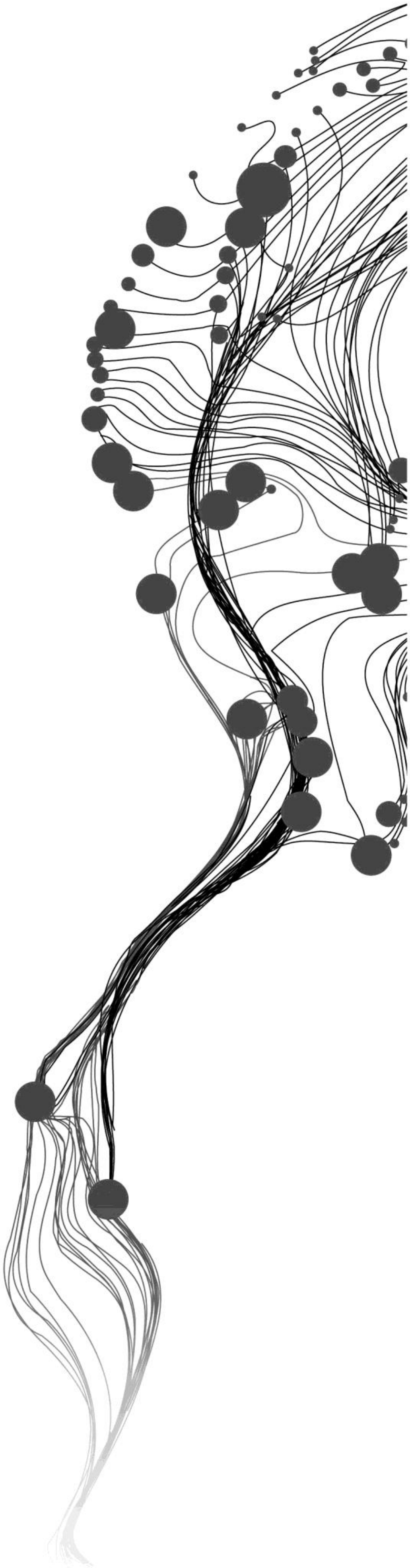
Hyperspectral Subspace Identification and Endmember Extraction by Integration of Spatial-Spectral Information

SOURABH PARGAL
April, 2011

SUPERVISORS:
Mrs. Shefali Agarwal
Dr. Harald van der Werff



Hyperspectral Subspace Identification and Endmember Extraction by Integration of Spatial-Spectral Information

SOURABH PARGAL
Enschede, The Netherlands, April, 2011

Thesis submitted to the Faculty of Geo-Information Science and Earth Observation of the University of Twente in partial fulfilment of the requirements for the degree of Master of Science in Geo-information Science and Earth Observation.
Specialization: Geoinformatics

SUPERVISORS:
Mrs. Shefali Agarwal
Dr. Harald van der Werff

THESIS ASSESSMENT BOARD:
Prof. Dr. Ir. A. Alfred Stein (Chair)
Dr. S.S. Ray (SAC, Ahmedabad)

Dedicated to my loving mother and father

ABSTRACT

This research work concentrates on understanding the concepts of hyperspectral signal subspace identification or dimensionality reduction and endmember extraction by the integration of spatial information with spectrally rich hyperspectral datasets. Signal subspace identification has become an integral part of a number of hyperspectral image processing techniques in which the data dimensionality is high and there is a lot of redundant information present in the dataset. Effectively the image signal information is usually concentrated in lower dimensional subspace. Signal subspace identification enables the representation of signal vectors in this lower dimensional subspace and aids in the correct inference of the dimensionality of the dataset. Hyperspectral subspace identification by minimum error (HySime) is an eigendecomposition based technique and does not depend on any tuneable parameters. HySime initializes by determining the signal and noise correlation matrices and then representing the subspace by minimizing the mean square error between the signal projection and the noise projection. The result is an estimate of the number of spectrally distinct signal sources or the inherent dimensionality of the dataset.

Most endmember extraction algorithms are based on the spectral properties of the dataset only to discriminate between the pixels. Endmembers with distinct spectral profiles or high spectral contrast are easier to detect, the endmembers having low spectral contrast with respect to the whole image are difficult to determine. The spatial-spectral integration approach searches for endmembers by analyzing the image in subsets such that it increases the local spectral contrast of the low contrast endmembers and increases their odds of selection. Spatial spectral integration process utilizes HySime to determine a set of locally defined eigenvectors explaining the maximum variability of the subsets of the image. The image data is then projected onto these locally defined eigenvectors which produces a set of candidate endmember pixels. The candidate endmember pixels, that the spectrally similar and having similar spatial coordinates are averaged together and grouped into different endmember classes.

The results highlights that HySime performs effectively in determining the number of spectrally distinct signal sources in the spaceborne hyperspectral datasets. The spatial-spectral integration results show that the endmember pixels obtained by imposing spatial constraints are cleaner and more representative of the land use land cover classes.

Keywords: Hyperspectral remote sensing, Dimensionality reduction, Signal subspace identification, Spatial-spectral integration, Endmember extraction, Spectral Unmixing

ACKNOWLEDGEMENTS

First and foremost, I want to extend my deepest gratitude to my thesis supervisors, Mrs. Shefali Agarwal and Dr. Harald van der Werff for their constant support, encouragement and guidance throughout the period of this research. With their constructive suggestions, sharp observations, feedbacks, timely supervision and motivational words, they have been constantly around while conducting this thesis work. What I have learnt from my supervisors was not only how to do research but also how to find the motivations.

I would also like to thank Mr. Prasun Kumar Gupta (Scientist-SC, IIRS) for his continuous help throughout the period of this course in the understanding of various aspects of programming in IDL and MATLAB and in the implementation of the algorithm for the completion of this project.

I would like to thank Dr P.S. Roy (Dean IIRS), Mr. P.L.N. Raju (In-charge, GID and Course co-ordinator, M.Sc. Geoinformatics) and all IIRS faculty and staff for providing such a nice infrastructure and environment to carry out the present research work.

I wish to extend my appreciation towards Dr. Nicholas Hamm for his valuable inputs and guidance during our tenure at ITC. And my gratitude and thanks to all ITC faculty and staff for making our stay in Enschede, Netherlands such a wonderful experience.

My special thanks to Dr. D.M. Rogge (Department of Earth and Atmospheric Sciences, University of Alberta, Canada) for his valuable inputs on various aspects for the implementation of the spatial-spectral integration. Also I would like to thank Dr. Jose M. Bioucas-Dias for making the code for HySime freely available for the researcher's community.

I would like to specially thank all my colleagues of Geoinformatics division Richard, Deepak, Shreyes, Tanvi and Preethi and all my P.G. Diploma friends, for always being around, their support and for creating such a peaceful environment for conducting this research work.

The research work was not possible in a timely manner without the gracious donation of computing and other resources by members of the Geoinformatics Division (GID) and Photogrammetry & Remote Sensing Division (PRSD) at IIRS.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Recent advances in remote sensing technology and the launch of a number of satellites have drastically increased space borne remote sensing capabilities which has greatly enhanced our understanding of a number of aspects of earth sciences. The multispectral sensors acquire electromagnetic energy in a small number of discrete spectral bands with comparatively large bandwidths which limits their ability for making precise earth surface studies. Hyperspectral sensors record reflected electromagnetic energy from the Earth surface across the electromagnetic spectrum extending from the visible wavelength region through the near-infrared and mid-infrared (0.3μm to 2.5μm) in tens to hundreds of narrow (in the order of 10nm) contiguous bands [1]. These contiguous bands are also referred to as spectral bands. As a result of such narrow bandwidths an almost continuous and detailed spectral response can be generated for a pixel which provides accurate and precise information about its constituents and is clearly an advantage over multispectral imaging. A hyperspectral image can be illustrated as an image cube with the two dimensions of the face of the cube represents the spatial information and the third dimension representing the spectral information. Figure 1.1 shows the Hyperion datacube and the spectrum.

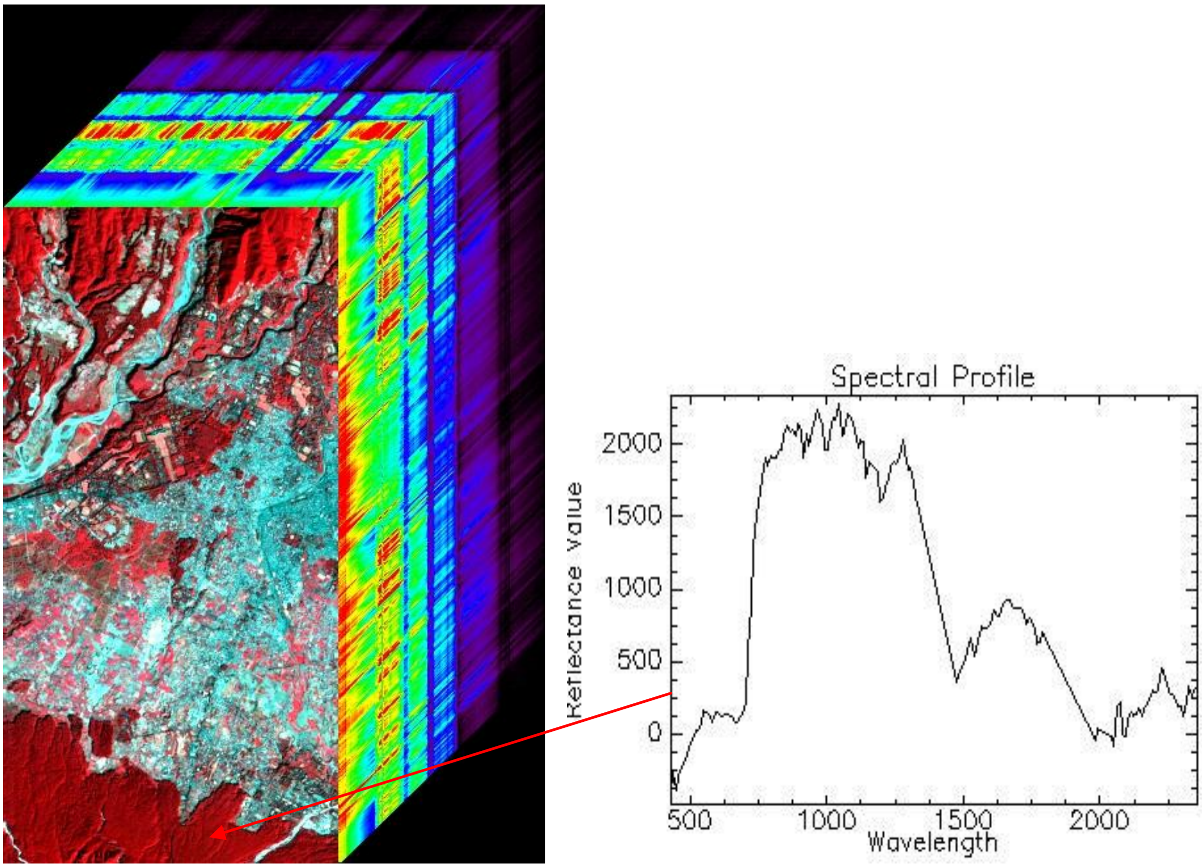


Figure 1.1 Hyperion Image cube of Dehradun area and reflectance spectrum

1.1. Problem context and outline

The availability and use of airborne hyperspectral data has been well studied and documented with a number of airborne sensors in operation since early eighties. With the launch of NASA's Earth Observing 1(EO-1) Hyperion instrument in the year 2000, a platform was created for exploiting the spaceborne hyperspectral imaging capabilities. Hyperion was the first hyperspectral sensor to provide a continuous spectral profile across the broad electromagnetic spectrum ranging from 400nm to 2500nm. The comparison of an airborne sensor, such as Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) and Hyperion datasets in terms of spectral information provide comparable results under optimum acquisition conditions viz. illumination, dark targets etc. [2]. The spatial resolution of airborne sensors (2-20 m depending upon flight altitude and sensor resolution) is however comparatively higher than that of spaceborne sensors (30 m in the case of Hyperion). The low spatial resolution of the hyperion sensor causes a problem of mixed pixels, a pixel which is formed when spectra of different underlying substances are combined into a mixture spectrum. Inspite of the limitations on the spatial resolution there are quite a few arguments which go in favour of spaceborne sensors. Firstly, they allow regular and repeated coverage over wider and restricted areas. Secondly, variations and distortions arising due to aircraft motion are reduced [3].

Due to the continuous spectrum for each pixel, the high-dimensional data space generated by hyperspectral sensors poses challenges in image processing and data analysis and is quite different from multispectral processing where there are only a few discrete bands Also the spaceborne hyperspectral remote sensing images are more affected by noise due to the narrow bandwidths, which can hamper the image interpretation and information extraction processes.

The spectrum received at the sensor can be thought of as the sum of spectral radiance energy (useful signal) and the noisy component. Image noise in remote sensing imagery can be regarded as the random variation in the brightness values in the image induced by the sensor circuitry [4], which is always independent of the atmospheric errors [5]. Atmospheric attenuation is due to the intervening atmospheric constituents, such as water vapours, aerosols etc., between the observed terrain and the sensor which affects the radiance energy received at the sensor.

Management of noise errors induced due to the sensor system and atmospheric attenuation forms the basis for applying pre-processing techniques, such as, bad band removal, destriping and atmospheric corrections, before proceeding to advanced processing for dimensionality reduction, endmember extraction or classification etc.

Hyperspectral datasets are spectrally overestimated and there is a lot of redundant information present even after the pre-processing steps. Effectively there is still noise present in the dataset and the useful signals usually occupy lower dimensional subspace which needs to be inferred. So there is need for exploration of dimensionality reduction (DR) methods which can effectively reduce noise in hyperspectral datasets with minimum loss of information.

1.2. Signal Subspace Identification

Although the presence of such large number of spectral bands does assist in effectively defining different classes; to have realistic multivariate statistical estimates, the size of the training data required increases exponentially with the increase in dimensionality of a dataset [6]. Also computations performed on an entire data cube with limited number of training samples may not give the desired classification accuracy. Considering the impracticality of using large training datasets, the alternate solution must be considered, which calls for dimensionality reduction for determining optimal lower dimensional subspace with a minimum loss of information and class separability. Signal subspace identification enables us to correctly identify the inherent dimensionality of the dataset, thereby increasing the efficiency of endmember extraction algorithms and allowing more efficient use of storage space and computational power [7]. The high dimensional hyperspectral images contain a lot of redundant information and the signal information is usually concentrated in lower dimensional subspaces. Thus signal subspace identification has become a necessary first step in number of hyperspectral processing algorithms such as target detection, classification and spectral unmixing.

A number of different approaches have been applied for reducing data dimensionality or subspace identification over the decades. Band selection or extraction takes the high correlation between spectral bands into consideration and selects a few spectral bands with high signal to noise ratio (SNR) [8]. Principal component analysis (PCA) [9], maximum noise fraction (MNF) [10], and Singular valued decomposition (SVD) [11], are projection techniques that aim at reducing the spectral information to lower dimensions.

PCA represents the signal in terms of power residing in the data, according to the magnitude of eigenvalues and the number of non-zero eigenvalues giving the dimensions of the dataset [12]. PCA neither computes any noise statistics nor does it optimize the SNR. PCA reorders the components according to decreasing image quality with the increasing component number but that is not always the case in reality [10]. MNF always orders components by image quality and maximizes the SNR, but requires prior knowledge about noise and signal covariance matrices [10]. SVD estimates the signal and noise covariance matrices and the subspace are identified by selecting the eigenvalues whose values are larger than the variance in our dataset [7]. As discussed in [7], limitations of MNF and SVD based approaches are; 1) the assumption that noise present in Hyperspectral datasets is independent and identically distributed (i.i.d) which is always not the case, and 2) there are always some random disturbances in the estimates of variance, eigenvalues and eigenvalues matrices of the signal correlation matrices. Also MNF and SVD assume the subspace dimensions are known beforehand, which is not the case in most applications [7]. The shift difference method for noise estimation in MNF has two weaknesses [7]: it assumes that adjacent pixels have almost same signal information and, for good noise estimation, shift difference method should be applied on a homogeneous area. Both these assumptions are not always valid.

The determination of the correct subspace dimensionality or the intrinsic dimensionality of hyperspectral datasets is a challenge. The intrinsic dimensionality of a dataset can be defined as the minimum number of parameters required to explain the properties of the acquired dataset [13]. Methods such as PCA [9] and factor analysis, are suitable for multispectral imagery as there are only a small no of bands, and uses the eigenvalues to determine the intrinsic dimensionality. The signal structure of the hyperspectral sensors, due to their high spectral resolution and a large number of contiguous bands, is largely unknown and may contain a number of unknown spectral sources which includes image endmembers (known or unknown), anomalies and other interference sources [13], which creates further issues in the correct determination of the intrinsic dimensionality.

1.2.1. Hyperspectral Subspace Identification by minimum Error (HySime)

This research work concentrates on a recently developed approach for dimensionality reduction or signal subspace identification (SSI), called Hyperspectral signal identification by minimum error (HySime), which is a minimum mean square error based approach to infer the subspace by minimizing the sum of projection power error and the noise power. This method was proposed in [7] and was applied on AVIRIS sensor. This method is eigen-decomposition based i.e. it decomposes or reduces the original signal into subsets of eigen vectors. The subspace obtained by HySime optimally represents the original signal with minimum error. HySime uses multiple regressions for the estimation of the noise and signal covariance matrices and is adaptive, i.e. it does not require any tuning parameters. Also it makes no assumptions about the noise being independent and identically distributed (i.i.d.) and the subspace dimensions.

For hyperspectral datasets a common approach for dimensionality reduction is the application of eigen decomposition based techniques, such as PCA, MNF or SVD. The difficulty in getting reliable noise estimation from these eigenvalues is that these eigenvalues are still representing the mixtures of the signal sources and the noise present in the data. When the signal sources are too weak their contribution towards the computation of eigenvalues is very less, which can be observed if there is no sudden drop in eigenvalues distribution [14]. HySime, as discussed in further sections, instead finds the subset of eigenvectors and the corresponding eigenvalues by minimizing the mean square error between the original signal and the noisy projection of it.

This study will focus on the results of HySime, in terms of signal subspace inferred, when applied to Hyperion datasets, and then a comparison of the results with the other mentioned techniques.

1.3. Endmember Extraction

1.3.1. Spectral Unmixing

Pixels values in spaceborne hyperspectral datasets, most of the times, have contribution from more than one type of ground objects due to their limited spatial resolution causing mixed pixel spectrum. Spectral unmixing aims at the decomposition of the mixed pixel spectrum into its constituent spectra, also called endmembers [12]. Each pixel in the hyperspectral image can be considered as being composed of linear combination of ground spectra or endmembers with each endmember contributing to the pixel spectra. Thus the spectral signature at each pixel in a L-dimensional hyperspectral image, i.e. the observed spectral vectors, $Y \in \mathbf{R}^L$, when p is the number of endmembers, can be expressed as,

$$y = x + n \tag{1.1}$$

where, y - L-dimensional pixel vector
x and n - L-dimensional signal and noise vectors respectively

Since the signal vectors lie in an unknown p-dimensional subspace, each signal vector is given as,

$$x = Ms = \sum_{i=1}^{p} m_i s_i \tag{1.2}$$

where, M - L×p matrix, whose columns are L×1 endmembers.
s – abundance fraction of each endmember in a pixel

In essence spectral unmixing can defined as the process of determination of the number of image endmembers and their pure signatures and the amount in which they appear in the given mixes pixel.

1.3.2. Spatial-Spectral Integration

Most of the endmember extraction techniques, such as pixel purity index (PPI) [15], N-FINDR [16] etc., rely on the spectral properties of the data alone for endmember extraction without giving any importance to the spatial arrangement of the pixels. Thus, while searching for endmembers the hyperspectral dataset is treated as an unordered collection of spectral measurements with no spatial arrangement [17] [18]. So there is a need for image representation of the data in the quest for endmember extraction as spatially adjacent data elements may be similar despite the differences induced by the noise.

Spatial context in hyperspectral processing is drawing attention of the researchers in this direction. Two of the most famous algorithms in this direction are the automated morphological endmember extraction (AMEE) algorithm [17] and the spatial spectral endmember extraction (SSEE) tool [19]. The AMEE method estimates for each pixel vector, a scalar quantity that gives some measure of the spectral similarity of adjacent pixels. This scalar quantity is then used to weigh the importance of the spectral information associated with each pixel in terms of its spatial context, i.e. distance from other spectrally similar pixels. The SSEE algorithm on the other hand extracts endmembers by partitioning the hyperspectral image into subsets thus enhancing the local spectral contrast of the endmembers, thus enhancing their chances of selection.

The SSEE model is adopted in this study for the integration of spatial spectral information for endmember extraction over AMEE, as AMEE has been primarily developed as a pre-processing method to run on full datacube before applying the conventional spectral based endmember extraction algorithms.

1.4. Data Set

1.4.1. Hyperion Sensor

Hyperion instrument onboard NASA's Earth Observation-1 (EO-1), launched on 21st November 2000 as part of NASA's New Millennium Program, is the first spaceborne Hyperspectral sensor for Earth Observation studies. It orbits the Earth in a sun-synchronous (polar) orbit at an altitude of 705km. The Hyperion is a Push-broom scanner with a high spectral resolution. It has 242 spectral bands spanning a spectral range from 0.4 to 2.5 µm, with a sampling interval of 10nm. The Spatial resolution is 30m (ground sample) with a swath width of 7.7 km and covers an area of 7.7x100 square km per image with high radiometric accuracy (12 bit quantization).

The Hyperion sensor has two spectrometers operating over different spectral ranges. One operates in Visible and near Infrared region (VNIR) i.e. 0.4 to 1µm having 70 bands and the other operates in Shortwave Infrared region (SWIR) i.e. 0.9 to 2.5µm having 172 bands. The overlap region between the two spectrometers between 0.9 to 1µm allows for cross calibration between two spectrometers. Also it helps in improving the signal to noise ratio.

The data in the form of cubes is put into Hierarchical Data Format (HDF) format and is archived. The dataset used for current analysis is radiometrically corrected Hyperion L1R radiance dataset [20].

1.4.2. Hyperion L1R data of Dehradun

The Hyperion image over Dehradun region was acquired on 25th December, 2006 at 05:08:45 AM. The dimensions of the acquired dataset are 256 (ground samples of 30m width) x 3407 (lines) x 242 (bands). The data ia acquired in a wavelength range to 355.589 nm to 2577.070 nm at approximately 10nm sampling interval and the signal to noise ratio is 65 – 130 dB. The scene characteristics of the hyperion image of Dehradun area are listen in Table 1.1.

Table 1.1 Scene Characteristics of Hyperion data of Dehradun Area
(Source: http://edcsns17.cr.usgs.gov/NewEarthExplorer)

Data Attribute	Attribute Value
Entity ID	EO1H1460392006359110PY
Acquisition Date	12/25/2006
Site coordinates	30.34020 N, 78.00660 E
NW Corner	30°40'36.48"N, 78°03'07.97"E
NE Corner	30°39'40.99"N, 78°07'45.03"E
SW Corner	29°46'24.74"N, 77°48'47.43"E
SE Corner	29°45'29.66"N, 77°53'22.00"E
Cloud Cover	0 to 9% Cloud Cover
Receiving Station	SGS

Data Attribute	Attribute Value
Scene Start Time	2006 359 05:08:45
Scene Stop Time	2006 359 05:13:05
Date Entered	1/2/2007
Target Path	146
Target Row	39
Sun Azimuth	153.720703
Sun Elevation	31.538009
Satellite Inclination	98.18
Look Angle	3.3268

1.4.3. Study Area

The city of Dehradun lies at 30°19' N and 78°20' E in the south central part of Dehradun district in the state of Uttaranchal. The Hyperion image strip highlighting the study area is given in Figure 1.2.

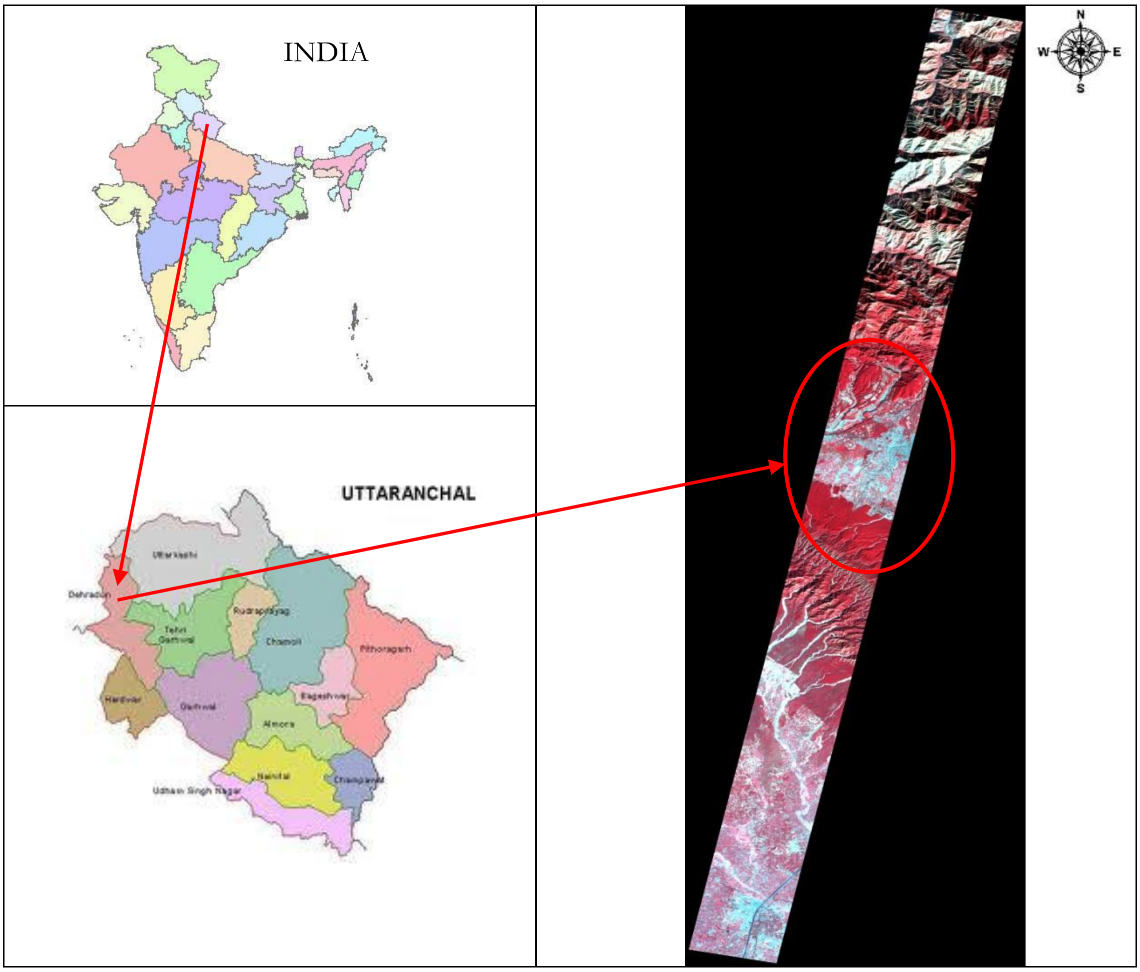


Figure 1.2 Dehradun City and its corresponding Hyperion Image (Scale: 1:100,000)

1.4.4. Linear Imaging Self Scanner (LISS-4)

The Linear Imaging Self Scanner (LISS-4) is a high spatial resolution camera onboard the Resourcesat-1 satellite launched by Indian Space Research Organisation (ISRO) in October, 2003. LISS-4 is a high resolution sensor with a spatial resolution of 5.8 meters and a swath width of 23.9 km from a sun synchronous orbit at an altitude of 817 km.

1.5. Research Identification

1.5.1. Problem Statement

The high dimensionality and the mixed spectrum of Hyperion sensor give us an opportunity to study the behaviour of different signal decomposition techniques and spectral spatial integration techniques for endmember extraction. Current endmember extraction techniques treat the hyperspectral datasets as unordered collection of spectral measurements without any spatial relationships. So there is a need of incorporating contextual information in the process of endmember extraction.

Only a few attempts exist in the literature which aims at integrating contextual spatial information with spectrally decomposed subspace in the process of endmember extraction, and none of these have been applied on spaceborne hyperspectral datasets, which opens up possibilities of more research in this area and is the primary goal of this research. The endmember extraction process could benefit by incorporating spatial information into spectrally rich hyperspectral datasets.

1.5.2. Research Objective

To identify an optimal hyperspectral signal subspace in spaceborne hyperspectral datasets with HySime and to pursue endmember extraction by integration of contextual spatial information with the spectrally decomposed subspace.

1.5.3. Research Questions

The following research questions have been formulated:

- Is the HySime signal decomposition technique more efficient than other existing techniques, in the context of spaceborne hyperspectral datasets?
- What will be the intrinsic dimension of the subspace identified by HySime?
- How to integrate spatial information with spectral subspace identified by HySime for endmember extraction?
- How will the integration of spatial and spectral information improve the classification and mapping accuracies?

1.6. Research Setup

The research work methodology is divided into three different parts:

- Pre-processing
- Hyperspectral subspace Identification
- Spatial-Spectral Integration for endmember extraction
- Spectral unmixing

1.6.1. Pre-processing

The pre-processing of dataset is a necessary first step in Hyperspectral Processing algorithms. Spaceborne hyperspectral datasets require careful data pre-processing because of their low spatial resolution which causes the mixing of spectral response of materials within a pixel. The various steps of pre-processing applied to the dataset in this work are bad band removal, abnormal pixel removal and destriping and atmospheric corrections.

1.6.2. Signal Subspace Identification

Signal subspace is estimated in two steps:

- **Noise estimation**

  Noise in the dataset is estimated using the multiple regression theory. These noise estimates become the input for the subspace identification algorithm.

- **Hyperspectral Subspace Identification by minimum error (HySime)**

  The dimension of the atmospherically corrected image is then reduced using the HySime algorithm which also gives an estimation of the number of endmembers present in the scene. HySime provides an estimation of the number of candidate endmember pixels in the dataset.

1.6.3. Spatial Spectral Integration

For analyzing the spatial and spectral properties of the candidate endmember pixels for endmember extraction, the model of the SSEE tool is adopted in this research work.

1.6.4. Spectral Unmixing

The extracted endmembers are used to unmix the hyperspectral data into the corresponding abundance fraction maps using the linear spectral unmixing module within ENVI™.

1.7. Thesis Organisation

The organization of the thesis is described in this chapter. The thesis contains a total of six chapters.

In ***chapter one***, problem context and outline, the problem statement, the research objectives, the research questions, the research setup and the thesis organization is described. In ***chapter two***, the literature review about different stages and various relevant aspects of the thesis is presented which includes most relevant works on Dimensionality reduction methods, spectral unmixing and previous works on different endmember extraction algorithms. In ***chapter three,*** the different pre-processing methods applied on the dataset to ready it for further processing are described. ***Chapter four*** is divided into two sections, first signal subspace identification contains the methodology on the signal subspace identification. The second section, detailed description of the spectral spatial endmember extraction algorithm used for this work is described. ***Chapter 5*** results obtained after following the proposed methodology are presented. In ***Chapter 6*** the conclusions derived from the results are presented and recommendations for this work are given.

2. LITERATURE REVIEW

The use and application of airborne hyperspectral imaging has been well studied and documented since the early eighties, but with the launch of the spaceborne Hyperion imaging spectrometer it was now possible to regularly obtain imaging spectroscopy data from the earth's orbit. Hyperion was a step forward in space based hyperspectral instrumentation and was designed as a technology demonstration instrument [21]. Although intended as technology demonstration and performance validation instrument for a period of one year, Hyperion is still providing data continuously. So with a number of spaceborne hyperspectral sensors planned to be launched in the next few years, EnMAP (Environmental Mapping and Analysis Program) to be launched in 2014 [22] by German Aerospace Center (DLR) and PRISMA [23] by Italian Space Agency (ASI) to be launched in 2012, the challenge will be either the development of new hyperspectral image processing techniques or refining the existing algorithms for the spaceborne hyperspectral datasets. This section will provide a brief overview of the existing hyperspectral processing algorithms and techniques.

2.1. Review of Dimensionality Reduction Methods

Dimensionality reduction or signal subspace identification has become a necessary pre-processing step in many hyperspectral processing and analysis algorithms. For accurate estimation of the signal subspace dimension, an effective noise estimation procedure is required so as to segregate noise from the signal component. A brief survey of the literatures reviewed for existing noise estimation methods and dimensionality reduction or signal subspace identification methods is presented in this section.

Jimnez & Landgrebe [6] *and Landgrebe* [24] have given two significant properties of high dimensional datasets; Firstly, high dimensional datasets are mostly empty and can be projected onto lower dimensional subspaces without consequential losses in terms of class separability. And secondly, the number of training samples required for statistical estimates increases exponentially with the increase in dimensionality of a dataset. Thus the need arises to project the high dimensional datasets onto appropriate subspace without losing the class separability information.

A band selection technique, using the process of feature weighting, was proposed by *Huang & He* [25], wherein the final spectral band components were selected based on the high correlation exhibited between the adjacent bands in the hyperspectral imagery. In hyperspectral data band selection was performed by pair wise separability criterion and matrix coefficient analysis. The criterion values for individual components were computed by Principal Component Transform (PCT). Sorting of bands for each class involved the evaluation of PCT coefficients and criterion values, determination of final weights for original bands and giving a threshold value for eliminating the redundant bands. The method was demonstrated to be better by comparison with two sequential searches and four feature weighting algorithms.

2.1.1. Principal component analysis (PCA)

PCT or PCA is one of the most popular tools for dimensionality reduction. As observed by *Green et. al.* [10], PCT does not provide an optimal ordering of components according to image quality due to varying noise characteristics from band to band. Principal component analysis (PCA) [9] is a linear transformation that maximizes the data variance by transforming the image data to a new coordinate system so that the original brightness values a are reprojected onto a new set of axis or dimensions. The greatest variance or spread obtained by the redistribution of points by any projection is associated with the first principal component. The second principal component explains the second greatest variance in the dataset and is orthogonal to the first principal component. For dimensionality reduction the orthogonal axis are identified by eigendecomposition of the covariance matrix of the data as given in the following equation [12],

$$\hat{\Sigma} = \frac{1}{N}\sum_{n=1}^{N}(y_n - \hat{\mu})(y_n - \hat{\mu})^T \tag{2.1}$$

where, $\hat{\Sigma}$ - sample covariance matrix,
y_n- image pixel vectors, $(y_1, y_2, \dots \dots, y_N)$,
$\hat{\mu}$ - sample mean vector,
and, N – number of pixels

‘

The eigenvalue decomposition of covariance matrix is represented as,

$$\hat{\Sigma} = U\Lambda U^T \tag{2.2}$$

where, U - eigenvector matrix,
and, Λ - diagonal eigenvalues matrix

The magnitude of the eigenvalues determines the power residing in the data and the eigenvalues are used to reorder the eigenvectors and retaining those representing the maximum variance in the dataset. The number of non zero eigenvalues gives the effective dimensionality of the data. PCA does not take noise statistics of the dataset into account, and does not construct the eigenvectors of the data in a way that optimizes signal to noise ratio [12], thus may not always give better results.

2.1.2. Singular Valued Decomposition

Scharf [11] showed that SVD maximizes the variance in the data i.e. the span of the eigenvectors whose corresponding eigenvalues are larger than the variance in the dataset give the estimate of the subspace dimension and are ordered in the decreasing order of significance.

Principal component analysis (PCA) as discussed in previous sections does not provide any noise statistics and thus may not be suitable for dimensionality reduction of high dimensional and noisy hyperspectral datasets.

A common practice in performing dimensionality reduction in of hyperspectral datasets consists of assuming that the noise is having zero mean and is i.i.d (uncorrelated). The correlation matrix for the observed signal vectors, R_y, is given by:

$$R_y = E(\Sigma + \sigma_n^2 I_L)E^T \tag{4.3}$$

where, E - eigenvector matrix of the signal correlation matrix
Σ - eigenvalues matrix of the signal correlation matrix, with the diagonal elements ordered in decreasing magnitude.

Thus the signal subspace dimensions, p, or the signal subspace estimate is given by the eigenvectors corresponding to the first few largest eigenvalues. The estimated signal subspace ⟨M⟩ is given by:

$$\langle M \rangle = \langle [e_1 \dots \dots e_p] \rangle$$

where, $e_1 \dots \dots e_p$ - eigenvectors spanning the subsapace

The expression 4.3 forms the basic idea behind the implementation of SVD based approaches for dimensionality reduction.

2.1.3. Maximum Noise Fraction and Noise Adjusted Principal Component Transform

The inability of PCT to reliably segregate noisy signals from high spectral resolution remote sensing data led to the development of MNF transform. *Switzer & Green* [26]*, and Green et. al.* [10] proposed the MNF transform which chooses the new components to maximize the SNR and orders them according to increasing image quality or decreasing noise. Maximum noise fraction (MNF) [10] computes the noise statistics information for effectively reducing the dimensionality of the dataset and removing the noise from the dataset.

MNF can be treated as two cascaded PCA's; the first is the transformation of the noise covariance matrix to an identity matrix also called as the noise whitening step. The second is the standard principal component transformation of the noise whitened dataset maximizing the signal to noise ratio (SNR) and thus segregating the signal from the noise. The noise statistics are calculated using the shift difference method also known as nearest neighbour difference [10]. MNF splits and projects the input image into two subspaces based on visual analysis eigenvalues and deciding the cut-off value: The first one is the Signal Subspace (signal plus noise) corresponding the largest eigenvalues and the second is the noise subspace corresponding to the lower eigenvalues.

If the estimates of noise correlation matrix ($\widehat{R_n}$) and the correlation matrix of observed vectors ($\widehat{R_y}$) are known, then MNF maximizes the SNR by the following expression,

$$\frac{U^T \widehat{R_n} U}{U^T \widehat{R_y} U} \tag{2.3}$$

where, U - eigenvector matrix and the component axis are given by the eigenvalues decomposition of the noise and signal covariance matrices.

$\widehat{R_n}$- noise correlation matrix
$\widehat{R_y}$ - correlation matrix of observed vectors

MNF requires prior knowledge of the signal and noise covariance matrices and uses near-neighbour difference to estimate the noise correlation matrix.

The nearest neighbour method for noise estimation is generally applicable for noise estimation in homogeneous areas as it assumes that the adjacent pixels in the dataset have the same signal information. And if noise is not present the correlation of the adjacent pixels should be zero and any variation is treated as noise. So for heterogeneous areas this variation in the signal information will be considered as noise thus disturbing the whole statistics. [27]. So it may be required to carefully select homogeneous areas for better noise estimation, which makes shift difference method not an appropriate method for estimating noise in the whole image.

Lee [28] proposed a method called Noise-adjusted Principal Components (NAPC) transform for dimensionality reduction of hyperspectral images, which is mathematically equivalent to MNF transform. NAPC transform is equivalent to two principal component transformations: First of the noise, and second of the transformed data set. The paper highlighted the first implementation of NAPC transform (or MNF transform) to high spectral resolution remote sensing dataset and proved the usability of NAPC transform (or MNF transform) for noise estimation and determination of the intrinsic dimensionality of data.

Xu & Gong [27] applied the NAPC transform to EO-1 Hyperion image. The noise structure of the Hyperion sensor is mostly unknown. The paper investigates a method to accurately estimate the noise structure, from the random noise present in the data, for the application of NAPC transform. A strategy is adopted to remove both striping noise and the low variance noise across all bands. The striping bands are first located followed by striping columns. The noise covariance structure is estimated either by a body of water such as a ocean or lake or by a piecewise chosen homogeneous site i.e. by generating a within site noise covariance matrix. It was observed that the noise estimation using water sites was more efficient than estimation from other homogeneous sites. The quality of the water vapour absorption bands improved considerably in the restored images.

The main limitations of the SVD based approaches and the MNF are that they assume the noise in hyperspectral datasets to be zero mean and uncorrelated which is not always the case is most datasets and more so in Hyperion data whose noise structure is largely unknown. So the signal subspace may not be given by the eigenvectors corresponding to the first few largest eigenvectors [7].

2.1.4. Estimating Spectrally Distinct Signal Sources

Chang & Du [13] introduces a new concept called virtual dimensionality (VD) defined as "the minimum number of signal sources that characterize the hyperspectral data". Due to the presence of many unknown signal sources in high spectral resolution hyperspectral sensors, the determination of the true dimensionality or intrinsic dimensionality (ID) becomes a difficult task. The signal sources identified by VD may also contain unknown sources such as unknown endmembers, natural signatures and anomalies. It uses multiple regression theory for the determination of noise covariance matrix. The number of spectral endmembers or VD is determined based on the Neyman-Pearson detection theory based thresholding method developed by Harsanyi, Farrand and Chang (HFC) which estimates the number of spectral signal sources in terms of their energies. Another method called noise whitened HFC (NWHFC) includes a noise whitening step [13]. The method provides an estimate of the number of spectrally distinct signal sources present in the hyperspectral data.

Bioucas-Dias & Nascimento [7] proposes a new approach called HySime which is a mean square error based approach for estimating the number of spectrally distinct signal sources in hyperspectral dataset. HySime is eigendecomposition based and uses SVD for the decomposition of signal and noise correlation matrices and then selects the subset of eigenvectors that span the subspace in the minimum mean square error sense. For noise estimation it uses multiple regression theory which performs better than the near neighbour difference used in MNF [10] and NAPC [28]. The experimental results showed that the HySime outperforms the other algorithms such as HFC and NWHFC although all the above methods generally overestimate the number of endmembers present in the scene.

The virtual dimensionality concept and the HySime are regarded as the two widely implemented methods available in literature, for estimating the signal subspace (or the number of endmembers) [29] . However, the advantage of HySime is that it does not require any input parameters. HySime has also been implemented for signal subspace identification by *Iordache et al.* in [30] and *Farzam & Beheshti* in [14]

2.2. Spectral Unmixing

In hyperspectral images, spectral mixing is the result of mixing of two or more spectrally distinct substances. The ground coverage of Hyperion is almost 900 square meters which allows disparate materials to occupy the same pixel. Spectral unmixing is the process by which we can identify the constituents of the mixed pixel and their proportions. The simplest and the most commonly assumed model for a mixed spectrum is a linear model. A single pixel can be portrayed as a checkerboard mixture, as illustrated in Figure 2.1 (a) and assuming that there is no multiple scattering between components, then the spectral response of the pixel is a linear combination of the fractional abundances (area covered by each endmember in the pixel) of the individual substances [12], hence the term Linear Mixture Model (LMM). If there are p endmembers, then the linear mixture model can be expressed as

$$x = \sum_{i=1}^{p} m_i s_{ij} + w_j = Ms + w \,, \qquad\qquad j = 1,2, \ldots \ldots, N \tag{2.4}$$

where, x - $L \times 1$ received pixel spectra

M - $L \times p$ matrix, whose columns are L×1 endmembers.

s - abundance fraction of each endmember in a pixel

w - $L \times 1$ additive noise

N - number of pixels in the image

To be physically meaningful the linear mixture model is subjected to following two constraints; the first is the non negativity constraint,

$$s_{ij} \geq 0$$

and the second is the full additivity constraint,

$$\sum_{i=1}^{p} s_{ij} = 1$$

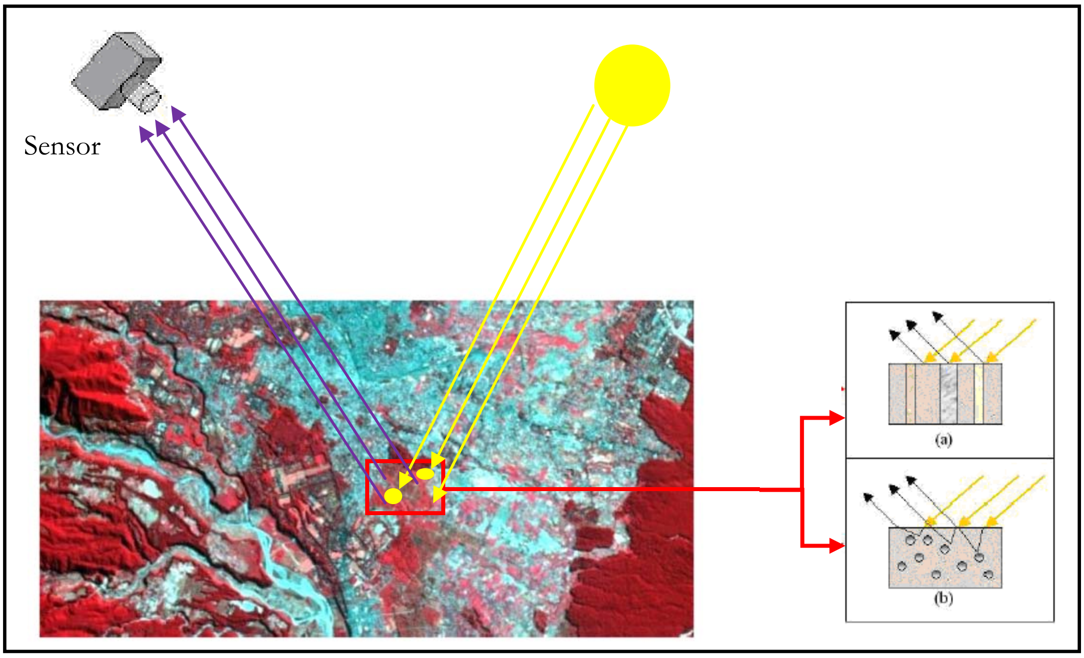


Figure 2.1 Mixing model illustration, a) Linear mixing (no multiple scattering) and b) Non Linear mixing scenario (multiple bounces due to intimate mixture)

2.3. Review of Endmember Detection Algorithms

When the pixel size is large then each individual pixel spectrum measured by the sensor may contain contributions from a number of different materials on the ground. The resultant product is a mixed spectrum and the pure constituents which contribute to this mixed spectrum are called endmember spectrum. By definition given by *Schowengerdt* [31] and cited by *Zortea & Plaza* [17], an endmember is an idealized pure signature for a class. A number of endmember detection algorithms are described in the literature. This section gives an overview of various endmember extraction algorithms.

Boardman [32] showed that geometric analysis of high dimensional data requires the treatment of pixels as vectors in N-dimensional space, N being the number of spectral bands, and then the projection of data onto lower dimensional subspace. Endmembers are determined by fitting a simplex around the complex hull of the data. The convex geometry model defines endmembers to be the vertices of a simplex that surround the pixels in an image. Fig. 2.1. shows a two dimensional scatter plot of a simplex in 2-D space.

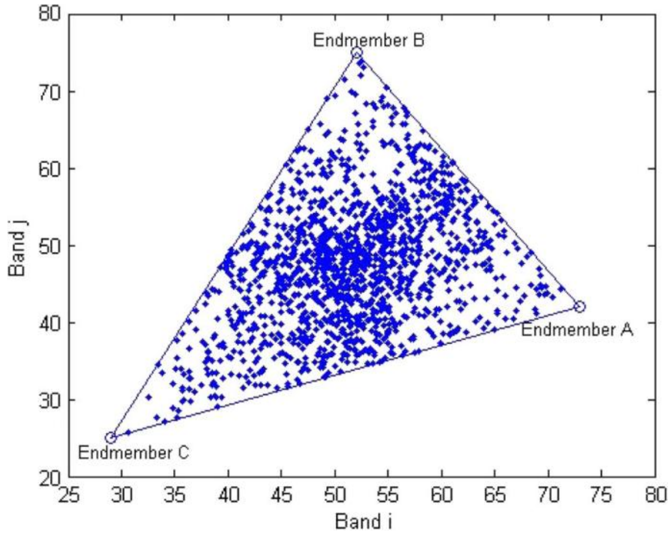


Figure 2.2 Two dimensional scatter plot showing a simplex in 2-D space

Most of the popular endmember extraction algorithms nowadays are based on geometric analysis of the image data. *Keshava & Mustard* [12] argued that the basic assumption for geometric endmember extraction is that endmembers are pure spectra in the image which lie at the extreme ends of the volume occupied by the data points. As shown in the Figure 2.1 the pixels lying at the extreme vertices of the simplex i.e. endmembers A, B and C are the most spectrally pure pixels, and those lying at the middle can be expressed as a linear combination of the these three pure spectra. This also forms the basic premise for linear spectral analysis (SMA) techniques or spectral unmixing.

Endmember extraction from remotely sensed hyperspectral images is increasingly becoming a first choice over spectral measurements in field or laboratory. Field and laboratory spectra usually acquired from the areas of individual's interest and have direct physical meaning for mapping purposes [19]. These physically meaningful endmembers may not represent all the endmembers present in the area. From satellite hyperspectral data we can extract pure or relatively pure endmember spectra, either by visual inspection or by applying one of the various endmember extraction techniques available.

2.3.1. Pixel Purity based Endmember Extraction Algorithms

A number of endmember extraction algorithms make the assumption that for each endmember, there exists, at least one pixel which belongs to that endmember only. With this comes the assumption that the spatial resolution of the imaging instrument does not combine the spectra of adjacent pixels [33], which is not practically possible for most of the hyperspectral sensors. The two popular algorithms based on the above assumption include the PPI algorithm [15] and the N-FINDR algorithm [16].

Winter [16] proposed the N-FINDR algorithm for endmember extraction from hyperspectral dataset. The algorithm determines a simplex of largest volume, within the dataset, containing the maximum number of pixels. The procedure initializes by appointing a set of randomly selected pixels as initial endmembers and calculates the volume. In order to refine the endmember estimate, the volume of the simplex is calculated by replacing each endmember by each pixel in the image. If the volume increases after replacement, the pixel is retained. The procedure continues until there is no further replacement of endmembers.

Boardman et al. [15] proposed the Pixel purity index (PPI) algorithm, is one of the most widely endmember extraction algorithm used for hyperspectral image analysis. It extracts the pure spectra or endmembers in the dataset by searching for set of vertices in the convex hull geometry of the dataset. First the dataset is transformed onto lower dimensions by using either PCA or MNF as the assumption here is that the endmembers lie in the first few principal components. The endmember pixels are obtained by repeatedly projecting the transformed data onto randomly projected vectors (*k*) in n-dimensional space. As the vectors are randomly generated the results depend upon the number of random projections. Pixels lying at the extremes of a random vector are assigned a purity value. The values are updated after each projection and the pixels having values more than a set threshold (*t*) are considered as "pure" pixels.

Despite being widely used PPI suffers from a number of limitations as discussed by *Chang & Plaza* [34]. One of the major limitations of the PPI algorithm is its sensitivity to the input parameters, *k* and *t*. Second problem is that the process to generate the initial random vectors may give us different set of endmember candidate in each run, making the process non repeatable owing to sensitivity to noise. The third concern is the amount of human intervention required to manually derive the final endmember set.

Another algorithm for endmember extraction based on the geometric analysis of the data is Vertex component analysis (VCA) [35] [36]. VCA is an unsupervised endmember extraction method and can be applied to hyperspectral datasets with or without dimensionality reduction, although it is generally preferred to reduce the dimensionality to reduce computational costs. VCA utilise two facts of geometric analysis: first, the image endmembers reside at the vertices of the simplex and, second, the affine transformation of a simplex is also a simplex. The algorithm starts by determining the subspace spanned by the endmembers using HySime and then projects the spectral vectors in a direction orthogonal to the determined subspace. The extreme ends of the projection correspond to the endmember spectra. The algorithm runs iteratively until all the endmembers are found. VCA algorithm was found to be performing better than PPI and better or comparable to N-FINDR. However the computational complexity of VCA was found to be least among the three algorithms.

Besides the above mentioned popular algorithms a lot of literature can be found on the subject of endmember extraction techniques such as the manual endmember extraction tool (MEST) by *Bateson & Curtiss* [37], the endmember optimization method by *Tompkins et al.* [38], the convex cone analysis (CCA) by *Ifarraguerri & Chang* [39].

However, all the techniques mentioned above take into account the spectral properties of the data only for endmember determination. Two of the most noted steps towards integrating spatial-spectral information are the automatic morphological endmember extraction (AMEE) [17] and the spatial-spectral endmember extraction tool (SSEE) [19].

2.3.2. Spatial adjacency based Endmember Extraction Algorithms

Zortea and Plaza [17] defines the AMEE algorithm as a pre-processing module that uses the spatial information and then uses the existing spectral endmember extraction techniques to effectively extract spectral endmembers, and helps in the accurate representation of the original hyperspectral scene. The AMEE algorithm does not require any dimensionality reduction thus using information from all the bands in the dataset. It searches for the most spectrally pure and mostly mixed pixel in a spatial neighbourhood using the morphological operators of dilation and erosion. It then assigns an eccentricity value to each spectrally pure pixel which is calculated as the spectral angle distance (SAD) between the most spectrally pure pixel and the mostly mixed pixel. The process is iterative and the eccentricity values of the selected pixels are update at each iteration. A threshold is applied to the resulting eccentricity image to obtain the final set of candidate endmembers which can be used as input to existing endmember extraction algorithms. The experiments in [17] with real and simulated datasets show that the AMEE algorithm by incorporating the spatial information effectively guides the traditional spectral endmember extraction algorithms to extract endmembers from hyperspectral datasets. There are a few issues however, firstly, the increase in processing time with the increase in maximum size of the spatial neighbourhood and secondly, the algorithm is able to select only one pixel per spatial neighbourhood as the candidate endmember pixel.

An approach used for the integration of spatial contextual information is the spatial-spectral endmember extraction tool (SSEE) proposed by *Rogge et al.* [19], which takes the advantage of the spatial properties of image endmembers by partitioning the image into subsets. Running the image endmember extraction process on subsets may result in the selection of endmembers having high local spectral contrast within the subset. The SSEE algorithm starts with the projection of the image pixel vectors onto the eigenvectors compiled by the singular valued decomposition (SVD) of the subsets of the input hyperspectral dataset. The pixel vectors lying at the extreme ends of the projection are identified as the candidate endmember pixels. The spatial and spectral characteristics of the candidate endmember pixels are analyzed by averaging the spectrally similar pixels (based on minimum SAD score or root mean square error) within a given spatial neighbourhood and become the updated candidate endmember pixels. Then each candidate endmember pixel is averaged with all other candidate endmember pixels within the window and the process is repeated iteratively until the end product is a set of endmember pixels that are spectrally and spatially distinct. Then the endmember pixels are ordered according to their spectral angle. The proposed tool was shown to perform better than the well known spectral based algorithms in extracting unique endmembers. The major benefit of SSEE when compared to pixel purity index is the use of non random vectors and thus the results are repeatable.

Rivard et al. [40] uses the SSEE algorithm for integrating spatial constraints in the endmember extraction process thus improving the relative spectral contrast of the endmembers. The SSEE results are then integrated with an iterative spectral mixture analysis (ISMA) tool to optimize the endmembers pixel wise and to give accurate estimation of the abundance fractions of endmembers.

2.3.3. Spectral Angle Distance (SAD)

The spectral angle distance (SAD) as explained in [41] computes the spectral similarity between a test (or pixel) spectrum, t, and the reference spectrum (target spectrum or laboratory spectrum or another pixel spectrum), r, and is expressed in terms of vector angle, φ, as:

$$cos\,\varphi = \frac{\sum_{i=1}^{n} t_i r_i}{\sqrt{\sum_{i=1}^{n} {t_i}^2}\sqrt{\sum_{i=1}^{n} {r_i}^2}} \tag{2.5}$$

where, φ - spectral angle, t - test or pixel spectrum
r - reference spectrum, n - number of bands

While computing the SAD each spectrum is considered a vector in the n-dimensional space, The output of spectral angle mapping for each pixel is an angular difference between the test and the reference spectrum measured in radians, ranging from zero radians to $\Pi/2$. The smaller the spectral angle more is the similarity between the test and the reference spectrum. Fig. 2.2 gives an example of the spectral angle between a pixel and the reference or target spectrum.

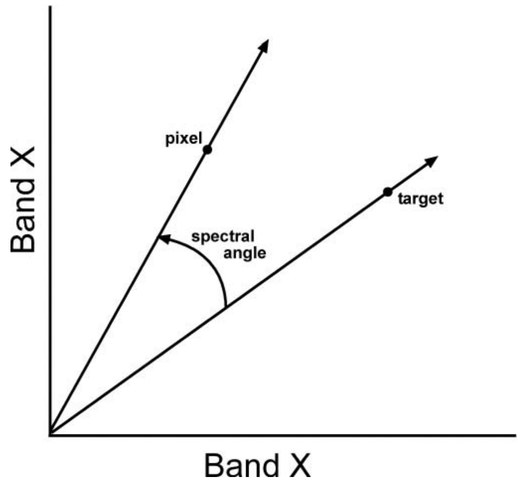


Figure 2.3 Spectral angle between target and the reference spectra

The spectral angle distance is preferred over other distance metrics as it is insensitive to illumination differences in a pixel. Any illumination change will change the magnitude of the vector but not the direction. Secondly, in the later stages of this study, unique image endmembers will be grouped based on the variation in their spectral response to represent various land use land cover classes.

3. DATASET AND PREPROCESSING

This chapter discusses about the dataset used in this thesis, its properties and the various pre-processing steps applied. The dataset used in this research work is the Hyperion level L1R dataset of the Dehradun area. The Dehradun area consists of different ground covers The study scene comprises of various land use land cover classes such as agricultural area, barren land, forest, settlement, tea garden, water body, settlements etc. A false colour combination (FCC) of the Hyperion image is shown in Figure 3.1.

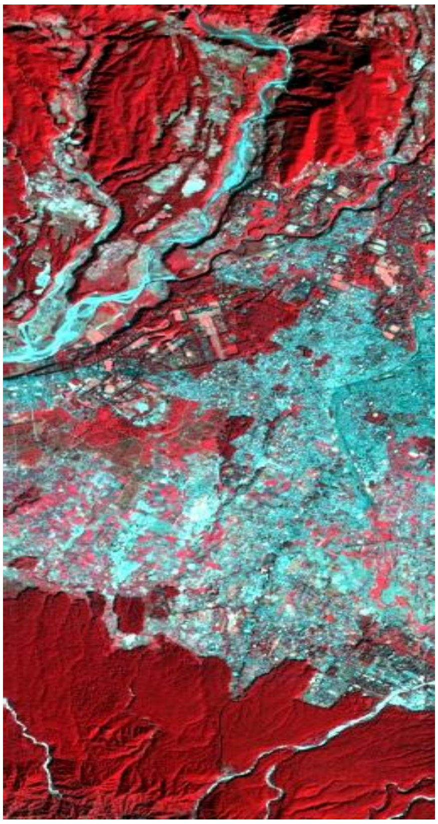


Figure 3.1 FCC of Hyperion data of Dehradun area (Scale: 1:100,000)

The Hyperion is a push-broom sensor with 242 contiguous, narrow bandwidth bands. Because of the huge volume of spectral data available, and the noise present the spaceborne hyperspectral dataset, it requires careful pre-processing for managing the noise. The pre-processing of dataset can be considered as the first step towards further interaction with the dataset.

The pre-processing approach adopted in this thesis involves:

- bad band removal i.e. removing the bands with no information,
- along track destriping and
- atmospheric corrections to convert the radiance to reflectance.

3.1. Bad Band Removal

Hyperion level L1R data has 242 bands out of which only 198 are nonzero i.e. a few were intentionally left unused (Bands 1 to 7 and 225 to 242) and others fall in the overlap region of the two spectrometers (Bands 58 to 76). Among the non zero bands, four band are still in the overlap region of the two spectrometers i.e. bands 56, 57 and 77, 78 out of which bands 77 and 78 were eliminated because of the higher noise levels present in those bands [42]. Then there are water vapour absorption bands which needs to be eliminated and are identified as bands120 to 132 (1346nm to 1467 nm), bands 165-182 (1800 to 1971 nm) and bands 221 (above 2356) and higher. Water vapour absorption bands absorb all the incident solar energy and can be easily identified visually. The list of bands which were eliminated is given in the table below:

Table 3.1 List of Unused Bands of the Hyperion Sensor, L1R product

Bands	Description
1 to 7	Not Illuminated
58 to 78	Overlap Region
120 to 132	Water Vapour Absorption Band
165 to 182	Water Vapour Absorption Band
185 to 187	Identified by Hyperion Bad Band List
221 to 224	Water Vapour Absorption Band
225 to 242	Not Illuminated

All the bands were visually examined and a list of bad bands was prepared. The Hyperion L1R dataset was imported into using the Hyperion import utility (hyperion_tools.sav, source: http://www.ittvis.com/), which is used to convert L1R dataset into ENVI formats containing wavelength, full width half maximum (FWHM) and bad band information. The bad band list generated by hyperiontools.sav was applied to the converted dataset to eliminate the bad bands which resulted in 158 bands to be used in further processing.

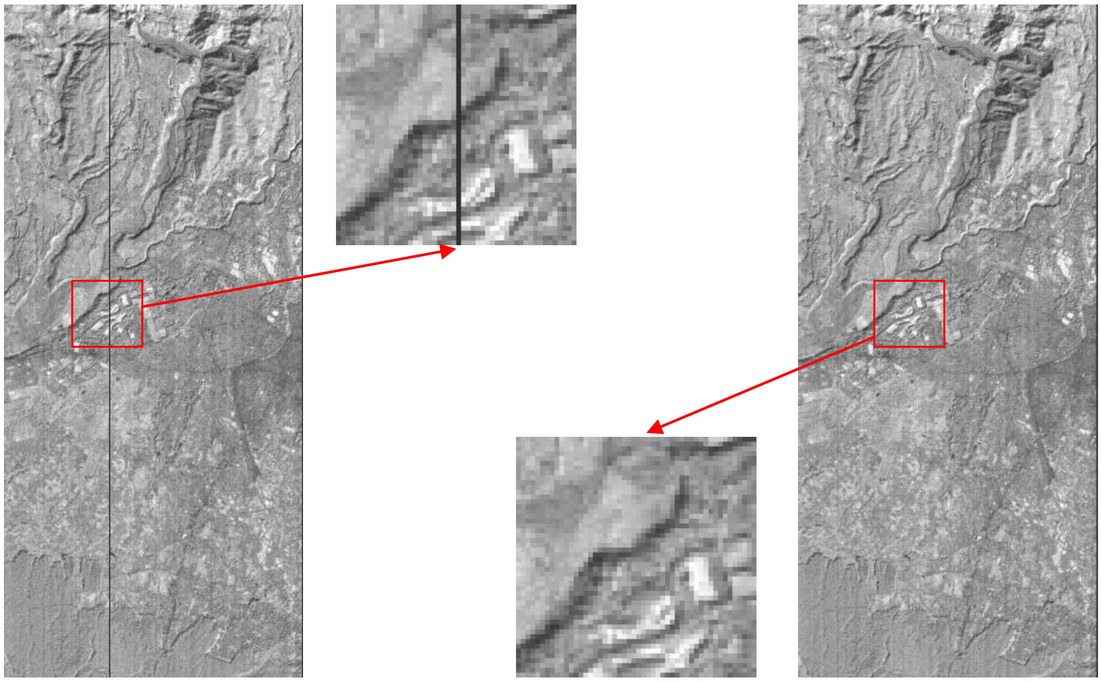
3.2. Along-track Destriping

There are a number of corrupted pixels and dark vertical stripes in the Hyperion datasets that are caused by calibration differences in Hyperion detector array and temporal variations in the detector's response [4]. The vertical stripes are in the along-track direction and appear as a series of stripes either along the whole length of the image or intermittently and are also referred to as striping noise. These vertical stripes and the corrupted pixels are referred to as abnormal pixels [43]. These abnormal pixels must be accounted for and corrected before further processing.

According to *Han et al.* [43] majority abnormal pixels in the Hyperion images appear as vertical stripes and can be classified into 4 categories:

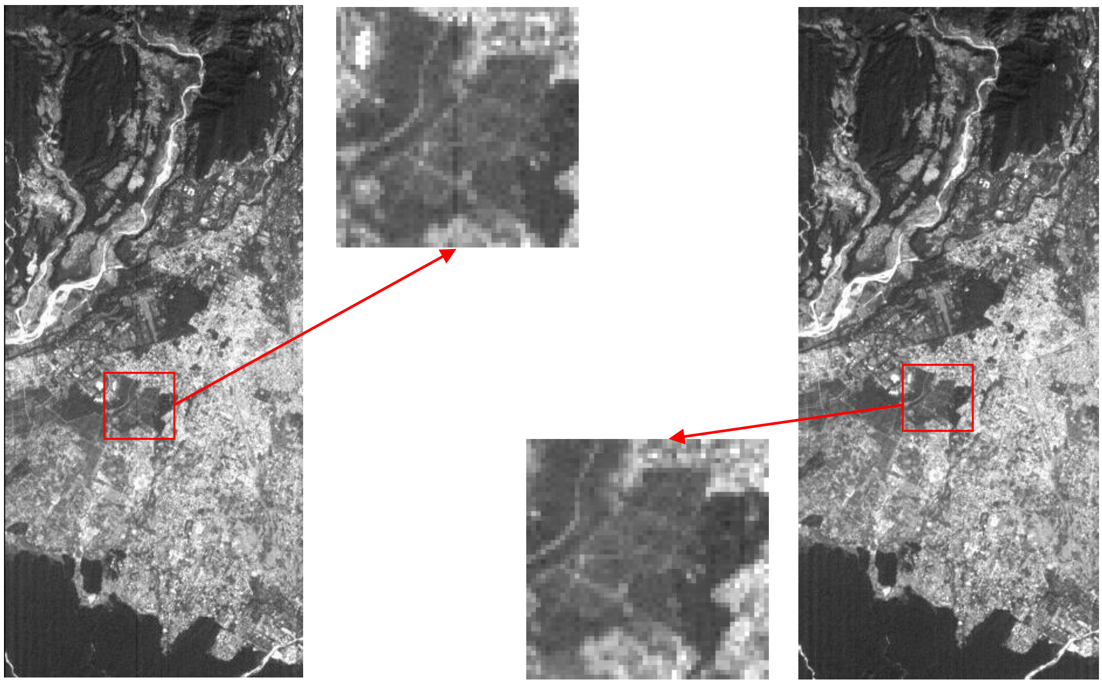
- Class1 - continuous with atypical DN values - extremely small DN values, usually zero
- Class2 - continuous with low DN values - low DN values compared to adjacent columns
- Class3 - intermittent with atypical DN values - extremely small DN values
- Class4 - intermittent with lower DN values - low DN values compared to neighbouring pixels

The figures below show examples of different types of abnormal pixels in the Hyperion data. Figure. 3.2. a) shows the Class 1 type of abnormal pixels by taking a spatial subset from the Hyperion image and Figure 3.2. b) shows the corrected image after correcting the image using Hyperiontools.sav.



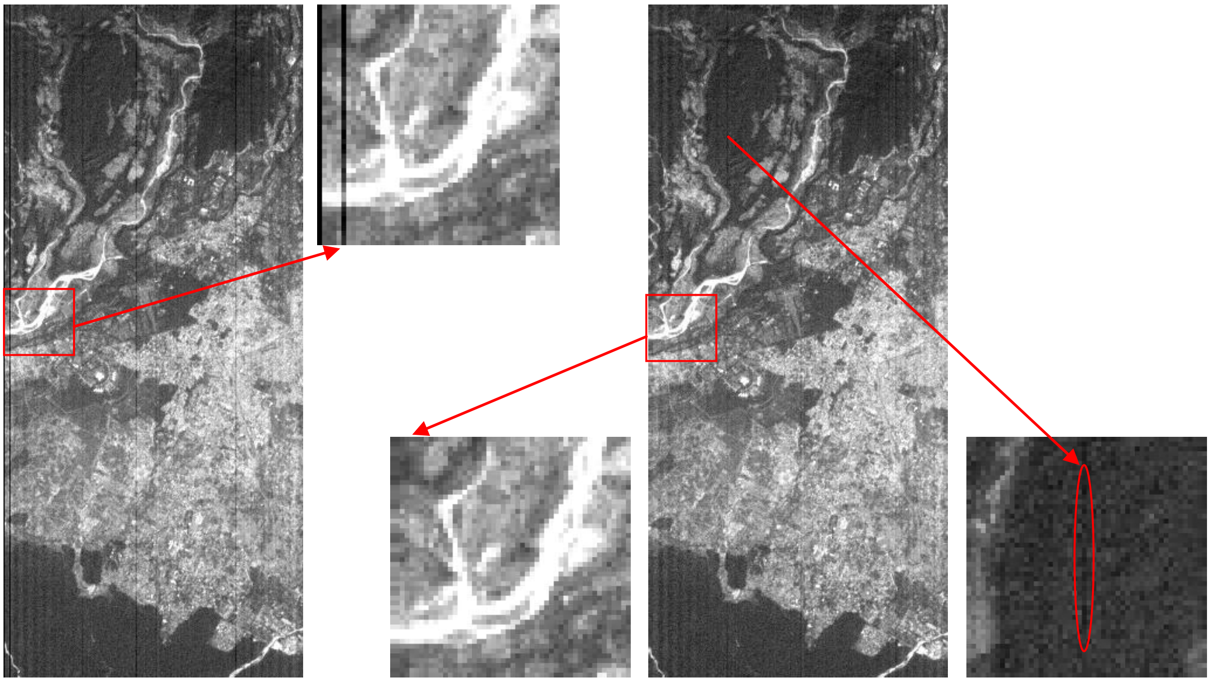
a) Original Band   b) Band after correction

Figure 3.2 a) Class 1 Abnormal pixels: Continuous with atypical DN values, Band 99 and b) Band after correction using Hyperion tools.sav



a) Original Band   b) Band after correction

Figure 3.3 a) Class 4 Intermittent pixels: Intermittent with atypical DN values, Band 14 and b) Band after correction using Hyperion tools.sav



a) Original Band b) Band after correction c) Uncorrected Pixels

Figure 3.4 a) Class 2 Abnormal pixels: Continuous with low DN values, Band 10, b) Band after correction using Hyperion tools.sav and c) Uncorrected pixels

The level L1R Hyperion dataset contains a number of bands containing a series of vertical stripes which are left for the user to correct according to its convenience. While generating the bad band list the hyperiontools.sav utility of ENVI uses the flag mask correction for detecting and correcting the continuous vertical stripes and the abnormal pixels with atypical values. Figures 3.2 (b), 3.3 (b) and 3.4 (b) show the output of the hyperiontools.sav utility, for band number 99, 14 and 10 respectively.

Even after destriping the image with the hyperiontools.sav utility, a lot of vertical stripes (continuous with low DN values and intermittent vertical stripes) were still remaining in different bands of the dataset as evident from the Figure 3.4.(c). The DN values of these vertical striping pixels or abnormal pixels are lower than their neighbouring pixels. Separate programs were created for the detection and correction of these abnormal pixels in the IDL environment of ENVI, following the algorithm proposed by *Goodenough et al.* in [44]. The IDL codes for identifying and correcting the abnormal pixels can be found in appendix A. The algorithm starts by traversing each band along the rows comparing the DN value of each pixel with the DN value of its immediate left and right pixels. If the DN value of a pixel is less than both the pixels then this pixel is labelled as abnormal pixel. As soon as an abnormal pixel is found, each column of each band is checked vertically to find the number of consecutive abnormal pixels. A column can be marked as a striping column if it satisfies two user defined conditions; first, the number of consecutive abnormal pixels should be above a user defined threshold value (five in this case) and second, the number of abnormal pixels should account for more than half of the total number of pixels in that column. If both conditions are satisfied then the column is marked as a striped column. The DN values of pixels in theses striping columns are replaced with the average DN values of the pixels of the adjacent columns on the left and the right. All of the 158 bands were also visually inspected for vertical stripes both continuous and intermittent and were combined with the results of the program output. Table 3.2 gives a list of vertical striped columns in each band of the Hyperion image.

Table 3.2 Detected striping columns

Band	Column	Band	Column
8	68, 125, 132, 162, 168, 172, 183 ,189, 198, 204, 237, 246	95 to 119	256
9	68, 148, 151, 166, 220, 223, 229	133 to 164	256
10	68, 131, 149, 158, 164, 206, 212	183 to 184	116, 117, 246, 247, 256
11	68, 195,	188	116, 117, 246, 247, 256
27, 28	47	189 to 191	116, 117, 213, 214, 246, 247, 256
39	177	192	256
54	13	193	196, 256
55	13, 17, 20, 37	194, 195	256
56	8, 13, 17, 20, ,33, 37	196 to 200	246, 247, 256
57	13, 18, 20, 32, 33	201	7, 256
79	256	202 to 208	256
80, 81	250, 256	209 to 210	88, 89, 256
82 to 93	256	211 to 220	256
94	91, 256		

After the removal of bad bands, removal of absorption bands and abnormal pixels the 158 band image is ready for further processing.

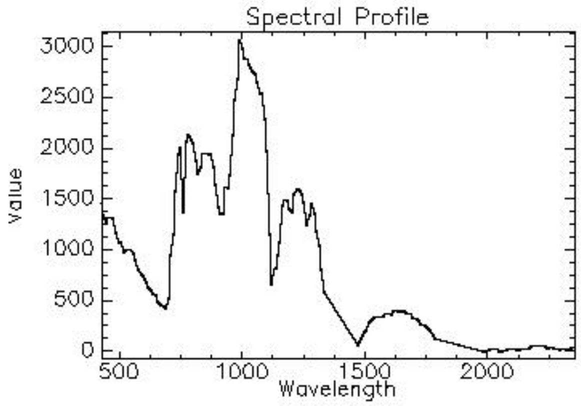
3.3. Atmospheric Corrections using FLAASH

After the removal of the bad bands and destriping the resized 158 bands were corrected for atmospheric errors using the FLAASH (Fast Line-of-Sight Atmospheric Analysis of the Spectral Hypercubes) model of ENVI's atmospheric correction module. The Hyperion image of Dehradun area, due to its time of acquisition, is badly affected by atmospheric error such as haze. Thus atmospheric corrections of Hyperion images are required for the reduction of the atmospheric influence on the reflectance and to filter out the target reflectance cleanly from the mixed signal [45]. Using FLAASH wavelengths ranging through visible, infrared and short wave infrared can be corrected for atmospheric errors. The different parameters which were applied for running the FLAASH model on hyperion image are listed in the table below

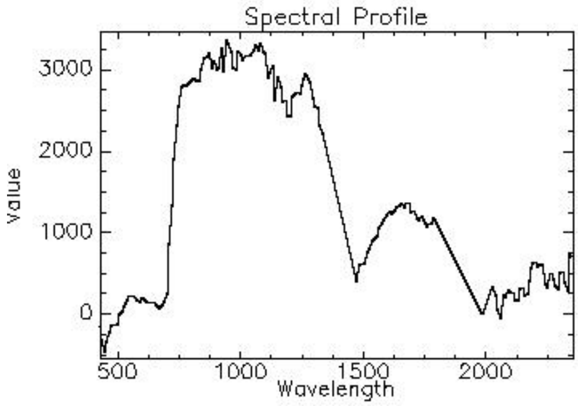
Table 3.3 FLAASH parameters for atmospheric corrections

Latitude	30º 20' 24.72"	Flight Date	25th December, 2006
Longitude	78° 0' 23.76"	Atmospheric model	Tropical
Sensor Type	Hyperion	Aerosol Model	Rural
Sensor Altitude	705 km	Aerosol Retrieval	2-Band (K-T)
Ground Elevation	0.600 km	Water Absorption Feature	1135 nm
Pixel Size	30 m	Initial Visibility	30 km
Flight Time (HH:MM:SS)	5:10:23		

After running FLAASH the haziness in the image is reduced and the image features are sharpened and the image looks to be better illuminated. The Figure 3.5 shows the spectral plots of a forest pixel before and after atmospheric corrections.



(a)



(b)

Figure 3.5 Spectral profile (Z-profile) of a randomly selected pixel, a) before Atmospheric corrections and b) after Atmospheric corrections with FLAASH

Atmospheric corrections transform the hyperspectral data to apparent surface reflectance. Atmospheric corrections are required for matching the image endmember spectra with the reference spectral libraries or ground data.

3.4. Spatial Subset

The spatial subset of the image is usually taken to extract the area of interest of the user. The raw Hyperion imagery of Dehradun area contained 256 samples and 3407 rows. For this study, a spatial subset consisting of 256 samples and 640 lines was extracted from the original Hyperion scene.

4. METHODOLOGY

The methodology chapter includes the theory and the steps undertaken to achieve the research objective. The various processes performed and the algorithms used will be described in this section. The flowchart of the overall methodology adopted for this work is given below in Figure 4.1

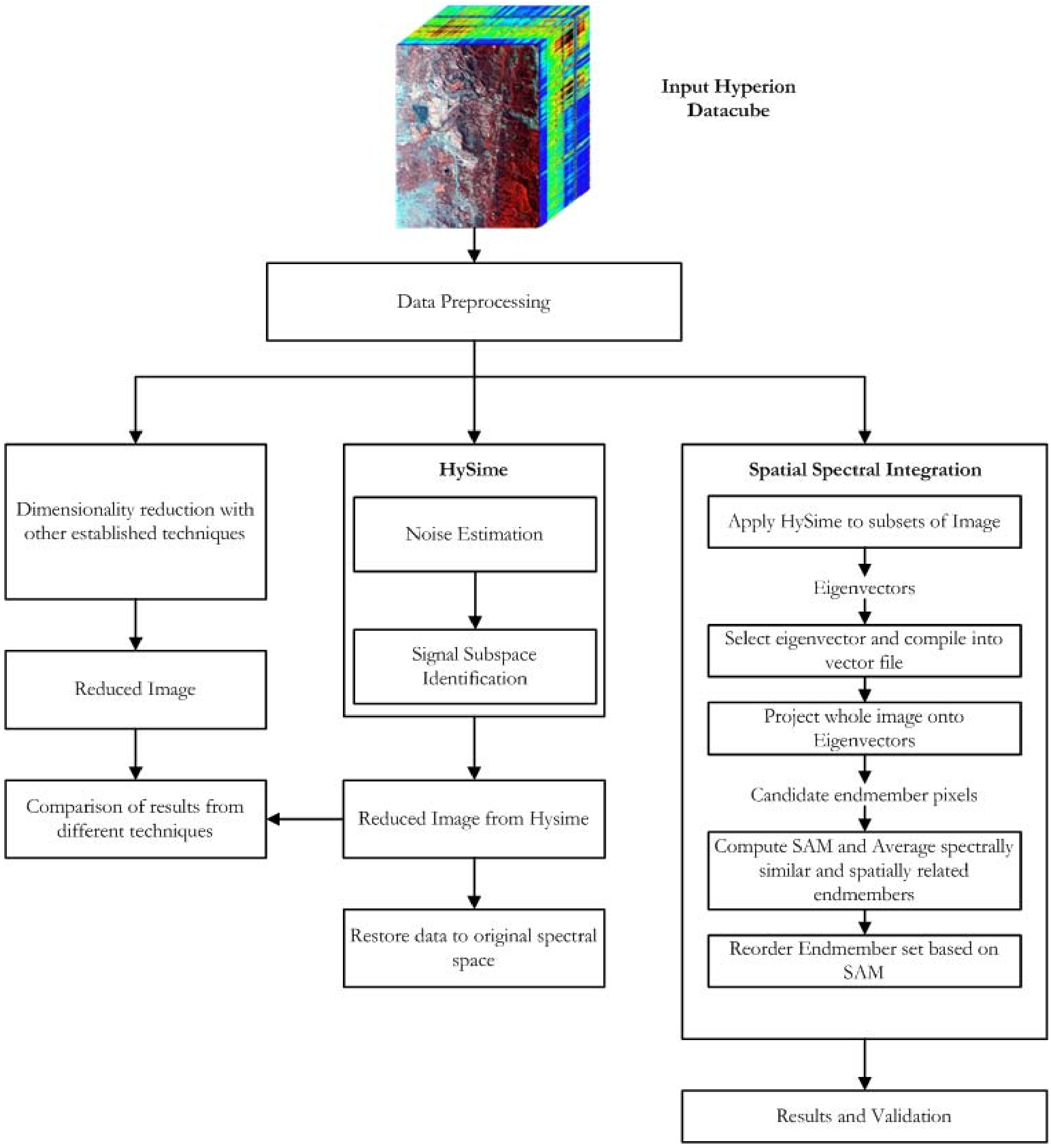


Figure 4.1 Methodology Flowchart

This chapter concentrates on the problem of dimensionality reduction (signal subspace identification) of hyperspectral datasets and the reasons for choosing HySime. It starts with the description of noise estimation and subspace identification algorithms. Then the further sections deal with the issue of integrating spatial information for endmember extraction.

4.1. Data Preprocessing

Data pre-processing the first step of this research work. The various pre-processing steps include bad band removal, along track destriping and atmospheric corrections, and are described in detail in the previous chapter.

4.2. Signal Subspace Identification:HySime

After pre-processing the data signal subspace estimation is performed using HySime. The preliminary code for the HySime algorithm was implemented in MATLAB environment by *Bioucas-Dias & Nascimento* [7] and is freely downloadable from the author's website (Source : http://www.lx.it.pt/~bioucas/code.htm) . The input to the algorithm is the AVIRIS Cuprite dataset in the '.mat' (MAT-file) format of MATLAB. MAT-files are binary files which can contain variables of different data types such as strings, matrices and multidimensional arrays. So the first step was to convert the spaceborne Hyperion dataset into MAT-file format, containing the variables, which becomes the input for the HySime algorithm. A small MATLAB code was written for reading the Hyperion data into MAT-file, and this MAT-file became the input to the HySime algorithm.

HySime [7] starts with the noise estimation step in which the noise correlation matrix of the data is computed. Then it calculates the signal correlation matrix and computes the eigenvectors by performing the eigen decomposition of the signal correlation matrix. The signal subspace is then derived by minimizing the sum of projection error power and noise power, which are decreasing and increasing functions of the subspace dimensions respectively.

Let us assume that the observed spectral vectors, $Y \in \boldsymbol{R}^{L}$, for the given hyperspectral scene are given by:

$$y = x + n \tag{4.1}$$

where x and n - L - dimensional vectors for signal and noise
and L is the number of bands.

The assumption here is that the signal vectors reside in an unknown p – dimensional subspace such that,

$$x = Ms \tag{4.2}$$

where, $p < L$,
M - $L \times p$ matrix, whose columns represent the image endmembers
and s - abundance fraction of the endmembers.

4.2.1. Noise Estimation

Estimation of noise from a dataset is a challenging task in image processing and particularly in remote sensing images. In case of Hyperion images the task of noise estimation becomes difficult because of the large dimensions and the noise structure of the Hyperion images, which is largely unknown. One of the most widely used noise estimation procedure for hyperspectral images is the nearest neighbour method [10], also known as shift difference method, used in the MNF [10] transform. The basic limitation of this approach, as discussed in previous sections, is that it requires a careful selection and prior knowledge of the homogeneous regions for obtaining quality noise estimates.

HySime uses a multiple regression based approach for noise estimation from hyperspectral images, which performs better than the shift difference method because of the high correlation exhibited between adjacent spectral bands. In this algorithm the multiple regression theory has been used to determine the noise correlation matrix from the hyperspectral dataset.

Let Y be an L×N matrix, where N is the number of observed spectral vectors and L is the number of bands. Then define a matrix, $Z = Y^T$, which is an N×L matrix, a N×1 vector, $z_i = [Z]_{:,i}$, where $[Z]_{:,i}$ is the i[th] column of Z, i.e. z_i contains the data read by the hyperspectral sensor at the i[th] band for all image pixels, and the N×(L-1) matrix $Z_{\partial i} = [z_1, \ldots, z_{i-1}, z_{i+1}]$.

Now if z_i is given by the linear regression equation,

$$z_i = Z_{\partial i}\beta_i + \xi_i \tag{4.3}$$

where, $Z_{\partial i}$ - data matrix of dimensions N×(L - 1)

β_i - regression vector of size (L – 1)×1

ξ_i - noise vector of size N×1

The least square estimate for the regression vector β_i is given by the equation,

$$\widehat{\beta}_i = (Z_{\partial i}^T Z_{\partial i})^{-1} Z_{\partial i}^T z_i \tag{4.4}$$

The noise estimates, $\widehat{\xi}_i$, are given by the equation,

$$\widehat{\xi}_i = z_i - Z_{\partial i}\,\widehat{\beta}_i \tag{4.5}$$

and the estimated noise correlation matrix, $\widehat{R}_n$, is given by,

$$\widehat{R}_n = [\widehat{\xi}_1, \ldots \ldots, \widehat{\xi}_N]^T [\widehat{\xi}_1, \ldots \ldots, \widehat{\xi}_N]/N \tag{4.6}$$

The algorithm for the noise estimation procedure is shown in Algorithm 1. The algorithm also contains a complexity reduction method included in Algorithm 1.

Algorithm 1: HySime, Noise Estimation [7]

Input: Observed Spectral Vectors, $Y = [y_1, y_2, \ldots \ldots, y_N] \in \mathbf{R}^{\mathbf{L}}$
Output: $\widehat{\xi}$, i.e. the estimated noise matrix of dimensions N×L and the noise correlation matrix, $\widehat{R}_n$

1) $Z = Y^T$, $\widehat{R} = Z^T Z$
2) $R' = \widehat{R}^{-1}$
3) for i=1 to L
4) $\widehat{\beta}_i = ([R']_{\partial_i,\partial_i}) - [R']_{\partial_i,i}[R']_{i,\partial_i}/[R']_{i,i})[\widehat{R}]_{\partial_i,i}$; regression vector
5) $\widehat{\xi}_i = z_i - Z_{\partial i}\,\widehat{\beta}_i$; noise estimates
6) end for
7) $\widehat{R}_n = [\widehat{\xi}_1, \ldots \ldots, \widehat{\xi}_N]^T [\widehat{\xi}_1, \ldots \ldots, \widehat{\xi}_N]/N$; Noise correlation matrix estimate

---

where, $[R']_{\partial_i,\partial_i}$ - matrix obtained by removing the ith row and ith column from $\widehat{R}$

$[R']_{i,\partial_i}$ - ith row of $[\widehat{R}]_{:,\partial_i}$

$[R']_{\partial_i,i}$ - $[\widehat{R}]^T_{i,\partial_i}$

4.2.2. Signal Subspace Identification

Signal subspace estimation starts by computing the noise and signal correlation matrices. A subset of the eigenvectors of the signal correlation matrix is used to represent the subspace. This signal subspace is determined by minimizing the mean square error between the original signal, x, and the noisy projection of it i.e. the observed spectral vector. The signal correlation matrix is given by, $\hat{R}_x$

$$\hat{R}_x = [\hat{x}_1, \dots \dots, \hat{x}_N][\hat{x}_1, \dots \dots, \hat{x}_N]^T/N \tag{4.7}$$

where, $\hat{x}$ - signal estimates obtained after subtracting the noise estimates from the original data.

The eigenvectors can be obtained by performing the eigen decomposition of the signal correlation matrix as given by,

$$\hat{R}_x = E\textstyle\sum E^T \tag{4.8}$$

where, $E = [e_1, \dots \dots, e_L]$ is the eigenvector matrix of $\hat{R}_x$,

and $\sum$ - eigenvalues matrix of the signal correlation matrix, with the diagonal values ordered in decreasing magnitude.

Now let the space $\mathbf{R}^L$ be decomposed into two orthogonal subspaces, the k-dimensional subspace, $\langle E_k \rangle$, be represented by $E_k \equiv [e_{i_1}, \dots \dots, e_{i_k}]$ and $\langle E_k \rangle^\perp$ be the orthogonal component of subspace $\langle E_k \rangle$, spanned by $E_k^T \equiv [e_{i_{k+1}}, \dots \dots, e_{i_L}]$.

Let $U_k = E_k E_k^T$ represent the projection matrix onto the subspace $\langle E_k \rangle$, then the projection of the observed spectral vectors, y, or the noisy projection of x, onto the subspace $\langle E_k \rangle$ is given by,

$$\hat{x}_k \equiv U_k y$$

The first order moment of $\hat{x}_k$ given x is,

$$\mathbb{E}[\hat{x}_k|x] = U_k\mathbb{E}[y|x] = U_k\mathbb{E}[x+n|x] = U_k x = x_k \tag{4.9}$$

where, x_k is the projection of the signal vectors onto the subspace $\langle E_k \rangle$.

And the second order moment of $\hat{x}_k$ given x is,

$$\mathbb{E}[(\hat{x}_k - x_k)(\hat{x}_k - x_k)^T|x] = \mathbb{E}[(U_k y - U_k x)(U_k y - U_k x)^T|x] = \mathbb{E}\left[\left(U_k n n^T {U_k}^T\right)|x\right] = U_k \hat{R}_n {U_k}^T \tag{4.10}$$

The mean square estimation between x and $\hat{x}_k$ is given by,

$$\begin{aligned} mse(k|x) &= \mathbb{E}[(x-\hat{x}_k)^T(x-\hat{x}_k)|x] = \mathbb{E}[(x - x_k - U_k n)^T(x - x_k - U_k n)|x] \\ &= (x-x_k)^T(x-x_k) + (U_k\hat{R}_n {U_k}^T)^T \end{aligned} \tag{4.11}$$

Since, $U_k x = x_k$, implies that $x - x_k = U_k{}^{\perp}$, which is the orthogonal component of the signal projection vector onto the subspace $\langle E_k \rangle$. Thus by using the projection matrix properties i.e. $U = U^T$, $U^2 = U$ and $U^T = I - U$, we have

$$mse(k) = \mathbb{E}[(U_k{}^{\perp}x)^T(U_k{}^{\perp}x)] + (U_k\hat{R}_nU_k{}^T)^T$$

$$= (U_k{}^{\perp}R_x)^T + (U_k\hat{R}_n)^T$$

$$= (U_k{}^{\perp}\hat{R}_y)^T + 2(U_k\hat{R}_n)^T + c \tag{4.12}$$

where c is an irrelevant constant. The signal subspace $\langle E_k \rangle$ is inferred by the minimization of the mean square error given by (4.13) with respect to all the permutations $\pi = \{i_{1\,......,}i_L\}$ and is given by the expression,

$$(\hat{k}, \hat{\pi}) = argmin\{(U_k{}^{\perp}\hat{R}_y)^T + 2(U_k\hat{R}_n)^T\} \tag{4.13}$$

where $\hat{k}$ is the estimate of the subspace and the subspace is spanned by $E_k \equiv [e_{i_1}, \dots \dots, e_{i_k}]$. Now the first term of the equation (4.12) corresponds to the projection error power which is a decreasing function of subspace dimension and the second term corresponds to the noise power and is increasing function of subspace dimension. As mentioned above $U_k = E_k E_k^T$ is a projection matrix and from matrices properties we know that $(AB)^T = (BA)^T$, then the minimization equation (4.14) can be written as

$$(\hat{k}, \hat{\pi}) = argmin\left\{c + \sum_{j=1}^{k}(-p_{ij} + 2\sigma_{ij}^2)\right\} \tag{4.14}$$

where c is a constant and

$$p_{ij} = e_{ij}^T\hat{R}_y e_{ij} \tag{4.15}$$

$$\sigma_{ij}^2 = e_{ij}^T\hat{R}_n e_{ij} \tag{4.16}$$

The term on the right hand side $(-p_{ij} + 2\sigma_{ij}^2)$ is represented by δ_{ij} and by including all the negative terms of $\hat{\delta}_i$, for $i = 1, \dots \dots, L,$ in the sum, the minimization of mean square error between projection power error and noise power is obtained. The estimate of the subspace dimension is given by the number of negative terms in $\hat{\delta}_i$. The pseudo code for HySime Subspace Identification is given in Algorithm 2.

HySime is computed for a subset size of 256x640x158 of the Dehradun area of the Hyperion Image. The subspace dimension, $\hat{k}$, for this subset is obtained by minimizing the mean square error between the signal projection power and the noise projection. The output of the signal subspace estimation step is a set of matrices containing the eigenvalues and the eigenvectors spanning the signal subspace.

Algorithm 2: HySime, Subspace Identification, [7]

Input: Observed spectral vectors $Y = [y_1, y_2, \ldots \ldots, y_N]$
Output: signal subspace

1) $Y = [y_1, y_2, \ldots \ldots, y_N]$, $\hat{R}_y \equiv (YY^T)/N$
2) $\hat{R}_n = \frac{1}{N}\sum_i \left(\hat{\xi}_i \hat{\xi}_i^{\,T}\right)$; Noise Estimates
3) $\hat{R}_x = \frac{1}{N}\sum_i \left(\left(y_i - \hat{\xi}_i\right)\left(y_i - \hat{\xi}_i^{\,T}\right)\right)$; Signal correlation matrix estimates
4) $E = [e_1, \ldots \ldots, e_L]$; Eigenvectors of the signal correlation matrix
5) $\delta_i = [\delta_1, \ldots \ldots, \delta_L]$
6) $(\hat{\delta}_i, \hat{\pi}) = sort(\delta)$; sorts the $\hat{\delta}_i$ terms in ascending order
7) $\hat{k}$ = number of negative terms, $\hat{\delta}_i < 0$; Signal subspace estimate
8) $E_k \equiv [e_{i_1}, \ldots \ldots, e_{i_k}]$; Eigenvectors spanning the signal subspace

4.2.3. HySime Components

The output of the HySime algorithm gives us the signal subspace estimates and the corresponding eigenvectors spanning the subspace i.e. $E_k \equiv [e_{i_1}, \ldots \ldots, e_{i_k}]$, sorted in descending order of their relevance. The HySime components, with the first component representing the maximum variance, corresponding to these eigenvectors can be computed by multiplying the eigenvector matrix, E, (can also referred to as transformation matrix or projection matrix) with the original image i.e. by projecting the original image by E. The components which are obtained will be ordered according to the decreasing variability. Each column of the eigenvector matrix produces a component image. The transformation can be achieved by the following expression [46],

$$newBV_{i,j,p}(Y^{HySime}) = \sum_{i=1}^{n}\left(E_{k,p} BV_{i,j,k}\right) \tag{4.17}$$

where, $E_{k,p}$ - Eigenvector matrix
$BV_{i,j,k}$ - Brightness value of pixel at i[th] row, j[th] column of the band k of original image
$newBV_{i,j,p}$ - Brightness value of pixel at i[th] row, j[th] column of the p[th] HySime component
Y^{HySime} - HySime component image

The HySime component image, Y^{HySime}, was obtained by using the HySime output of eigenvectors spanning the subspace in the expression 4.19. The number of components in the HySime component image, Y^{HySime}, will generally be equal to the subspace dimension inferred. However visual inspection of the components obtained must be carried out to decide the number of component images to be used for any further analysis.

4.2.4. Inverse HySime for Hyperspectral image restoration

Once the HySime components have been obtained and the noise segregated the Hyperion data can be restored to its original spectral space without noise. The noise free original spectral space consisting of the noise less signals only can be achieved by performing an inverse HySime transform. As the original image data *Y* was transformed into HySime components, Y^{HySime} in the HySime space, the inverse transformation can be achieved by inverting the projection matrix E and multiplying it with

Y^{HySime} . The expression for restoration of signals to original spectral space is given in expression 4.18 and was implemented as an extension to the original HySime algorithm.

$$Z = (E^T)^{-1} \, Y^{HySime} \tag{4.18}$$

where, $Y^{HySime} = \left(y_1^{HySime}, \dots \dots, y_p^{HySime}, 0, \dots \dots, 0\right)$ - is the HySime component image
p - the number of 1st components selected for restoring to the original spectral space
and Z - the restored image

4.3. Spatial-Spectral Endmember Extraction

Two of the most noted attempts in this direction of integrating spatial information with the spectrally rich hyperspectral datasets are Automatic morphological endmember extraction (AMEE) [17] and spatial-spectral endmember extraction tool (SSEE) [19]. An approach based on the SSEE model is adopted for this study and is discussed further in this chapter.

Image endmember extraction techniques based on spectral based approaches differentiate between the signals sources based on the spectral properties only. No weightage is given to the spatial distribution of the image endmembers. In spectral based endmember extraction algorithms, the endmembers having high spectral contrast (distinct spectral features) are detected easily; however problems are encountered in the detection of endmembers having low spectral contrast with respect to the whole image [19]. It might be possible that the endmembers having low spectral contrast with the whole image may be having high local spectral contrast thus increasing their chances of detection when a algorithm is applied to utilize the local contrast of the endmembers. This is the idea behind the SSEE [19] model used in this work. The SSEE model divides the hyperspectral image into subsets so as to enhance the local spectral contrast of the endmembers, thus improving their odds for selection. The algorithm for spatial spectral endmember extraction from step 2 onwards was written in IDL programming tools within ENVI™ environment.

The spatial spectral endmember extraction process consists of four steps:

- Determination of the set of eigenvectors explaining maximum variance from the image subsets
- Projection of the image data onto these eigenvectors
- Spatial constraints to average spectrally similar candidates
- Ordering of endmembers with respect to their spectral similarity and spatial location

4.3.1. Step 1: Eigenvector Determination

Step 1 of the spatial spectral integration is to obtain a set of eigenvectors that explain maximum spectral variability of a given image. In this work we have used HySime for obtaining this set of eigenvectors from subsets of the image, instead of SVD as in the original SSEE algorithm. HySime is selected owing to its better estimation of the eigenvectors which explain the variability of the data.

The two major input parameters in this step are the subset size and the HySime threshold value, i.e. the number of eigenvectors to be retained from each subset. Subset of the images is used to increase the relative spectral contrast of the endmembers. HySime is applied to subsets of an image which are square and non-overlapping. For each subset the eigenvectors accounting for majority (around 99.9%) of the spectral variability are retained and compiled into a single eigenvectors matrix, E_{ev} where ev is the total number of eigenvectors retained. These eigenvectors represent the local high contrast endmembers of the respective image subsets.

Figure 4.2 shows an example of an image with three regions (A) and square and non overlapping image subsets created subsequently (B), for determining eigenvectors.

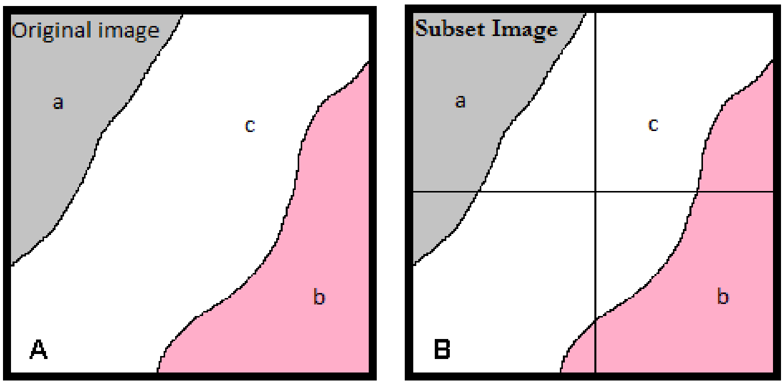


Figure 4.2 Step 1: (A) Original Image, (B) Image subsets (Source: Rogge et al. [19])

The size of the subset can be varied based on the characteristics of the image. Larger subsets can be used for homogeneous image regions, whereas for heterogeneous or complex image regions smaller subsets should be used. Also the use of larger subsets limits the ability to extract low contrast endmembers from the image. And the use of smaller subsets would result into a large number of eigenvectors being retained. Keeping this in mind a subset size of 32 pixels was chosen for this study. The minimum number of vectors retained from each subset should be 2 and maximum should be the number of eigenvectors explaining 99.9 % of spectral variability. However after visually examining the eigenvectors matrix spanning the subspace it was found that 99.9% of the variability was explained by first few eigenvectors only, so the minimum number of vectors to be retained from each subset was set accordingly.

4.3.2. Step 2: Projecting Image data onto Eigenvectors

In step 2, the full image data is projected onto each eigenvector in the compiled eigenvector matrix. The data cloud lies along the eigenvector and the endmembers or extreme pixels are those pixels that are lying at either extremes of the projection. These pixels are labelled as the candidate endmember pixels and or just candidate pixels. The following expression was used for the projection of the observed spectral vectors (image data), $Y = [y_1, y_2, \dots \dots, y_N]$, onto the compiled eigenvector matrix, E_c^T,

$$P_{mat} = E_{ev}^T\, Y \tag{4.19}$$

where, E_c^T - transpose of the compiled eigenvector matrix
Y - Observed spectral vectors (image data)
P_{mat} - matrix obtained after projecting Y onto E_{ev}^T

The projection the image data onto the compiled eigenvector matrix results into another matrix of projected data. Then it looks into the matrix for pixels on the extremes of the projection and pixels at the extremes of the vectors are assigned a hit. Pixels receiving more hits are designated as candidate endmember pixels.

Figure 4.3 depicts the candidate endmember pixels obtained after projecting the image data onto the eigenvectors and selecting the extreme pixels. The pseudo code for step 2 is given in pseudo code 1 and the IDL code is presented in Appendix B.

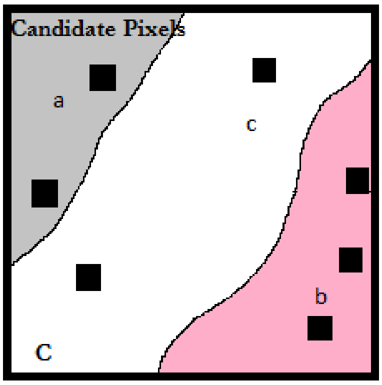


Figure 4.3 (C) Candidate Endmember Pixels (black squares) (Source: Rogge et al. [19])

Pseudo code 1: Spatial Spectral Endmember Extraction, Step 2

Input: Observed Spectral Vectors (Image data), Y, and Compiled Eigenvector matrix from Step1, E_c and number of pixels to be retained at each extreme

Output: Candidate endmember pixels,

1) $E_{Trans} = E_{ev}^T$; transpose of eigenvector matrix
2) $P_{mat} = E_{ev}^T\, Y$; computation of projection matrix
3) $for\ k\ =\ 0\ to\ ev - 1$
4) $\quad Extreme_{pix\_id\_} = sort(P_{mat})$; Finds the extreme pixels
5) $\quad Assign\ hit\ value\ equal\ to\ 1\ to\ the\ extreme\ pixels$
6) $endfor$
7) Number of pixels having more hits are designated as candidate endmember pixels

4.3.3. Step 3: Spatial Analysis

In step 3 a sliding window of the size equal to the subset size is used to scan the image. The sliding window scans the image checking whether the centre pixel of the image is a candidate endmember pixel or not. As soon as a candidate endmember pixel is encountered, spectral angle is computed between the candidate endmember pixel (reference pixel) and all the other pixels (target pixels) in the window according to the equation 2.1. Those pixels for which the spectral angle is less than the given threshold value for minimum spectral angle are the most similar to centre pixel and are labelled as the updated candidate endmember pixels as shown in Figure 4.4.

The centre pixel i.e. the candidate endmember pixel is then averaged with all the other endmember pixels that are spectrally similar within the window. The average value is assigned to the centre endmember pixel. For reducing the effects of noise and for determination of endmembers which are spectrally similar and spatially similar, the averaging process may be repeated for n number of iterations, thus reducing the variance of the endmember clusters. An example of the updated candidate endmember pixels and the averaging window being used is shown in figure 4.4 (D) and 4.4 (E) respectively. The pseudo code for step 3 is shown in pseudo code 2.

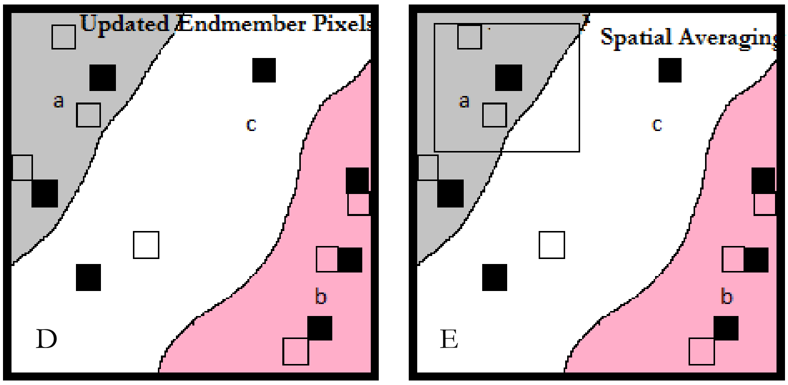


Figure 4.4 (D) Updated Candidate endmember pixels (empty squares), (E) Spatial averaging (Source: Rogge et al. [19])

Pseudo code 2: Spatial Spectral Endmember Extraction, Step 3

Input: Image Data, Candidate endmember pixels (CEM) data from step 3, threshold value for spectral angle (t_{sa}), sliding window size, number of iterations (n)
Output: Averaged candidate endmember pixel spectra, number of averaged candidate endmember pixels (count)

1) Select sliding window size = subset size defined in step 1
2) count = number of candidate endmember pixels,
3) Place window on the CEM (reference endmember pixel, centre pixel)
4) $for\ iterations\ =\ 0\ to\ n$,
5) $for\ i\ =\ 0\ to\ count-1,$
   compute spectral angle between candidate endmember pixel (reference pixel) and all the other pixels (target pixels)

$$\text{Spectral angle, } cos\ \varphi = \frac{\sum_{i=1}^{n} t_i r_i}{\sqrt{\sum_{i=1}^{n} t_i^2}\sqrt{\sum_{i=1}^{n} r_i^2}}$$

6) if $\varphi < t_{sa}$
7) update the target pixel as candidate endmember pixels
8) $end\ for$
9) Compute average and store the average spectrum in centre pixel
10) $end\ for$

So after the averaging process the candidate endmembers will contain the average spectrum of the endmember pixels within the window thus reducing the noise and interclass variability of the candidate endmember pixels. These endmembers are then utilized further for classification and mapping purposes.

4.3.4. Step 4: Reordering endmembers

The set of endmembers obtained from step 3 is then reshuffled based on spectral angle. The spectrum of the first endmember is compared with the spectrum of all the other endmembers and the endmembers are ordered in the ascending order of their spectral angle. The reordering process is recursive in nature in the sense that each endmember is compare with every other endmember and the reordered endmembers can be stored as a matrix along with their spatial coordinates. The reordered endmembers still retain their image coordinates. The reordered endmembers are then grouped into endmember classes based on their spectral and spatial similarity. The endmembers are then visually examined for spectral similarity and similar spatial coordinates and are grouped into different endmember classes.

The pseudo code for the implementation of step 4 of spatial spectral endmember extraction method is given in the pseudo code 3.

Pseudo code 3: Spatial Spectral Endmember Extraction, Step 4

Input: Averaged Endmember spectra, count
Output: Reordered endmembers based on spectral angle

1) Assign the first endmember in the average spectrum file as first spectra (reference spectra)
2) $for\ i\ =\ 0\ to\ count-1$
3) $\quad for\ j\ =\ i+1\ to\ count-1$
4) $\quad\quad$ Compute φ between the first spectrum and all other endmember spectra
5) $\quad end\ for$
6) $end\ for$
7) The reordered list is stored as a matrix
8) Endmembers can be grouped according to their spectral and spatial similarity

4.4. Spectral Unmixing

Once the average spectrum of the endmembers grouped by their similar spectral and spatial characteristics has been obtained, the next step is to unmix the data into the respective endmember classes using the spectra of the obtained endmember pixels. To determine whether the obtained endmembers by spatial spectral integration show any meaningful spatial distribution spectral unmixing is applied using the ENVI software. The result of the unmixing step will be the fraction abundance images for each representative class.

4.5. Validation

The validation of the results will be performed by visually examining the endmembers obtained from the spatial integration process against a high resolution dataset of Dehradun, IRS P6 LISS-4 image of Dehradun area and field photographs of the features captured during earlier field visits and available from the Photogrammetry & Remote Sensing Division (PRSD) at IIRS.. The locations of the pure classes obtained are mapped to show the physical representation of the endmembers.

5. RESULTS AND DISCUSSIONS

This chapter focuses on the results obtained from the methods discussed in chapter 4. Different section discusses about different experiments on Hyperion dataset and the results obtained from HySime and the spatial spectral endmember extraction. The chapter follows the methodology adopted and discusses the results obtained in each step.

5.1. HySime: Noise Estimation and Eigenanalysis

The signal subspace estimation is performed on the Hyperion (radiance) dataset of Dehradun area, to determine the number of spectrally distinct sources in hyperspectral signal or the intrinsic dimensionality of the dataset. The spatial subset size used for the experiment is 256 bands×640 lines and the spectral subset of 158 bands is taken.

For a reliable estimation of the signal subspace or dimensionality of the dataset the random noise present in the dataset must be estimated accurately, to segregate it from signal information. For hyperspectral datasets a common approach for dimensionality reduction or signal subspace identification is the application of eigen decomposition based techniques, such as PCA, MNF or SVD. The eigenvalues obtained from these techniques are still representing the mixtures of the signal sources and the noise present in the data, which pose difficulty in obtaining reliable noise estimates as the signal sources are too weak and their contribution towards the computation of eigenvalues is very less.

As mentioned in chapter 4, HySime uses a multiple regression theory based approach for the estimation of noise from hyperspectral datasets and efficiently segregates noise from the signals. This can be explained by observing the eigenvalues and determining the spectral energy explained by the first few eigenvalues. The eigenvalues are ordered in the decreasing order of their magnitude. By calculating the cumulative percentage of all the eigenvalues, the percentage of variability explained by the first few eigenvalues can be determined.

From the Table 5.1 and the subsequent graphs in Figure 5.1, it can be observed that in case of HySime and MNF there is a sudden drop magnitude after the first few eigenvalues indicating the presence of most of the spectral energy in the first few components. Whereas in case of SVD there is gradual decrease, thus indicating the presence of noise mixed with the signal information, as evident from Figure 5.1.

Table 5.1 below shows the first 20 eigenvalues obtained from MNF, SVD and HySime and the corresponding percentage of cumulative spectral energy explained by these eigenvalues.

From the Table 5.1 we can observe that in case of HySime the first 7 eigenvalues contains more than 99.95% of the total spectral energy contained in the dataset. On the other hand the percentage of spectral energy explained by the first 7 eigenvalues of MNF and SVD is 96.63% and 83.47% respectively which indicates the presence of noisy signals. Thus in case of HySime the bulk of the spectral energy is explained by a small number of eigenvectors corresponding to the first 7 eigenvalues. This can be further explained by plotting the accumulated signal energy against eigenvalues index numbers in decreasing order. Figure 5.1 (a), (b) and (c) show the plots of percentage of spectral energy against the eigenvalues index for MNF, SVD and HySime respectively.

Table 5.1 Percentage Spectral Energy explained by Eigenvalues (MNF, SVD and HySime)

Sl No	MNF Eigenvalues	%age of Spectral Energy explained by Eigenvalues	SVD Eigenvalues	%age of Spectral Energy explained by Eigenvalues	HySime Eigenvalues	%age of Spectral Energy explained by Eigenvalues
1	1541662.38	77.02	3321243.40	65.48	168314510.00	97.61
2	214695.35	87.75	444897.85	74.26	3019030.13	99.36
3	119027.33	93.70	222513.90	78.64	755039.31	99.80
4	26006.09	94.99	110716.90	80.83	185715.55	99.91
5	16907.72	95.84	56405.54	81.94	46271.27	99.94
6	9912.30	96.33	44104.73	82.81	22031.18	99.95
7	5966.85	96.63	34034.12	83.48	12632.67	99.96
8	4920.32	96.88	27954.65	84.03	10526.32	99.96
9	4237.02	97.09	27318.29	84.57	9352.28	99.97
10	3550.42	97.27	24012.09	85.04	6016.99	99.97
11	3220.14	97.43	22257.25	85.48	4782.03	99.97
12	3083.21	97.58	21234.01	85.90	4354.03	99.98
13	2770.45	97.72	19330.88	86.28	3009.91	99.98
14	2450.91	97.84	18784.18	86.65	2837.40	99.98
15	2384.32	97.96	18012.87	87.01	2657.69	99.98
16	2134.66	98.07	17570.90	87.35	2362.34	99.98
17	1951.66	98.17	15149.22	87.65	2318.32	99.99
18	1653.91	98.25	14540.29	87.94	2104.39	99.99
19	1610.21	98.33	14269.19	88.22	1658.81	99.99
20	1492.47	98.40	13861.15	88.49	1346.20	99.99

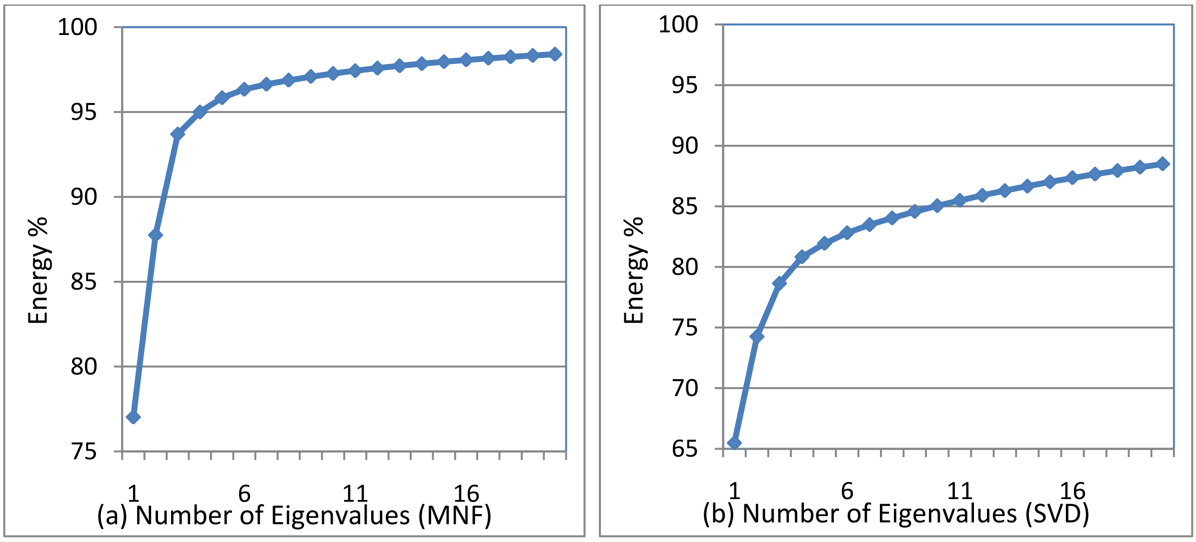


Figure 5.1 Percentage of spectral energy explained vs. number of eigenvalues (a) MNF and (b) SVD

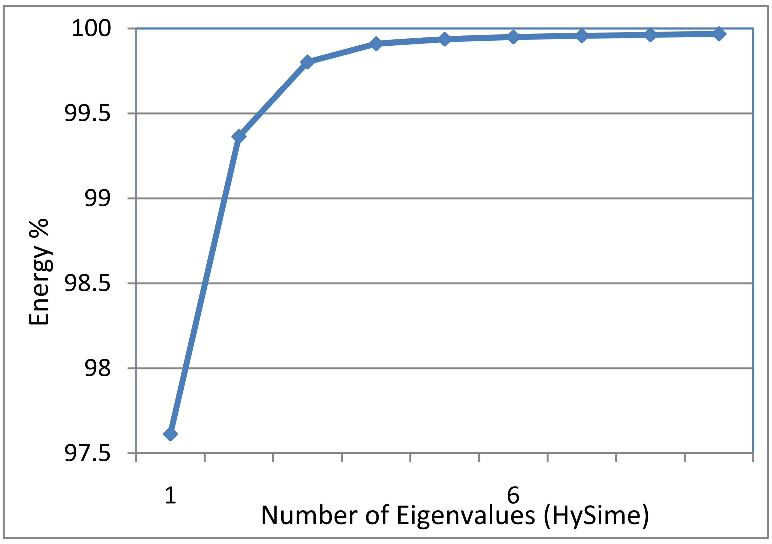


Figure 5.2 Percentage of spectral energy explained vs. number of eigenvalues, HySime

5.2. HySime: Signal Subspace Estimation

Once the noise has been estimated the next step is the estimation of the true dimensionality or the signal subspace. As mentioned in chapter 4 the signal dimension is estimated by minimizing the mean square error between the original signal and the noisy projection of it, as in expression 4.16 and the subspace dimension, $\hat{k}$, is given by the number of negative terms in the minimization. Figure 5.3 below shows a plot of the mean square error as a function of subspace dimension $\hat{k}$. The minimization for the Hyperion subset of 256×640×158 is obtained for $\hat{k}$ = 26 (depicted by a small circle), after which the mean square error curve begins to rise again.

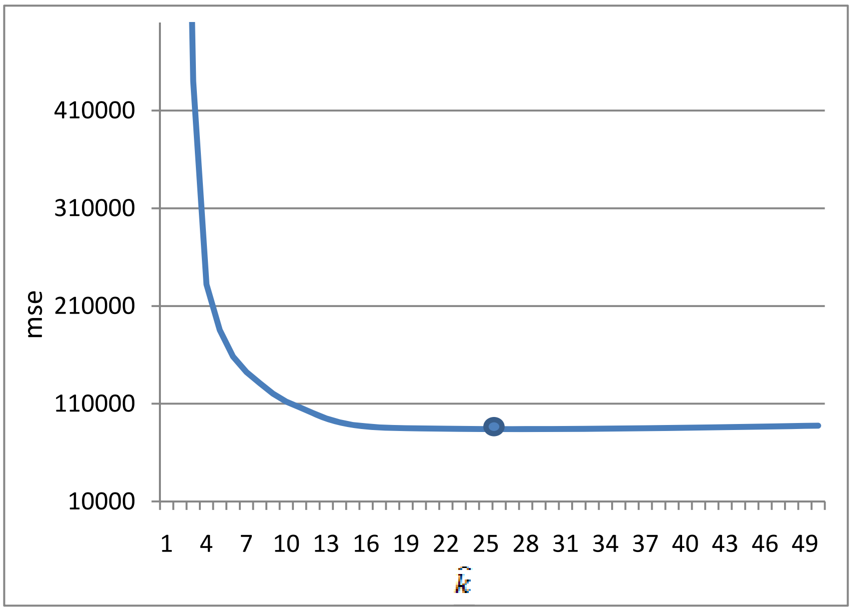


Figure 5.3 Mean square error vs. $\hat{k}$. for the Hyperion data of Dehradun area

The Figure 5.4 shows the mean square error vs. $\widehat{k}$ plot generated by the MATLAB implementation of HySime. The plot shows the mean square error curve along with the projection error and the noise power error as a function of subspace dimension.

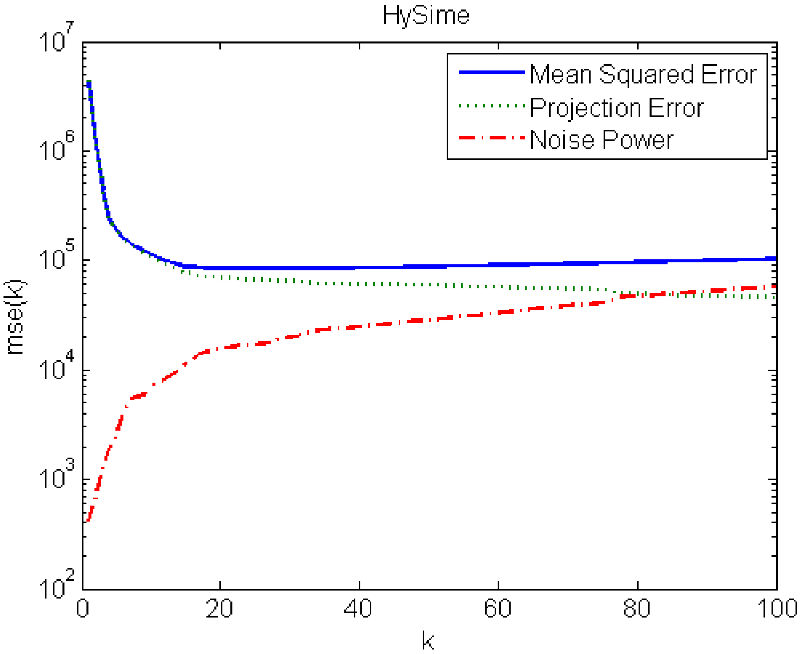


Figure 5.4 Mean square error vs. $\widehat{k}$ plot: MATLAB output

The signal subspace dimension, $\widehat{k}$, is estimated to be 26, which means there are 26 spectrally distinct signal sources in the dataset. However it may not always be the case. The subspace dimension is largely overestimated by HySime for spaceborne hyperspectral data and the reason for this is the spectral variability of the Hyperion dataset and a number of unknown signals present in the hyperspectral dataset such as unknown noise sources or interferers, unknown signatures etc. And the highly mixed spectrum of the Hyperion sensor and the area in question (Dehradun) adds to the problem. Thus HySime provides only an estimate of the number of spectrally distinct signal sources present in the spaceborne hyperspectral datasets and this can be used as a reference for determining the intrinsic dimensionality of the Hyperion dataset. Also the estimate of the number of spectrally distinct signal sources can be used further as an input for the existing endmember extraction algorithms.

5.3. HySime Components

Once the dimensionality of the Hyperion dataset is determined the different HySime components and the HySime component image, Y^{HySime}, corresponding to the eigenvectors spanning the subspace i.e. $E_k \equiv [e_{i_1}, \dots \dots, e_{i_k}]$, can be generated using expression 4.17. The cut-off value for the number of components to be selected for further analysis can be decided by examining the individual components visually or by using the components corresponding to the eigenvalues explaining maximum variability in the data. Based on both visual examination and from eigenvalues the first six bands of Y^{HySime} were found to be containing the maximum variability. Figure 5.5 shows the first five HySime components obtained after performing HySime. From table 5.1 it can be observed that the only the first five components explain about 99.94% of variability. So most of the signal information is in the first 5 or 6 components only and rest of the components being mostly noisy and therefore not used for analysis.

Figure 5.6 shows the first five components generated by MNF transform. A simple visual comparison also illustrates the better performance of HySime against MNF.

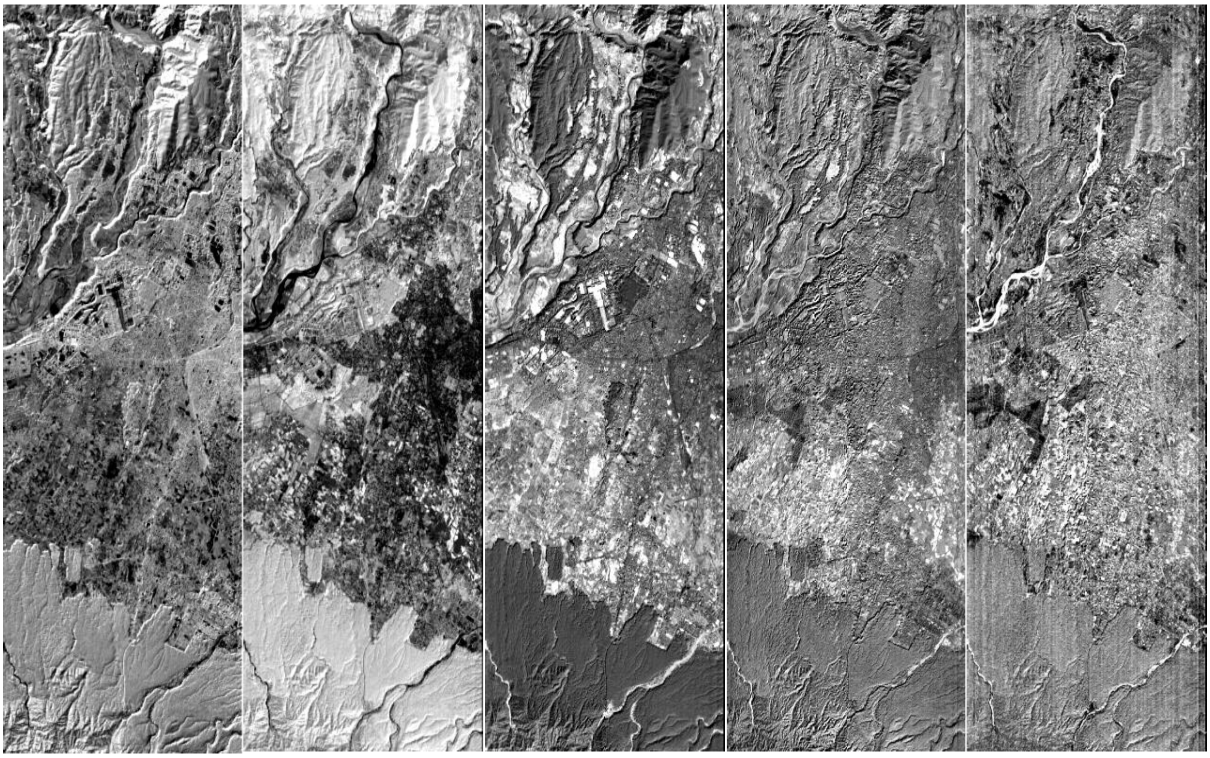


Figure 5.5 The first 5 HySime components, Hyperion

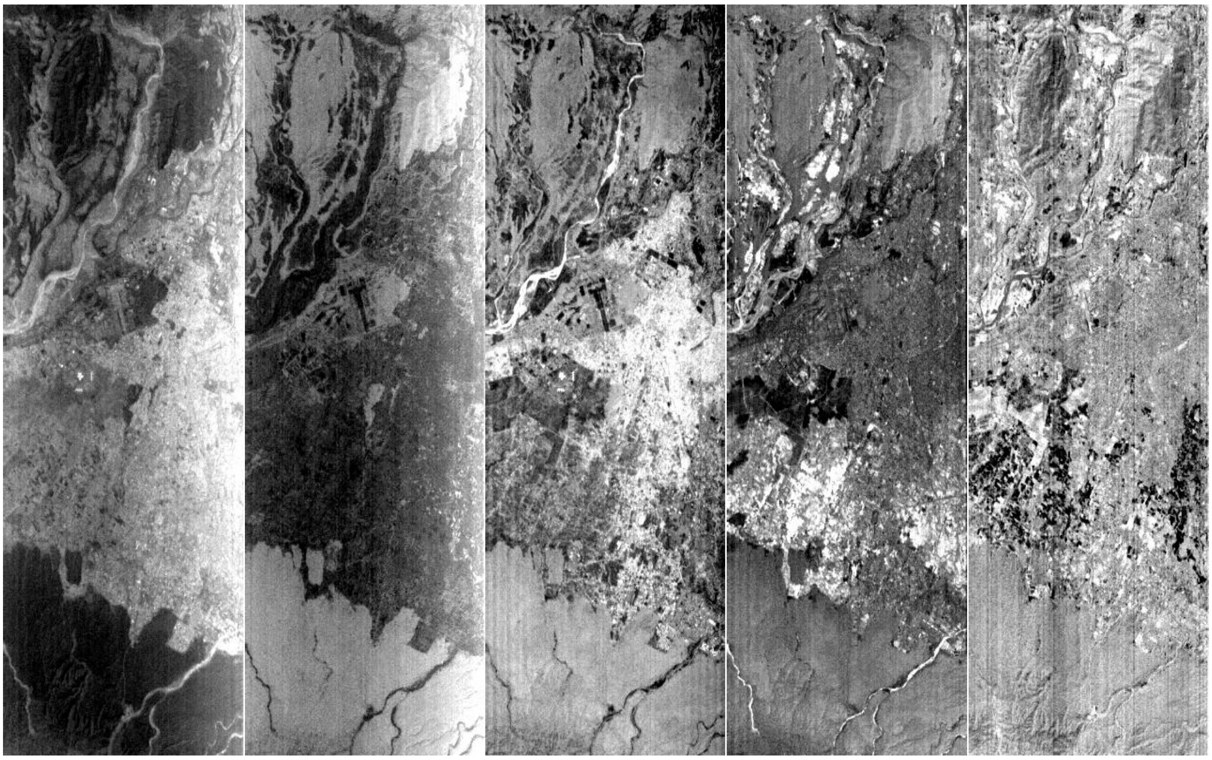
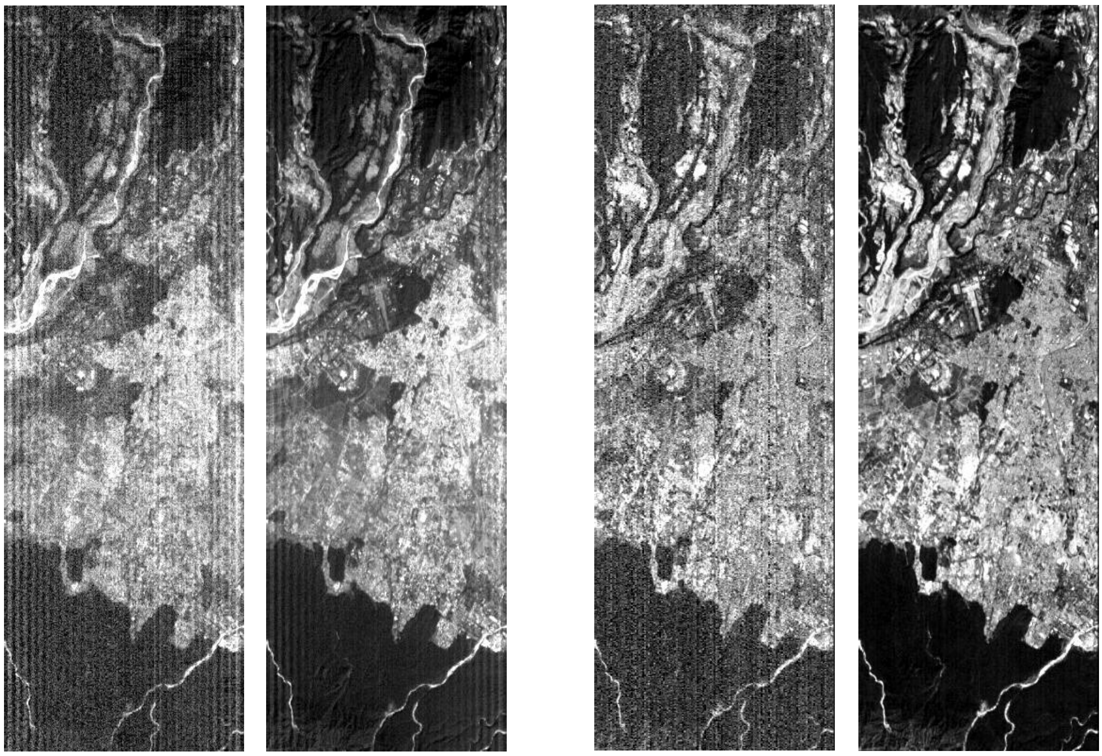


Figure 5.6 The first 5 MNF components, Hyperion

5.4. Inverse HySime

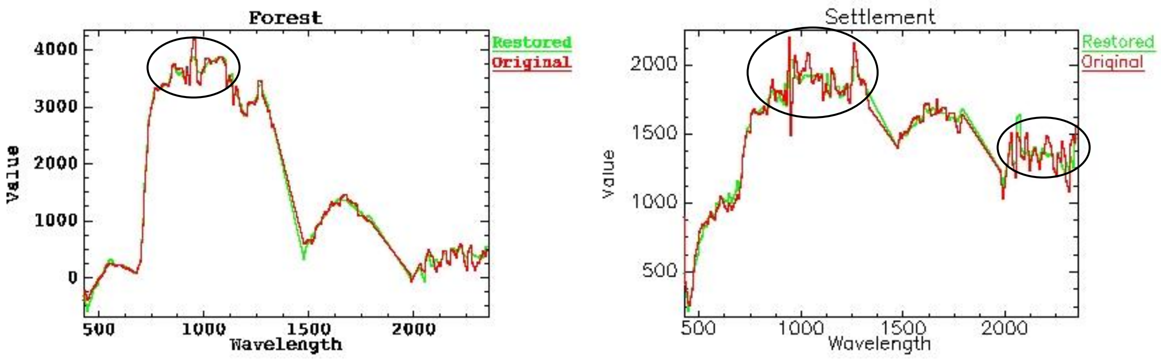
To illustrate the effective noise estimation procedure employed by HySime, the inverse HySime transform was also computed using the expression 4.18 to transform the image back into its original spectral space. The transformed image after performing the inverse HySime transform is much more noise free and cleaner than before restoration which can be observed by visual examination of different bands of Hyperion image before and after inverse HySime transform. Figure 5.7 and show the restored images for band 8 and band 220 of the Hyperion sensor against the images before restoration.



(a) (b) (c) (d)

Figure 5.7 Original spectral image of band 8 (a) and band 220 (c) and the corresponding images after restoration (b) and (d)

Figure 5.8 (a) shows spectra of a pixel selected from the forest class and Figure 5.8 (b) show the spectra of a pixel selected from settlement class before and after image restoration using Inverse HySime.



(a) Forest (b) Settlement

Figure 5.8 Spectral profile of Hyperion image before and after image restoration by Inverse HySime

5.5. Spatial-Spectral Iintegration

The integration of spatial spectral information has been approached in a way to increase the local spectral contrast of the endmembers. So the endmembers having low spectral contrast with respect to the whole image can have high local contrast within a given region, thus increasing their odds of selection. The integration is achieved by adopting the SSEE model for spatial-spectral integration as described in chapter four.

The **step1** of the spatial spectral integration process produces a set of eigenvectors and compiled into a single vector file by selecting the eigenvectors from the image subsets that explain 99.9% variability of the subset data. After examining the eigenvectors, the number of subsets to be retained from each subset was set to 4. These eigenvectors represent the local high contrast endmembers within the subset. An example showing the compiled eigenvector matrix for the first two subsets is given in Table 5.2.

Table 5.2 Eigenvector matrix for the first two subsets of Hyperion Image (first 20 values out of 158)

Eigenvectors, Subset 1				Eigenvectors, Subset 2			
-0.1048	-0.2540	-0.1949	-0.1143	-0.1056	-0.2989	-0.0216	0.0695
-0.1013	-0.2461	-0.1814	-0.1081	-0.1012	-0.2870	0.0102	0.0909
-0.0969	-0.2320	-0.1664	-0.1052	-0.0966	-0.2689	-0.0082	0.1145
-0.0978	-0.2335	-0.1545	-0.0919	-0.0972	-0.2689	-0.0024	0.0891
-0.0962	-0.2301	-0.1414	-0.0832	-0.0951	-0.2632	0.0015	0.0823
-0.0968	-0.2324	-0.1335	-0.0816	-0.0954	-0.2637	-0.0006	0.0844
-0.0866	-0.2084	-0.1014	-0.0620	-0.0841	-0.2302	0.0055	0.0614
-0.0835	-0.2024	-0.0820	-0.0472	-0.0805	-0.2185	0.0125	0.0539
-0.0811	-0.1959	-0.0637	-0.0341	-0.0776	-0.2056	0.0141	0.0464
-0.0754	-0.1773	-0.0442	-0.0140	-0.0715	-0.1830	0.0208	0.0268
-0.0773	-0.1718	-0.0311	0.0082	-0.0731	-0.1764	0.0199	0.0019
-0.0766	-0.1604	-0.0218	0.0258	-0.0723	-0.1624	0.0184	-0.0134
-0.0758	-0.1525	-0.0098	0.0413	-0.0714	-0.1507	0.0184	-0.0262
-0.0737	-0.1436	0.0044	0.0535	-0.0688	-0.1385	0.0230	-0.0386
-0.0684	-0.1358	0.0195	0.0546	-0.0631	-0.1266	0.0264	-0.0437
-0.0634	-0.1319	0.0368	0.0548	-0.0575	-0.1179	0.0400	-0.0505
-0.0617	-0.1319	0.0523	0.0566	-0.0549	-0.1125	0.0467	-0.0476
-0.0584	-0.1291	0.0657	0.0636	-0.0514	-0.1030	0.0517	-0.0586
-0.0570	-0.1284	0.0782	0.0642	-0.0493	-0.0995	0.0565	-0.0662

The **step 2** of the spatial spectral integration process projects the image data onto the compiled eigenvector set and the output is a set of initial candidate endmember pixels. The white dots in the figure 5.9 (b) represents the spatial distribution of the initial candidate endmembers estimated by the projection of the image data onto eigenvectors. These are the pixels that lie at the extremes of the projection and are retained. The vectors onto which the image data is projected are non random in nature and it makes the whole process repeatable, and this is the primary benefit of this approach over PPI.

In **step 3**, a sliding window equal to the size of the subset scans the image, traversing the initial candidate endmember pixels. By placing the window with the candidate endmember pixel at its centre the spectral angle distance (SAD) is computed between the centre pixel and all the other pixels in the window. The pixels within the window which were similar, based on minimum spectral angle distance were labelled as updated candidate endmember pixels. The threshold for the minimum spectral angle for spectral similarity was set to 0.1 radians in this study. Figure 5.9 (c) shows the spectral angle distance (SAD) score image and gives an illustration how the sliding window is moving.

The pixels which were similar based on the minimum spectral angle are then averaged and their value stored in the centre pixel. This process can be repeated for a number of iterations. The numbers of iterations performed were 5. After the number of iterations of the averaging process an image containing the average spectra of endmembers after averaging process is generated. These endmembers are much cleaner and have reduced interclass variance.

Figure 5.9 below show the images of the initial candidate endmembers from step 2 and the spectral angle distance score map from step 3 along with an false colour combination (FCC) of the dataset.

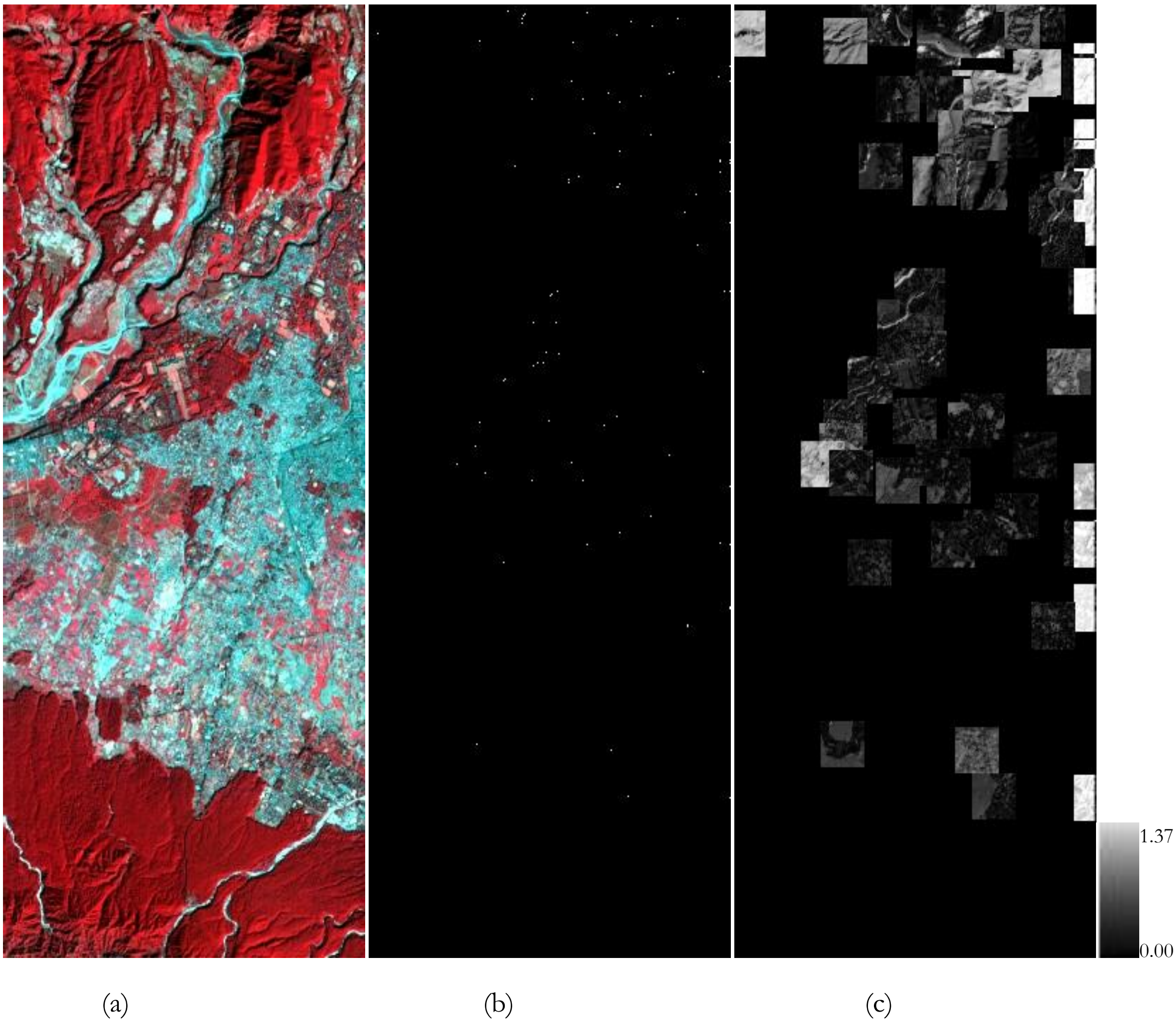
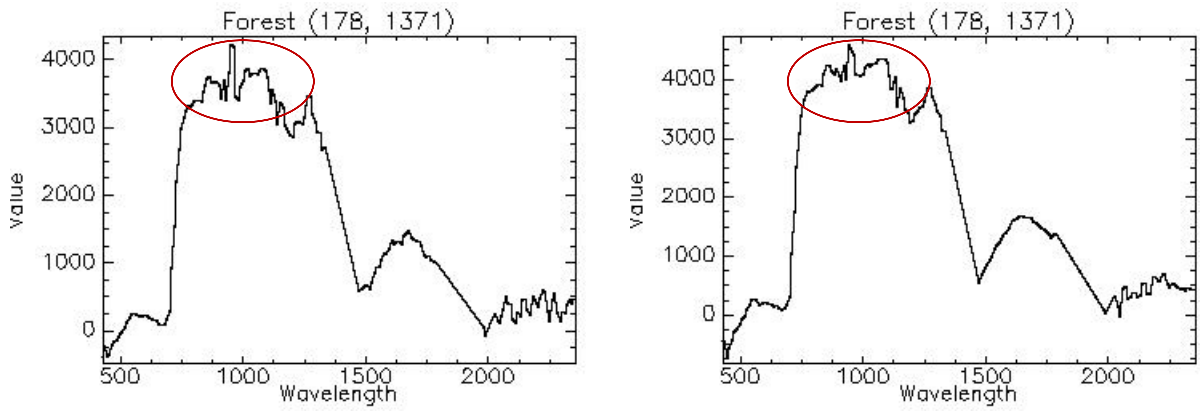


Figure 5.9 a) FCC of the Hyperion Image, b) Spatial distribution of the Candidate endmember pixels, and c) spectral angle distance score (in radians) of endmember pixels within subset size

Step 3 results in a set of averaged candidate endmember pixels which are spectrally similar and are also defined spatially owing to the averaging process. The endmembers obtained after the averaging process are more noise free than before. This can be illustrated by plotting the spectral curve of the particular endmember before and after averaging process. Figure 5.11 shows the spectral profile of an endmember pixel taken from forest class, a) original endmember spectra and b) is the spectra obtained after averaging. The curve has been smoothened around the 1000 nm, 1650 nm and 2150 nm range.



(a) original spectrum (b) spectra after averaging

Figure 5.10 Spectra of a forest class endmember, (a) Original spectra and (b) Spectra after averaging process in step 3.

In **step 4**, the endmembers obtained from step 3 are reordered based on their spectral angle. The first endmember is designated as the first endmember and then its spectrum is compared with all other endmembers. The most similar endmember with the lowest spectral angle is the second in the list and so on. A sample of the matrix containing the spectral angle distance of each endmember with all other endmembers is shown in Table 5.3.

Table 5.3 Spectral angle distance between the endmember pixels

	1	2	3	4	5	6	7	8	9	10
1	0.0000									
2	0.0725	0.0000								
3	0.0843	0.0766	0.0000							
4	0.0903	0.0796	0.0763	0.0000						
5	0.0985	0.0827	0.0822	0.0828	0.0000					
6	0.0987	0.0808	0.0795	0.0816	0.0695	0.0000				
7	0.0987	0.0854	0.0839	0.0845	0.0828	0.0729	0.0000			
8	0.1075	0.0982	0.0882	0.0931	0.0877	0.0780	0.0841	0.0000		
9	0.1017	0.0943	0.0913	0.0913	0.0821	0.0810	0.0820	0.0825	0.0000	
10	0.0883	0.0849	0.0845	0.0914	0.0855	0.0818	0.0870	0.0898	0.0779	0.0000

The image coordinates of the endmembers are also stored. The endmembers are then reordered manually, according to their spectral similarity and spectrally similar endmembers with similar spatial coordinates are grouped together into endmember classes. Once reordered the endmembers were then grouped into several different land use land cover classes such as Forest, Dry river, settlements, Grass/Shrubs, fallow land, agricultural crop fields etc. A number of endmembers in the obtained set were noisy and were rejected.

The results obtained during different stages of the spatial-spectral integration process for the Hyperion Dehradun dataset are listed below in Table 5.2. The location of the dominant endmembers, which were identified after the reordering of the endmember spectra and grouping based on similar spatial coordinates, is also recorded in Table 5.5.

Table 5.4 Spatial-spectral integration results

	Subset Size			
	128×128	**64×64**	**32×32**	**16×16**
Number of Image subsets	10	40	160	640
HySime Threshold	99.9% (4 eigenvectors/ subset)	99.9% (4 eigenvectors/ subset)	99.9% (4 eigenvectors /subset)	99.9% (4 eigenvectors /subset)
Step 1, Number of Vectors	40	160	640	2560
Step 2, Number of Candidate Endmembers	34	59	128	Too many vectors, Program halted
Step 3, Number of Averaged Endmember pixels	34	59	128	
Step 4, Number of Unique endmembers	Identified manually by grouping spectrally similar and spatially adjacent endmembers			

Table 5.5 Pure endmembers extracted for different LULC classes and their image coordinates

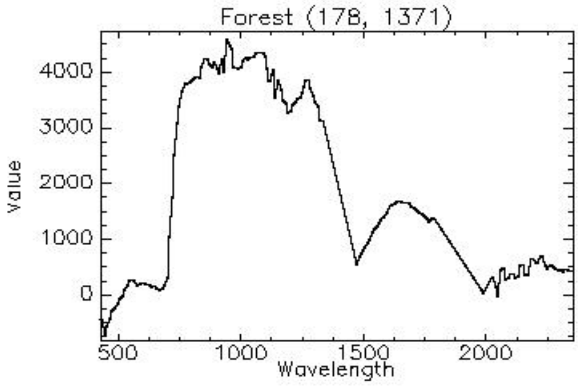
LULC Class	Image Coordinates
Forest	178, 1371
Agricultural crop land	226, 1667
Grounds with grass	124, 1491
Settlement	212, 1558
Dry River Bed	142, 1370
Fallow Land	109, 1357

5.6. Identified Endmembers : Visual Analysis

In this section a visual interpretation of the results obtained for different land use land cover classes is presented to discriminate between various classes. The spectral profiles of the extracted endmembers are shown with their spatial coordinates. The spectral profile, averaged spectrum obtained from step 3, for each endmember is plotted till the 950 nm range for easier visual discrimination. The Hyperion image and the LISS-4 image of the land use land cover class which these endmembers are representing are also shown. The field photographs captured earlier which are representations of these land use land cover classes are also presented.

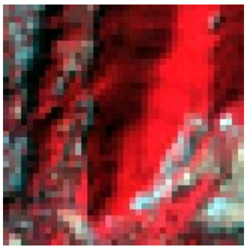
5.6.1. Forest Class

The screen shot shows the image of forest with mainly Sal vegetation, taken during earlier field visits. The spectral profile shows a steep rise at 700 nm which reaches to a value of around 4000 before decreasing again. Then there exists small peaks in the near infrared (NIR) region. Figure 5.11 shows the averaged spectral profile of the forest class and its reference images.

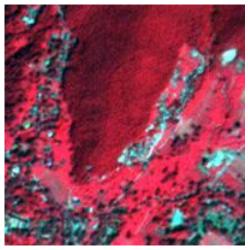


(a)

(b)



(c)

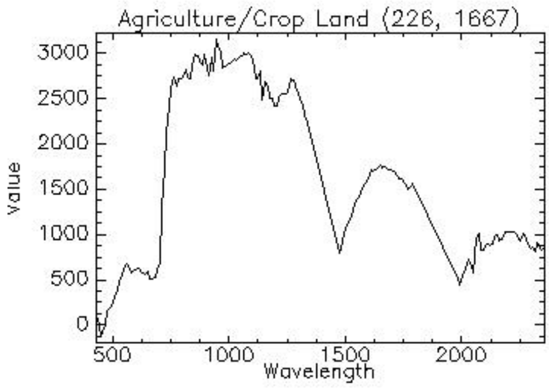


(d)

Figure 5.11 (a) Spectral profile of Forest class, (b) snapshot of Sal forest, (c) FCC of the Hyperion Image and (d) LISS-4 image of the same area

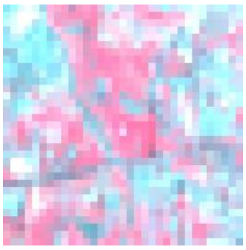
5.6.2. Agriculture/Crop Land

The screen shot shows the image of agricultural land or crop field in the Southern part of Dehradun. The spectral profile shows a steep rise at 700 nm which reaches to a value of around 3000 before decreasing again. Then there exists small peaks in the near infrared (NIR) region. Figure 5.12 shows the averaged spectral profile of the agriculture/crop land class and its reference images.

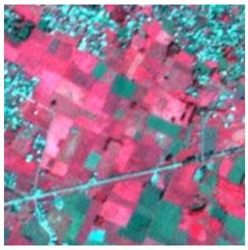


(a)

(b)



(c)



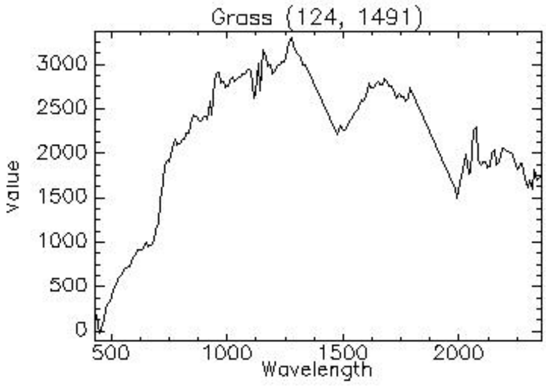
(d)

Figure 5.12 a) Spectral profile of Agriculture class, (b) snapshot of crop field, (c) FCC of the Hyperion Image and (d) LISS-4 image of the same area

5.6.3. Grounds with grass

The snapshot shows the grass in the lawns of the Forest research Institute (FRI) in Dehradun. The spectral profile of the averaged endmember is shown in figure 5.13 (a). The curve rises slowly till a wavelength of 675 nm and then steeply till 750 nm and peaks at a value of 3000 at 950 nm.

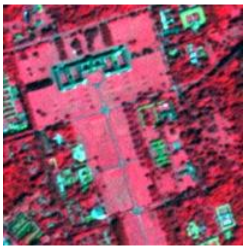
Figure 5.13 shows the averaged spectral profile of a type of grass in Dehradun region and its reference images.



(a)

(b)

(c)



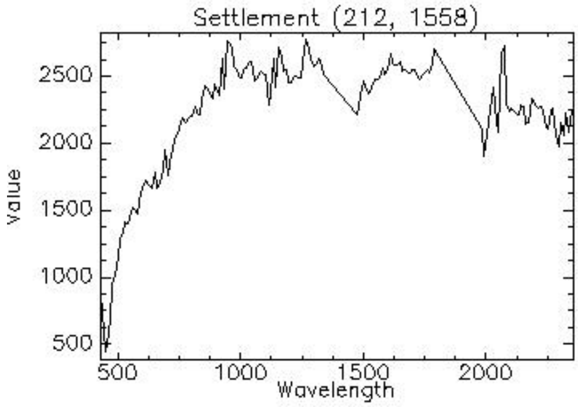
(d)

Figure 5.13 (a) Spectral profile of grass, (b) a snapshot of grass in FRI, (c) Hyperion zoom image of FRI and (d) LISS-4 zoom image of FRI

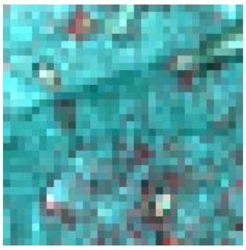
5.6.4. Settlement

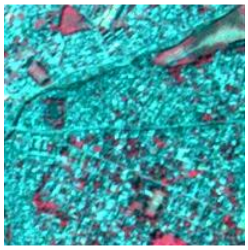
The snapshot shows a typical building feature present in the Dehradun city. The roof top mostly consists of sand, concrete and clay. The spectral curve of the building of Hyperion image rises steadily to a value of 2500 at around 850 nm and then there are a lot of peaks and dips.

Figure 5.14 shows the averaged spectral profile of a building in Dehradun region and its reference images.



(a) (b)





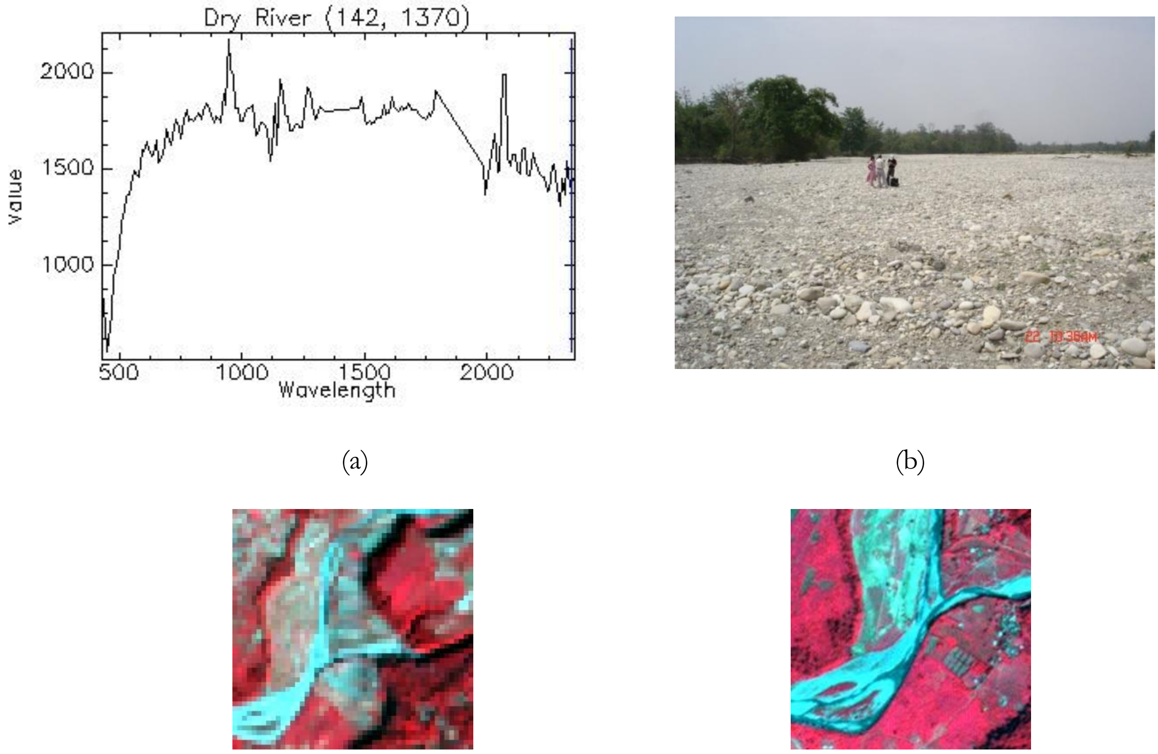
(c) (d)

Figure 5.14 a) Spectral profile of settlement class, (b) a snapshot of typical building in Dehradun city, (c) Hyperion zoom image of settlement class and (d) LISS-4 zoom image of settlement class

5.6.5. River Bed

The snapshot shows the dry river bed of the Tons river in Dehradun and flows from North east to south west direction in the area. The river is seasonal in nature and the river bed is dry most of the year. The river bed is characterised by cyan colour due to high silica content in dry river bed. The spectral curve of the river rises steeply to 1500 at around 575 nm and after that there are a lot of undulations.

Figure 5.15 shows the averaged spectral profile of a dry river bed of Tons river in Dehradun region and its reference images.



(a) (b)

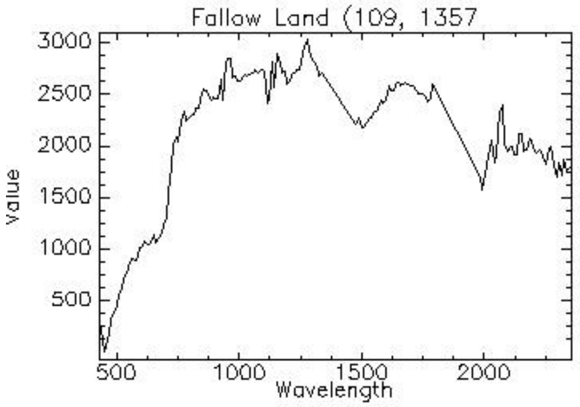
(c) (d)

Figure 5.15 a) Spectral profile of dry river bed, (b) a snapshot of Tons river in Dehradun, (c) Hyperion zoom image of dry river and (d) LISS-4 zoom image of dry river bed.

5.6.6. Fallow land

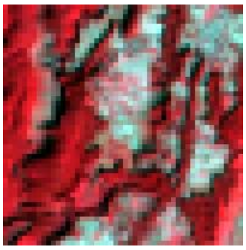
The snapshot shows the land cover class called Fallow land, in the northern part of Dehradun and is characterised by mixed shade of red and cyan colours. The land cover class is called fallow land as it is deficient of any crop. The spectral curve of the fallow land rises steadily till 750 nm without any dips. After that a few dips and peaks can be observed in the near infrared region.

Figure 5.16 shows the averaged spectral profile of fallow land near Tons river in Dehradun region and its reference images.

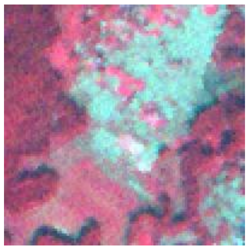


(a)

(b)



(c)

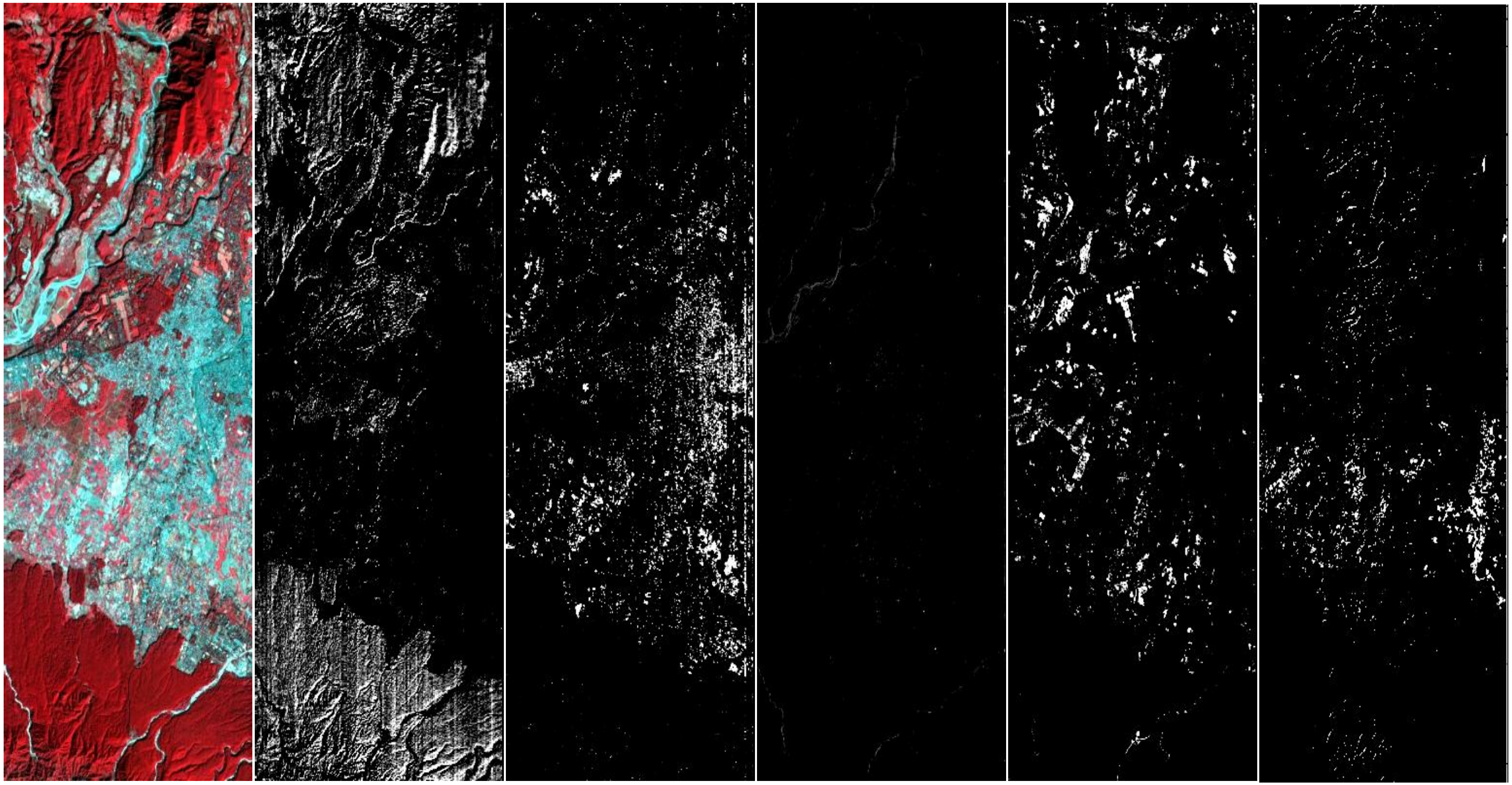


(d)

Figure 5.16 a) Spectral profile of fallow land, (b) a snapshot of fallow land, (c) Hyperion zoom image of fallow land and (d) LISS-4 zoom image of fallow land.

5.7. Spectral Unmixing

The spectra of the identified endmembers are stored as spectral library and are used for performing linear spectral unmixing in ENVI™. Linear spectral unmixing is performed using the endmember pixels whose averaged spectrum is shown in previous section. The outputs of linear spectral unmixing are fractional abundance images that show the abundance of that material in a particular pixel. The abundance fraction maps of the five endmember classes, i.e. forest, dry river, fallow land, crop fields and settlements, derived after spatial-spectral with a subset size of 32×32 and HySime threshold value of 99.9% are shown in Figure 5.13 along with a false colour composite of the study scene. The black background represents values equal to zero values. The forest and the settlements have an abundance fraction greater than 50% and the rest of the three classes have abundance fraction greater than 75%.



(a) (b) (c) (d) (e) (f)

Figure 5.17 Fraction Images generated after linear spectral unmixing, a) FCC of Dehradun Area, b) Forest class (abundance >50%), c) Settlements (abundance>50%), d)Dry river (abundance>75%), e) Fallow Land (abundance>75% and f) Crop land (abundance>75%)

From the fraction images we can observe the five different land use land cover classes against the FCC of the study scene. From the fraction image for forest it can be seen that the forest in the lower part of the image is well delineated, the fractions of forest spectra are lower owing to mixing with other spectra. Figure 5.17(c) is the fraction image for settlement class and shows a general outline of the urban area. In Figure 5.17 (d) the dry river bed of the rivers can be clearly identified. Figure 5.17 (e) is the fraction image of fallow land. Here a lot of other classes such as grounds with grass, shrubs etc. have been mixed with fallow land. The Figure 5.17 (f) highlight the fraction image of agriculture/ crop land class. The different agricultural fields in the southern part of the Dehradun are well defined in this image.

6. CONCLUSIONS

This research work concentrates on two aspects of hyperspectral image processing; first, signal subspace identification or the estimation of the number of spectrally distinct signal sources present in spaceborne hyperspectral dataset using a recently proposed approach called hyperspectral subspace identification by minimum error (HySime) and second, the integration of spatial information with these spectrally distinct signal sources with the aim to improve the extraction of the pure image endmembers. This chapter summarizes the results obtained by following the methodology adopted for this research work.

6.1. Is the HySime signal decomposition technique more efficient than other existing techniques, in the context of spaceborne hyperspectral datasets?

From the results obtained in chapter five, it was found that the HySime algorithm has an edge over the other mentioned techniques such as MNF and SVD for estimating the signal subspace in spaceborne hyperspectral imagery. HySime is eigen-decomposition based and uses the eigenvalues to determine the percentage of variability explained by the first few eigenvectors as shown in section 5.1. The comparison of spectral energy explained by HySime with other techniques gave evidence that HySime performs better in the presence of strong random noise as in case of high dimensional hyperspectral datasets. This also proves the utility of the multiple regression based noise estimation procedure employed by HySime to estimate the noise correlation matrix. The effectiveness of the noise estimation procedure of HySime was also illustrated by transforming the HySime components back to its original spectral space in section 5.4. The restored image was cleaner and more noise free than the original image.

6.2. What will be the intrinsic dimension of the subspace identified by HySime?

The signal subspace dimension or the estimate of the number of spectrally distinct signal sources inferred by HySime, $\hat{k}$, was found to be 26. However after looking at the eigenvalues and the HySime components it was found that only the first few components corresponding to the largest eigenvalues were useful for further analysis. HySime largely overestimates the subspace dimension, a conclusion that was also inferred in [7]. This is because of the presence of largely unknown noise structure of Hyperion, other noise sources or interferers or the presence of unknown spectral signatures in the study scene.

6.3. How to integrate spatial information with spectral subspace identified by HySime for endmember extraction?

The spatial information is incorporated in the endmember extraction process by partitioning the image into subsets thus enhancing the local spectral contrast of the endmembers which are having low spectral contrast with respect to the whole image. HySime is used to extract the local eigenvectors from the subset that enables the selection of both low and high contrast endmembers from a given scene. A total of 128 initial candidate endmembers were identified. By averaging the endmembers which are spectrally similar based on minimum spectral angle and also spatially related based on their similar spatial coordinates, the resultant averaged endmembers are both spectrally and spatially defined. The averaging process reduces the noise and their spectral profiles are smoother than before as illustrated by Figure 5.11

of section 5.5. The result is a set of all 128 endmembers which are both spectrally and spatially defined. However the set of unique pixels from the averaged endmember set was extracted manually by reordering the endmember set based on minimum spectral angle and identifying the endmembers which are spectrally similar and spatially related and grouping them into different land use land cover classes.

6.4. How will the integration of spatial and spectral information improve the classification and mapping accuracies?

The endmember set obtained after reordering and grouping into different spectral classes resulted in the delineation of the endmembers into six broad classes as given in Table 5.5 of chapter 5. The validation for the extracted endmembers was performed visually by comparing them against the original Hyperion spectra and their spatial locations were confirmed by comparisons with field photos taken during earlier field visits against the high resolution data set i.e. IRS-P6 LISS4 image of Dehradun area.

The Hyperion dataset was unmixed using these unique endmembers using the linear spectral unmixing approach of ENVI™ fractional abundance maps were generated. The endmembers represented distinct spatial regions which can be visually identified. Although visually identifiable the various classes were still a mixture of different spectra owing to complex land use land cover classes present in the study scene and the low spatial resolution of the Hyperion sensor. However the method proved to be effective in reducing the noise and the interclass variability of the candidate endmember pixels, thus making them more efficient to be used for unmixing.

The result obtained from the spatial-spectral integration highlight the method is capable of extracting endmembers from the spaceborne hyperspectral datasets which are both spatially and spectrally defined.

6.5. Recommedations

The integration of spatial information with spectrally rich hyperspectral datasets is an active area of research. In this thesis the approach has been tested on spaceborne hyperspectral dataset. A few recommendations for spatial-spectral integration in the context of spaceborne hyperspectral datasets are:

1) The HySime approach can be tested for Non-Linear mixing scenarios where the incident solar radiations are multiply scattered and the mixing between different substances is non-linear.
2) The subsets used in this study were non-overlapping and square. However depending upon scene characteristics, overlapping subsets may be used for averaging such that all the pixels are compared with all other pixels, explaining how the endmembers are related in neighbouring subsets.

LIST OF REFERENCES

[1] T. Lillesand, R. W. Kiefer, and J. W. Chipman, *Remote sensing and image interpretation*, 5th ed. New York: Wiley, 2004.

[2] F. Kruse, J. Boardman, and J. Huntington, "Comparison of airborne hyperspectral data and EO-1 Hyperion for mineral mapping," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 41, no. 6, pp. 1388-1400, 2003.

[3] P. Varshney, *Advanced image processing techniques for remotely sensed hyperspectral data*. Berlin ;;New York: Springer, 2004.

[4] N. Acito, M. Diani, and G. Corsini, "Subspace-Based Striping Noise Reduction in Hyperspectral Images," *IEEE Transactions on Geoscience and Remote Sensing*, 2010.

[5] N. Bhatia, "Hyper spectral image denoising : interscale orthonormal wavelet shrinkage using spatial - spectral domain," M.Sc Thesis, University of Twente Faculty of Geo-Information and Earth Observation ITC, 2010.

[6] L. Jimenez and D. Landgrebe, "Supervised classification in high-dimensional space: geometrical, statistical, and asymptotical properties of multivariate data," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 28, no. 1, pp. 39-54, 1998.

[7] J. Bioucas-Dias and J. Nascimento, "Hyperspectral Subspace Identification," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 46, no. 8, pp. 2435-2445, 2008.

[8] S. DeBacker, P. Kempeneers, W. Debruyn, and P. Scheunders, "A Band Selection Technique for Spectral Classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 2, no. 3, pp. 319-323, 2005.

[9] I. Jolliffe, *Principal Component Analysis*. New York: Springer-Verlag, 2002.

[10] A. Green, M. Berman, P. Switzer, and M. Craig, "A transformation for ordering multispectral data in terms of image quality with implications for noise removal," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 26, no. 1, pp. 65-74, 1988.

[11] L. Scharf, *Statistical signal processing : detection, estimation, and time series analysis*. Reading Mass.: Addison-Wesley Pub. Co., 1991.

[12] N. Keshava and J. Mustard, "Spectral unmixing," *Signal Processing Magazine, IEEE*, vol. 19, no. 1, pp. 44-57, 2002.

[13] C. Chang and Q. Du, "Estimation of Number of Spectrally Distinct Signal Sources in Hyperspectral Imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 3, pp. 608-619, 2004.

[14] M. Farzam and S. Beheshti, "The Noiseless code-length concept in subspace estimation for low SNR hyperspectral signals," in *Communications, Computers and Signal Processing, 2009. PacRim 2009. IEEE Pacific Rim Conference on*, pp. 425-430, 2009.

[15] J. W. Boardman, F. A. Kruse, and R. O. Green, *Mapping Target Signatures Via Partial Unmixing Of Aviris Data*. Campus Box 216 University of Colorado, Emdder, CO S03W-0216: Center for the Study of Earth from Space Cooperative Institute for Research in the Environmental Sciences, 1995.

[16] M. E. Winter, "N-FINDR: an algorithm for fast autonomous spectral end-member determination in hyperspectral data," in *Proceedings of SPIE*, pp. 266-275, 1999.

[17] M. Zortea and A. Plaza, "Spatial Preprocessing for Endmember Extraction," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 47, no. 8, pp. 2679-2693, 2009.

[18] V. Madhok and D. Landgrebe, "Spectral-Spatial Analysis of Remote Sensing Data: An Image Model and A Procedural Design," PURDUE UNIVERSITY, 1999.

[19] D. Rogge, B. Rivard, J. Zhang, A. Sanchez, J. Harris, and J. Feng, "Integration of spatial-spectral information for the improved extraction of endmembers," *Remote Sensing of Environment*, vol. 110, no. 3, pp. 287-303, Oct. 2007.

[20] K. Simon, "Hyperion Level 1G (L1GST) Product Output Files Data Format Control Book (DFCB)," Department of the Interior U.S. Geological Survey, Apr-2006.

[21] J. Pearlman, P. Barry, C. Segal, J. Shepanski, D. Beiso, and S. Carman, "Hyperion, a space-based imaging spectrometer," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 6, pp. 1160-1173, 2003.

[22] L. Guanter, K. Segl, and H. Kaufmann, "Simulation of Optical Remote-Sensing Scenes With Application to the EnMAP Hyperspectral Mission," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 7, pp. 2340-2351, 2009.

[23] C. Galeazzi, A. Sacchetti, A. Cisbani, and G. Babini, "The PRISMA Program," in *IGARSS 2008 - 2008 IEEE International Geoscience and Remote Sensing Symposium*, pp. IV - 105-IV - 108, 2008.
[24] D. Landgrebe, "Hyperspectral image data analysis," *Signal Processing Magazine, IEEE*, vol. 19, no. 1, pp. 17-28, 2002.
[25] R. Huang and M. He, "Band Selection Based on Feature Weighting for Classification of Hyperspectral Data," *IEEE Geoscience and Remote Sensing Letters*, vol. 2, no. 2, pp. 156-159, 2005.
[26] P. Switzer and A. A. Green, *Min / Max autocorrelation factors for multivariate spatial imagery*. Department of Statistics, Stanford University: , 1984.
[27] B. Xu and P. Gong, "Noise estimation in a noise-adjusted principal component transformation and hyperspectral image restoration," *Canadian journal of Remote Sensing*, vol. 34, no. 3, pp. 271-286, Aug. 2008.
[28] J. Lee, A. Woodyatt, and M. Berman, "Enhancement of high spectral resolution remote-sensing data by a noise-adjusted principal components transform," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 28, no. 3, pp. 295-304, 1990.
[29] G. Martín and A. Plaza, "Region-Based Spatial Preprocessing for Endmember Extraction and Spectral Unmixing," *IEEE Geoscience and Remote Sensing Letters*, 2011.
[30] M. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Sparse Unmixing of Hyperspectral Data," *IEEE Transactions on Geoscience and Remote Sensing*, 2011.
[31] R. Schowengerdt, *Remote sensing: models, and methods for image processing*, 2nd ed. San Diego: Academic Press, 1997.
[32] J. Boardman, "Geometric mixture analysis of imaging spectrometry data," in *Proceedings of IGARSS '94 - 1994 IEEE International Geoscience and Remote Sensing Symposium*, pp. 2369-2371.
[33] A. Zare, "Hyperspectral Endmember Detection And Band Selection Using Bayesian Methods," The Graduate School Of The University Of Florida, 2008.
[34] C. Chang and A. Plaza, "A Fast Iterative Algorithm for Implementation of Pixel Purity Index," *IEEE Geoscience and Remote Sensing Letters*, vol. 3, no. 1, pp. 63-67, 2006.
[35] J. Nascimento and J. Bioucas-Dias, "Vertex component analysis: a fast algorithm to unmix hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 4, pp. 898-910, 2005.
[36] J. M. P. Nascimento, "Unsupervised Hyperspectral Unmixing," PhD Thesis, UNIVERSIDADE T¶ECNICA DE LISBOA, INSTITUTO SUPERIOR T¶ECNICO, 2006.
[37] A. Bateson and B. Curtiss, "A method for manual endmember selection and spectral unmixing," *Remote Sensing of Environment*, vol. 55, no. 3, pp. 229-243, Mar. 1996.
[38] S. Tompkins, J. F. Mustard, C. M. Pieters, and D. W. Forsyth, "Optimization of endmembers for spectral mixture analysis," *Remote Sensing of Environment*, vol. 59, no. 3, pp. 472-489, Mar. 1997.
[39] A. Ifarraguerri and C. Chang, "Multispectral and hyperspectral image analysis with convex cones," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37, no. 2, pp. 756-770, 1999.
[40] B. Rivard, D. Rogge, J. Feng, and J. Zhang, "Spatial constraints on endmember extraction and optimization of per-pixel endmember sets for spectral unmixing," in *Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, 2009. WHISPERS '09. First Workshop on*, pp. 1-4, 2009.
[41] F. D. Van Der Meer, *Imaging spectrometry : basic principles and prospective applications*. Dordrecht ;;Boston: Kluwer Academic Publishers, 2001.
[42] B. Datt, T. McVicar, T. Van Niel, D. Jupp, and J. Pearlman, "Preprocessing eo-1 hyperion hyperspectral data to support the application of agricultural indexes," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 6, pp. 1246-1259, 2003.
[43] T. Han, D. Goodenough, A. Dyk, and J. Love, "Detection and correction of abnormal pixels in Hyperion images," in *IEEE International Geoscience and Remote Sensing Symposium*, pp. 1327-1330.
[44] D. Goodenough et al., "Processing hyperion and ali for forest classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 6, pp. 1321-1331, 2003.
[45] Jinguo Yuan and Zheng Niu, "Evaluation of atmospheric correction using FLAASH," in *2008 International Workshop on Earth Observation and Remote Sensing Applications*, pp. 1-6, 2008.
[46] J. Jensen, *Introductory digital image processing : a remote sensing perspective*, 2nd ed. Upper Saddle River N.J.: Prentice Hall, 1996.

APPENDIX A

A.1. Abnormal pixel detection: IDL Code for Along Track Destriping of Hyperion Images

```
threshold1 = 5                                         ;Number of continuous bad pixels in a column
threshold2 = fix(0.50 * input_rows)                    ;Minimum number of bad pixels required to label a column
print, 'Threshold1 is ', threshold1
print, 'Threshold2 is ', threshold2

for b = 0, input_bands-1 do begin                      ;Identify bad columns based on threshold values
  for c = 1, input_cols-2 do begin
    counter = 0L
    bad_candidate = intarr(input_rows)
    for r = 0, input_rows-1 do begin
      if input_file_data[b,c,r] lt input_file_data[b,c-1,r] $
          AND input_file_data[b,c,r] lt input_file_data[b,c+1,r] then begin
        counter += 1
        bad_candidate[r] = 1
      endif
    endfor
    if counter ge Threshold2 then begin
      ;print, b, c, counter
      for i = 0, input_rows-5 do begin
        if bad_candidate[i]+bad_candidate[i+1]+bad_candidate[i+2] $
          +bad_candidate[i+3]+bad_candidate[i+4] eq 5 then begin
          ;print, 'yes'
          print, b+1, c+1;, counter                   ; Print Bands and Bad Columns
          break
        endif
      endfor
```

A.2. Abnormal pixel correction: IDL Code IDL Code for Along Track Destriping of Hyperion Images

```
print, 'averaging Bad Columns pixels', systime()
 for i = 0, iterations-1 do begin
   b = bad_cols_data.field1[0, i]-1                                  ; Bad Column Band
   c = bad_cols_data.field1[1, i]-1                                  ; Bad Column Location
   if NOT(c eq 0 OR c eq input_cols-1) then begin
     for r = 0, input_rows-1 do begin
       input_file_data[b, c, r] = round((input_file_data[b, c-1, r]+ $   ; Input_file_data - Input Hyperion data
                                  input_file_data[b, c+1, r])/2)     ; Replacing Bad band by average of
                                                                     ; adjacent bands
     endfor
   endif else begin
     if c eq 0 then begin
       for r = 0, input_rows-1 do begin
         input_file_data[b, c, r] = input_file_data[b, c+1, r]
       endfor
     endif else begin
       for r = 0, input_rows-1 do begin
         input_file_data[b, c, r] = input_file_data[b, c-1, r]
       endfor
     endelse
   endelse
 endfor
   print, 'processing pixels complete ', systime()
```

APPENDIX B

B1: Spatial-Spectral Integration, Step 2 - IDL code for the Projection of image data onto Eigenvectors

```
input_file = dialog_pickfile(filter='*.tif')                          ; Input Image Data
is_file_ok = query_tiff(input_file)
if is_file_ok then print, 'Tif file is ok' else return
print, 'Reading tif file    ', systime()
img_data = read_tiff(input_file, interleave=2)
print, 'Tif file read complete    ', systime()

vec_file = 'F:\Research Work\DATA\~\eigen_vec_compiled_640.csv'       ; Input Eigenvector file
openr, 100, vec_file
vec_data = fltarr(640, 158)
readf, 100, vec_data
close, 100


ppi = subspace_proj_step2(img_data, vec_data, 2)                      ; Calls function to project Image
                                                                      ; onto Eigenvectors

temp = where(ppi gt 0, count)
print, count                                                          ; Counts Number of Hits
ppi[temp] = 255

output_file = dialog_pickfile()                                       ; Write Output as Tiff file
   if output_file eq '' then return
   print, 'writing into file', systime()
   write_tiff, output_file,  ppi,  /double,  /short, /signed
   print, 'Done', systime()
```

B2: Spatial-Spectral Integration, Step 2 - Function to project the Image data onto Eigenvectors

```
function subspace_proj_step2, ip_img, vec_data, thresh_value  ;ip_img - image data
                                                              ;vec_data - compiled eigenvectors
                                                              ; thresh_value - number of pixels retained
                                                              ; at each end of the vector
sz=size(ip_img)
Bands=sz[3]
ip_img_new=reform(TEMPORARY(ip_img), long(sz[1])*sz[2],Bands)

sz_vec=size(vec_data)
dummy=bytarr(long(sz[1])*sz[2],sz_vec[1])
hits =intarr(long(sz[1])*sz[2])

unit_vecs=transpose(vec_data)                                 ; Transpose Eigenvectors

proj_all=unit_vecs##ip_img_new                                ; project image data onto eigenvectors

for i=0L, sz_vec[1]-1 do begin
  extrem_idx=array_sort1(proj_all[*,i], thresh_value)         ; Calls function to sort the projection matrix and
                                                              ; to find the extreme pixels
 dummy[extrem_idx,i]=1
endfor

for j=0L, long(sz[1])*sz[2]-1 do hits[j]=total(dummy[j,*])    ; Counts the number of Hits for each extreme pixel
help, hits
ppi=reform(hits,sz[1], sz[2])

return, ppi
```

B3: Spatial-Spectral Integration, Step 2 - Function to sort the array and to find the extreme pixels

```
function array_sort1, arr, thres
  if thres eq 1 then begin
    temp = max(arr, sub_max)                          ;Finds max value in an array and its subscripts
    temp = min(arr, sub_min)                          ;Finds min value in an array and its subscripts

    rt_value = [sub_max, sub_min]
  ENDIF
  if thres eq 2 then begin
    temp = max(arr, sub_max)                          ;Finds max value in an array and its subscripts
    I = indgen(n_elements(arr))
    if sub_max eq 0 then I[sub_max] = I[sub_max+1] $
    else I[sub_max] = I[sub_max-1]
    temp = max(arr[I], sub_max1)                      ;Finds 2nd max value in an array and its subscripts

    temp = min(arr, sub_min)                          ;Finds min value in an array and its subscripts
    I = indgen(n_elements(arr))
    if sub_min eq 0 then I[sub_min] = I[sub_min+1] $
    else I[sub_min] = I[sub_min-1]
    temp = min(arr[I], sub_min1)                      ;Finds 2nd min value in an array and its subscripts
    rt_value = [sub_max, sub_max1, sub_min, sub_min1]
  endif
  return, rt_value
end
```