CONF-970465--12

Methodology for hyperspectral image classification using novel neural network

Suresh Subramanian[a], Nahum Gat[a], Michael Sheffield[a], Jacob Barhen[b], Nikzad Toomarian[c]

[a]Opto-Knowledge Systems Inc. (OKSI), 1737 3rd Street, Manhattan Beach, CA 90266
[b]Oak Ridge National Laboratory, Bethel Road, Bldg. 6010/214, Oak Ridge, TN 37831
[c]Jet Propulsion Laboratory, 4800 Oak Grove Drive, Pasadena, CA 91109

ABSTRACT

A novel feed forward neural network is used to classify hyperspectral data from the AVIRIS sensor. The network applies an alternating direction singular value decomposition technique to achieve rapid training times (few seconds per class). Very few samples (10-12) are required for training. 100% accurate classification is obtained using test data sets. The methodology combines this rapid training neural network together with data reduction and maximal feature separation techniques such as principal component analysis and simultaneous diagonalization of covariance matrices, for rapid and accurate classification of large hyperspectral images. The results are compared to those of standard statistical classifiers.

Keywords: Neural Networks, Hyperspectral Image Classification, Feature/Class Separation[1].

1. INTRODUCTION

An important issue associated with classifying hyperspectral images is the large size of data produced by current hyperspectral imaging systems. While sensors like LANDSAT's TM produce data cubes with only 7 bands per pixel, more recent sensors record the spectra of individual pixels with increasing spectral resolution. Sensors like AVIRIS, HYDICE and the TRW Hyperspectral Imager (HSI) measure pixel spectra in 224, 210 and 348 bands respectively. Ultraspectral sensors based on Fourier transform spectrometers, perform measurements at even higher resolution. Using such high resolution data for image classification increases the dimensionality of the problem, and hence the complexity and computation time, significantly.

The advantages of neural network (NN) based approaches for classifying hyperspectral images have been recognized for a while.[1-6] NNs are considered to be powerful classification tools because of their nonlinear properties and the fact that they make no assumptions about the distribution of the data. This feature is useful in cases where no simple phenomenological model exists to accurately describe the underlying physical process that determines the data distribution. Yet the use of NN for hyperspectral image classification has been limited primarily due to the inordinately long time required to train NN. During training, feed forward networks use a gradient descent method for least squares error back propagation.[7] Recently, radically different approaches to error minimization have been considered that employ a Householder transform combined with QR factorization or other iterative schemes.[8-11] The NN used in this work is a feed forward type of network that builds on the least squares paradigm. However, it introduces solutions involving a sequence of alternating directions singular value decompositions (ADSVD) for error minimization, which drastically reduces the error convergence time.[12] As a result, rapid training of the NN on large hyperspectral data sets is now possible.

---

[1] Further author information -
[a] S.S. (correspondence): Email - suresh.subramanian@worldnet.att.net, Tel: (310) 372 6665
[a] N.G., M.S., Email - oksi@cerfnet.com, Tel: (310) 372-6665
[b] J.B. , Email - barhenj@ornk.gov, Tel: (423) 574-7131
[c] N.T., Email - nikzad.toomarian@jpl.nasa.gov, Tel: (818) 354-7945

DISCLAIMER

In addition to the rapid convergence property of the ADSVD NN, training time is significantly decreases by using data reduction schemes. Techniques like principal component analysis and simultaneous diagonalization of covariance matrices decrease the training time by reducing the problem dimension (by factor of 20 to 50). They also improve classification accuracy by enhancing the separation between classes. A substantial reduction in the training time of feed forward networks is also attained by utilizing the concept of sub-networks. The idea here is to train a single network to identify one particular class only instead of using a one network to identify all classes. Taken together, the data reduction and sub-network schemes not only reduces the training time drastically but also improve the classification accuracy.[13] This combined methodology of the ADSVD NN together with data reduction/feature separation and using the sub-network concept yielded excellent results. For a limited test set selected from the Moffett Field image acquired by the AVIRIS sensor (224 bands), we achieved extremely rapid training times (few seconds per class) and 100% classification accuracy, using no more than a dozen pixels/class for training. All computations were performed on a PC platform (200 MHz Pentium Pro with 64 Mbytes RAM).

The next section discusses the transformation techniques employed in this work. Following that, we present the ADSVD NN classification results of the Moffett field data cube acquired using the AVIRIS sensor. Our results are then compared with those of conventional classifiers available in remote sensing software tools like ENVI.

2. ANALYSIS

Principal component analysis (PCA) is a linear coordinate transformation technique to represent a data set in a reference frame where the variables (spectral bands in our case) are no longer correlated.[14-16] The process of determining the new coordinate axes involves diagonalizing the covariance matrix of the data. Since the off diagonal elements of this matrix measure the correlations between bands, we compute a coordinate transformation in which these terms are zero; i.e. a transformation that diagonalizes the covariance matrix. From linear algebra, there exists an orthonormal transformation capable of diagonalizing the covariance matrix as it is symmetric.[17,18]

The new axes (or principal components) are organized so as to indicate the directions of decreasing variance in the data. Hence by retaining only the first few principal axes (corresponding to directions of maximum variance) to represent the data and discarding the rest, we obtain not only maximal feature separation but also a substantial reduction in the dimensionality of the data set. Physically, the reduced dimensionality means that in order to identify K distinct classes of objects with spectra in N bands (where $N >> K$), we require at the most K distinct coordinate axes, and often fewer than K. PCA computes these axes.

With data classification as the final goal, the covariance matrix is computed using all the pixels from the selected regions of interest (ROIs) for each class. Note that each pixel represents an N dimensional vector x (N = # bands = 224 for the AVIRIS sensor). Let K be the number of classes, M_k be the number of pixels in class k and M be the total number of pixels where $M = \sum_{k=1}^{K} M_k$. The mean vector of this set, m is denoted by

$$m = E(x) = \frac{1}{M} \sum_{i=1}^{M} x_i \qquad (1)$$

where $E(x)$ is the expectation of x. The covariance matrix of the data set (ROIs), Σ_x, is given by

$$\Sigma_x = E\{(x-m)(x-m)^T\} = \frac{1}{M} \sum_{i=1}^{M} (x_i - m)(x_i - m)^T \qquad (2)$$

Diagonalizing this covariance matrix gives the required transformation matrix, G. The eigenvectors (columns of G) corresponding to the largest few eigenvalues correspond to the axes of maximal variance. The data show very little variance (i.e. no useful separation amongst classes) along the remaining principal directions (eigenvectors). Hence data along these axes are not useful for classification. The transformed data may therefore be computed after zeroing all but the first few eigenvectors:

$$y = G^T(L)x \qquad (3)$$

where L is the number of eigenvectors retained in the transformation. Hence the dimension of a transformed pixel, y, is L and the data reduction factor is N/L.

An important point to note about PCA is that while it computes the axes of maximum overall variance, there is no guarantee that it will actually increase the separation between a particular pair of classes. It only guarantees to maximize the overall variance given by $\sigma_x^2 = G\Sigma_x G^T$. In order to ensure maximum separation between classes, the function to be maximized must be sensitive to the class structure. Hence a suitable function would be one that not only maximizes the variance amongst classes but also simultaneously minimizes the variances within all the classes. One such a function is given by [14,19,20]

$$\sigma^2 = \frac{\sigma_A^2}{\sigma_W^2} = \frac{\sigma_A^2}{\sum_{k=1}^{K} \sigma_k^2} \tag{4}$$

σ_A and σ_W are the amongst class and within class variances respectively and σ_k is the variance of class k. These terms are related to their corresponding covariance matrices Σ_A and Σ_W by

$$\sigma_A^2 = G\Sigma_A G^T \quad \text{and} \quad \sigma_W^2 = G\Sigma_W G^T \tag{5}$$

where G is the transformation matrix that diagonalizes both Σ_A and Σ_W. The detailed expressions for the Σ's are

$$\Sigma_A = E\{(m_k - m_0)(m_k - m_0)^T\} = (\tfrac{1}{K}) \sum_{k=1}^{K} (m_k - m_0)(m_k - m_0)^T \tag{6}$$

$$\Sigma_W = \{\sum_{k=1}^{K} (M_k - 1)\, \Sigma_k / M\} \quad \text{where}$$

$$\Sigma_k = E\{(x_k^i - m_k)(x_k^i - m_k)^T\} = \left(\tfrac{1}{M_k}\right) \sum_{i=1}^{M_k} (x_k^i - m_k)(x_k^i - m_k)^T$$

m_0 is the overall means of all the pixels, m_k is the mean of class k and x_k^i is the i^{th} pixel belonging to class k. The expression in equation (4) is maximized by setting

$$\frac{d\sigma^2}{dG} = 0 \qquad \Rightarrow \quad (\Sigma_A - \sigma^2 \Sigma_W)G = 0 \tag{7}$$

This is a generalized eigenvalue (GEV) equation where σ^2 are the eigenvalues and the matrix G contains the eigenvectors of the transformation. The constraint $G\Sigma_W G^T = I_{NxN}$ (NxN identity matrix) renders the distribution of all the classes spherical. This process also known as 'whitening' is only useful when the distributions of various classes deviate significantly from a Gaussian. If for example the members of each class are affected by random gaussian noise only, they will have a spherical distribution to start with. Hence in this case, a GEV transformation will not separate the classes better than the PCA. Lastly, for K classes we obtain $K-1$ non zero eigenvalues from the GEV and hence $K-1$ distinct axes (provided N > K). The eigenvectors corresponding to the remaining N-K zero eigenvalues belong to a degenerate orthogonal subspace.[17,18]

3. RESULTS

3.1 Maximum Feature Separation

Figure 1 shows an image of the Moffett field data cube recorded using the AVIRIS sensor in 224 bands (0.4 μm to 2.4 μm spectral range). This data are 12 bit digital numbers without any atmospheric corrections for absorption and scattering. Eight regions of interest (ROIs) are marked in the image for classification (table 1) Five of these regions (1, 4, 5, 6, 7) are different water bodies ranging from clear water (1) to an evaporation pond (7). The water bodies in between have different levels of clarity (due to different amounts of particle suspension). Region 3 corresponds to concrete structures. Regions 2 and 8 are dense and sparse vegetation. Figure 2 shows the average spectrum of each of the ROIs marked in figure 1. The spectra of some of the ROIs, for example regions (2, 8), (4, 7) and (5, 6) look very similar. The entire image is classified to identify pixels belonging to these 8 regions (classes). Similar classes of pixels have deliberately selected to make the classification problem difficult. Since the average spectra of the

pixels from these classes look very similar, any classification in the spectral domain will most likely produces significant errors. Classification is therefore performed in the PCA and GEV transformed domains.

Figure 3 illustrates the advantage of classifying the image in the PCA or GEV transformed feature space, where we show the average 'transformed spectra' of the 8 ROIs in these domains. Only 8 bands are retained in the PCA case and 7 for the GEV. The choice of 8 bands for the PCA transform is motivated by the fact that there are only 8 classes while for the GEV we retain only 7 bands because there are only 7 (K-1 bands where K = 8) non-zero eigenvalues arising from the simultaneous diagonalization process. We obtain a data reduction factor of about 30 from these transformations. For classification, we need only the first 4-5 bands where the classes are separated. The last few bands have been retained to illustrate the fact that the variance amongst classes is negligible along these axes.

Both the PCA and the GEV transforms enhance the separation between classes as compared with the 224 band spectral space (figure 2), but the interclass separations are clearly greater using the GEV transformation (figure 3(b)). This is confirmed by calculating the average interclass separations for the spectral, PCA and GEV spaces using the angular distance and Fisher discriminant distance as measures (table 2). The columns labeled 1-8 indicate the average distance between class i and all other classes. The angular separation between classes increases dramatically from the spectral to the GEV domain where the pixels are nearly collinear in the former and highly separated in the later (90° separation is orthogonality condition). The Fisher distance is given by the ratio $\frac{d_{ij}}{\sqrt{\sigma_i^2+\sigma_j^2}}$ where d_{ij} is the Euclidean distance between classes i and j and the σ's are their respective variances, all calculated in the appropriate domain. The Fisher distances for all classes but 2 and 3 are greater for the GEV than the PCA. The overall average (last column) also clearly indicates that the GEV transformation affords greater class separation than the PCA. A direct outcome of this is the improved classification results by the NN obtained after training on the GEV transformed data than on the PCA transformed set.

3.2 Image Classification by ADSVD Neural Network

The implementation details of the ADSVD NN are discussed elsewhere.[12] For the present it suffices to state that it has as many input nodes as the number of bands of the input data (8 for PCA and 7 for GEV), one hidden layer and a single output node. The approach we adopt is to compute the PCA and GEV transformation matrices using the ROI pixels. The entire data cube is then transformed using the appropriate transformation. NN classification is then performed using the PCA and GEV transformed data cube. We also employ 8 sub-networks for training and classification. Each of which is trained to identify one particular class only and reject the rest.

Without independent ground truth to validate the NN classification results, a small fraction of the ROI pixels were used to train the networks. The remaining pixels from the ROIs were then used to test the classification performance of the NN. Specifically, 8-12 randomly selected pixels from each class were used to train the NNs. As a result, a reasonable number of pixels were left in the ROIs to test the network classification accuracy.

Table 3 shows the results of testing the NN classifier on the ROIs in the PCA and GEV domains. The classification threshold is set at 0.5 (middle range of NN's logistical activation function). A single pixel is processed by each network and a 'winner take all' scheme is employed to assign a pixel to a particular class. A pixel is 'correctly' classified if it's network activation level is the highest of all networks and exceeds the threshold. A 'miss' occurs when a pixel gets incorrectly classified. Lastly, a pixel is 'unclassified' if it's NN has the highest output level of all networks but it is still less than the threshold. The sum of the correct, missed and unclassified columns equals the total number of pixels per class. A missed pixel in one class appears as a falsely classified pixel elsewhere. Hence the sum of the 'missed' column equals that of the 'falsely' classified one.

The PCA transformed data are correctly classified for all but classes 2, 5 and 6. There are a few misses in classes 2. However, most pixels in class 6 are unclassified and a large number of pixels from class 6 are incorrectly classified into 5. On the other hand, the NN classifies the GEV transformed data with 100% accuracy. This is not surprising that the different classes are very well separated in the GEV transformed feature space as is depicted in figure 3(b) and table 2.

Table 4 show the results of the entire Moffett Field image classified by the NN in the PCA and GEV transformed domains using a threshold level of 0.5. The classified images are not included as they are not useful for viewing without full color representation.[21] While the overall spatial structure of the image is reconstructed in both the PCA and the GEV classified images, table 4 indicates that the NN identifies more pixels in each class with the GEV transformed cube than with the PCA transformed one. The NN barely detects any pixels in class 6 from the PCA cube while in the GEV cube, it assigns over 6% of the pixels to this class. From figure 1, it is apparent that there is a considerable number of pixels belonging to class 6 in the image. The poor NN result for class 6 on the PCA transformed cube is consistent with the previous result obtained while verifying NN performance on the PCA transformed ROI's, where also the NN failed to detect class 6. Overall, the NN classifies about 50% of the GEV transformed pixels whereas it classifies only 30% of the pixels in the PCA case. The unclassified pixels are concentrated in the central portion of the image in both cases indicating the need to add more training classes from to this region.

Lastly, we note that the total computation time for the PCA/GEV transformation on the entire cube is about 10 minutes. The NN takes about 1 minute per class to process the entire cube. Training times is insignificant.

3.3 Supervised ENVI Classification

ENVI classifiers like the parallelepiped (PP), minimum distance (Min-D) and spectral angle mapper (SAM) are applied to the AVIRIS cube in the PCA transformed domain, using the same ROI's for training as before. In order to compare the ENVI results with those from the NN, the ENVI classification thresholds are adjusted till approximately 50% of the total number of pixels are classified (which is the number of pixels classified by the NN using the GEV transformed cube). The lack of ground truth makes it difficult to make absolute performance comparisons but some of obvious problems of each method can nevertheless be identified.

Table 5 summarizes the results of the ENVI classifiers. The PP and Min-D methods assign about 30% of the pixels to class 1. However, a large number of pixels assigned to this class are from the central region of the cube where no ROIs were selected. In contrast, the NN classifications of the PCA and GEV cubes for class 1 take a middle ground with 6% and 14% respectively. Further, the NNs leave the central regions of the image as 'unclassified'. The SAM assigns a negligible number of pixels to class 1, which is clearly incorrect. In class 2, the results from all classifiers are more or less in agreement. In class 3 (concrete), the Min-D classifier detects very few pixels whereas it is clear from the image that a large number of pixels belong to this class. The other statistical classifiers tend to agree with the NN assignments in this case. With classes 4 and 5, all the classifiers (NN and ENVI) assign more pixels to 4 than to 5 although the ratio of pixels assigned to the two classes varies. The SAM violates this trend and assigns over 3 times as many pixels to class 5 (20%) than to class 4 (6%). With class 6, the ENVI classifier more or less agree with the NN results from the GEV cube. The NN fails to detect class 6 in the PCA transformed cube. Min-D identifies very few elements belonging to classes 7 and 8 in the image while PP and SAM identify larger numbers. The latter's results are consistent with those of the NN.[21]

ENVI takes about 1-2 minutes to classify the image in the PCA domain. The PCA transformed cube (8 bands) was created externally before feeding it ENVI. Classification in the spectral domain takes 10-15 minutes depending on the distance measure.

4. SUMMARY

In this paper we have demonstrated the advantage of the GEV technique for class separation and its effect on classification accuracy. The ADSVD NN which has a very rapid error convergence rate facilitates extremely fast training of hyperspectral data sets. This NN is also highly robust in that it requires very few samples (10-12) to encode the features of a particular class. Training the NN in the GEV transformed space not only speeds up training by reducing the data dimension (by a factor of 30-50) but also improves classification accuracy by maximizing class separation. The use of sub-networks also contributes to improved classifcaton accuracy and reduced training time. All the above techniques lead to a methodology that makes the NN a very fast, robust and practical tool for supervised hyperspectral image classification.

Finally, more work needs to be done in terms of validation and comparison of the various techniques. While absolute comparison of ADSVD NN with statistical classifiers is only possible with ground truth, we have qualitatively identified some obvious cases where the statistical classifiers produce erroneous results. Previous studies using synthetic hyperspectral data also verified that the ADSVD NN outperformed the statistical classifiers.[13,21]

ACKNOWLEDGMENTS

The authors thank Dr. Risa Wu from NASA Stennis Space Center (Mississippi) for providing the AVIRIS data set and the ROI's discussed in this paper. We also thank Dr. Robert Cromp, the NASA program manager who supervised the NASA STTR Phase I contract (# NAS5-32935, Jan-June 1996) under which this work was done.

REFERENCES

1. P. Blonda, *et al*, "Feature extraction and pattern classification of remote sensing data by a modular neural network", *Optical. Eng*, **35**, 536 (1996).
2. T. Yoshida, S. Omatu, " Neural network approach to land cover mapping", *IEEE Trans. Geosci. and Remote Sensing*, **32**, 1103 (1994).
3. Y.C. Tseng *et al*, "A dynamic neural network for remote sensing applications", *IEEE Trans. Geosci. and Remote Sensing*, **32**, 1096 (1994).
4. J.A. Bendicktsson, P.H. Swain, O.K. Ersoy, "Conjugate gradient neural networks in classification of very high dimensional remote sensing data", *Int. Jnl. Remote Sensing*, **14**, 2883 (1993).
5. R. Li, H. Si, " Multi-spectral image classification using improved backpropagation neural networks, *Proc. IEEE IGARSS-92*, **2**, 1078 (1992).
6. J.A. Bendicktsson, P.H. Swain, O.K. Ersoy, "Neural network approaches versus statistical methods of classification of multisource remote sensing data", *IEEE Trans. Geosci. and Remote Sensing*, **28**, 540 (1992).
7. J.L. Mcclelland, D.E. Rumelhart, *Parallel Distributed Processing*, Vol. 1, and 2, MIT Press (1988).
8. M. Azmi, R. Liu, "Fast learning of multilayer neural networks using recursive least squares", *IEEE Trans. Signal Proc.*, **40**, 446 (1992).
9. F. Biegler-Konig, f. Barmann, "Learning algorithm for multilayered neural networks based on linear least squares", *Neural Networks*, **6**, 127 (1993).
10. B. Cetin, J. Barhen, J. Burdick, "Terminal repeller unconstrained subspace tunneling (TRUST) for fast global optimization", *Jnl. Optmization Theory and Applications*, **77**, 97 (1993).
11. J. Barhen, "Learning without local minima", *IEEE World Congress on Computational Intelligence*, **7**, 4592, (1994).
12. J. Barhen, N. Gat, "Rapid convergence of feed forward multilayered neural networks using alternate direction singular value decompositions", (manuscript in preparation).
13. N. Gat, S. Subramanian, M. Sheffield, N. Toomarian, J. Barhen, NASA STTR Phase I Program Final Report (contract # NAS5-32935), Mission to planet earth, "Novel neural network technology for very fast analysis of hyperspectral imagery", (Opto-Knowledge Systems Inc., June1996).
14. I.T. Jollife, *Principal Component Analysis*, Springer Verlag New York, (1986).
15. J.A. Richards, *Remote Sensing and Digital Analysis*, Springer Verlag, 2nd ed., (1995).
16. R.A. Schowengerdt, *Techniques for Image Processing and Classification in Remote Sensing*, Academic Press (1983).
17. G.H. Golub, C.F. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press (1996).
18. R. Horn, C.R. Johnson, *Matrix Analysis*, Cambridge University Press (1985)
19. C.W. Therrein, *Discrete Time Signals and Statistical Signal Processing*, Prentice Hall, (1992).
20. T. Parsons, *Voice and Speech Processing*, McGraw-Hill, (1987).
21. For latest updates on this work please see the "Hyperspectrum Newsletter" at the following web-site: (http://www.techexpo.com/WWW/opto-knowledge/hyperspectrum).

Class #	Pixels/ Class	Material
1	318	clear water
2	221	dense vegetation
3	24	concrete
4	149	water with suspended solid material
5	179	water (not as clear as 1)
6	183	water with suspended solid material (not as much as 4)
7	212	evaporation pond
8	107	sparse vegetation

Table 1: ROIs for classification.

Figure 1: Moffett field data cube recorded by AVIRIS sensor in 224 spectral bands covering the range 0.4 μm to 2.4 μm. The RGB composite was created using bands 45, 31 an 18 respectively. The patches alongside the numbers are the 8 regions of interest.

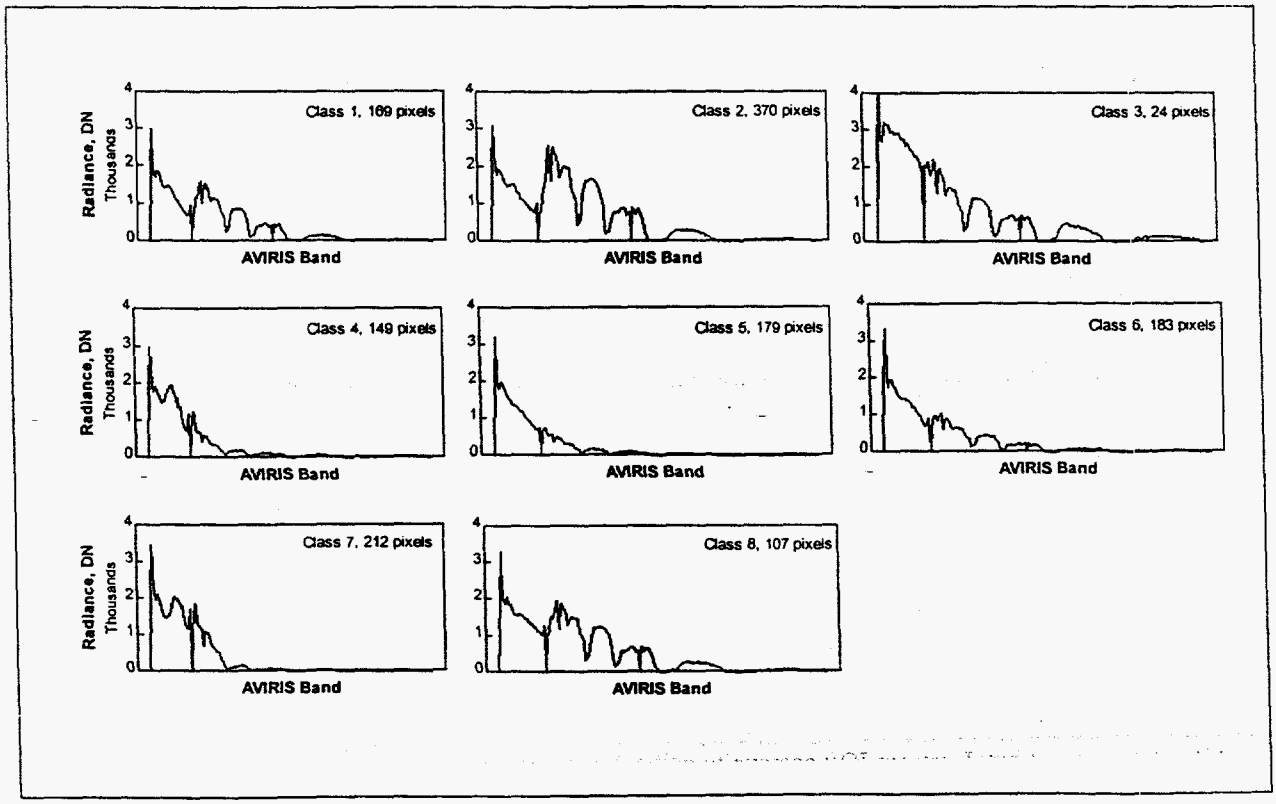

Figure 2: Average spectrum of each class created using ROI pixels.

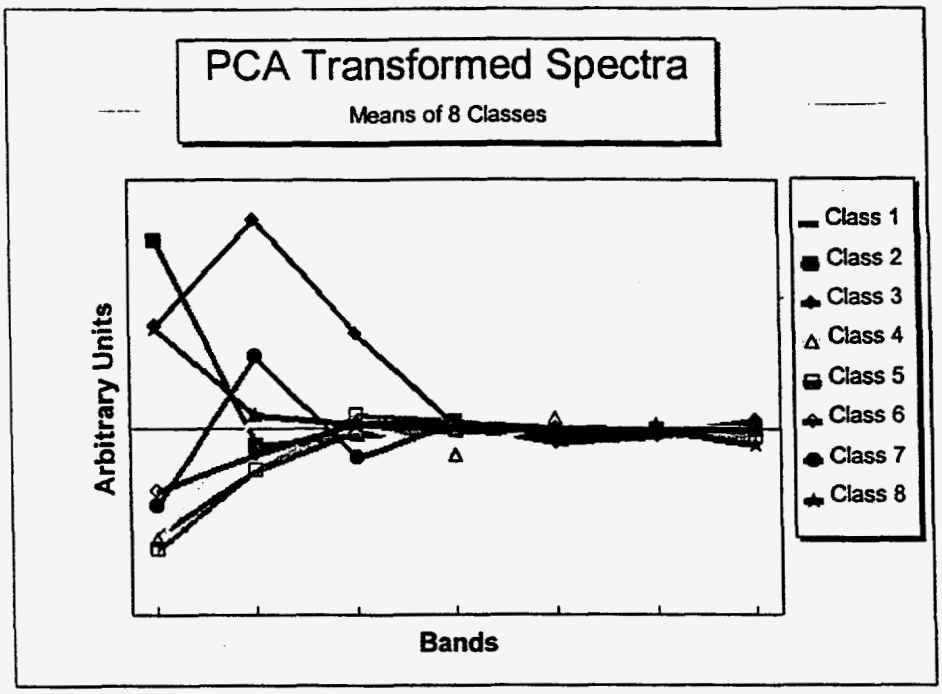

Figure 3(a): Principal component representation of average ROI spectra.

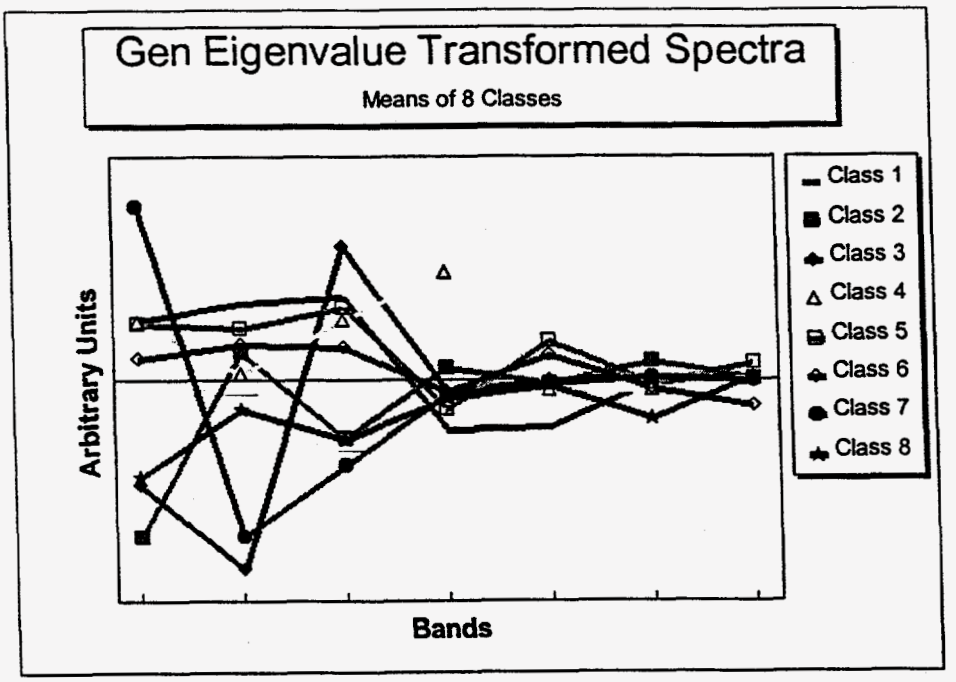

Figure 3(b): Generalized eigenvalue transform representation of average ROI spectra. Interclass separation is better than with PCA and all classes look distinct.

Class #	1	2	3	4	5	6	7.00	8.00	Overall Avg
Average Angular Distance (degrees)									
SPECTRAL	19.70	31.92	18.33	21.21	21.82	16.56	20.53	23.60	21.71
PCA	32.75	33.25	67.89	37.32	33.02	33.93	56.18	30.74	40.64
GEV	57.79	55.67	75.09	68.43	55.22	57.31	71.26	52.35	61.64
Average Fisher Distance (dimensionless)									
SPECTRAL	86.98	154.83	83.41	78.99	80.03	60.80	99.84	80.06	90.62
PCA	122.10	203.19	126.01	104.56	105.35	75.99	136.78	102.80	122.10
GEV	162.65	198.15	56.55	165.26	140.00	128.41	210.36	111.69	146.63

Table 2: Class separation of the average ROI spectra measured using the angular separation and Fisher discriminant distance measures in the spectral, PCA and GEV domains. The separations increase as we go from spectral to GEV domains.

Classification Domain	Class #	Total # Pixels Per Class	Correct	Missed	UnClassified	False Alarms
PCA	1	169	169	0	0	4
	2	370	366	4	0	0
	3	24	24	0	0	0
	4	149	149	0	0	0
	5	179	177	1	1	97
	6	183	2	100	81	0
	7	212	212	0	0	0
	8	107	107	0	0	4
GEV	1	169	169	0	0	0
	2	370	370	0	0	0
	3	24	24	0	0	0
	4	149	149	0	0	0
	5	179	179	0	0	0
	6	183	183	0	0	0
	7	212	212	0	0	0
	8	107	107	0	0	0

Table 3: ADSVD neural network classification of the ROI pixels in the PCA and GEV domains.

	PCA Classified NN		GEV Classified NN	
Class #	# Pixels Classified	%	# Pixels Classified	%
1	19,936	6.3	44,718	14.3
2	4,046	1.3	6,338	2
3	8,678	2.7	16,390	5.2
4	12,065	3.8	33,979	10.9
5	5,701	1.8	7,963	2.5
6	30	~0	20,799	6.6
7	2,436	0.8	5,801	1.8
8	41,867	13.3	18,383	5.8
Total Classified	94,732	30.1	154,371	49.1
Unclassified	219,709	69.9	159,997	50.9

Table 4: NN classification of the entire PCA and GEV transformed images at 0.5 threshold. The NN classifies nearly 50% of the pixels in the GEV cube and only 30% in the PCA case.

	PCA Domain		
	Min-D	PP	SAM
Class #	% Classified	%	%
1	34.8	33.8	~0
2	2.2	1	2.3
3	0.2	6.1	6.3
4	3.6	2.2	6.1
5	3	1.1	20.8
6	3.9	5.1	9.4
7	0.2	0.7	0.6
8	0.2	0.5	1.4
Total Classified	48.1	50.5	46.9
Unclassified	51.9	49.5	53.1

Table 5: ENVI classified result in the PCA domain. Different thresholding parameters were used in each case so as to make the total number of classified pixels approximately equal to 50% (same as NN classified results of GEV cube). The following threshold parameters were used: (Min-D ($\sigma = 4.0$), PP ($\sigma = 4.0$), SAM ($\theta = 0.27$ radians).