results obtained in all numerical experiments applied to both hyperspectral images. In this table the columns under "noise" give the results for the case that all spectral bands (original or transformed) are considered as noise, implying that for all bands the Lempel-Ziv compression is used, "signal" marks the case that all bands are processed by the FlexWave software, and "30 sig" means that the 30 bands with the highest spatial correlation are processed by FlexWave and the remainder by Lempel-Ziv, using the Unix "compress" command. In Table 1 it is also indicated which wavelet filter was used, whether or not the spectral decorrelation by means of PPCA was applied, and the error tolerance in percent of the standard deviation of the input band. Here an error tolerance of zero represents the lossless case.

The results obtained and shown in Table 1 represent about 150 hours of processing time on a Sun UltraSparc Server. If all bands were processed by means of FlexWave, one session took 5 hours, in case of 30 signal bands it took about one hour, and if all bands were compressed with the Unix "compress" command, it took about half an hour. Timings during a session showed that Unix "compress" was about 15 times faster than FlexWave. The PPCA spectral decorrelation algorithm is so fast, that the difference between with (five decorrelation steps) and without PPCA (zero decorrelation steps) is hardly noticeable. Already the calculation of the root mean squared errors in all bands between reconstructed and original data took more time.

Comparing the figures in Table 1 one can see that data compression is always possible, also in the reversible (zero error tolerance) case, although the obtained compression factor is not high. It varies from 1.38 for Jasper Ridge, no spectral trans-

form, and all bands compressed with Lempel-Ziv, to 2.29 for Cuprite, spectral transform and all bands processed with FlexWave using the 3-5 wavelet filter.

Table 1. Compression factors obtained for various combinations of input conditions

Error tolerance	Spectral transform	Wavelet filter	Jasper Ridge			Cuprite		
			noise	30 sig	signal	noise	30 sig	signal
0%	N	9-7	1.38	1.44	1.79	1.50	1.60	2.01
		3-5		1.44	1.79		1.60	2.01
	Y	9-7	1.98	2.06	2.19	2.13	2.21	2.27
		3-5		2.06	2.20		2.21	2.29
1%	N	9-7	2.41	2.47	2.42	2.16	2.29	2.70
		3-5		2.46	2.34		2.27	2.58
	Y	9-7	2.92	3.01	2.33	3.07	3.15	2.35
		3-5		2.99	2.32		3.14	2.35
2%	N	9-7	3.07	3.14	2.83	2.74	2.90	3.23
		3-5		3.10	2.70		2.87	3.01
	Y	9-7	3.89	3.95	2.48	4.07	4.12	2.46
		3-5		3.91	2.44		4.09	2.43

In most cases the wavelet filter type does not make much difference. In the cases where it does, this is probably related to coincidental factors, such as a just more favourable transformed data range for either of the two filters.

In the reversible (lossless) case it turns out that the combination of spectral decorrelation by means of PPCA and the spatial wavelet transform of FlexWave applied to all spectrally transformed bands gives the highest compression factors, about 2.2 for Jasper Ridge and 2.3 for Cuprite. If only the 30 signal bands with the highest spatial correlation are processed by Flex-Wave, and the remaining noisy bands by Lempel-Ziv, then there is only a small decrease in the achieved compression ratio in the case of PPCA-transformed data, and a much greater decrease if this spectral transform is omitted.

Quasi-lossless compression, with tolerated errors of 1% and 2% of the standard deviation of the input images, of course gives higher compression ratios than lossless compression, but for Lempel-Ziv coding this increase is much stronger than for the EZT-coder of FlexWave. However, this is partly explained by the fact that also the actual errors were greater in the case of Lempel-Ziv coding, because for noisy images a less accurate quantisation was applied.

Comparing the cases “noise”, “30 sig” and “signal” in Table 1, one can observe that for quasi-lossless compression the case “30 sig” always gives the highest compression ratios, and that applying the PPCA spectral decorrelation always increases the compression ratio. So the idea of processing only the least noisy images by FlexWave and the remainder by a much simpler method gives optimum results, especially when combined with the PPCA spectral decorrelation, and overall proc-essing time is greatly reduced. One might wonder why the case “30 sig” gives better results than either of the other two cases, and why it would not end up somewhere in between. This can be explained as follows, however. For noisy bands the compression ratio achieved by means of Lempel-Ziv is higher than by means of FlexWave. For the signal bands however the reverse is true. Now the case “30 sig” will be better than “signal” because the noisy bands are compressed with the method that is more suitable for them (Lempel-Ziv), and “30 sig” will also be better than “noise”, because in that case the signal bands are compressed by the then more suitable FlexWave method. A worse result in the case of “30 sig” would only be possible if many times the least efficient compression method had been chosen. However, this is not very likely to hap-pen. In one sample, of the 30 “signal” bands, 27 had a higher compression ratio when compressed with FlexWave, and only three when compressed with Lempel-Ziv, so in 90% of the cases the right choice has been made.

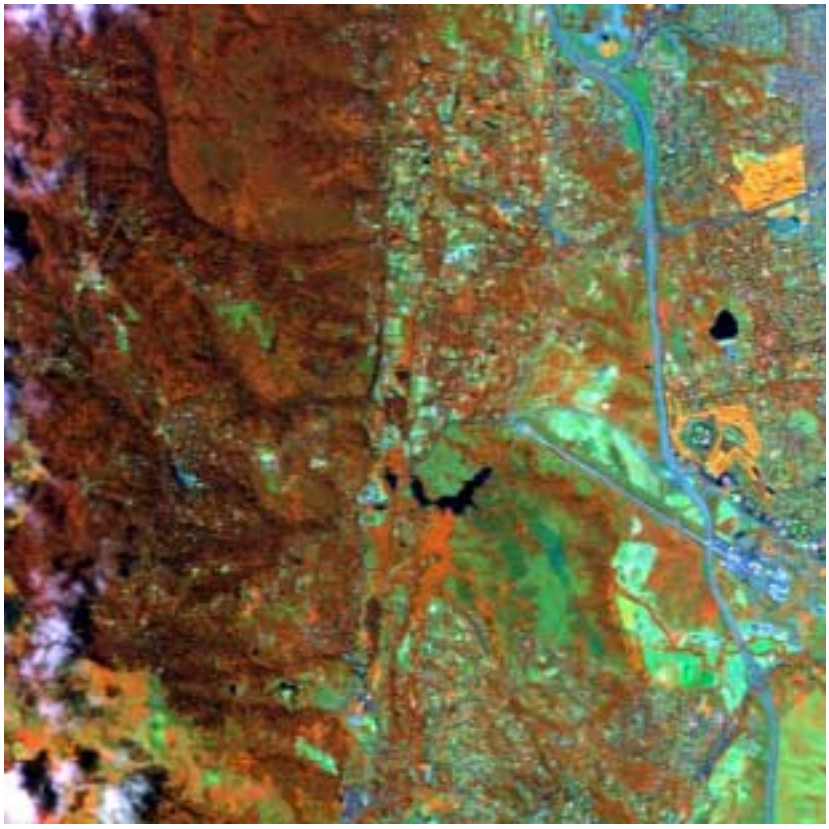

Fig. 4 Jasper Ridge image

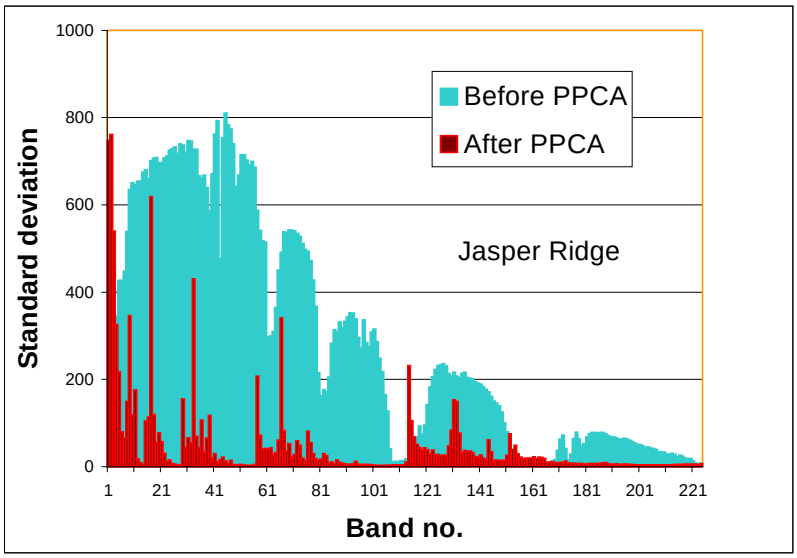

Fig. 5 Jasper Ridge standard deviations before and after PPCA

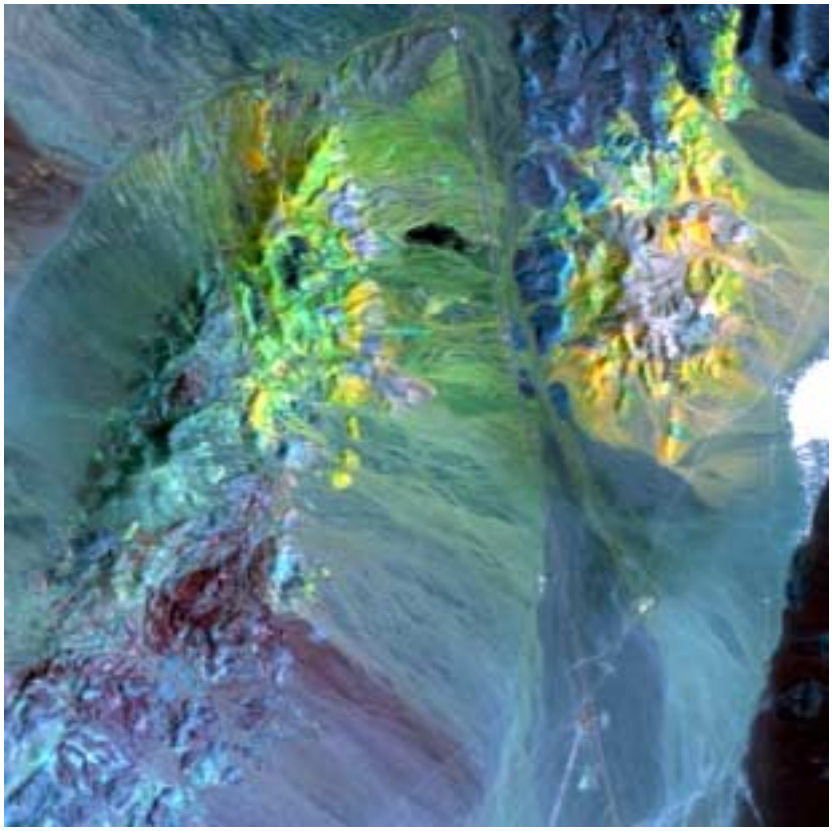

Fig.6 Cuprite image

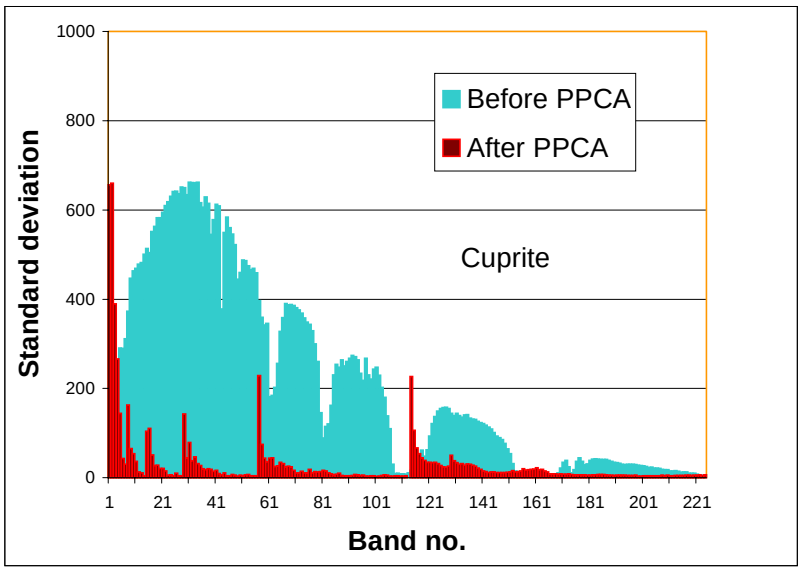

Fig. 7 Cuprite standard deviations before and after PPCA

8. CONCLUSIONS

The FlexWave wavelet compression software package from IMEC has been successfully integrated with software developed by NLR for the spectral decorrelation of hyperspectral image data. The integrated package is controlled by means of a graphical user interface also developed at NLR. This should allow the evaluation of lossless and quasi-lossless image data compression methods by potential users of imaging spectrometer data.

Pairwise Principal Components Analysis (PPCA) is a very effective method of spectral decorrelation of hyperspectral image data. It can be applied in a reversible form using lifting techniques, it is very fast, and close to optimum in terms of achieved decorrelation.

Quantisation in the FlexWave software has been realised successfully, by taking into account the normalisation factors associated with each wavelet filter and each subband image. In this way the tolerated reconstruction error specified by the user is actually approximated reasonably well.

For noisy images the much simpler Lempel-Ziv coder achieves higher compression ratios than does FlexWave. After spectral decorrelation many noisy images result and these can better be compressed by means of Lempel-Ziv. This also saves processing time, as FlexWave was found very slow.

FlexWave and PPCA form a good combination, as they achieve spatial as well as spectral decorrelation, and if the PPCA-transformed bands with the highest spatial correlation are compressed using FlexWave, and the more noisy bands using Lempel-Ziv, the highest compression ratios are obtained.

From several numerical image data compression experiments applied to two AVIRIS hyperspectral image scenes (Jasper Ridge and Cuprite), it could be concluded that the results were not very much different between both scenes, despite the large differences in landscape. Also the wavelet filter type did not make much difference.

The obtained compression ratios varied from about two for lossless compression to about four for an error tolerance of 2% of the input standard deviation.

ACKNOWLEDGEMENTS

The results presented in this paper have been obtained as part of the ISAC project, under ESTEC Contract No. 11964/NL/FM/(SC). The AVIRIS images were provided free-of-charge by Jet Propulsion Lab., Pasadena, USA.

REFERENCES

1. W. Sweldens, “The lifting scheme: a new philosophy in biorthogonal wavelet constructions”, in *Wavelet Applications in Signal Processing III,* A.F. Laine and M. Unser (eds.), pp. 68-79, 1995.
2. A.R Calderbank, I. Daubechies, W. Sweldens and B.-L. Yeo, “Wavelet transforms that map integers to integers”, Preprint, Dept. of Mathematics, Princeton Univ., August 1996.
3. I. Daubechies and W. Sweldens, “Factoring wavelet transforms into lifting steps”, Preprint, Bell Laboratories, Lucent Technologies, September 1996.
4. J. Shapiro, “Embedded image coding using zerotrees of wavelet coefficients”, *IEEE Trans. on Signal Proc.,* 41, No. 12, pp. 3445-3462, 1993.