STUDY OF SOME IMFORMATION EXTRACTION TECHNIQUES FOR HYPERSPECTRAL IMAGING

Ph.D. THESIS

by

N. PRABHU

DEPARTMENT OF CIVIL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE – 247 667 (INDIA)

OCTOBER, 2014

STUDY OF SOME IMFORMATION EXTRACTION TECHNIQUES FOR HYPERSPECTRAL IMAGING

A THESIS

Submitted in partial fulfilment of the
requirements for the award of the degree

of

DOCTOR OF PHILOSOPHY

in

CIVIL ENGINEERING

by

N. PRABHU

DEPARTMENT OF CIVIL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE – 247 667 (INDIA)

OCTOBER, 2014

INDIAN INSTITUTE OF TECHNOLOGY ROORKEE ROORKEE

CANDIDATE’S DECLARATION

I hereby certify that the work which is being presented in this thesis entitled “STUDY OF SOME IMFORMATION EXTRACTION TECHNIQUES FOR HYPERSPECTRAL IMAGING” in partial fulfilment of the requirements for the award of the Degree of Doctor of Philosophy and submitted in the Department of Civil Engineering of the Indian Institute of Technology Roorkee is an authentic record of my own work carried out during a period from July, 2010 to October, 2014 under the supervision of Prof. Manoj K. Arora, Director, PEC University of Technology, Chandigarh and Dr. R. Balasubramanian, Associate Professor, Department of Computer Science and Engineering, Indian Institute of Technology Roorkee, Roorkee.

The matter presented in the thesis has not been submitted by me for the award of any other degree of this or any other Institute.

(N. PRABHU)

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

(R. Balasubramanian)
Supervisor

(Manoj K. Arora)
Supervisor

Date: October , 2014

ABSTRACT

Remote sensing is an art and science of acquiring information about earth surface without actual contact with the earth surface. It gives data in the form of images which comprises of finite number of elements known as pixels. Such images which are obtained in from remote sensing needs to be processed and the easy and effective way is to do it digitally. Hence the field of digital image processing refers to processing of digital images by means of a computer. Digital image processing plays a key role in the field of image processing, image analysis, image enhancement and coding with a wide range of applications such as, classification, target/anomaly detection, mineral identification, super resolution mapping, etc. In remote sensing, the digital image processing methods are used for deriving the information from airborne and satellite borne images.

Remote sensing community has utilized the digital images obtained from airborne and satellite borne sensors and has made enormous progress in recent years. Based on the sensor specifications the digital image produced by each sensor varies in spatial, spectral and radiometric resolutions. Other components like dimension of the image, temporal resolution of the image, swath width of the image, etc., plays secondary role in some types of applications. Due to variation in spectral resolution we have panchromatic (single band), multi-spectral (few bands less than 10) and hyperspectral (100s of bands) data are available. In this research, the study has been done for hyperspectral data. The land use land cover classes on the earth's surface have different physical characteristics. Particularly, for a hyperspectral data, due to coarse spatial resolution pixels may contain two or more classes. Hence classification of hyperspectral data is a way to extract useful information from it.

But due to the large spectral dimension of hyperspectral data, it suffers while doing processing. Redundant information from hyperspectral data are carried every time. Also one of the important properties of hyperspectral data is that the neighbouring bands of hyperspectral data are highly correlated. Hence due to the collective reasons the hyperspectral data needs to be reduced. To reduce the hyperspectral data, in this research, the feature extraction techniques have been employed. The two feature extraction techniques employed here are wavelet based feature extraction and PCA based feature extraction. In wavelet based feature extraction three wavelet transforms have been used, namely, Haar wavelets, Daubechies wavelets and Coiflets wavelets. These wavelets along with its sub classes have also been utilized. For each of the sub classes, decomposition upto 4 levels have been performed. In case of PCA based feature extraction two prominent techniques have been used, segmented PCA and spectrally segmented PCA. The evaluation of the feature extraction techniques has been done by calculating the classification accuracy. Also a study has been made to observe the duration of feature extraction, duration of classification, number of extracted features, relation between classification accuracy and number of reduced features, etc.

Once the features have been extracted from hyperspectral data, now it is ready for performing classification. Even though the number of bands reduced to significant level the algorithms used for multi-spectral classification are incapable of producing useful information. Hence for classification of hyperspectral data few algorithms exist in literature. For hyperspectral data also, the classification may be per pixel and sub pixel. By incorporating training pixels if the classification is done then it is called as supervised classification while with no help from training data if classification is done then it is unsupervised classification. Also if the statistical parameter extracted from the data are used then it is parametric classification and if not it is called as non-parametric classification. In this research all types of these classifications are touched and they are support vector machines (SVM) which is supervised, per pixel and non-parametric classifier while linear mixture model (LMM) is also supervised and non-parametric but sub pixel classification. Finally, the independent component analysis (ICAMM) which is parametric classification unsupervised but produces sub pixel outputs. The information extractions by each of these classification techniques are useful in one or the other way.

To have a study about the performance of these techniques, three hyperspectral datasets have been taken. Two AVIRIS datasets with different spatial resolutions 4m and 20 m while one HYPERION dataset with spatial resolution 30m. The 20m spatial resolution dataset is named as dataset II covers by most of the region by vegetation and the classes are almost crisp. The 4m resolution data from HYPERON is named as dataset III which covers Roorkee and its surroundings covers by urban area and vegetation. Also some small classes like barren land, sand, etc., are also present in the dataset. Moreover the classes are not crisp enough to retrieve them back easily because of the nature of the land cover and the coarse spatial resolution. The next dataset is again from AVIRIS sensor but its airborne and hence a fine spatial resolution of 4m data is available. This data is over San Diego Naval Station and has many classes whose spectral signatures are almost

similar. The study of classification algorithms has been made on these three hyperspectral datasets.

When per pixel classification produces thematic map in which each pixel has been allotted to one and only one class while in sub pixel classification the fraction abundance of each class present in each pixel has been estimated. But that is not suffice until the spatial location has been estimated at sub pixel level. A solution to this problem may be achieved by super resolution mapping. In super resolution mapping, every pixel is divided into a specified zoom factor and each divided portion (sub pixel location) is filled by unique class thus making the original data into much finer spatial resolution. In this research, a novel super resolution mapping algorithm has been proposed, named as Pixel Filling Algorithm, to split each of the pixels and by gathering fractional abundance information from neighbouring pixels each sub pixel location may be filled by class values. This algorithm has been compared with existing super resolution mapping algorithm named as Pixel Swapping Algorithm.

Due to the novelty of the pixel filling algorithm, a synthetic dataset has been generated of dimension 45x60. This dataset has been reduced by two low-pass filters, 3x3 and 5x5, to convert the data to dimensions 15x20 and 9x12 respectively. Now if super resolution mapping algorithm has been applied on the reduced dataset by appropriate zoom factors, 3 and 5, respectively then it is expected to get back the original data of dimension 45x60.

At every stage of information extraction, the accuracy assessment has been performed. The feature extracted hyperspectral data are classified by SVM and then the conventional error matrix based accuracy assessment has been used. For sub pixel classification techniques the fuzzy error matrix based accuracy assessment has been performed. Finally, for super resolution mapping algorithms, the conventional error matrix based accuracy assessment has been performed but by taking three types of testing samples.

Among the feature extraction techniques studied here, Daubechies wavelets perform better in extracting useful information from hyperspectral data. The second level decomposition is better in both accuracy wise and feature reduction wise. Since for all the datasets only the first two decomposition levels give better overall classification accuracies the sub pixel classifications have been performed only to the first two decomposition levels and that too only for Daubechies and its sub classes. In LMM has two solutions, constrained and unconstrained and there is no significant classification accuracies between them. The accuracy comes to be around 67%, 62% and 61% respectively for dataset II, dataset III and dataset IV. The ICAMM too extracts sub pixel information but due to unsupervised nature it retrieves some classes which are small enough to collect pure pixels from the three hyperspectral datasets. From dataset II, ICAMM retrieves railway track and non-metallic road and from dataset III, it retrieves noise free water body while from dataset IV it retrieves cylindrical drum, aircrafts, etc. In this way, each algorithm proves that it is superior over the other.

The results for super resolution mapping have been analysed for the two algorithms, the proposed pixel filling algorithm and pixel swapping algorithm. The pixel filling algorithm performs well in super resolving mixed pixels which are having complicated boundaries and simple boundaries while the pixel swapping works well for only classes having linear boundaries. The accuracy for pixel filling and pixel swapping algorithms has been given in ordered pairs (for easy comparison) (92%, 98%), (96%, 90%), (97%, 91%) for datasets II, III, IV respectively for zoom factor 3 while for zoom factor 5 it is (91%, 94%), (90%, 70%), (90%, 70%) respectively. For dataset I, the overall accuracy of super resolved image by pixel filling algorithm gives 90.7% and 72.7% for zoom factors 3 and 5 respectively. Also the time required to perform super resolution mapping via pixel resolution mapping takes less than 10 seconds while by pixel swapping algorithm it takes more than a minute. Both in accuracy wise and duration for super resolution wise pixel filling algorithm is better than pixel swapping algorithm.

Acknowledgements

First and above all, I praise **Lord**, *the Almighty*, to whom I owe my very existence. This work will never be accomplished without **His** blessings. I thank the ALMIGHTY LORD for giving me strength, health, perseverance and determination to carry out this research. This thesis appears in its current form due to the assistance and guidance of several people. Therefore, I would like to offer my sincere thanks to all of them.

Foremost, I would like to express my sincere gratitude to my research supervisors **Dr. Manoj K. Arora**, Director, PEC University of Technology, Chandigarh, (Professor, on lien from Department of Civil Engineering, IIT Roorkee, Roorkee, India) and **Dr. R. Balasubramanian,** Associate Professor, Department of Computer Science and Engineering, IIT Roorkee, for the continuous support of my Ph.D. study and research, for their patience, motivation, enthusiasm and immense knowledge.

I am deeply grateful to **Prof. Manoj K. Arora** for his outstanding support, wise guidance and invaluable suggestions both professional and personal. He always encouraged me to pursue my ideas, offered innumerable creative insights and guided me with unmatched wisdom. He helped me to understand this research work through dividing all objectives into small parts and completing them step by step. This research would not have been possible without his active support.

I owe my most sincere gratitude to the members of the student's research committee, Dr. R. P. Gupta, Professor, Department of Earth Sciences and Dr. K. S. Hariprasad, Professor, Department of Civil Engineering, IIT Roorkee for their detailed review, constructive criticism and excellent advice during the preparation of this thesis. ***I am grateful to faculty and staff members, Geomatics Engineering Group, for providing laboratory facilities during the course of this work.***

I owe my sincere thanks to Dr. Sangeeta Khare and Ms. Kiran Chauhan both from DEAL – DRDO, Dehradun, for their continuous constructive criticism in project work which is part of my thesis work, makes a better way of understanding the concept.

The present research was carried out under Quality Improvement Programme, AICTE, New Delhi, India. I am very grateful to AICTE for providing generous funds for the research work. Also I thank my parent institution, Sona College of Technology, Salem, India for giving me permission to do Ph.D., at IIT Roorkee.

During this work, I have collaborated with many colleagues for whom I have great regard, and I wish to extend my warmest thanks to Dr. Hemendra Singh Gusain, Scientist D, SASE DRDO, Dr. Mohit Srivastava, Associate Professor, Chandigarh University, Chandigarh, Mr. Manoj Kuri, Assistant Professor, Government Engineering College, Bikaner, Dr. Kamal Kumar, Assistant Professor, PEC University of Technology, Chandigarh and Dr. Atanu Bhatacharaya, Research Fellow at TU Dresden.

I also thank M. Tech. Geomatics Engineering students Ms. Shweta Bansal, Mr. Saurabh Vijay, Mr. Debojit Biswas, Ms. Kavita Mitkari and Ms. Roshni Sanyal and Mr. Kapil Gupta, MCA student for their help and support.

I would like to thank my friends and research scholars, Ms. Deepti Yadav and Ms. Varinder Saini and all those who have helped me directly and indirectly with my work.

I acknowledge the support provided by my friends Mr. A.B. Danie Roy, Mr. Franklin F.R. Fredrick, Mr. A. Sathish Kumar, Mr. G.B. Ramesh Kumar, Mr. R. Siva Chidambaram, Ms. Shermi Roy, Dr. N. Sivakumar, Mr. M. Prince, Dr. Germin Nisha, Mr. K. Senthil, Mr. S. Udaya Kumar, Dr. A. Manju, Dr. P. Purushotaman, Mr. Arun Mondal, Ms. Sananda Kundu, Dr. Dinesh Singh and Mr. M. Suresh for all their help and hospitability during my stay at IIT Roorkee, which never feels me away from home.

I acknowledge the support provided by my friends Dr. P.K. Sundaramurthy, Dr. R. Rajkumar, Mr. B. Samuel Thavamani, Mr. V. Rajesh, Mr. S. Marikumar, Mr. A. Chellaram Malaravan, Mr. A. Saravanan, Ms. S.R. Latha and Mr. B. Venkatesh for their continuous encouragement.

My special thanks to Dr. S. Emmanual Jebarajan, my inspiration, because of whom my thinking gets a shape and Mr. T. Thomas and Dr. David Amirtharajan for their sincere prayers, which helps me in consistent progress.

My special sincere, heartfelt gratitude and indebtedness is due to my brother **Mr. N. Ravi,** my sister **Ms. N. Mercy,** my sister-in-law **Ms. P. Gnana Sheeba,** my brother-in-law **V. Karthick,** my dearest **master Roger, master Steven** and **master Kevin** for their sincere prayers, constant encouragement. I also warmly thank and appreciate my relatives, friends and well wishers for their moral support in all aspects of my life.

I express my earnest regards to my father-in-law **Mr. M. Joseph Stephen Raj**, my mother-in-law **Ms. J. Josephine** and brother-in-laws **Mr. J. George Raja** and **Mr. J. Prabhu,** for encouraging me to achieve my goals.

I would like to pay high regards to my family members and relatives for their blessings, patience and moral support. I want to express my gratitude and deepest appreciation to my father **Mr. A. Natarajan,** my mother **Ms. N. Jeyaseeli**, my beloved wife **Ms. J. Tansi Sagaya Hema** and my sweet little daughter **baby Sherine Giftriya** for their great patience and understandings. Their encouragement and contribution lifted me to reach this place. I dedicate this thesis to my adoring sweet little daughter **Sherine Giftriya.**

Roorkee
October 27, 2014 **(N. Prabhu)**

Contents

Page No.

List of Figures

List of Tables

List of Abbreviations

AIS	Airborne Imaging Spectrometer
ANN	Artificial Neural Network
AVIRIS	Airborne Visible/Infra-Red Imaging Spectrometer
BAO-SAM	Band Add-On Spectral Angle Mapper
CASI	Compact Airborne Spectrographic Imager
DAFE	Discriminant Analysis Feature Extraction
DAIS	Digital Airborne Imaging Spectrometer
DBFE	Decision Boundary Feature Extraction
EMD	Euclidean Minimum Distance
EO	Earth observation
FCC	False Colour Composite
FE	Feature Extraction
FERM	Fuzzy Error matrix
GA	Genetic Algorithm
GMM	Gaussian Mixture Model
HNN	Hopfield Neural Network
HYDICE	Hyperspectral Digital Imagery Collection Experiment
HyMap	Hyperspectral Mapper
HYPERION	High Resolution Hyper Imager
ICA	Independent Component Analysis
ICAMM	Independent Component Analysis Mixture Model
JM	Jeffries–Matusita
JPL	Jet Propulsion Laboratory
KLT	Karhunen-Loeve Transformation
KPCA	Kernal Principal Component Analysis
LAI	Leaf Area Index
LMM	Linear Mixture Model
LWIR	Long-Wave Infra-Red
MLC	Maximum Likelihood Classification
MNF	Minimum Noise Fraction
MP	Mixed Pixel
MWIR	Mid-Wave Infra-Red
NASA	National Aeronautics and Space Administration
NIR	Near Infra-Red
OA	Overall Accuracy

OSP	Orthogonal Subspace Projection
PC	Principal Component
PCA	Principal Component Analysis
PF	Pixel Filling
PP	Projection Pursuit
PPI	Pixel Purity Index
PSFE	Prototype Space Feature Extraction
PTS	Pixel to be Super Resolve
RBF	Radial Basis Function
ROC	Receiver Operating Characteristics
SAM	Spectral Angle Mapper
SBFS	Sequential Backward Floating Selection
SBS	Sequential Backward Selection
SCM	Spectral Correlation Mapper
SFFS	Sequential Forward Floating Selection
SFS	Sequential Forward Selection
SGA	Spectral Gradient Angle
SIM	Spectral Information Measure
SPCA	Segmented Principal Component Analysis
SRM	Super Resolution Map
SSPCA	Spectrally Segmented Principal Component Analysis
SVMs	Support Vector Machines
SWIR	Short-Wave Infra-Red
VIS	Visible
VNIR	Visible-Near Infra-Red
WEF	Wavelet Energy Feature
WFE	Wavelet Feature Extraction

Chapter 1

Introduction

1.1 Motivation

Changes on the earth surface are bound to happen. In the last few decades, global warming has been a big threat to the living species on the earth. Drought at one place and flood at another place are common phenomena. Two-third of the earth surface is covered by water whereas the remaining one-third is covered by land masses, which include a number of land use land cover types; built-up areas, vegetation, forest, inland water bodies, rivers, roads, deserts, barren lands, soil, sand, etc. Conventional way of gathering information about the land use land cover via field surveys is time consuming and expensive. Moreover, the information at spatial level may not be extracted expediently from field surveys.

During last 4 decades or so, remote sensing has been found to be one of the cheapest and efficient ways in mapping and monitoring the land use land cover on the earth surface. The remote sensing sensors can collect data in spectral bands of visible to microwave region of the electromagnetic spectrum. Sensors which are capable of acquiring information in a single but wide spectral range produce panchromatic data while multispectral sensors are capable of collecting information in less number of bands (also called as features). No doubt, multispectral images provide more information than panchromatic but it is available in discrete bands. Due to significant improvement in sensor technology, the spectral bands can further be sub-divided into few more bands. Sensors which are capable of collecting information in hundreds of narrow width contiguous bands are now in existence. These sensors are called as imaging spectrometers and collect data as hyperspectral images.

1.2 Hyperspectral Imaging

A hyperspectral image is often represented as a hyperspectral image cube (Figure 1.1). In this cube, x and y axes represent the spatial position of the image while the z axis represents the spectral dimension. The number of bands acquired is the spectral dimension of the hyperspectral data. Hyperspectral sensors deal with imaging narrow spectral bands over a contiguous spectral range and produce the spectra of all pixels in the scene. So a sensor with only 20 bands can also be hyperspectral when it covers the range from 500 to 700 nm with 10-20 nm wide bands (while a sensor with 20 discrete bands covering the VIS (visible), NIR (near infra-red), SWIR (short-wave infra-red), MWIR (mid-wave infra-red), and LWIR (long-wave infra-red) is considered as multispectral) (Gonz´alez *et al.*, 2010). Some prevalent hyperspectral sensors are DAIS 7915 (Digital Airborne Imaging Spectrometer with 79 bands), HyMap (Hyperspectral Mapper with 125 bands), HYDICE (HYperspectral Digital Imagery Collection Experiment with 210 bands), HYPERION (High Resolution Hyperspectral Imager with 220 bands), CASI 2 (Compact Airborne Spectrographic Imager with 288 bands), AVIRIS (Airborne Visible/Infrared Imaging Spectrometer with 224 bands).

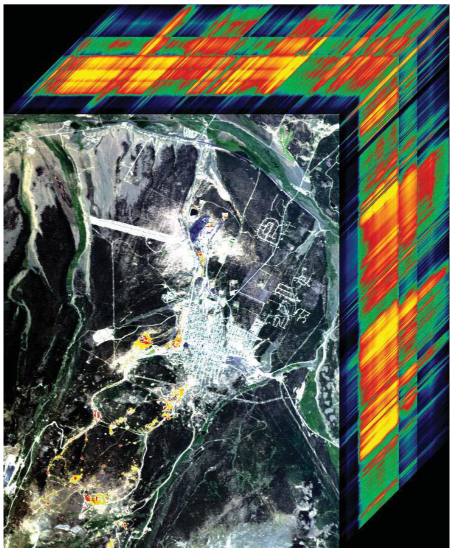


Figure 1.1: The hyperspectral cube. This cube is an AVIRIS hyperspectral image of the Leadville mining district in Colorado. The front of the cube is a true color composite, with areas containing secondary minerals from acid mine drainage highlighted in red, orange and yellow. This cube has been processed using ENVI (**Source:** Shippert, 2004)

Hyperspectral imaging is a class of techniques commonly referred to as spectral imaging or spectral analysis. The perception of hyperspectral imaging originated at NASA’s Jet Propulsion Laboratory (JPL) in California, when they developed instruments such as the Airborne Imaging Spectrometer (AIS), then termed AVIRIS. As a result, each pixel of an image becomes pixel vector collected by a hyperspectral sensor and can be seen as a spectral signature or spectral cure or fingerprint of the underlying materials within the pixel (Figure 1.2) (Gonz´alez *et al.*, 2010).

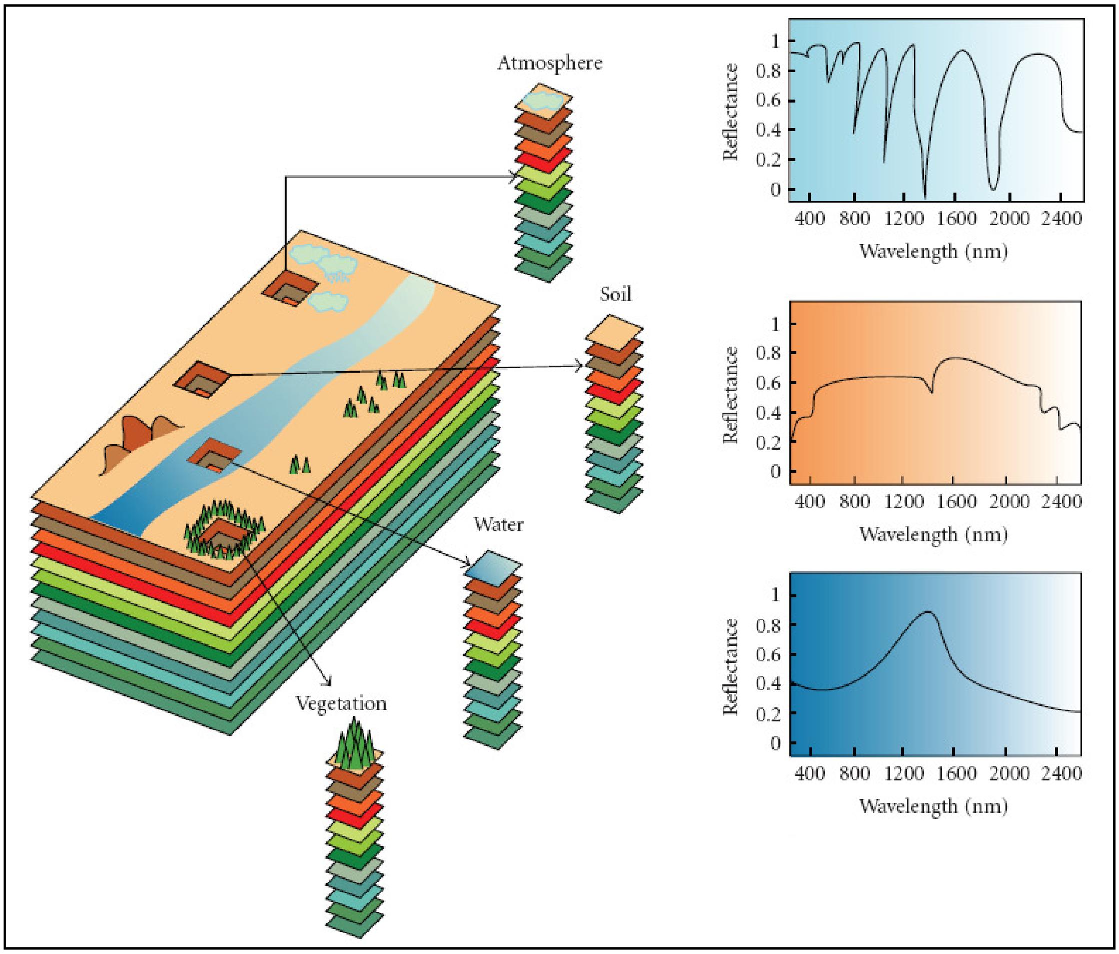


Figure 1.2: The concept of hyperspectral imaging (Source: Gonz´alez *et al.*, 2010)

Several imaging tools have been developed for hyperspectral data processing in recent years, covering tasks such as data compression, dimensionality reduction, land cover classification, multi-source image fusion techniques, spectral mixture analysis (both linear and non-linear), etc. The primary assumption governing clustering and classification techniques is that each pixel vector comprises the response of a single underlying material. However, if the spatial resolution of the sensor is not high enough to separate different materials, these can jointly occupy a single pixel. The resulting spectral measurement will be a mixed pixel, that is, a composite of the few individual pure spectra. For instance, in Figure 1.2, it is likely that the pixel labeled as vegetation is actually a mixture of vegetation and soil or of different types of vegetation canopies. To deal with this problem, linear spectral mixture analysis (Foody and Arora, 1996; Lu *et al.*, 2004; Kasetkasam *et al.*, 2011) has been used. In fact, spectral mixture analysis has been a fascinating exploitation goal since the earliest days of hyperspectral imaging. No matter the spatial resolution, in natural environments the spectral signature for a nominal pixel is invariably a mixture of the signatures of the various materials found within the spatial extent of the ground instantaneous field view of the sensor.

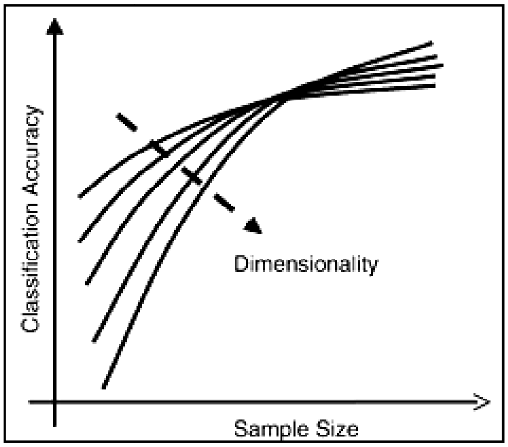


Figure 1.3: Curse of dimensionality **(Source:** Hsu**, 2007)**

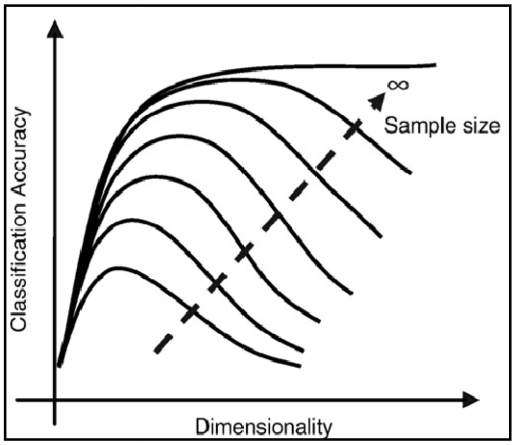


Figure 1.4: Hughes phenomenon **(Source:** Hsu**, 2007)**

Since hyperspectral images contain rich and fine spectral information, an improvement of land use/cover classification accuracy is highly expected from the utilization of such images. However, the traditional statistics-based classification methods which have been successfully applied to multispectral data in the past are not as effective to hyperspectral data. One major issue is that the number of spectral bands is too large relative to the number of training samples (Figure 1.3).

Due to the small number of training samples and the high number of features available in remote sensing applications, reliable estimation of statistical class parameters is another challenging goal (Foody *et al.*, 2004). As a result, with a limited training set, classification accuracy tends to decrease as the number of features increases. This is known as the Hughes phenomenon (Hughes, 1968) (Figure 1.4). High dimensional spaces have been demonstrated to be mostly empty, thus making density estimation even more difficult (Plaza *et al.*, 2009).

1.3 Issues in Dimensionality Reduction

Reducing the high dimension data set into lower dimension without sacrificing significant information of interest is thus one of the significant steps in hyperspectral image processing. In hyperspectral image classification, effective features are those which are most capable of preserving class separability (Hsu, 2000). Here features represent bands. After feature reduction, a subset of features is obtained. But, preserving the useful information from hyperspectral sensor is the key issue in feature reduction, which demands for development of appropriate algorithms. The meaning of reduction of hyperspectral data means the reduction in the spectral dimension by keeping the spatial dimension unaltered. So this reduction is called as feature reduction or dimension reduction. Feature reduction of hyperspectral data may be done by two ways, either by feature selection or by feature extraction.

(i) feature selection – selecting an optimum number of subsets from original data that allow the user to extract the land cover information as accurate as possible.
(ii) feature extraction - allows transforming the original data into a new feature space, from which the salient features (i.e., reduced feature space) are taken for further processing to derive information.

Feature selection has the advantage of preserving the relevant original information from the data (Adolfo Martínez-Usó *et al.*, 2007; Archibald and Fann, 2007). Some of the feature selection methods used for reducing hyperspectral data are sequential forward selection (SFS), sequential backward selection (SBS), sequential forward floating selection (SFFS), sequential backward floating selection (SBFS) (Somol *et al.*, 1999), distance measures (Bhattacharya distance, Mahalonobis distance, JM distance, etc.,), divergence analysis, steepest ascent, fast constrained search, feature similarity measure, graph searching algorithms, neural networks, genetic methods, tabu search meta-heuristics, spectral distance metrics, parametric feature weighting, spatial autocorrelation, band ratioing, wavelet based feature selection, etc. Standard feature selection methods based on class separability measures such as divergence, JM distance and transformed divergence may not be used effectively as a result of several considerations (Chang 2007),

(i) First, the number of permutations of band subsets is intensively large for hyperspectral data and hence practically impossible.
(ii) Second, the resulting subsets may not be as information-rich as the features consisting of linear combinations of bands generated by transformation-based methods for feature reduction (Richards and Jia, 1999).

Optimal search algorithms identify the subset that contains a prefixed number of features and that is the best in terms of the adopted criterion function, whereas suboptimal search algorithms select a good subset that contains a prefixed number of features but that is not necessarily the best one. Due to their combinatorial complexity, optimal search algorithms may not be used when the number of bands is larger than a few tens. In these cases, (which obviously include hyperspectral data), the use of suboptimal algorithms become mandatory (Serpico, 2007).

The goal of feature extraction is to reduce the number of dimension substantially without sacrificing significant information (Hsu, 2007). Feature extraction may require the whole (or most) of the original data representation to extract the new features, forcing to always obtain and deal with the whole initial representation of the data. In addition, since the data are transformed, some critical information may have to be compromised and distorted (Martínez-Usó *et al.*, 2007). But feature extraction methods are more effective than feature selection methods (Serpico and Moser, 2007).

Some of the standard feature extraction techniques are principal component analysis ((PCA) (Richards, 1993; Jensen and Walts, 1997; Schowengerdt, 1997), segmented principal component analysis (SPCA) (Jia and Richards, 1999), the decision boundary method (Lee and Landgrebe, 1993), independent component analysis (ICA) (Hyvarinen *et al*, 2000; Robila *et al.*, 2000), orthogonal sub-space projection (OSP) (Harsayani and Chang, 1994), projection pursuit (PP) (Ifarraguerri and Chang, 2000), wavelet transformation (Cheriyadat and Bruce, 2003; Zhang *et al*., 2014), discriminant analysis feature extraction (DAFE) (Landgrebe, 2003; Fukunaga, 1990), penalized discriminant analysis (Hastie *et al*., 1995; Ye *et al*., 1999), kernel Fisher discriminant (Muller *et al*., 2001; Baudat and Anouar, 2000; Mika *et al*., 1999), projection pursuit (PP) (Jimenez and Landgrebe, 1999), decision boundary feature extraction (DBFE) (Lee and Landgrebe, 1993).

The PCA involves the eigen value analysis of the co-variance matrix obtained from the given data (Cheriyadat and Bruce 2003, Gonzalez and Woods, 2001). The popularity of PCA is due to its simplicity and ease of use. However, the PCA may not be an optimal method for feature extraction in classification and target detection applications. Cheriyadat and Bruce (2003) have demonstrated the ineptness of PCA in extracting discriminating features from certain data distributions. The higher order principal components (PCs) do not always guarantee retention of the discriminatory information present in the original feature space. It has also been shown that for certain cases of data distribution, the transformed features derived using PCA do not provide better discrimination than the original subset of features. Under a supervised classification problem, the authors investigate the use of a class specific co-variance matrix as the basis of transformation, which leads to different Karhunen-Loeve transformation (KLT) methods. When the original bands are highly correlated, PCA works efficiently, while for poorly correlated data it yields little change (Shah *et al*., 2002)

The special characteristic of hyperspectral data is the peaks and valleys of spectral response of classes. But PCA fails to retain these sharp variances in the spectral response of classes (Cheriyadat and Bruce, 2003). Higher order principal components do not always retain desired distinct features. Therefore, alternative feature extraction methods for better performance in target detection and supervised classification with hyperspectral data has been suggested by Tsai *et al*. (2007).

When feature extraction techniques have been applied to hyperspectral data, the principal components analysis (PCA) outperforms those feature extraction techniques that are based on class statistics. But there are limitations in using PCA. First, it requires high computational load. Second, the variances of spectral bands in the short wavelength region are much higher than the remaining bands if the data are not calibrated. Finally, PCA operates on global statistics and thus it may overlook the local variances that are helpful for the detection of targets and anomalies (Cheriyadat and Bruce, 2003).

Hyperspectral data have the property in which the correlations between neighboring spectral bands are generally higher than for bands further apart, with high correlations appearing in blocks. If the conventional PCA is modified so that the transformation is carried out by avoiding the low correlations between the highly correlated blocks, the efficiency of the PCA will be improved. Highly correlated bands are selected as the subgroups in a segmented PCA (SPCA) scheme so that the PCA works efficiently, since it depends on redundancy reduction. Moreover, the difference between the transformed data obtained by conventional PCA and that from the new scheme will be minimized when the new bands corresponding to the high eigen values in all subgroups are kept. The solar spectrum weighting imposed on each band within a subgroup, corresponding to a narrow region of wavelength, tends to be relatively uniform. Therefore, the bands in a subgroup with similar variance will not suffer from the bias problem that occurs with a conventional PCA (Jia and Richards 1999).

The classical feature extraction techniques contribute little to class separability. Canonical analysis and discriminant analysis feature extraction are often useful when applied to multispectral data, but have disadvantages when applied to hyperspectral data. These include difficulty in calculating covariance matrices with high dimensional data, and unreliability of extracted features when the classes have similar means or when a class has a very different mean from other classes (Richards and Jia, 2006; Hsu, 2007).

Recently, a multi-resolution technique from wavelets has been introduced to study hyperspectral data for feature extraction (Moon and Merenyi, 1995; Bruce *et al.* 2002). Wavelets have been used to detect crop zinc stress assessment (Liu *et al.*, 2011) for hyperspectral image. A wavelet is a mathematical function used to divide a signal or vector into different frequency components, affording analysis of each component with scale dependent resolution. A wavelet transform is the representation of a function by wavelets. The strength of the wavelet transform for hyperspectral feature extraction lies in this ability to analyse signal at different resolution or scales (Bruce *et al.,* 2001). The advantages of multi-scale representation of hyperspectral data are twofold. First, subtle variation in spectral features in the original hyperspectral data may be detected at different scales (Hsu, 2007). Second, the useful information is represented by fewer wavelet features, effectively compressing the data (Bruce *et al.*, 2002; Peng *et al.*, 2013; Banskota *et al.*, 2011).

Although some wavelet-based methods, such as the matching pursuit, the non-linear wavelet feature extraction (WFE) and the best basis algorithms (Kumar *et al.*, 2001) are based on the best approximation for data representation, they are effective for classification. Especially, the nonlinear wavelet-based methods are more effective for classification than linear methods. In some circumstances, the matching pursuit basis had better results than the best wavelet packet basis (Hsu, 2007). However, it has been demonstrated that the reduction of features will be in powers of two may lead to either reducing more features with loss of some useful information or taking more features for further processing (Richards and Jia, 2006).

1.4 Image Classification

Similar to classification of multispectral data, here also the classification of hyperspectral data may be done by two ways, per pixel classification and sub pixel classification. In per pixel classification, each pixel is allotted to one and only one class while in case of sub pixel classification, the proportion of class availability in each pixel is determined.

1.4.1 Issues in Per Pixel Classification

In hyperspectral image analysis, especially in classification and detection applications, spectral characterization plays a more crucial role. To determine spectral variability, similarity, discrimination or divergence, many spectral measure criteria that calculate different distance metrics have been proposed over the past few decades. These include maximum likelihood decision metric, spectral angle mapper (SAM), spectral correlation mapper (SCM), spectral information measure (SIM), Euclidean minimum distance (EMD), spectral gradient angle (SGA), and band add-on spectral angle mapper (BAO-SAM). Appropriate distance metrics employed in hyperspectral data processing for classification and detection application can produce accuracy classification by describing spectral characteristics in mathematical or physical meaning properly (Wang *et al.*, 2009).

In N dimensional multi-(or hyper) spectral space, a pixel vector x has both magnitude (length) and an angle measured with respect to the axes that defines the coordinate system of the space (Richards and Jia, 1996). In the Spectral Angle Mapper (SAM) technique for identifying pixel spectra, only the angular information is used. SAM is based on the idea that an observed reflectance spectrum can be considered as a vector in a multidimensional space, where the number of dimensions equals the number of spectral bands. If the overall illumination increases or decreases (due to the presence of a mix of sunlight and shadows), the length of this vector will increase or decrease, but its angular orientation will remain constant (Richards and Jia, 1996). The limitations of SAM are:

(i) The SAM algorithm may not distinguish between positive and negative correlations because a spectral angle value measured by SAM may be generated from two spectral vectors with random interrelation (Wang *et al*., 2009).

(ii) Further, due to sensitivity to the additive factor in the feature space, the SAM holds high false alarm rate generally (Wang *et al*., 2009). The SAM may also fail if the vector magnitude is important in providing discriminating information, which it will in many instances.

Therefore, during last decade or so, a number of machine learning algorithms have become popular in producing image classification for hyperspectral data. These include Artificial Neural Network (ANN) (Foody *et al*., 1997, Carpenter *et al*., 1999) and, decision tree classification (Richards and Jia, 1999; Pal, 2006; Watanachaturaporn *et al*., 2008), support vector machines (Brown *et al*., 2000; Watanachaturaporn and Arora, 2004; Sun *et al*., 2013). These techniques are attractive as they do not suffer from the problem of dimensionality as observed in many statistical classifiers. There are several studies conducted on the use of ANN for per pixel classification from multispectral data (Foody *et al*., 1997).

However, ANN has two limitations particularly in reference to hyperspectral data,

(i) First, as in any neural network, the number and sizes of the hidden layers need to be set. Generally, the problem is over specified and some form of pruning is used to generate a minimum network that will solve the problem at hand. Nevertheless, the issue is not straightforward (Chang, 2007).

(ii) Second, a very large number of iterations are often required to find a solution which leads to computation overload.

Support Vector Machines (SVMs) are large margin classifiers that exploit the principles of the statistical learning theory (Vapnik 1998). If an L2-norm regularizer is used, the optimization problem related to the learning of SVMs can be represented as a quadratic convex optimization problem with inequality constraints. For such optimization problems in nonlinear optimization theory, duality is preferred. Thus, SVMs are often solved in dual representation by introducing Lagrange multipliers. However, this is not mandatory since one can also implement SVMs in the primal representation (Chapelle, 2007; Bruzzone, 2007; Chi et al., 2008).

Although, SVM appears to be the most appropriate classifier for any hyperspectral image classification problem, however, its major limitation lies in the selection of appropriate kernel function, selection of the suitable multiclass method and the choice of appropriate value of the parameters for the selected kernel function. Moreover, SVMs also have high algorithmic complexity and extensive memory requirements due to quadratic programming in applications requiring a large datasets. The computational complexity also increases in case of non-linear SVM when the data are projected into higher dimension it's computationally expensive.

1.4.2 Issues in Sub Pixel Classification

In per-pixel classification, each pixel is assigned to one and only class, whereas in case of sub-pixel classification the fraction abundance tells the proportion of available classes in each pixel. The hyperspectral sensors which are efficient to collect fine details from the ground, are useful to identify the type of vegetation, the type of mineral and the objects present in the image. For example, Apan *et al.* (2004) used EO-1 Hyperion hyperspectral imagery to detect disease in sugarcane. Various researchers used hyperspectral data to find fine details from the image. Asner and Heidebrecht (2002) used hyperspectral data to unmix the land cover classes vegetation, soil and dry carbon cover in arid region. Here they compared the output with multispectral data output. Benediktsson and Kanellopoulos (1999) and Benediktsson *et al*., (2005) used hyperspectral data for classification which covers urban area and to separate the bright and dark regions, and applied mathematical morphology as a pre-processing step. Okin *et al*. (2001) discussed about some practical difficulties on hyperspectral vegetation discrimination in arid and semiarid environments.

From the literature, it can also be deciphered that the typical algorithms used for sub-pixel classification of hyperspectral data are linear mixture modeling (Lu *et al*., 2003; Kasetkasem, 2011; Shanmugam *et al*., 2006, Shanmugam and Abhishekh, 2006), orthogonal subspace projection (Kwon and Nasrabadi, 2005), independent component analysis mixture model (Shah *et al*., 2004), etc.

The linear mixing model has been widely used due to its strong tie between the mathematical foundations of the model and the physical processes of mixing that result in much of the variance seen in hyperspectral imagery. However, issues arise when the basic assumptions of the model are violated and it fails to accurately represent the nature of hyperspectral imagery. One such situation is nonlinear mixing. Another reason is the assumption that mixtures of a small number of deterministic spectra can be used to represent all of the non-noise variance in hyperspectral imagery.

Depending upon whether or not the constraints are imposed, there are unconstrained or constrained approaches and hence two solutions. The end member spectra are generally assumed to be pure. However, for certain land cover studies obtaining required number of end member spectra is quite impossible. This is particularly true when the signatures are extracted directly from an image scene with no precise knowledge of ground truth. So the orthogonal subspace projection tackles some of these limitations.

The idea of OSP is to divide the p substances into two classes, desired substance class and undesired substance class. Without loss of generality, we assume that the desired substance class contains only one single substance and undesired substance class consists of the remaining p – 1 substances.

It is also well-known that linear classifiers fail when the data are not linearly separable. However, by transforming the original data into a much higher dimensional space (feature space) by using an appropriate non-linear mapping, the mapped data will probably become linearly separable in the high-dimensional feature space where a linear classifier can be applied (Sebastiano, 2001).

In real imagery, when light reflects off surfaces that are composed of an intimate mixture of various material components that cause multiple bounces, spectral mixing tends to become nonlinear (Keshava and Mustard, 2002; Keshva, 2003). But both LMM and OSP are linear algorithms based on a linear mixture model, which do not exploit the higher order correlations between the spectral bands nor it addresses the nonlinear mixing of the spectral signatures that are encountered in real data. Therefore, LMM and OSP are not flexible enough to fully exploit the complex data structure encountered with real hyperspectral imagery. The higher order statistics which are useful in discriminating classes is not possible by LMM and OSP. So ICAMM, an unsupervised classifier, which exploits the higher order statistics, came into existence.

The ICA mixture model (ICAMM) algorithm vfiews the observed data as a mixture of several mutually exclusive classes. Each of these classes is described by a linear combination of independent components with non-Gaussian densities. The ICAMM algorithm finds independent components and the mixing matrix for each class using an extended information-maximization learning algorithm and computes the class membership probability for each pixel. The pixel is allocated to the class with the highest posterior class probability to produce the classification map (Hyvärinen and Oja, 2000, Shah *et al.*, 2004). Nevertheless, some limitations of ICAMM are (Hyvärinen and Oja 2000),

(i) The independent components may have at most one non-Gaussian distribution.

(ii) It requires lots of computation in finding the fraction abundance, particularly if the number of bands is more than the number of classes.

(iii)We cannot determine the variances (energies) of the independent components.

(iv)We cannot determine the order of the independent components (Hyvärinen and Oja 2000). Since the extraction of mixing matrix starts from random initialization of mixing matrices, the clusters permute with respect to the other clusters. The order of fraction abundance values need not same for every run.

The fraction maps derived from the sub pixel classification may provide more useful land cover information as compared with that derived from hard classification techniques. However, the spatial distribution of each class in these mixed pixels may not be ascertained from sub-pixel classification outputs. Only the building areas could be estimated more precisely using sub-pixel classification technologies, whereas the boundaries of the urban buildings could not be determined (Ling and Fu, 2009). To find details upto that level, super resolution mapping may be helpful. The art of producing fine spatial resolution map from a coarse spatial resolution is called super resolution mapping.

1.5 Issues in Super Resolution Mapping

Super resolution mapping is a promising technology for prediction of the spatial distribution of each class at the sub-pixel scale. This distribution is often determined based on the principle of spatial dependence and from fraction images derived with sub-pixel classification technology. This technology uses the fraction maps derived with sub pixel classifications as input and converts them into high resolution maps based on the land cover spatial pattern, which is often described with the maximum spatial dependence principle. The super resolution mapping has its origin by converting a single gray scale image into finer by simply dividing the pixels and rearranging according to the neighbouring pixels. Current super-resolution mapping methods include the Hopfield neural network (HNN) (Tatem *et al.*, 2003), linear optimization (Verhoeye and Wulf, 2002), genetic algorithm (Mertens *et al.*, 2003), feed-forward neural network (Mertens *et al.*, 2004), Markov random field (Kasetkasem *et al.*, 2005), pixel swapping (Atkinson

2005; Thornton *et al*., 2006), simulated annealing (Makido *et al*., 2007) and geostatistical methods (Boucher and Kyriakidis, 2006; Boucher and Kyriakidis, 2007).

Super resolution mapping of a hyperspectral image has been done by artificially multiple-sub-pixels shift of the original data. Also, by fusion of multi-observation images by sub-pixel shifted to get more accurate image of higher spatial resolution than the original observations. This can be done by iteratively back propagation algorithm (Peleg *et al*., 2002; Mianji *et al*., 2009a; Mianji *et al*., 2009b; Lu *et al*., 2010).

To map a certain features alone like roads, water bodies, by giving the characters of that particular feature the super resolution map has been obtained (Foody *et al*., 2005, Ling *et al*. 2008, Akgun *et al*., 2005). Super resolution mapping algorithm based on an MRF model has also been proposed. It is assumed that a super resolution map (SRM) has MRF properties, i.e., two adjacent pixels are more likely to belong to the same land cover class than different classes. By integrating this fact into the model, a large number of misclassified pixels, which often appear as isolated pixels, are removed from the resulting SRM (Kasetkasem *et al*., 2005). By using multiple sub-pixel shifted remotely sensed images super resolution map will be obtained (Ling *et al*., 2010). Low-resolution pixels in these remotely sensed images contain different land-cover fractions that can provide useful information for super-resolution land cover mapping. Here, by constructing a Hopfield Neural Network (HNN) model the sub-pixels are mapped by sub-pixel shift of the original image (Ling *et al*., 2010). Maximum spatial dependence is the goal of the proposed model, and the fraction maps of all images are constraints added to the energy function of HNN (Ling, 2009). The drawbacks of this technique are:

(i) The information on accurate boundary feature (i.e., the reference data) may not be readily available.

(ii) It does not consider the spatial dependence within and between pixels.

A spatial-spectral data fusion technique was discussed by Mianji *et al*., (2010). In this algorithm, the four main steps used are endmember extraction, spectral unmixing, training of the SRM algorithm and super resolution mapping. Here training data are taken from high spatial resolution hyperspectral image. Based on the assumption of spatial correlation of the land cover classes, simulated annealing is used to optimize a function where spatial proximity of pixels belonging to the same land cover class is preferred (Villa *et al*., 2011).

1.6 Research Gaps

From the brief review presented above, a few research gaps in extracting information from hyperspectral data have been identified as,

1. A number of feature extraction algorithms for hyperspectral have been developed each having its own limitations. The efficacy of these techniques on hyperspectral data have yet to be established. There appears to be no work conducted in the direction of evaluating the quality of features obtained from feature extraction techniques.

2. The computational load of SVM is heavy and the parameter fixation of SVM has always been an issue to investigate.

3. The ICAMM for various types of datasets with various spatial resolutions needs to be explored. The classifier has been studied less for extracting fraction outputs from hyperspectral data.

4. While producing super resolution map from a hyperspectral image, the basic assumption is that the spatial proximity of pixels belonging to the same land cover class is preferred. But classes may fall in the center of the pixel also, which has not been studied earlier.

5. Targets/anomalies which fall at the inner portion of a pixel via super resolution mapping have not been addressed.

6. Attention is required to perform accuracy assessment for super resolved image only for super resolved mixed pixels.

1.7 Objectives of the Research

The main objective of this research is to produce a fine spatial resolution map after applying various feature extraction techniques, per pixel and sub-pixel classification in a sequential manner. In this process, several specific research objectives have been defined,

1. Investigating the use of wavelet and PCA based techniques for feature extraction.

2. Assessing the usefulness of SVM for extracting information at per pixel level from hyperspectral data.

3. Study and implementation of supervised and unsupervised techniques for extracting information at sub-pixel level from hyperspectral data.
4. Development of a novel algorithm to produce super resolution map from hyperspectral data.
5. Accuracy assessment of extracting information from hyperspectral data at each stage.

1.8 Overview of Methodology

The methodology framed to fulfill the aims of these research objectives includes several processing steps, namely, feature extraction, per pixel and sub-pixel classification and super resolution mapping of hyperspectral data and is portrayed in Figure 1.5. Three hyperspectral datasets have been used for experimental purposes which are from different sensors, different spatial and spectral resolutions and different types of datasets. Two datasets from AVIRIS sensor with spatial resolution 20 m and 4 m respectively and one HYPERION dataset which is of spatial resolution 30 m. Details of these datasets have been provided in Chapter 3.

1.8.1 Feature Extraction of Hyperspectral Data

From the three hyperspectral data sets, the features are reduced by two feature extraction techniques: wavelet based and PCA based approaches. Three families of wavelets, namely, Haar, Daubechies, and Coiflets have been used. The features are extracted by decomposing each of the pixel vectors of hyperspectral data at 1-level decomposition to 4-level decompositions. A total of 32 different extractions have been performed for each dataset. The other feature extraction technique is based on PCA applied to segments depends upon the spectral nature of the classes. The spectral region is divided into four segments namely, visible, NIR, SWIR – I and SWIR – 2. Now PCA is applied on each of the segments and only first few PCs from each segment have been collected for further investigation.

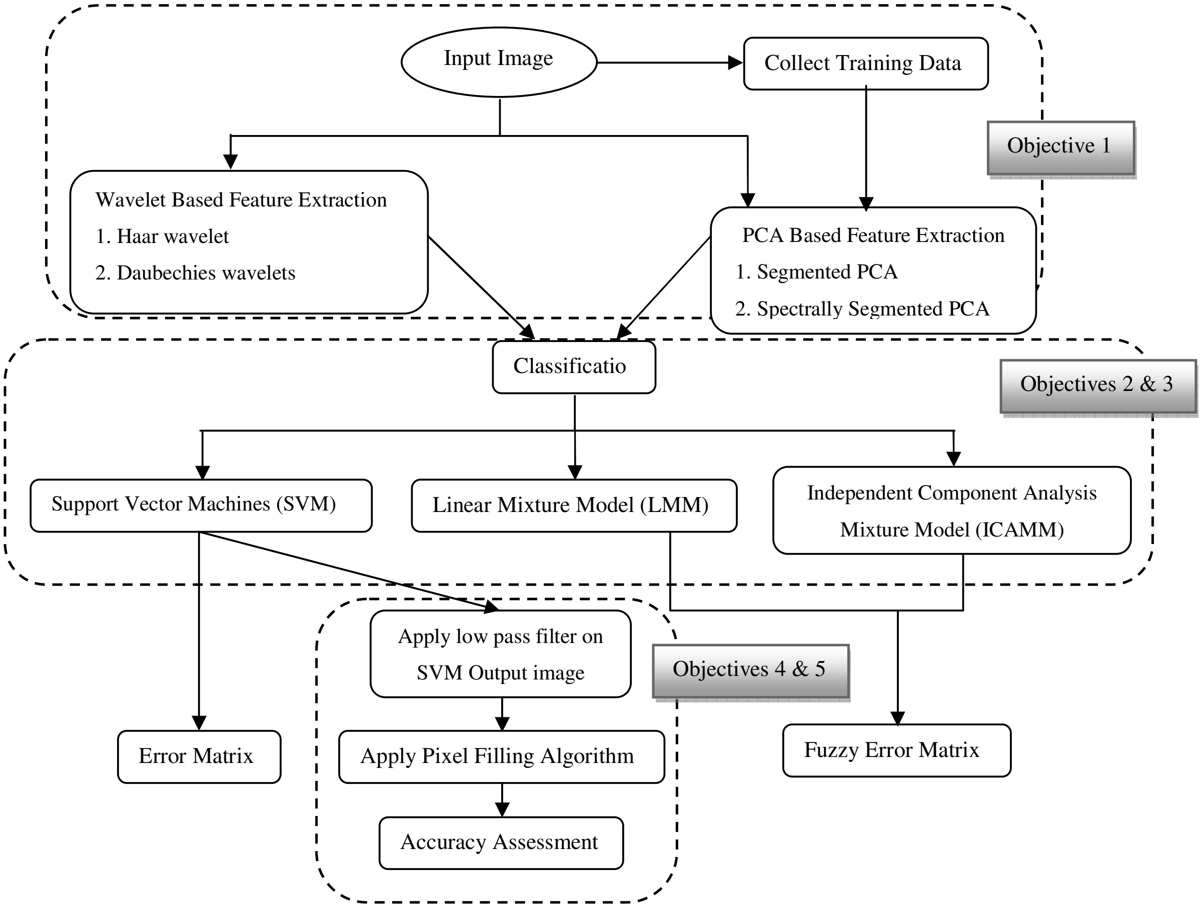

Input Image
Collect Training Data
Objective 1
Wavelet Based Feature Extraction
1. Haar wavelet
2. Daubechies wavelets
PCA Based Feature Extraction
1. Segmented PCA
2. Spectrally Segmented PCA
Classificatio
Objectives 2 & 3
Support Vector Machines (SVM)
Linear Mixture Model (LMM)
Independent Component Analysis Mixture Model (ICAMM)
Apply low pass filter on SVM Output image
Objectives 4 & 5
Error Matrix
Apply Pixel Filling Algorithm
Fuzzy Error Matrix
Accuracy Assessment

Figure 1.5: The overall methodology adopted in this research

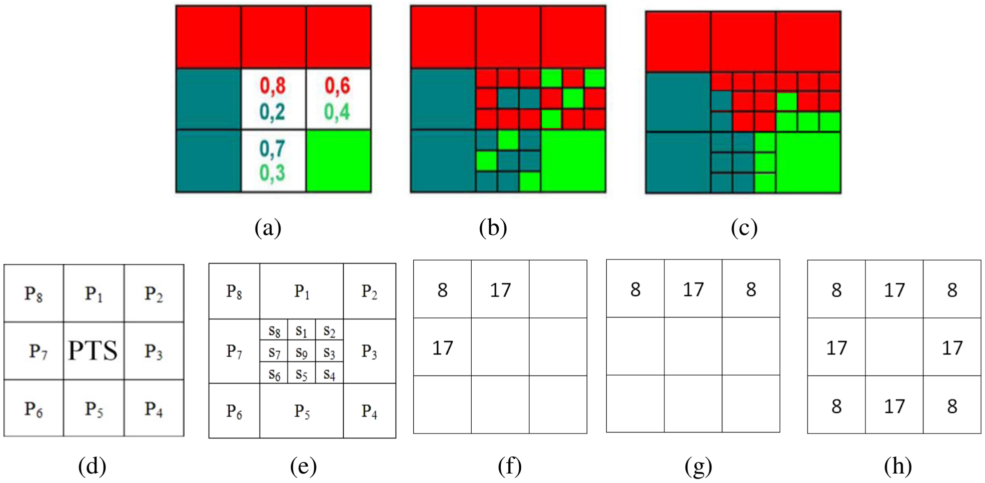
1.8.2 Classification of Hyperspectral Data

The extracted features from each of the datasets form the input data to the per- and sub-pixel classification algorithms. The supervised per pixel classification, supervised sub pixel classification and unsupervised sub pixel classification by SVM, LMM and ICAMM respectively have been performed.

The main advantage of using SVM classifier for hyperspectral data classification is that it requires less training samples. Particularly here, in dataset I, number of available pixels for classes' alfalfa, grass/pasture-mowed and oats are very less whose number of training pixels are 22, 18 and 11 respectively. For dataset II, the number of pixels for class sand is less and for dataset III class vegetation is having only 32 pixels for training.

Here, one-against-one SVM classifier has been used with tolerance value 0.0001 and penalty value one to extract the classes.

For sub-pixel classification, since ICAMM exploits higher order statistics, it has been used to extract classes from the datasets. Here in dataset I, there is a highway running across east to west on the top of the dataset for which pure pixels are not available. Also, in dataset III, some metallic objects like aircraft, metallic cylinder may not be extracted using LMM but possibly by ICAMM. In this study the proportionality constant involve in updating the mixing matrix has been studied for various spatial resolution, various textured datasets.



(a) (b) (c)

(d) (e) (f) (g) (h)

Figure 1.6: (a) Abundance fraction got from spectral unmixing, (b) Random initialization of sub-pixels according to zoom factor, (c) Final map after super resolution mapping algorithm is performed, (d) PTS and its surrounding pixels, (e) Arrangement of subpixels of PTS, (f), (g), (h) respective weights to be convolute with neighbouring pixels.

1.8.3 Super Resolution Mapping

To super resolve each of the pixels in the hyperspectral data, the class proportion of each of the pixels are required. The spatial proximity of the neighbouring pixels is considered. The pixel to be super resolved is divided into z^2 sub pixels where z is the zoom factor (z is odd). First total number of sub pixels to be mapped to each of the classes is calculated and then a linear combination is calculated by taking the number of classes to be mapped multiplied with the weights, where the weights depends upon the spatial location of the sub pixel. In this algorithm, for zoom factor z, the total number of weights are (z + 1)/2. These weights vary from 8 to 17. For z = 3, there are 2 weights 8 and 17. For z= 5, there are 3 weights and they are $w_1 = 8$, $w_2 = 13$ and $w_3 = 17$. In the PTS, the corner sub pixels depend upon the corresponding diagonal pixel and the two pixels adjacent to PTS on the same side in which it presents.

There are 4 types of sub pixel locations in PTS and they are filled in their own way. They are (i) middle sub pixel (only one sub pixel), (ii) corner sub pixels (denoted by C) (4 sub pixels), (iii) pixels which falls on the middle line (both horizontal and vertical) but not middle sub pixel, (iv) Remaining sub pixels. For zoom factor z, there could be only (z + 1)/ 2 levels. Then weight matrices (Figure 1.6) for each class have been calculated by linear combination of the number of sub pixels along with the level at each sub pixel location. Greater the value the sub pixel is occupied by the class attributes from the corresponding matrix.

Since SVM output has been subsampled by 3x3 or 5x5 low-pass filter, only the best output from SVM classified for all the datasets are taken for super resolution mapping.

The quality of derived outputs at various stages of processing has been assessed based on conventional error matrix (Story and Congalton, 1986; Congalton, 1991) based accuracy assessment procedures, namely, overall accuracy, user's accuracy, producer's accuracy and kappa coefficients. The quality of feature extraction has been assessed by finding out the accuracy of classification (by SVM) obtained from the extracted features used. For sub pixel classification, fuzzy error matrix has been generated for each fraction output. Due to the non availability of reference or the ground data, the best classification output from SVM has been taken as reference data. In case of super resolution mapping, the accuracy of super-resolved map at fine spatial resolution has again been assessed using conventional error matrix based accuracy measures. Also small patches of mixed pixels, mixed by 2 classes, mixed by 3 classes have been taken as testing pixels for spatial accuracy assessment.

1.9 Organization of the Thesis

The thesis has been systematically organized into seven chapters including the current chapter on introduction. In this chapter, the motivation of the work has been given followed by a brief introduction to hyperspectral imaging. Some issues on the processing of hyperspectral data have been discussed. From that a set of research gaps has been identified. Research gaps helped in framing the research objectives. A skeleton of the overall methodology to fulfill the objectives is also given.

A brief literature review has been given in *Chapter 2* for all the various processing tasks. Literature review has been done on three major topics including feature extraction of hyperspectral data, classification of hyperspectral data and super resolution mapping of hyperspectral data. A description on PCA and wavelet based feature extraction techniques for feature extraction of hyperspectral data and per pixel and sub pixel classification algorithms has been provided. Review on SVM classification, LMM classification and ICAMM classification algorithms have been given. Finally, a review on super resolution mapping has been given for hyperspectral data.

In *Chapter 3*, introduction of the datasets has been given which is followed with a detailed methodology which includes training data, testing data and implementation details.

The feature extraction of hyperspectral data part has been described in *Chapter 4*. Besides the description of feature extraction of hyperspectral data, wavelet based and PCA based techniques have been elaborated. The mathematical background and implementation of those techniques has been given. Finally results got from feature extraction techniques have been discussed.

In *Chapter 5*, mathematical background and techniques of the three classifiers SVM, LMM and ICAMM have been given. Then the methodologies of each classifier have been described followed with accuracy assessments and observations from the result for all the hyperspectral datasets have been explained. The chapter concludes with a discussion of results.

A separate chapter has been dedicated to super resolution mapping and it is *Chapter 6*. The description of a novel super resolution mapping algorithm, developed here, has been given. The details of pixel swapping algorithm have also been provided described. It followed by description of the methodology, experiments, results and discussion on super resolution mapping from hyperspectral data.

Finally, the conclusions derived from this research have been narrated in *Chapter 7*. A few recommendations and possibility of future work have also been suggested in this chapter. At the end of this thesis, lists of references and recent publications have been given.

Chapter 2

Literature Review

2.1 Introduction

The urge to know more about the earth surface is ever increasing. Remote sensing has become a viable technology to extract information about the earth surface on almost real time basis. The remote sensing data are now available at many different spatial, spectral, radiometric and temporal resolutions. The spectral resolution has been enhanced from a panchromatic image to hyperspectral images and in future to ultraspectral also. The hyperspectral sensors, which are capable of acquiring information in contiguous, narrow width wavelength bands and that too in hundreds of bands offer lot of information about earth's surface. These data are being used in a number of applications such as studying types of vegetation species (Tipping, 2000, 2001), (Okin, 2001), coastal vegetation mapping (Schmidt *et al.*, 2004), detecting disease in vegetation (Apan *et al.*, 2004), mineral exploration (Neville *et al.*, 2003), soil classification, military target applications (Arora and Tiwari, 2013), urban environment, (Pathak and Dikshit, 2004a; Pathak and Dikshit, 2004b), estimating grassland biomass, (Clevers et al., 2007), retrival of images (Sawant, *et al.*, 2006) etc. Therefore, to extract information for a range of applications, visual interpretation of hyperspectral data may not efficient. The data have to undergo several digital image processing operations to derive desired, effective and quality information in an efficient manner.

However, the voluminous hyperspectral data are fraught with difficulties of long processing time. Also, at times the data acquired in neighbouring contiguous bands may be highly correlated. It is therefore necessary to reduce the number of bands without compromising on the quality of information extraction. Further, typically, supervised image classification is applied to extract the information from hyperspectral data, which is highly dependent on the size of the training data which in turn depends upon the number of bands used for classification. Sufficient training data are required to be input to the classification process in order to avoid Hughes' phenomenon (Hughes, 1968). Collection of pure training data has always been a difficult task due to time, money and other constraints. It is therefore expedient to reduce the feature space thereby reducing the training data size requirement and hence satisfy the Hughes' phenomenon.

The feature reduction of hyperspectral data may be possible by feature extraction (FE); projecting the original data set onto adequate sub-spaces. Feature extraction algorithms are categorized into two main approaches: supervised and unsupervised feature extraction. Principal component analysis (Fukunaga, 1990), independent component analysis (Hyvarinen and Oja, 2000), unsupervised linear feature extraction (Rodriguez *et al.*, 2007) and maximum noise fraction (Green *et al.*, 1988) are examples of unsupervised feature extraction techniques. The most common feature extraction technique employed in remote sensing data analysis is the principal component analysis (PCA). Since PCA operates on global statistics, it may overlook local variances that are helpful in classification. PCA may not necessarily be an optimal method for feature extraction of hyperspectral data (Cheriyat and Bruce, 2003) especially for target detection and supervised classification applications. Jia and Richards (1999) presented a segmented PCA (SPCA) procedure to extract the best principal components. However, if the goal is to detect a specific plant type, it may not be an efficient way to segment spectral bands. To identify a predetermined plant species a spectrally segmented PCA system specifically designed (Tsai *et al.*, 2007) for hyperspectral remote sensing imagery. Also wavelets have been used to extract important features and signature classification of hyperspectral data (Bruce *et al.*, 2002). Once the feature reduction of hyperspectral data has been achieved, then the classification algorithms may be applied on the reduced datasets. By mapping each pixel to single land cover class, a per-pixel classification may be obtained whereas finding the proportion of class present in each pixel results into a sub-pixel classification. Some of the commonly used per pixel classification algorithms include Maximum Likelihood Classification (MLC) (Song *et al.*, 2005), Decision Tree Classification (Brodley *et al.* 1996), Spectral Angle Mapper (SAM), Spectral Correlation Mapper (SCM), Spectral Information Measure (SIM), Euclidean Minimum Distance (EMD), Spectral Gradient Angle (SGA), Band Add-On Spectral Angle Mapper (BAO-SAM), Support Vector Machines (SVM) (Melgani and Bruzzone, 2002; Huang *et al.*, 2002), Evidential Reasoning (ER), Artificial Neural Networks (ANN) (Foody and Arora, 1996; Mohan, 2000; Reddy and Mohan, 2005, Foody *et al.*, 1997), Decision-Tree Regression (Min *et al.*, 2005; Goel et al., 2003), Logistic Regression (Cheng *et al.*, 2006), relevance vector machine (Demir and Erturk, 2007), etc. Each of these algorithms has its own advantages and disadvantages when handling hyperspectral data. In a similar way, few sub pixel classification algorithms have also been used in classification of hyperspectral data. Some of the algorithms used in information extraction of hyperspectral data are MLC in soft form, Linear Mixture Model (LMM) (Foody and Arora, 1996; Heinz and Chang, 2001; Kasetkasem *et al.*, 2011; Dias *et al.*, 2012), Fuzzy set based methods, Independent Component Analysis Mixture Model (ICAMM), (Hyvarinen *et al.*, 2001; Robila and Varshney, 2002; Wang and Chang, 2006), probabilistic SVM (Platt, 1999; Villa *et al.*, 2011), etc.

The class proportions (also known as fractional abundance) of each pixel may be useful information in many applications but the spatial location of a class within each pixel is not available from the sub-pixel classification outputs. The solution to this problem may be obtained through super resolution mapping. Thus, the output from super resolution mapping is a finer resolution image from the coarse spatial resolution hyperspectral data. Super resolution mapping technique may be performed using regression-type algorithms and spatial optimization based algorithms.

As the focus of this research is on feature extraction, sub-pixel classification and super resolution mapping of hyperspectral data, a brief literature covering these aspects has been provided in this chapter.

2.2 Feature Extraction of Hyperspectral Data

The hyperspectral data may either be reduced by feature selection (Jain and Zongker, 1997, Bruzzone and Serpico, 2000) or feature extraction (Liu and Motoda, 1998; Serpico *et al.*, 2003; Zortea and Haertel, 2004; Tsai *et al.* 2007). Feature extraction allows transforming the original data into a new feature space, from which the prime features (i.e., reduced feature space) are taken for further processing to derive useful information.

Some of the prevalent feature extraction techniques include principal component analysis (PCA) (Richards, 1993; Jensen and Walts, 1997, Schowengerdt, 1997), segmented principal component analysis (SPCA) (Jia and Richards, 1999), the decision boundary method (Lee and Landgrebe, 1993), independent component analysis (ICA) (Hyvarinen *et al.*, 2001, Robila *et al.*, 2000), orthogonal sub-space projection (OSP) (Harsayani and Chang, 1994), projection pursuit (PP) (Ifarraguerri and Chang, 2000), wavelet transformation (Cheriyadat and Bruce, 2003), etc. These techniques are capable of compressing the information present in the original hyperspectral data into a subset of uncorrelated components.

In this research, two types of feature extraction techniques namely wavelet based feature extraction and PCA based feature extraction techniques have been used. Hence, a review on these two techniques applied on hyperspectral data has been given.

2.2.1 Wavelet Based Feature Extraction of Hyperspectral Data

The advantage of using wavelets in hyperspectral data is their capability of reducing the dataset exponentially. The reduction in the number of bands of hyperspectral data will depend on the level of decomposition. Higher the number of decomposition level, more will be the reduction. The wavelets are generated from a base function called 'mother wavelet'. The convolution of any bounded function with mother wavelet produces wavelet coefficients. These wavelet coefficients are used in extracting features from hyperspectral data. The wavelet transform is used to dissect the signal or pixel vector of a hyperspectral data into different frequency components and then depending upon the frequency components they are used in further processing. Wavelet analysis of a pixel vector of a hyperspectral data is performed by scaling and shifting the wavelet function which produces the wavelet coefficients. By doing so, there occurs a similar pattern between the spectral curve and the wavelet coefficients (Blackburn, 2007). The earlier uses of wavelet transformations can be found in signal processing. Of late, wavelets have also been used in remote sensing for various image processing related tasks such as image compression (Lee *et al.*, 1994), image texture feature analysis (Fukuda and Hirosawa, 1999), feature extraction (Pittner and Kamarthi, 1999; Simhadri *et al.*, 1998) and image fusion (Nu´n˜ez *et al.*, 1999; Wang et al., 2012), soil moisture retrieving using hyperspectral data (Peng *et al.*, 2013).

Wavelet based transforms provide transformation of a pixel vector of a hyperspectral data from the time domain to the time-frequency domain. The performance of wavelets on dimensionality reduction of hyperspectral data has been studied in the last decade (Bruce *et al.,* 2002; Kaewpijit *et al.,* 2003; Cheriyadat and Bruce, 2003; Wang *et al.*, 2012; Singh *et al.*, 2012; Chen *et al.*, 2009) and wavelet feature-based classification (Koger *et al.,* 2003; Pu and Gong 2004; Zhang *et al.,* 2014; Schmidt *et al.,* 2007).

Bruce *et al.* (2002) applied Haar, Daubechies and Coiflets wavelets on hyperspectral data for vegetation mapping and found that wavelet based features resulted in significant increase in the classification accuracy as compared to the spectral band selection and PCA based reduction. Among 1000 spectral bands for consideration, after feature extraction by different mother wavelets, the number of bands reduced to nearly

50% since 1-level decomposition was performed. The wavelet based techniques produced higher overall accuracy than PCA based feature extraction for per pixel classification as well as sub pixel classification. For PCA based feature extraction the per pixel classification accuracy was 67.9 percent while that for wavelets it ranged from 91.2 percent to 98.6 percent. For sub pixel classification, the accuracy was around 50 percent for PCA while for wavelets it varied from 77 percent to 97 percent (Bruce *et al*., 2002).

Kaewpijit *et al.* (2003) considered only discrete orthonormal bases of wavelets and compared the wavelet based feature extraction with PCA. The experiments were conducted on three hyperspectral datasets, two AVIRIS data and one AISA data. Here, upto 5 levels of decomposition were performed. In comparison to PCA based feature extraction, the wavelet based feature extraction produced better classification accuracy for most of the classifications. Further, as the decomposition level increased the classification accuracy decreased.

The discrimination capability for different classes in the original signal and the wavelet based decomposed signal may be different, since projection of the signal onto a wavelet function can separate the low pass and high pass signals from hyperspectral pixel vectors (Bruce *et al*., 2002). By adopting the receiver operating characteristics (ROC) curves, Bruce *et al.* (2002), selected the optimum subset wavelet coefficients as the reduced feature set, and then on the reduced dataset a maximum-likelihood classifier was applied to evaluate the effectiveness of the extracted features. The reduction of number of bands was exponential when applying wavelet transform when the decomposition level increased. Kaewpijit *et al.* (2003) also proved that wavelets based feature extraction technique was efficient in dimension reduction for hyperspectral data. Here the decomposition level is selected automatically. The wavelet based technique has been effective for detecting the weed pitted morning glory in soybean crops (Koger *et al.,* 2003). Pu and Gong (2004) conducted an experiment to analyse the effectiveness of wavelet based feature extraction as a spectral feature extraction method for mapping forest crown closure and leaf area index (LAI) with Earth Observing (EO)-1 Hyperion data. By performing the wavelet energy feature (WEF) as a pre-processing step, Zhang *et al.* (2005) performed two classification techniques, namely the maximum-likelihood and hidden Markov models, to classify soil texture.

Not only for classification of hyperspectral data but also for other applications like quantifying vegetation pigment concentrations (Blackburn, 2007), wavelet decomposition has been used. The wavelets used here are Reverse Biorthogonal Wavelet 3.3 (rbio3.3), Coiflets 4 (Coif4), Symlets 8 (sym8), Meyer Wavelet (dmey) and Biorthogonal Wavelets (bior various sub-classes – small change in the scaling function of the corresponding mother wavelet). Here the decomposition level has been from 3 to 5 found that wavelets are much useful in quantifying pigment concentration of vegetation across leaf and canopy scales (Blackburn, 2007).

Hence, in remote sensing applications, a few wavelets, namely, Haar, Daubechies wavelets (only one or very few sub-classes), Symlet wavelets (only one sub-class), etc., have been studied. However, an in-depth analysis of these wavelets for feature extraction and classification of hyperspectral data has been lacking.

2.2.2 PCA Based Feature Extraction of Hyperspectral Data

PCA is the most widely used technique for feature extraction in remote sensing community because of its simplicity but it operates on the global statistics and thus may overlook local fluctuations which are helpful in discriminating classes and detection of targets (Cheriyadat and Bruce, 2003, Tsai *et al.*, 2007). Due to this, an advanced version of PCA by segmenting the spectral bands with the help of correlation matrix, named as SPCA, was introduced (Jia and Richards, 1999). This technique is not much effective for detecting a specific plant type.

ICA is also a statistical tool which exploits higher order statistics in reducing the data. The implementation of reducing technique for hyperspectral data by ICA is by maximizing the non-Gaussianity of each component. But ICA is iterative based and it takes lot of computation time in reducing the data (Robila *et al.*, 2000). In ICA, all the required components are retrieved at simultaneously while in case of projection pursuit the retrieval is done by component by component (Ifarraguerri and Chang, 2000). Also finding the optimum number of components is inexplicable. Taking more components leads to redundancy while leaving some components becomes sub-optimal. But for hyperspectral data, PCA and its variant has not been studied much for feature extraction. Hence an advanced version of PCA may be required to extract information from hyperspectral data.

2.3 Classification Techniques of Hyperspectral Data

The classification of hyperspectral data may be performed to extract information at pixel level, either by mapping each pixel to a single land cover class (per pixel classification) or by finding the proportion of class present in each pixel (sub pixel classification). A number of algorithms have been proposed in the last decades for classification of hyperspectral data. Algorithms which are useful for classification of multi-spectral data might not be useful in hyperspectral data due to the high dimensionality of hyperspectral data and other issues, as discussed in Chapter 1.

2.3.1 Per Pixel Classification of Hyperspectral Data

Some of the techniques are the Maximum Likelihood Classifier (MLC) (Song *et al.*, 2005), Spectral Angle Mapper (SAM) technique (Sohn and Rebello, 2002; Sohn *et al.*, 1999), for identifying pixel spectra, only the angular information is used, machine learning algorithms include ANN (Foody *et al.*, 1997; Carpenter *et al.*, 1999), decision tree classification and support vector machines (Brown *et al.*, 2000) have been used. There are several studies conducted on the use of ANN for per pixel classification (Foody *et al.*, 1997). There is a range of different network architectures (Aleksander and Morton, 1990; Davalo and NaÈõm, 1991) and a range of potential applications in remote sensing. Feed-forward artificial neural networks can be trained to learn by example and are attractive for supervised classification (Schalkoff, 1992). They also can readily accommodate multisource data acquired at different levels of measurement precision, are free of distribution assumptions, and can process data rapidly once trained (Peddle *et al.*, 1994). Such networks have been used for image classification (e.g., Kanellopoulos *et al.*, 1992)

In this research, one per pixel classifier (SVM) which is non-parametric and supervised in nature has been used. Also, two more classifiers for sub pixel classification; a supervised classifier, LMM, and an unsupervised classifier, ICAMM, have been used. Hence a brief literature on these classification algorithms has been given.

2.3.1.1 Support Vector Machines (SVM)

Support Vector Machines (SVMs) are large margin classifiers that exploit the principles of the statistical learning theory (Vapnik, 1998; Watanachaturaporn and Arora, 2004; Watanachaturaporn *et al.*, 2008). If an L2-norm regularizer is used, the optimization problem related to the learning of SVMs can be represented as a quadratic convex optimization problem with inequality constraints. For such optimization problems in nonlinear optimization theory, duality is preferred. Thus, SVMs are often solved in dual representation by introducing Lagrange multipliers. However, this is not mandatory since one can also implement SVMs in the primal representation (Chapelle 2007). The most advantage of using SVMs in the field of remote sensing is the ability to produce accurate classification even with the small training dataset (Mantero *et al.*, 2005). Even though

SVMs have more benefits, few challenges are also there. The major challenge is the choice of kernels. A plenty of options are available, but some of the kernel functions may not provide optimal SVM solution for remote sensing applications. Experimental evidence indicates that kernels such as polynomial kernels for higher order and radial basis function applied on SVM-based classification of remote sensing data produce different results (Zhu and Blumberg, 2002). A good review on SVM kernels and their functionality is presented in numerous papers (e.g., Kavzoglu and Colkesen, 2009). A detailed survey on SVMs applied on various types of remote sensing datasets and for various applications has been given in Mountrakis *et al.* (2010). The incorporation of localized, highly sensitive transformations to capture fine changes in hyperspectral signatures has been addressed by Sahoo *et al.* (2007). They compared the so called S-transform to classifiers without it and found encouraging results. By incorporating border training samples Demir and Erturk (2009) produced an improvement to classification of hyperspectral data by SVM in a two step classification process.

Bazi and Melgani (2006) performed experiment on selecting the most appropriate feature subspace and model selection based on a genetic optimization framework using three feature selection methods including steepest ascent, recursive feature elimination technique, and the radius margin bound minimization method. The genetically optimized SVM using the support vector count as a criterion resulted in the best performance for both simulated and real-world AVIRIS hyperspectral data.

Pal (2006) explored methods for feature selection based on SVMs. He addressed the unreasonably large computational requirements as a major drawback of exhaustive search methods in practical applications. Also the researcher justified that non-exhaustive search procedure is useful in selecting features with high discriminating power from large search spaces. SVM-based methods combined with genetic algorithm (GA) produced marginal better accuracy when compared with the random forest feature selection method in land cover classification problems with hyperspectral data.

A modified recursive SVM was proposed by Zhang and Ma (2009) who applied the approach to reduce AVIRIS dataset. The reduced dataset produced marginally better result but the computational burden was higher than other methods. A hybrid classification by integration of SVMs with pairwise decision trees was proposed by Chen *et al.* (2008). Here the proposed method produced similar results from both the methods..

A new classification scheme was proposed by Tarabalka *et al.* (2009) by highlighting both the spatial and the spectral characteristics of hyperspectral data. In their method, they combined the pixel-wise SVM classification results with the segmentation map based on partitional clustering using the majority voting strategy. This hybrid type of approach was useful when large spatial structures were included in the dataset or when separability of spectral signatures of classes was more. For vegetation classification, Gualtieri and Cromp (1998) evaluated performance of SVM on hyperspectral AVIRIS data. SVM performed better than other classifiers applied on the same dataset. SVM also outperformed Gaussian maximum likelihood classification and k-NN techniques in forest species classification (Dalponte *et al.*, 2009). Dalponte *et al.* (2009) classified a complex forest scenario of a hyperspectral data. SVMs were used in classification of crops using HyMap hyperspectral imagery (Camps-Valls *et al.* (2004)). SVMs performed better than typical neural networks in terms of three factors namely, accuracy, simplicity, and robustness. They also found that SVMs were not as sensitive to training sample size, and SVMs were able to successfully detect noisy bands. Hyperspectral image data of a cornfield, acquired through airborne mission (Compact Airborne Spectrographic Imager) was used in conjunction with the SVM method in automatic detection of weeds and nitrogen in the field (Karimi *et al.*, 2006). The discriminant features were based on the general remote sensing principle: corn exhibits different spectral responses depending on the type or method of weed control used and nitrogen application rates.

Karimi *et al.* (2006) while using CASI data of cornfields for automatic detection of weeds and nitrogen by SVM. Linear support vector machines were reported to be useful in classification of hyperspectral remote sensing data whose elements had been extracted using a technique called kernel principal component analysis (KPCA) (Fauvel *et al.*, 2007). Although only the basic SVM was employed in the set of experiments, the improved feature provided a significant clue on the effectiveness of SVMs especially when applied on reliably clean datasets.

An experiment was performed by Melgani and Bruzzone (2004) for classification of AVIRIS data using SVMs and radial basis function (RBF) neural networks and the K-nearest neighbor classifier. They observed that SVMs outperformed the other techniques and concluded that SVMs were effective approaches to hyperspectral remote sensing data. Another experiment was performed on Landsat 7ETM+and hyperspectral data by Pal and Mather (2005) to compare SVM classification with MLC and back propagation ANN.

Here also SVM outperformed both the classifiers for the two datasets. Hence the usage of SVM for classification of hyperspectral data has a significant role in extraction techniques of hyperspectral data. Moreover, the training of SVM requires less pixels compared to MLC technique. Hence in this research, SVM has been used for analysing feature extraction and per pixel classification techniques.

2.3.2 Sub Pixel Classification of Hyperspectral Data

The sensors providing data at coarse spatial resolutions such as AVIRIS, Hyperion and MODIS increase the possibility occurrence of mixed pixels. Hence, the conventional per pixel classification algorithms that allocate one class to each pixel may not be appropriate for the classification of mixed pixels. Alternative approaches for classifying mixed pixels are, therefore, sought. Some sub-pixel classification techniques include, maximum likelihood classification (MLC) (Jia and Richards, 1994) in soft form, linear mixture model (LMM) (Foody and Arora, 1996; Lu *et al.*, 2003; Kasetkasam *et al.*, 2011), fuzzy set based methods (Chanussot *et al.*, 2006), artificial neural networks (ANN) (Foody and Arora, 1996), support vector machines (SVMs) (Platt, 2000) in soft form, decision-tree regression (Xu *et al.*, 2005), logistic regression (Cheng *et al.*, 2006), orthogonal subspace projection (Kwon, 2005) and independent component analysis mixture model (Lee *et al.*, 1999; Lee *et al.*, 1999; Shah *et al.*, 2004), etc.. So, an algorithm which is capable of estimating the class member probability for each pixel is required.

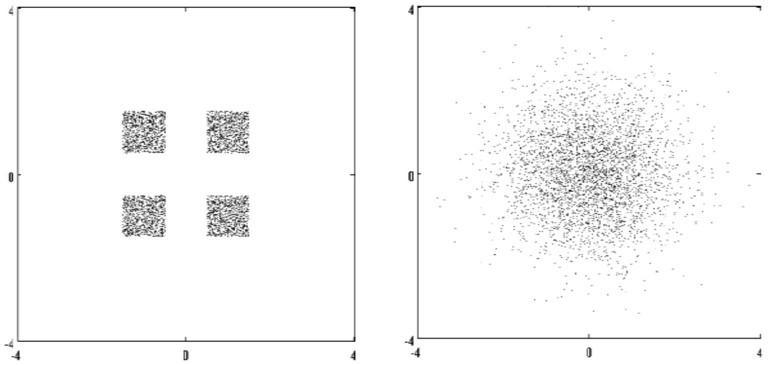
Unlike in per-pixel classification algorithms, here in sub-pixel classification the fraction abundance for each pixel have been determined. The hyperspectral sensors which are efficient to collect fine details from the ground, are useful to identify the type of vegetation, the type of mineral and the objects present in the image. Apan *et al.* (2004) used to detect disease in sugarcane using EO-1 Hyperion hyperspectral imagery. Various researchers used hyperspectral data to find fine details (Asner and Heidebrecht, 2002; Benediktsson and Kanellopoulos, 1999; McGwire *et al.*, 2000; Okin *et al.*, 2001; Shah *et al.*, 2004). The LMM approach assumes that the spectrum measured by a sensor is a linear combination of the spectra of all components within the pixel (Adams *et al.*, 1995; Roberts *et al.*, 1998; Ustin *et al.*, 1998). Both LMM & OSP are algorithms based on a linear mixture model, which do not exploit the higher order correlations between the spectral bands nor it addresses the nonlinear mixing of the spectral signatures that are encountered in real data. Therefore, LMM & OSP are not flexible enough to fully exploit the complex data structure encountered with real hyperspectral imagery. So ICAMM which exploits the higher order statistics came into existence. The fraction maps derived from the soft classification may provide more useful land cover information as compared with that derived from hard classification technologies. If needed, the pixel is allocated to the class with the highest posterior class probability to produce thematic map (Hyvärinen and Oja, 2000; Shah *et al*., 2007).

2.3.2.1 Linear Mixture Model (LMM)

The linear mixture model (LMM) algorithm is based on the concept of spectral unmixing. The LMM approach assumes that the spectrum measured by a sensor is a linear combination of the spectra of all components within the pixel (Manolakis *et al*., 2001). Both LMM and OSP are linear algorithms based on a linear mixture model, which do not exploit the higher order statistics in the data nor it addresses the nonlinear mixing of the spectral signatures that are encountered in real data. Therefore, LMM and OSP are not flexible enough to fully exploit the complex data structure encountered with real hyperspectral imagery. Less work has been performed for classification of hyperspectral data. Hence a deep analysis for various spatial, spectral resolution images needs to be classified by using LMM.

2.3.2.2 Independent Component Analysis Mixture Model (ICAMM)

An approach for unsupervised classification of hyperspectral images is based on a mixture model, where the distribution of the entire data is modeled as a weighted sum of the class-conditional densities. If the classes are modeled as multivariate Gaussian distribution the mixture model is known as Gaussian Mixture Model (GMM) (Duda *et al*., 2001) while, if the classes are modeled as multivariate non-Gaussian distribution then the mixture model is Independent Component Analysis Mixture Model (ICAMM). GMM works by considering the lower order statistics of the data. Classification is done on the basis of second order statistical parameters derived from spectral characteristics of each class, which may not be sufficient in discriminating the within-class variation. For example, in Figures 2.1(a) and 2.1(b), both the spectral data sets are having same mean and same variance values; however, there are four clusters in Figure 2.1(a) and only one cluster in Figure 2.1(b) (Shah and Varshney, 2004). Thus, second order statistics are not capable of revealing the information available in the data. In ICAMM, kurtosis, a higher order statistics, plays an important role in making the clusters as independent as possible. The ICAMM algorithm finds the linear transformation from the data to independent components for each cluster.



(a) (b)

Figure: 2.1: Two data sets having identical second order statistics

Even sometimes the sub pixel proportion of each class available in each pixel of a hyperspectral data is not enough but the spatial arrangement of the classes in sub pixel level is required. This could be possible by performing super resolution mapping algorithm. A study on super resolution mapping algorithms has been given in the subsequent section.

2.5 Super Resolution Mapping

The sub pixel classification gives the proportion of classes available in each of the mixed pixels, but fails to locate the classes at sub-pixel level. Super resolution mapping is a promising technology for prediction of the spatial distribution of each class at the sub pixel scale. The super resolution mapping has its origin by converting a single gray scale image into finer by simply dividing the pixels and rearranging according to the neighbouring pixels. Current super-resolution mapping methods include the Hopfield neural network (HNN) (Tatem *et al.*, 2001), linear optimization (Verhoeye and Wulf, 2002), genetic algorithm (Mertens *et al.*, 2003), feed-forward neural network (Mertens *et al.*, 2004), Markov random field (Kasetkasem *et al.*, 2005), pixel swapping (Atkinson, 2005; Thornton *et al.*, 2006), simulated annealing (Makido *et al.*, 2007), inverse Euclidean algorithm (Tiwari *et al.*, 2011) and geostatistical methods.

Super resolution mapping of a hyperspectral image has been performed by artificially multiple-sub-pixels shift of the original data. The geostatistical approach described in Villa *et al.* (2011) has been studied for mixed pixels mixes with only two classes. Here the algorithm has been applied on a region where sea meets the shore. So this is almost considered as linear boundary classification problem. So a general super resolution mapping algorithm is required which maps mixed pixels mixed with two or more classes. The main limitation of spatial regularization by simulated annealing (Thornton *et al*., 2006) is in its incapability of mapping linear features. Since the objective function used in this algorithm is the cost function to be minimized, where the cost function here is the perimeter of the areas belonging to the same class. So, error in one pixel will be carried over to the other pixels also. So an algorithm in which the super resolution mapping of one pixel does not affect the super resolution mapping of the other is to be considered. The drawback of the spatial pixel swapping algorithm (Shen *et al*., 2009) is it is also incapable of super resolving linear pixels. In case of target detection using pixel swapping algorithm (Arora and Tiwari, 2013) the binary class problem may be extended to multi-class problem.

Collectively, by considering the limitations of the above said super resolution mapping algorithms, we propose an algorithm, based on inverse Euclidean distance, which works better for pixels having linear boundary and multi-class classification problem.

Also, by fusion of multi-observation images by sub-pixel shifted to get more accurate image of higher spatial resolution than the original observations. This can be done by iteratively back propagation algorithm (Peleg *et al*., 2002; Lu *et al*., 2010). To map a certain features alone like roads, water bodies, by giving the characters of that particular feature the super resolution map was obtained (Foody *et al*., 2005; Ling *et al*., 2008). Super resolution mapping algorithm based on an MRF model was also proposed. It is assumed that a super-resolution map (SRM) has MRF properties, i.e., two adjacent pixels are more likely to belong to the same land cover class than different classes. By integrating this fact into the model, a large number of misclassified pixels, which often appear as isolated pixels, are removed from the resulting SRM (Kasetkasem *et al*., 2005). By using multiple sub pixel shifted remotely sensed images super resolution map will be obtained (Ling *et al*., 2010). Low-resolution pixels in these remotely sensed images contain different land-cover fractions that can provide useful information for super-resolution land cover mapping. Here, by constructing a Hopfield Neural Network (HNN) model the sub-pixels are mapped by sub-pixel shift of the original image (Ling *et al*., 2010). Maximum spatial dependence is the goal of the proposed model, and the fraction maps of all images are constraints added to the energy function of HNN (Ling 2009). A spatial-spectral data fusion technique has been discussed by Mianji (2010). In this algorithm, the four main steps used are endmember extraction, spectral unmixing, training of the SRM algorithm and super resolution mapping. Here training data are taken from high spatial resolution hyperspectral image. Based on the assumption of spatial correlation of the land cover classes, simulated annealing has been used to optimize a function where spatial proximity of pixels belonging to the same land cover class is preferred (Villa *et al.*, 2011). Here a novel algorithm to perform super resolution mapping of hyperspectral data which has more class is required.

2.6 Summary of Review

In this chapter, the following review on hyperspectral data has been given

1. A brief literature on feature extraction of hyperspectral data has been given and more specifically the uses of the two techniques, wavelet based feature extraction technique and PCA based feature extraction technique have been discussed.
2. The two major divisions of classification techniques, per pixel and sub pixel classification have been given.
3. In particular, a review of SVM, LMM and ICAMM for classification of hyperspectral data has been given.
4. Finally, a review on super resolution mapping for hyperspectral data has been given.

Chapter 3

Experimental Datasets and Methodology

3.1 Introduction

In order to achieve the research objectives, three hyperspectral datasets have been taken at varied spatial, spectral and radiometric resolutions to extract information from the algorithms developed. Further, as the emphasis of the research is on assessment of algorithms, one synthetic dataset has also been created. Detailed description of these datasets has been given in this chapter. The spectral, spatial and radiometric resolutions of these datasets have been given in Table 3.1.

Table 3.1: The specifications of experimental datasets

Dataset No.	Dataset	Sensor	Dimension (pixels)	Spatial Resolution (m)	Spectral Resolution (bands)
Dataset 1	Synthetic Data	-	45x60	-	-
Dataset 2	Indiana Pine	AVIRIS	145x145	20	224
Dataset 3	Roorkee & its Surroundings	Hyperion EO-I	250x250	30	242
Dataset 4	San Diego	AVIRIS	140x150	4	224

3.2 Description about Training and Testing Data

The two important components of remote sensing data are the training data and testing data. The training data are useful in classification of the data while testing data helps in validation of the classified data. The descriptions of these two types of data have been discussed in this section.

3.2.1 Training Data

Training is one of the important steps in supervised classification of hyperspectral. From the available classes, pure pixels have been taken whose quantity is proportion to the area covered by each class in the dataset. Hence, the number of pixels used in the training stage varies for each class. Collecting pure pixels from a dataset depends upon the spatial resolution of the dataset, available classes, size of the classes, etc. Particularly, when the data is of coarse spatial resolution, the domination of mixed pixels may be more and extraction of collecting training data becomes complex. Generally, for each class, instead of collecting a big block training data, few small patches at various locations, if available, for the training data may be collected. The details of training samples and testing/reference samples for each of the three datasets have been given in Table 3.2, Table 3.3 and Table 3.4.

3.2.2 Reference Data

To validate the performance of feature extraction of hyperspectral data, first it has been classified by a per pixel classifier SVM. Then based on the accuracy assessment obtained through error matrix based accuracy measures, the quality of information extraction has been assessed. To perform accuracy assessment, a set of samples which are pure pixels from the data, whose class is known, is collected. Such pure pixels are collectively known as reference data or testing samples. The number of testing samples collected for each class for each image has been given in Table 3.2, Table 3.3 and Table 3.4.

3.3 Description of Experimental Datasets

The datasets used in fulfilling the objects of this research have been discussed in this section. One synthetic data and three hyperspectral data have been used. The descriptions of these datasets have been given in this section.

3.3.1 Experimental Dataset I: Synthetic Data

A synthetic data of dimension 45x60 pixels has been created by imitating the spectral signatures from AVIRIS San Diego dataset (described as Experimental Dataset IV). Five classes, namely, vegetation, steel, metallic road, aircraft and concrete road (or simply called as concrete) have been considered to map from this dataset. The area of interest (.aoi) for each class has been demarcated on the image and the spectral responses are collected. For each class, the average of the spectral response in each band is obtained. As, there are 5 classes and 189 bands, 5 pixel vectors each of length 189 are generated. A 2D matrix of dimension 45x60 pixels consisting of values from 1 to 5 representing class labels is also created. Finally, a 3D image with pixel vectors of length 189 each has been created, by associating number 1 (which is in 2D matrix) to pixel vector 1. Similar exercise has been done for other class labels 2 to 5. An FCC of the image has been given in Figure 3.1. Clearly, the classes are very distinct from each other in this synthetic data. The resolution of image has been coarsened by applying low-pass filters of sizes 3x3 and 5x5 (Figure 3.2 and Figure 3.3). These coarse resolution images shall be subjected to super resolution algorithm.

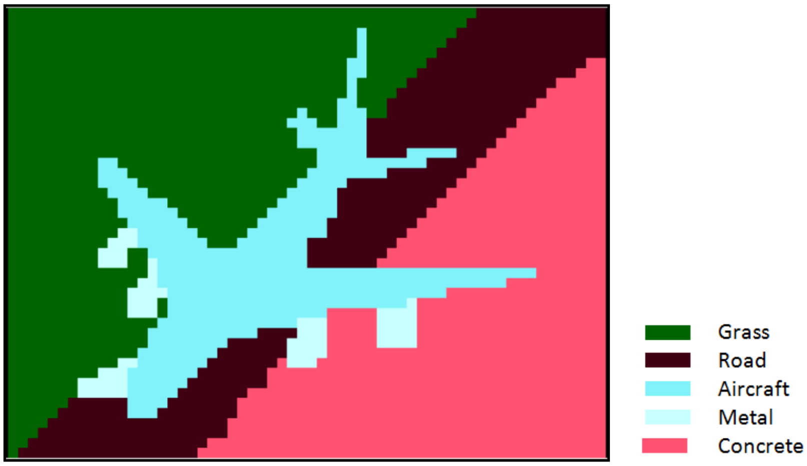


Figure 3.1: Synthetic data having five classes. Band combination is 170, 80, 20. Dimension of the image is 45 x 60 pixels.

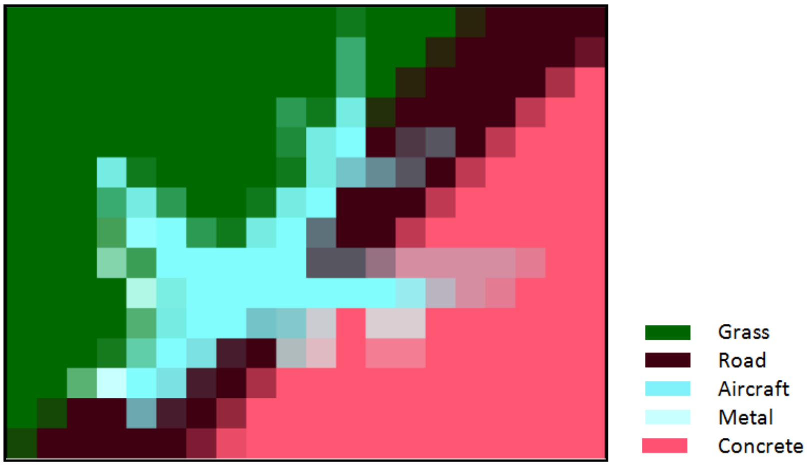


Figure 3.2: Synthetic data sub-sampled by 3x3 low-pass filter. Band combination is 170, 80, 20. Dimension of the image is 15 x 20 pixels.

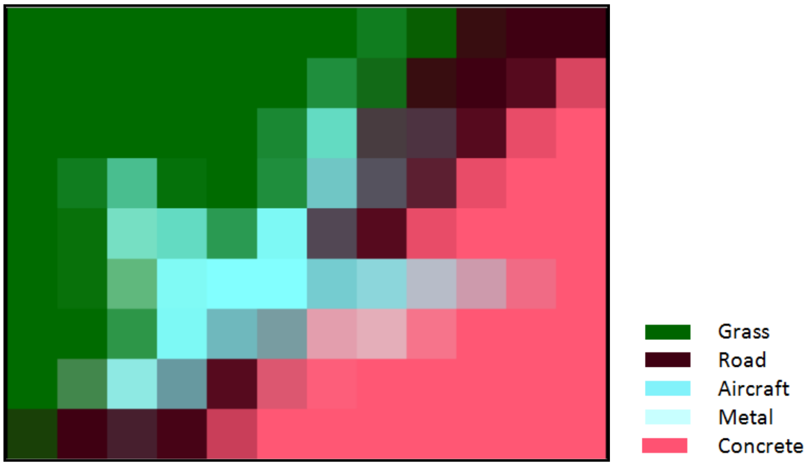
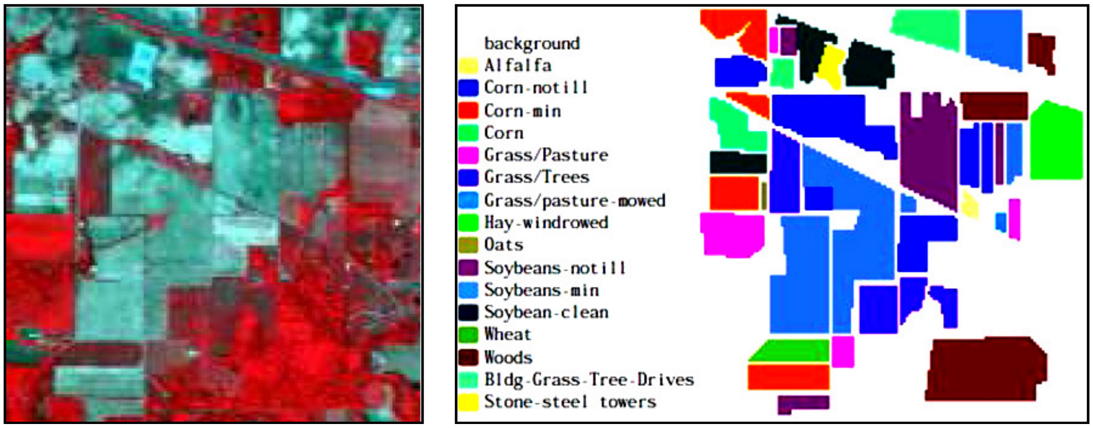


Figure 3.3: Synthetic data sub sampled by 5x5 low-pass filter. Band combination is 170, 80, 20. Dimension of the image is 9x12.

3.3.2 Experimental Dataset II: AVIRIS Dataset at 20m Spatial Resolution

This dataset contains an image from AVIRIS hyperspectral sensor (*www.ece.purdue.edu/~biehl/MultiSpec)*. The image is taken over NW Indiana's Indian Pine test site and has been used in many similar studies. From the original 224 spectral bands, four bands containing no data have been discarded. From this set, 38 water absorption bands have also been removed. Thus, 182 bands have been used in the experiment. Additional advantage of this data is the availability of reference image of the area covered, which can be used to extract the training and testing samples required for classification and accuracy assessment. Two third of the area is covered with agriculture and the remaining one third is covered with forest or other natural perennial vegetation. However, due to the early season of data collection, the cultivated land appears to have very little canopy cover as yet. There is a major dual lane highway, a rail line crossing near the top and a major secondary road near the middle, all in a NW and SE direction. A false color composite (FCC) of the image and the reference image are given in Figure 3.4.



(a) (b)

Figure 3.4: (a) FCC (Red: Band 50, Green: Band 27, Blue: Band 17) of AVIRIS image (dataset II) and (b) The reference image for dataset II.

A total of 16 classes have been identified and training and testing samples have been collected for each class, which scales with the area cover of the respective classes. In this dataset, there are 15 classes of different vegetation species and one class is stone-steel tower. Few classes are having less number of pixels like alfalfa, oats, etc. The number of training and testing samples used in classification of this dataset has been given in Table 3.2.

Table 3.2: Number of training and testing samples for experimental dataset – II

S.No.	Class Name	Number of Training Samples	Number of Testing Samples
1	Alfalfa	22	26
2	Corn-notill	341	234
3	Corn-min	279	144
4	Corn	72	132
5	Grass/pasture	188	298
6	Grass/trees	210	184
7	Grass/pasture-mowed	18	24
8	Hay-windrowed	173	315
9	Oats	11	18
10	Soy-notill	291	330
11	Soy-min till	601	408
12	Soy-clean	199	140
13	Wheat	66	126
14	Woods	445	527
15	Bldg-grass-trees-drives	142	80
16	Stone-steel towers	49	62
	TOTAL	**3107**	**2948**

Here for the purpose of classification, 14.78 percent of training pixels have been used while 14.02 percent of pixels for testing. There may be a few pixels which are common to both training and testing samples because of the classes may be having less number of pixels.

3.3.3 Experimental Dataset III: Hyperion Dataset at 30m Spatial Resolution

The third dataset is an image from the sensor Hyperion of EO-1 (Earth Observation-1) acquired over Dehradun area on 25th December, 2006. The data are collected in 220 unique spectral bands covering 0.357–2.576 µm spectrum. The Level 1 product has a total of 242 bands but only 198 bands have been calibrated. Because of an overlap of data between the VNIR and SWIR regions, there are only 196 unique bands. Calibrated bands are 8–57 for the VNIR, and 77–224 for the SWIR. The bands that are not calibrated are set to zero. A subset of image covering Roorkee area and surroundings has been extracted to perform experiments in this research. An FCC of the subset is shown in Figure 3.5.

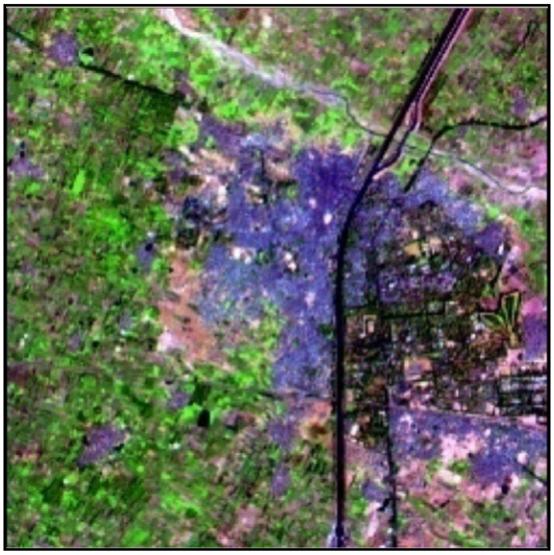


Figure 3.5: The FCC of Roorkee and its surrounding area by Hyperion sensor (Red: Band 70, Green: Band 20, Blue: Band 2).

Table 3.3: Number of training and testing samples for experimental dataset – III

S.No.	Class Name	Number of Training Samples	Number of Testing Samples
1	Trees	125	181
2	Vegetation	88	59
3	Barren Land	85	104
4	Sand	28	28
5	Urban	420	467
6	Water	131	66
	TOTAL	**877**	**905**

This dataset has only 6 major classes which are trees, vegetation, barren land, sand, urban and water. As the spatial resolution is 30, there are mixed pixels. Hence, for classification, 1.4 percent training pixels have been used while 1.45 percent of pixels for testing. Both the datasets are mutually exclusive.

3.3.4 Experimental Dataset IV: AVIRIS Dataset at 4m Spatial Resolution

This AVIRIS dataset is a 400x400 pixel hyperspectral image of naval air station in San Diego, California collected by the AVIRIS sensor in 224 bands and was received as part of the ENVI 4.2 tutorials. Due to the effects of illumination source and the atmosphere, the raw radiance spectra obtained by any hyperspectral sensor is required to be converted into reflectance spectrum. This true reflectance spectrum can only be compared with any laboratory spectra available for identification of the targets. The true colour composite of the image is given in Figure 3.6. A subset of the image of dimension 140x150 has been taken for experiment purpose.

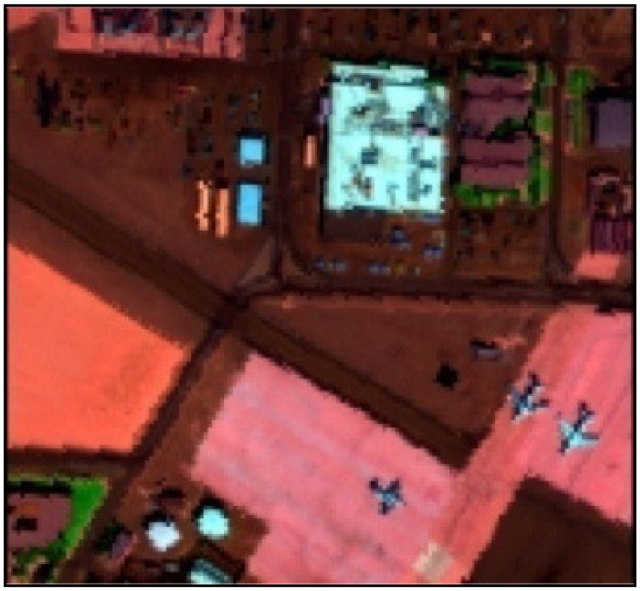


Figure 3.6: Subset of false colour composite (Red : Band 170, Green : Band 80, Blue : Band 20) of Naval Air Station, San Diego, California.

Table 3.4: Number of training and testing samples for experimental dataset – IV

S.No.	**Class Name**	**Number of Training Samples**	**Number of Testing Samples**
1	Vegetation	32	18
2	Road Type 1	70	49
3	Road Type 2	86	42
4	Concrete 1	124	65
5	Concrete 2	235	89
6	Concrete 3	74	36
7	Building Type 1	54	59
8	Building Type 2	72	32
9	Building Type 3	48	36
	TOTAL	**795**	**426**

Here for classification 3.79 percent pixels from the whole dataset have been taken for training while 2.03 percent of pixels for testing. Here also there are no pixels which are common to both training and testing samples. The predominant classes are vegetation, two types of roads, three types of concrete and three types of building types. Even though the spatial resolution of the dataset is 4m, the with-in class variation is very less.

3.2 Methodology

In this section, the methodology followed to achieve the objectives has been given. In this research, the algorithms developed for information extraction from hyperspectral data has followed four image processing tasks in sequential manner. These tasks are,

(i) feature extraction,
(ii) per-pixel classification,
(iii) sub-pixel classification and
(iv) super resolution mapping.

First, the hyperspectral data has been reduced by applying two unique feature extraction algorithms, namely, segmented PCA and wavelet based feature extraction. The reduced datasets are then classified by using the most widely used per pixel classification technique, namely, Support Vector Machines (SVM) for hyperspectral data. The SVM classification forms the basis of assessing the performance of various feature extraction algorithms. The reduced datasets having acceptable accuracy values are then subjected to sub-pixel classification by LMM and ICAMM. The sub-pixel classification outputs have been assessed using fuzzy error matrix based accuracy measures. The information from sub-pixel outputs via LMM or ICAMM then forms the basis of super-resolution mapping by the proposed Pixel Filling Algorithm. The outputs of super resolution mapping have been assessed using three techniques, given as,

(i) Computation of accuracy of super resolved image with respect to reference data as used for assessing a per pixel classification
(ii) Pixel-to-pixel comparison of super resolved image with SVM classified image.
(iii)Analysing the portions of super resolved image where there is likelihood of mixed pixels.

A flow chart for overall methodology has been given in Figure 3.7, which has been followed with a detailed description in the subsequent paragraphs.

For feature extraction using wavelets, three wavelet transforms, namely, Haar, Daubechies and Coiflets wavelets have been used. For each wavelet, the first 4 sub-classes have also been taken. Haar wavelet is the first subclass of Daubechies wavelets. Therefore, we get 4 sub-classes for Daubechies wavelets and 4 sub-classes for Coiflets. These 8 sub-classes are decomposed upto first 4 decomposition levels. Hence there will be a total of 32 feature extracted datasets generated from wavelet based feature extraction. Here, Daub2_4 means Daubechies wavelets 2nd sub-class and level of decomposition is 4. The other feature extraction algorithm is spectrally segmented PCA to get a reduced dataset. Thus, there are 34 reduced datasets, which have been used for classification. The flowchart describing the methodology followed in feature extraction has been given in Figure 3.8. To validate the feature extraction techniques, the reduced datasets are classified using SVM and their accuracy assessed using conventional error matrix based overall accuracy.

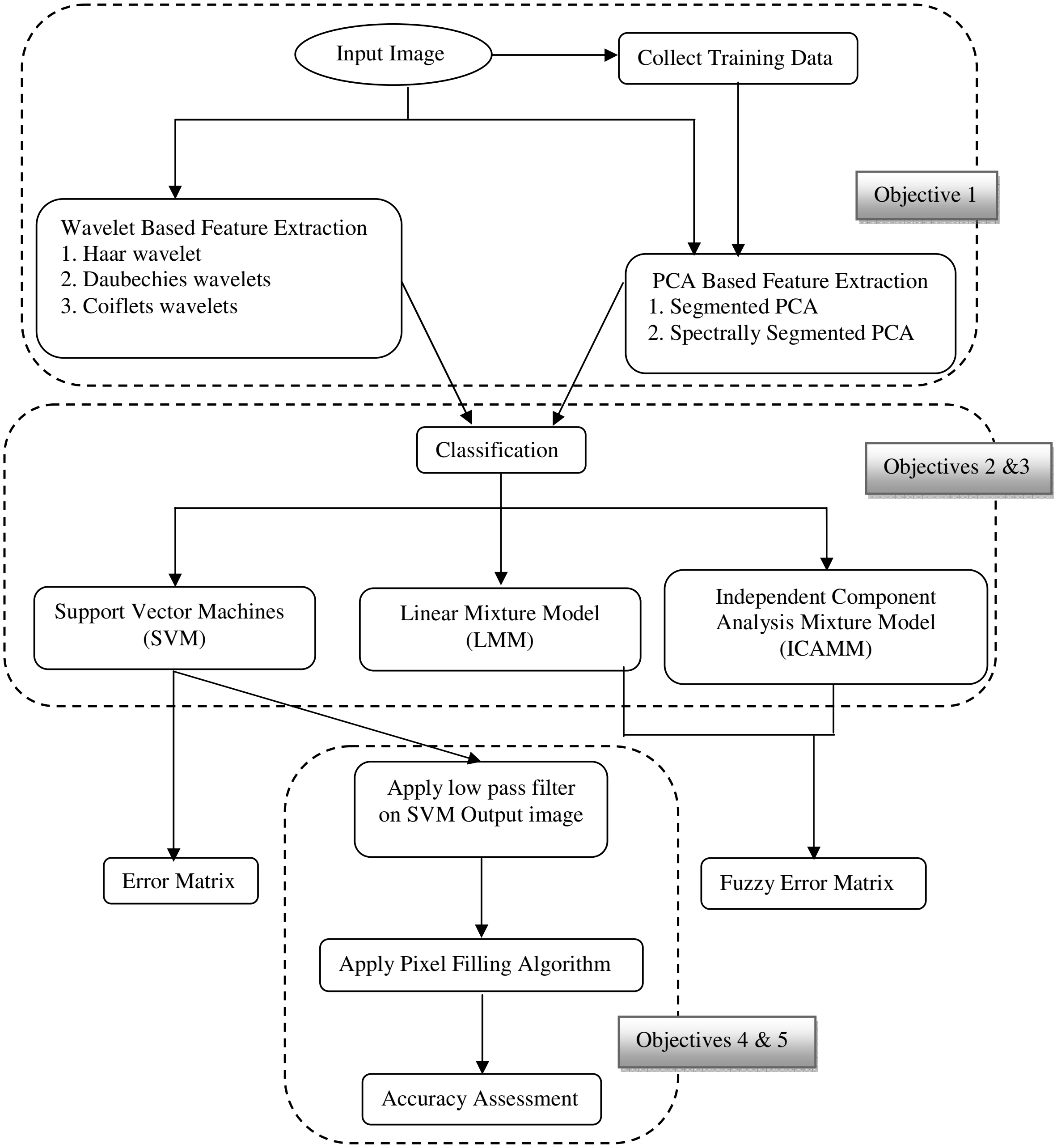

Input Image
Collect Training Data
Objective 1
Wavelet Based Feature Extraction
1. Haar wavelet
2. Daubechies wavelets
3. Coiflets wavelets
PCA Based Feature Extraction
1. Segmented PCA
2. Spectrally Segmented PCA
Classification
Objectives 2 &3
Support Vector Machines (SVM)
Linear Mixture Model (LMM)
Independent Component Analysis Mixture Model (ICAMM)
Apply low pass filter on SVM Output image
Error Matrix
Fuzzy Error Matrix
Apply Pixel Filling Algorithm
Objectives 4 & 5
Accuracy Assessment

Figure 3.7: The overall methodology adopted in this research.

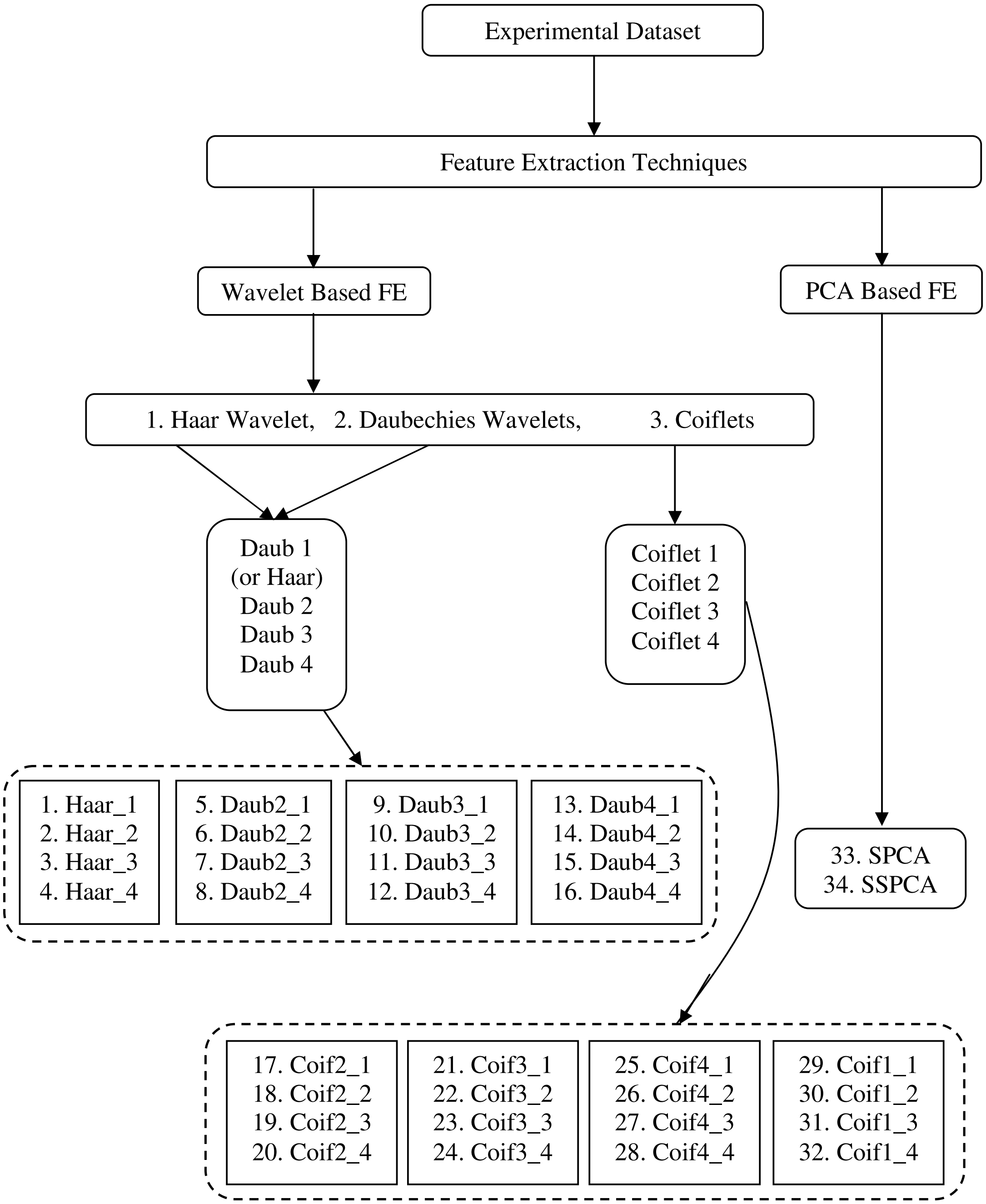


Figure 3.8: Flowchart for the feature extraction.

For information extraction, three classification algorithms have been performed on the selected datasets. In the feature extraction part, the per pixel classification using SVM has been performed to all the reduced datasets. Now, only for the reduced datasets whose classification accuracies are acceptable, have been classified by using sub pixel classification algorithms. Then the accuracy assessment of sub-pixel classification algorithms have been performed by fuzzy error matrix (FERM) based measures. One of the main requirements of FERM is the soft reference data (i.e., proportion of classes on the reference data). Due to non-availability of soft reference data from any source, the best classified SVM output has been taken as soft reference data. For example, if there are 6 classes in a dataset and a pixel has been classified as class 3 by SVM, then this information has been used as soft reference data represented as a vector of length 6 and denoted as (0 0 1 0 0 0). A flowchart for this part of methodology has been given in Figure 3.9.

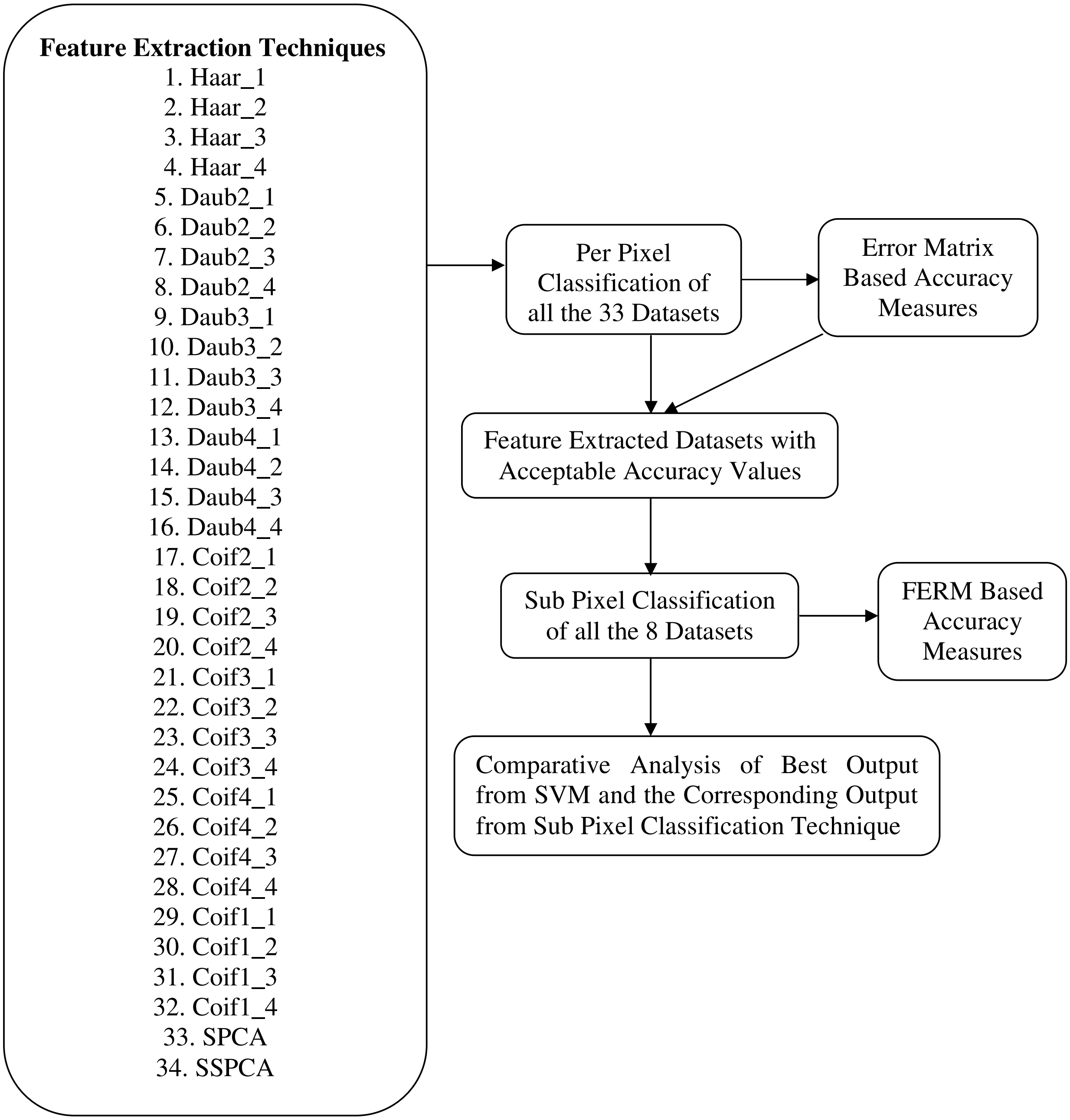

Feature Extraction Techniques
1. Haar_1
2. Haar_2
3. Haar_3
4. Haar_4
5. Daub2_1
6. Daub2_2
7. Daub2_3
8. Daub2_4
9. Daub3_1
10. Daub3_2
11. Daub3_3
12. Daub3_4
13. Daub4_1
14. Daub4_2
15. Daub4_3
16. Daub4_4
17. Coif2_1
18. Coif2_2
19. Coif2_3
20. Coif2_4
21. Coif3_1
22. Coif3_2
23. Coif3_3
24. Coif3_4
25. Coif4_1
26. Coif4_2
27. Coif4_3
28. Coif4_4
29. Coif1_1
30. Coif1_2
31. Coif1_3
32. Coif1_4
33. SPCA
34. SSPCA
Per Pixel Classification of all the 33 Datasets
Error Matrix Based Accuracy Measures
Feature Extracted Datasets with Acceptable Accuracy Values
Sub Pixel Classification of all the 8 Datasets
FERM Based Accuracy Measures
Comparative Analysis of Best Output from SVM and the Corresponding Output from Sub Pixel Classification Technique

Figure 3.9: Flowchart for classification of hyperspectral data.

Once the hyperspectral data has been classified, then a super resolution mapping has been performed. The output from sub pixel classification is taken as input to the super resolution mapping using pixel filling algorithm. For this purpose, the sub-pixel outputs generated in following two ways have been considered.

(i) The sub-pixel output having the highest classification accuracy, as obtained in the classification stage of information extraction.

(ii) Degrading the per pixel classification produced from SVM at a pre-defined zoom factor (3, 5, 7, etc.) and finding out the proportion of classes within each pixel of the degraded image.

The methodology followed in obtaining a super resolution map has been given as a flowchart in Figure 3.10.

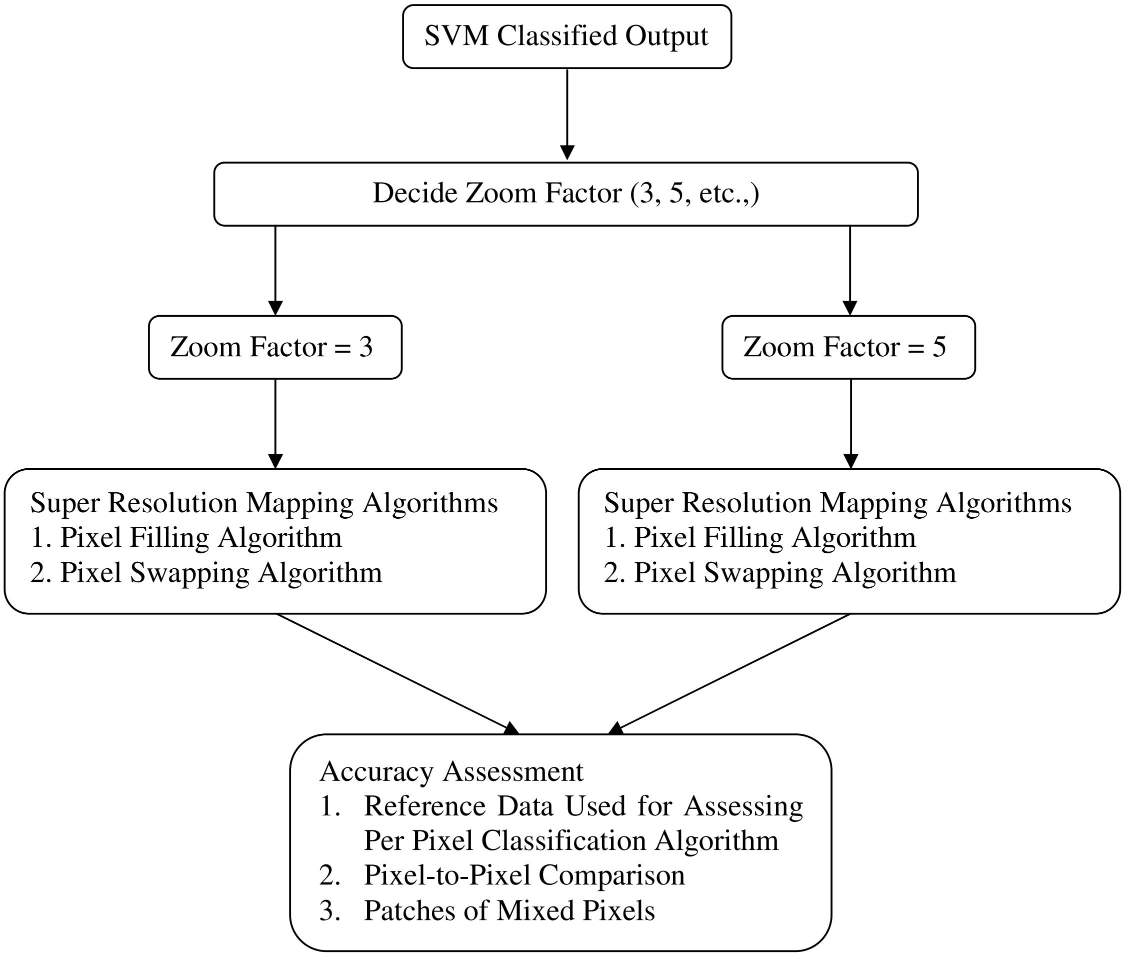

SVM Classified Output
Decide Zoom Factor (3, 5, etc.,)
Zoom Factor = 3
Zoom Factor = 5
Super Resolution Mapping Algorithms
1. Pixel Filling Algorithm
2. Pixel Swapping Algorithm
Super Resolution Mapping Algorithms
1. Pixel Filling Algorithm
2. Pixel Swapping Algorithm
Accuracy Assessment
1. Reference Data Used for Assessing Per Pixel Classification Algorithm
2. Pixel-to-Pixel Comparison
3. Patches of Mixed Pixels

Figure 3.10: Flowchart for super resolution mapping techniques

The super-resolution outputs have been assessed in three different ways,

(i) conventional error matrix construction based on some testing samples taken for validation of per pixel classification algorithm,

(ii) conventional error matrix construction based on pixel-to-pixel comparison of the whole classified image,

(iii)conventional error matrix construction for patches of mixed pixels (mixed by 2 classes or 3 classes or 4 classes) respectively.

The reference data taken for classification have been given in Figure 3.11 and Figure 3.12. In Figure 3.11, since it is a thematic map, it is clearly visible that which training pixels are pure, which are sharing common boundaries with its neighbours and which pixels falls at the center of the class.

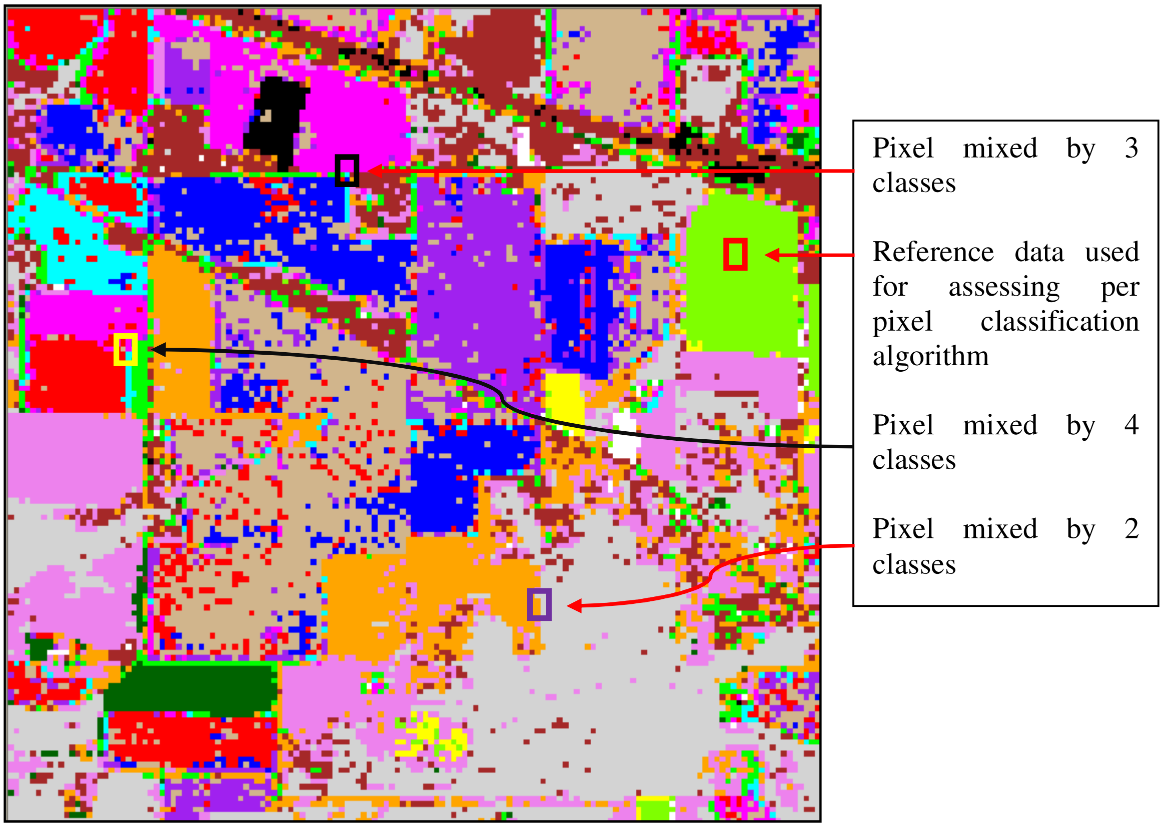
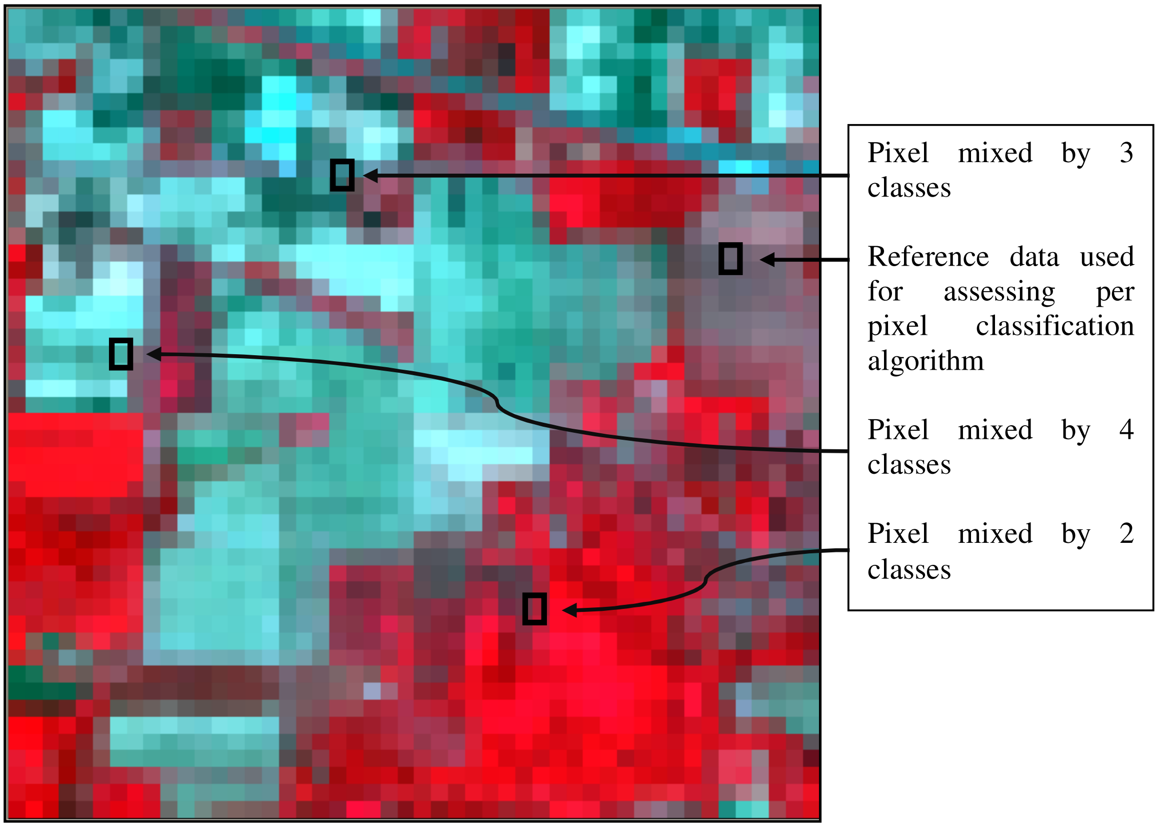


Figure 3.11: SVM classified pseudo image of dataset II along with reference data described in the adjoining box.



Pixel mixed by 3 classes

Reference data used for assessing per pixel classification algorithm

Pixel mixed by 4 classes

Pixel mixed by 2 classes

Figure 3.12: FCC of dataset II along with reference data described in the adjoining box.

To execute the algorithms and for applying them on hyperspectral images, coding have been done in MATLAB. For accuracy assessment of hyperspectral classification algorithms, VC++ has been used while for displaying images both MATLAB and ERDAS have been used.

3.5 Summary

In this chapter, the experimental datasets and the different components of methodology to achieve the research objectives were described. In the following chapters, the results from feature extraction, classification and super-resolution mapping will be analysed and discussed.

Chapter 4

Feature Extraction of Hyperspectral Data: Mathematical Background, Methodology and Results

4.1 Introduction

The advantage of hyperspectral data is that it collects information throughout the visible and near infrared (VNIR) region in a contiguous manner. Hyperspectral data have found their usage in extracting a number of information such as object and target/anomaly detection, mineral identification, segmentation and classification. Although, more useful information from hyperspectral data than multispectral data may be obtained, there are three major issues at various levels; storage of hyperspectral data redundancy of the data and large training sample requirement. The first issue is that the hyperspectral data is of huge volume and high dimension. There is thus always a challenge to process such a massive data. The second issue in the hyperspectral data is that the data have a tendency that the neighbouring bands are highly correlated. So taking all the bands for processing is not advisable. The third major issue of hyperspectral data processing is that in some applications like classification by means of statistical methods, large amount of training samples are required for higher accuracy, which is termed as curse of dimensionality. Also more training samples leads to poor classification accuracy, which is termed as the Hughes phenomenon. But often collecting the required number of training samples from field is time and cost intensive. Therefore, the hyperspectral data needs to be reduced without loss of information or not much loss of information. A typical spectral signature of vegetation has been given in Figure 4.1, collected in the region visible to near infrared (VNIR). It contains many peaks and valleys and those are very important in differentiating from one kind of species to another kind. Thus, when the spectral bands are narrow and information acquisition is contiguous, it increases the number of bands and leads to high dimensionality. Moreover to show the tiny difference in the spectral signature between bands, it requires high radiometric resolution which further increases the capacity requirement of the hyperspectral data. Hence it requires lots of storage space to keep the collected data.

Once the hyperspectral data is stored, their processing is also a challenging task. For a given algorithm, the data may be input as pixel vectors, row vector along spatial coordinate, band wise, etc. Due to huge volume and high dimension dataset, the processing time will be exponentially large. Often, carrying huge data in processing stage leads to low performance of the processing algorithms.

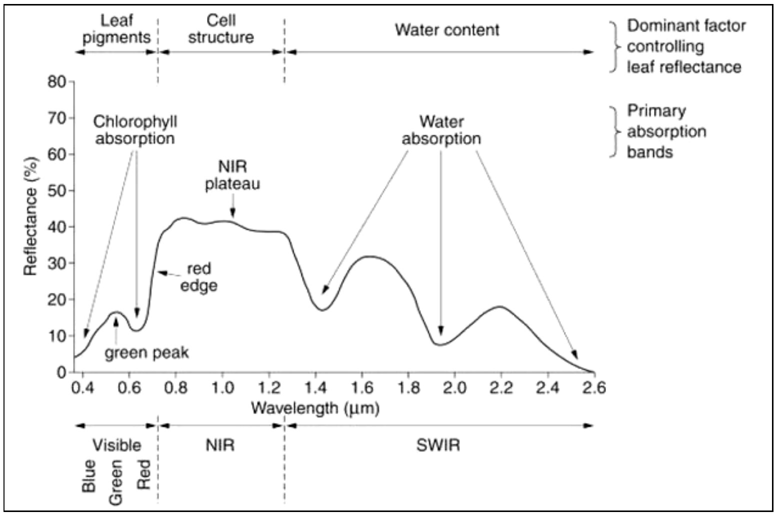


Figure 4.1: A typical spectral reflectance curve of vegetation

Second, the spectral signature given in Figure 4.1 is continuous and tells that the nearby bands values are close enough to each other which lead to high correlation between the neighbouring bands in the hyperspectral dataset. Therefore, some of the wavebands may provide redundant information. This redundancy may also be eliminated without any effective loss of information content within the hyperspectral dataset.

The third issue in hyperspectral data comes in case of supervised classification, the number of training samples required is proportional to the number of bands involved in classification. As the spectral dimension increases, requirement for the number of training pixels also increases, which at times is difficult to fulfill. Thus, this may also lead to the curse of dimensionality or Hughes phenomenon (Hughes 1968). For any reason, the hyperspectral data needs to be reduced.

4.2 Feature Reduction Techniques

Reducing the high dimension data set into lower dimension without sacrificing significant information of interest is thus one of the significant steps in hyperspectral image processing. In hyperspectral image classification, effective features are those which are most capable of preserving class separability (Hsu 2000). Here feature represents bands. After feature reduction, a subset of features is obtained. But, preserving the useful information is the key issue in feature reduction, which demands for development of appropriate algorithms. The reduction of hyperspectral data means the reduction in the spectral dimension by keeping the spatial dimension unaltered. Feature reduction of hyperspectral data may be done by two ways; feature selection and feature extraction.

Selecting a subset of features/bands from the original data, where the classes are highly separable is known as feature selection. Feature extraction allows transforming the original data into a new feature space, from which the prime features (i.e., reduced feature space) are derived for further processing to derive information.

4.2.1 Feature Selection

Feature selection has the advantage of preserving the relevant original information from the data (Martínez-Usó *et al.*, 2007). Some of the feature selection methods used in hyperspectral data reduction are sequential forward selection (SFS) (Jain and Zongker, 1997), sequential backward selection (SBS) (Pudil *et al.*, 1994), sequential forward floating selection (SFFS), (Pudil *et al.*, 1994) sequential backward floating selection (SBFS) (Somol *et al.*, 1999; Pudil *et al.*, 1994), distance measures (Bhattacharya distance, Mahalonobis distance, Jeffries–Matusita (JM) distance (Serpico and Moser, 2007) , etc.,), divergence analysis, steepest ascent (Bruzzone and Serpico, 2000), fast constrained search (Bruzzone and Serpico, 2000), feature similarity measure, graph searching algorithms, neural networks, genetic methods, tabu search meta-heuristics (Korycinski *et al.*, 2003a; Korycinski *et al.*, 2003b), spectral distance metrics (Keshava, 2004), parametric feature weighting, spatial autocorrelation, band ratioing, wavelet based feature selection, mutual information based feature selection (Guo *et al.*, 2006), orthogonal subspace projection (Du and Yang, 2008) etc. Standard feature selection methods based on class separability measures such as divergence, JM distance and transformed divergence may not be used effectively as a result of several considerations (Chang 2007).

4.2.2 Feature Extraction

Feature extraction may require the whole (or most) of the original data representation to extract the new features, forcing to always obtain and deal with the whole initial representation of the data. In addition, since the data are transformed, some critical information may have to be compromised and distorted (Martínez-Usó *et al.*, 2007). But feature extraction methods are more effective than feature selection methods (Serpico and Moser, 2007).

Some of the standard feature extraction techniques are principal component analysis (PCA) (Richards, 1993; Jensen and Walts, 1997; Schowengerdt, 1997) and minimum noise fraction (MNF) (Belluco *et al.*, 2006, Bakos and Gamba, 2011). Others feature extraction methods are segmented PCA (SPCA) (Jia and Richards, 1999), orthogonal subspace projection (OSP) (Harsanyi and Chang, 1994), singular value decomposition subset selection, discriminant analysis feature extraction (DAFE) (Lee and Landgrebe, 1993), prototype space feature extraction (PSFE), penalized discriminant analysis (Hastie *et al.*, 1995; Yu *et al.*, 1999), kernel Fisher discriminant (Jin *et al.*, 2010), Fisher's canonical transform projection pursuit, decision boundary feature extraction (DBFE), independent component analysis (ICA) (Wang and Chang, 2006,), etc.

As can be seen, there are many feature extraction techniques that may be used for hyperspectral data, the description of each of them cannot be provided here. In this thesis, the study has been focused onto two feature extraction techniques based on markedly different mathematical background. The first one is wavelet based feature extraction which is a multi-level decomposition technique. On the other hand, statistical transformation based feature extraction technique has also been executed to extract important information from the hyperspectral data. Description of these two techniques has been given in the following section.

4.3 Wavelet Based Feature Extraction Techniques

Wavelets view the hyperspectral data in a different way by taking weighted average of the neighbouring band values or digital numbers of each pixel vector. The hyperspectral data are transformed from the original feature space into a scale space plane using the wavelet transform to extract the significant spectral features. The wavelet transform can focus on localized signal structures with a zooming procedure. The decomposition of a pixel vector is achieved by applying two filters act on the signal; averaging filter to produce a vector named as trend and the other vector named fluctuations, to give details coefficients (Walker, 2008).

Wavelets have some mathematical properties in addition to multi-resolution analysis, orthogonality, compact support, etc. and are given as follows:

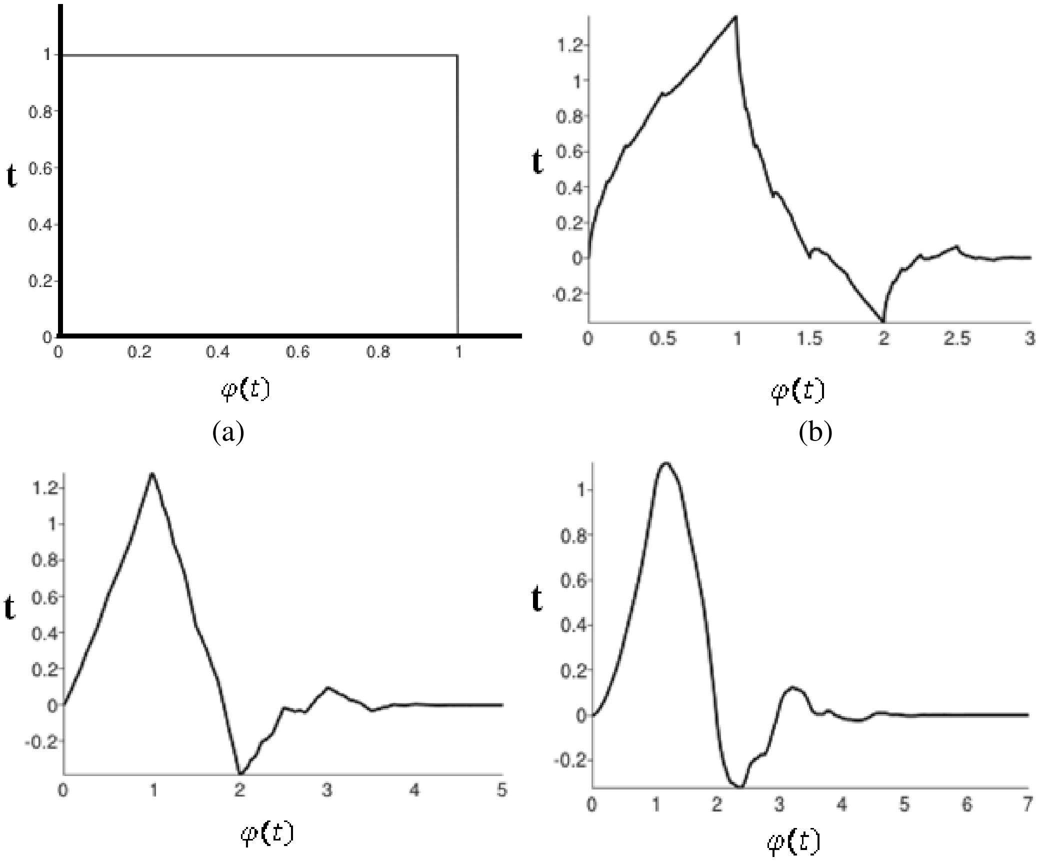
a) Wavelets are important tools in mathematics which are useful for decomposition of given signals into various frequencies.

b) Transformation in one pixel vector does not affect the other pixel vectors.

c) Wavelets are very simple to handle.

d) The wavelet series are usually orthogonal to each other.

e) Reconstruction is possible.



(a)

(b)

(c)

(d)

Figure 4.2(a) – (d): The first 4 sub-classes of Daubechies scaling functions, (a) Haar or Daubechies 1, (b) Daubechies 2, (c) Daubechies 3, (d) Daubechies 4. Here $\varphi(t)$ is the scaling function depends on time **t**.

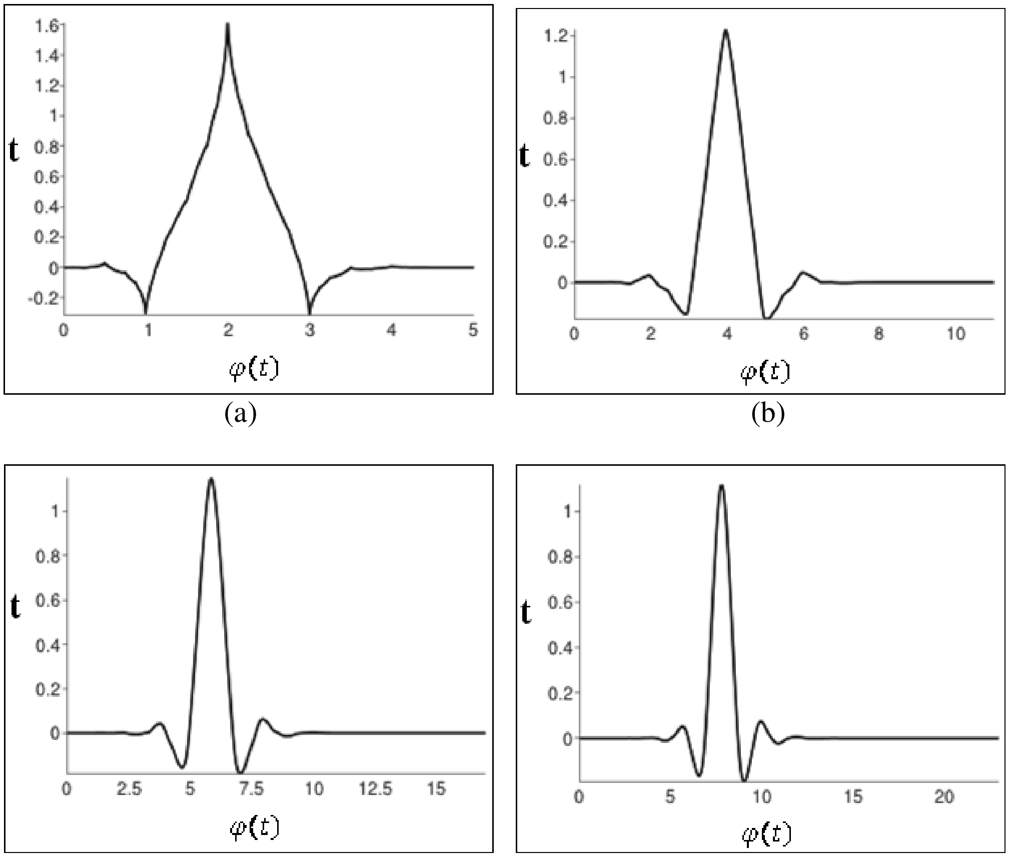
The 1-D mother wavelet function Ψ (in continuous form) may be defined as,

$$\Psi_{s,\tau}(t) = \frac{1}{\sqrt{s}}\,\Psi\left(\frac{t-\tau}{s}\right), \qquad (4.1)$$

where $s, \tau \in \mathbf{R}$ such that $s \neq 0$ and satisfies the admissibility condition,

$$\int_{-\infty}^{\infty} \Psi(t)dt = 0 \qquad (4.2)$$

where s and τ are the dilation (scale) and translation (position) parameters respectively.



(a) (b)

(c) (d)

Figure 4.3(a) – (d): The first 4 sub-classes of Coiflets scaling functions, (a) Coiflets 1, (b) Coiflets 2, (c) Coiflets 3, (d) Coiflets 4. Here $\varphi(t)$ is the scaling function depends on time **t**.

This may happen only when the function values are zero at most of the places and the other values 'wave' around zero. The set of values at which the function is non-zero is called the support of the function. By imposing additional mathematical restrictions, a number of wavelet families may be created. Already, the formulation of a few wavelets, namely, Haar, Daubechies, Coiflets, Symlets, BiorSplines, ReverseBior, Gaussian wavelets etc. exists in the literature. Also, within each family of wavelets, there is a range of sub-classes of different orders which are different from each other in respect of their scaling and the wavelet functions (Blackburn, 2007). Figures 4.2 and 4.3 provide a selected number of scaling functions for the first four sub-classes of Daubechies and Coiflet wavelets respectively. As each of the families behaves differently with the hyperspectral data, because of the scaling function for all these wavelets are different, the mathematical background of these wavelets could be helpful. Also, the decomposition level is one of the most important parts in applying the wavelet transform, in which the number of features extracted decreases when level of decomposition increases.

4.3.1 Mathematical Details of Haar Wavelet Transform

The Haar wavelet or Daubechies 1 wavelet is the simplest of all the wavelets. The Haar Mother wavelet is denoted by $\mathit{\Psi}(t)$and is defined as,

$$\mathit{\Psi}(t) = \begin{cases} 1 & 0 \le t < 1/2, \\ -1 & \frac{1}{2} \le t < 1, \\ 0 & otherwise \end{cases} \tag{4.3}$$

The scaling function of Haar wavelet is denoted by $\varphi(t)$ (Figure 4.2) as,

$$\varphi(t) = \begin{cases} 1 & 0 \le t < 1, \\ 0 & otherwise \end{cases} \tag{4.4}$$

The scaling numbers, α_1 and α_2, may be generated from equation (4.4). The curve given in Figure 4.2(a) and obtained from equation (4.3) may be divided into two parts. In both the parts, the values $\varphi(t)$ are 1. Normalisation of these results into $\frac{1}{\sqrt{2}}$. That is $\alpha_1 = \frac{1}{\sqrt{2}}$ and $\alpha_2 = \frac{1}{\sqrt{2}}$. And now these scaling numbers are used to create vectors called as scaling signals, defined as,

$$\boldsymbol{V_1^1} = (\alpha_1, \alpha_2, 0, 0, 0, \ldots 0) = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, \ldots, 0\right)$$

$$\boldsymbol{V_2^1} = (0, 0, \alpha_1, \alpha_2, 0, 0, \ldots 0) = \left(0, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, \ldots, 0\right)$$

$$\boldsymbol{V_3^1} = \left(0, 0, 0, 0, \alpha_1, \alpha_{2,} 0, 0, \ldots 0\right) = \left(0, 0, 0, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, \ldots, 0\right)$$

.

.

.

$$\boldsymbol{V_{N/2}^1} = (0, 0, 0, \ldots 0, \alpha_1, \alpha_2) = \left(0, 0, \ldots, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right) \tag{4.5}$$

Using these Haar scaling signals, the values $a_1, a_2, \dots a_{\frac{N}{2}}$ for the first trend are expressed as scalar products: $a_m = \mathbf{f}.\boldsymbol{V}_m^1$, for m = 1, 2, … , N/2, where f is the signal to be decomposed. Then all the a_i values are collected to form the reduced signal of length N/2 and is given by,

$$\mathbf{a_1} = \left(a_1, a_2, \dots a_{\frac{N}{2}}\right)$$

Hence, the pixel vector $\mathbf{f}$ which is of length N has been decomposed into two vectors of length N/2 each, which is mentioned as averaging part and detail part (detail part is not used in further processing and hence derivation is also not given). Similarly, each pixel vector of the hyperspectral data has been also decomposed and only the approximation part vectors are collected. Hence, a hyperspectral data of dimension L x M x N is now reduced to L x M x N/2. Thus, the wavelet transform when applied on pixel vector reduces the spectral dimension and not spatial dimension.

The same concept may be extended to every other levels of decompositions also. After performing a 1-level Haar scaling decomposition of a signal **f**, a first approximation part vector or first trend $\mathbf{a_1}$ is obtained. Again apply 1-level decomposition on $\mathbf{a_1}$ to compute a second trend $\mathbf{a_2}$.

The spectral curves of hyperspectral pixel vectors signal consists of peaks and valleys such that the difference between the number of peaks and number of valleys is at most 1. But the sharpness of peaks and valleys considered to be important. In general, for all wavelet transforms, when the number of decomposition level increases, the length of the signal decreases since every time only the approximation part vector is taken which is half in length that of the pixel vector.

4.3.2 Mathematical Details of Daubechies Wavelet Transform

Haar transform is the first sub-class of Daubechies transforms which differs in the scaling functions and wavelets. The other Daubechies sub-classes also follow the same procedure as that of Haar. The most obvious difference between them is the length of the support, which means the number of non-zero terms of their scaling signals. The explanation on formation of scaling signals for all sub-classes of Daubechies is similar to that of Haar transformation.

However, here a brief description for Daubechies 2 scaling function is given. The graph of the scaling function of Daubechies 4 is given in Figure 4.2(d). The scaling numbers, denoted by $\alpha_1, \alpha_2, \alpha_3$ and α_4, are given in equation (4.6).

$$\alpha_1 = \frac{1+\sqrt{3}}{4\sqrt{2}} = 0.48, \quad \alpha_2 = \frac{3+\sqrt{3}}{4\sqrt{2}} = 0.84, \quad \alpha_3 = \frac{3-\sqrt{3}}{4\sqrt{2}} = 0.22, \quad \alpha_4 = \frac{1-\sqrt{3}}{4\sqrt{2}} = 0.13 \quad \textbf{(4.6)}$$

The derivation of values in equation (4.6) is beyond the scope of this thesis. Using these 4 scaling numbers, the 1-level scaling signals can be obtained as:

$$\boldsymbol{V_1^1} = (\alpha_1, \alpha_2, \alpha_3, \alpha_4, 0, 0, 0, \dots 0) = (0.48, 0.84, 0.22, -0.13, 0, 0, 0, \dots 0)$$

$$\boldsymbol{V_2^1} = (0, 0, \alpha_1, \alpha_2, \alpha_3, \alpha_4, 0, 0, \dots 0) = (0, 0, 0.48, 0.84, 0.22, -0.13, 0, 0, \dots 0)$$

$$\boldsymbol{V_3^1} = \left(0, 0, 0, 0, \alpha_1, \alpha_{2,}\alpha_3, \alpha_4, 0, 0, \dots 0\right) = (0, 0, 0, 0, 0.48, 0.84, 0.22, -0.13, 0, 0, \dots 0)$$

$$\boldsymbol{V_{N/2-1}^1} = (0, 0, \dots, 0, \alpha_1, \alpha_2, \alpha_3, \alpha_4) = (0, 0, \dots, 0, 0.48, 0.84, 0.22, -0.13)$$

$$\vdots$$

$$\boldsymbol{V_{N/2}^1} = (\alpha_3, \alpha_4, 0, 0, 0, \dots \alpha_1, \alpha_2) = (0.22, -0.13, 0, 0, 0, \dots 0.48, 0.84) \qquad (4.7)$$

These scaling signals hold few properties:

$$\text{(i) } \alpha_1^2 + \alpha_2^2 + \alpha_3^2 + \alpha_4^2 = 1$$

$$\text{(ii) } \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = \sqrt{2} \qquad (4.8)$$

From Figures 4.2(a) to 4.2(d), it can be seen that the Daubechies scaling functions are not symmetric. Therefore, the corresponding scaling numbers (i.e., weights) are different and may also contain few negative numbers. Here also, for the signal **f**, the first level approximation part vector $\mathbf{a_1}$ and the second level approximation part vector $\mathbf{a_2}$ may be computed as explained for Haar transform in sub-section 2.1. Here also for first level decomposition, the length of first level approximation part vector $\mathbf{a_1}$ is half that of the original signal and the second level approximation part vector $\mathbf{a_2}$ is 1/4th of the original signal.

4.3.3 Mathematical Details of Coiflets Wavelet Transform

The Daubechies scaling functions are neither symmetric nor asymmetric while the Coiflets family is asymmetric (near to symmetry). Here also, there are many sub-classes, which are identified by the respective scaling functions. The graphs of first four sub-classes of Coiflets are given in Figures 4.3(a) to 4.3(d). Similar to earlier wavelet transforms, the scaling signals are formed from the scaling numbers $\alpha_1, \alpha_2, \alpha_3$, etc. Appropriate values of $\alpha_1, \alpha_2, \alpha_3$, etc., are taken from Table 4.1.

Table 4.1: The scaling coefficients of the first four sub-classes of Daubechies and Coiflets wavelets. (Walker, 2008)

Haar	$\alpha_1 = 0.7071067812$ $\alpha_2 = 0.7071067812$		Coif 1	$\alpha_4 = 0.3848648469$ $\alpha_5 = -0.0727326195$ $\alpha_6 = -0.0156557281$	$\alpha_1 = -0.0727326195$ $\alpha_2 = 0.3378976625$ $\alpha_3 = 0.8525720202$
Daub 2	$\alpha_1 = 0.4829629131$ $\alpha_2 = 0.8365163037$ $\alpha_3 = 0.2241438680$ $\alpha_4 = -0.1294095226$		Coif 2	$\alpha_7 = -0.0764885991$ $\alpha_8 = -0.0594344186$ $\alpha_9 = 0.0236801719$ $\alpha_{10} = 0.0056114348$ $\alpha_{11} = -0.0018232089$ $\alpha_{12} = -0.0007205494$	$\alpha_1 = 0.0163873365$ $\alpha_2 = -0.0414649368$ $\alpha_3 = -0.0673725547$ $\alpha_4 = 0.3861100668$ $\alpha_5 = 0.8127236354$ $\alpha_6 = 0.4170051844$
Daub 3	$\alpha_1 = 0.3326705530$ $\alpha_2 = 0.8068915093$ $\alpha_3 = 0.4598775021$ $\alpha_4 = -0.1350110200$ $\alpha_5 = -0.0854412739$ $\alpha_6 = 0.0352262919$		Coif 3	$\alpha_{10} = -0.0823019271$ $\alpha_{11} = 0.0345550276$ $\alpha_{12} = 0.0158805449$ $\alpha_{13} = -0.0090079761$ $\alpha_{14} = -0.0025745177$ $\alpha_{15} = 0.0011175188$ $\alpha_{16} = 0.0004662170$ $\alpha_{17} = -0.0000709833$ $\alpha_{18} = -0.0000345998$	$\alpha_1 = -0.0037935129$ $\alpha_2 = 0.0077825964$ $\alpha_3 = 0.0234526961$ $\alpha_4 = -0.0657719113$ $\alpha_5 = -0.0611233900$ $\alpha_6 = 0.4051769024$ $\alpha_7 = 0.7937772226$ $\alpha_8 = 0.4284834764$ $\alpha_9 = -0.0717998216$
Daub 4	$\alpha_1 = 0.2303778133$ $\alpha_2 = 0.7148465706$ $\alpha_3 = 0.6308807679$ $\alpha_4 = -0.0279837694$ $\alpha_5 = -0.1870348117$ $\alpha_6 = 0.0308413818$ $\alpha_7 = 0.0328830117$ $\alpha_8 = -0.0105974018$		Coif 4	$\alpha_{13} = 0.0393344271$ $\alpha_{14} = 0.0250822618$ $\alpha_{15} = -0.0152117315$ $\alpha_{16} = -0.0056582867$ $\alpha_{17} = 0.0037514362$ $\alpha_{18} = 0.0012665619$ $\alpha_{19} = -0.0005890208$ $\alpha_{20} = -0.0002599746$ $\alpha_{21} = 0.0000623390$ $\alpha_{22} = 0.0000312299$ $\alpha_{23} = -0.0000032597$ $\alpha_{24} = -0.0000017850$	$\alpha_1 = 0.0008923137$ $\alpha_2 = -0.0016294920$ $\alpha_3 = -0.0073461663$ $\alpha_4 = 0.0160689440$ $\alpha_5 = 0.0266823002$ $\alpha_6 = -0.0812666997$ $\alpha_7 = -0.0560773133$ $\alpha_8 = 0.4153084070$ $\alpha_9 = 0.7822389309$ $\alpha_{10} = 0.4343860565$ $\alpha_{11} = -0.0666274743$ $\alpha_{12} = -0.0962204420$

By using the scaling numbers given in Table 4.1 for various wavelet transforms and sub-classes, the 1-D pixel vector **f** is decomposed to produce various levels of trends.

4.3.4 Reconstruction of Wavelet Transformation

Just now we have seen how a hyperspectral pixel vector is decomposed into averaging (trend) and detail (fluctuation) coefficients by wavelet techniques. From the decomposed signals, it is possible to retrieve the original signals. The method is called as reconstruction of wavelets or inverse wavelet transform. In decomposition of signals, filtering and down-sampling happens while in reconstruction filtering and up-sampling will happen.

The reconstruction of wavelets can be done in many ways. One such way is to assume the detail coefficient value obtained at the final level to be zero and to find the averaging coefficients from lower level to higher level to reach the original signal $\boldsymbol{f}$. Another way is to assume all the detail coefficients to be zero and to find the averaging coefficients from lower level to higher level to reach the original signal $\boldsymbol{f}$.

Since wavelets have the property of multi-resolution analysis, the spectral decomposition of each pixel vector has been performed and studies have been made on various decomposition levels. Here every pixel vector has been taken and the wavelet transformation has been applied on them individually. Another transformation based on statistical method has also been performed. Here the whole dataset has been transformed into another dataset of same dimension. The mathematical concept of wavelets and PCA are entirely different. The transformation is pixel wise in the former case and in the latter case it is collective information transformation. Therefore, the study on these two techniques for feature extraction of hyperspectral data may be useful. The basic mathematical structure of PCA based feature extraction techniques has been given in the next section.

4.4 PCA Based Feature Extraction Techniques

The PCA is a multivariate statistical tool which generally projects the data into orthogonal space. PCA, an orthogonal subspace projection is performed on the hyperspectral images and produces a new sequence of uncorrelated bands or components. Usually the first few components contain the most of information, and the later components that tend to show little variance which are usually ignored. Therefore, the essential dimensionality of the classification space will be reduced. Although, this method can effectively provide high classification accuracy, it is sensitive to noise and has to be performed with the whole dataset (Schowengerdt, 1997).

In PCA, the second order statistics (covariance) are considered for transforming the original data. Let x represents the vector of a pixel's digital number value in an image of N bands. The transformation is defined in equation (4.9) as,

$$Y = A^t x,, \qquad (4.9)$$

where, A is the normalized matrix of eigen vectors of the covariance matrix. The image covariance matrix, C, is an N-by-N matrix and can be constructed by finding covariance between band to band and by keeping each band as a pixel vector. Now, when x_i represents a pixel vector for i = 1, 2, …, K and let m be the mean vector, then the mean and covariance matrix can be described as,

$$m = E\{x\} = \frac{1}{K}\sum_{i=1}^{K} x_i \quad (4.10)$$

$$C = E\{(x_i - m)(x_i - m)^t\} = \frac{1}{K-1}\sum_{i=1}^{K}(x_i - m)(x_i - m)^t \quad (4.11)$$

PCA may be regarded as an optimal method for data compression but it is not necessarily an optimal method for feature extraction. Particularly, in case of applications such as image classification and target/anomaly detection, PCA is inept in extracting discriminating feature. Also, as PCA operates on global statistics it may overlook local variances that are helpful in detection problems. To improve PCA for better feature extraction, Jia and Richards (1999) developed a segmented PCA to extract best principal components from hyperspectral data.

4.4.1 Segmented Principal Component Analysis (SPCA)

In segmented PCA, the spectral bands are segmented by taking the correlation between the neighbouring bands. Since in hyperspectral data, correlation between neighbouring bands is higher than bands far away, the high correlations bands appear in white coloured blocks, as shown in Figure 4.4.

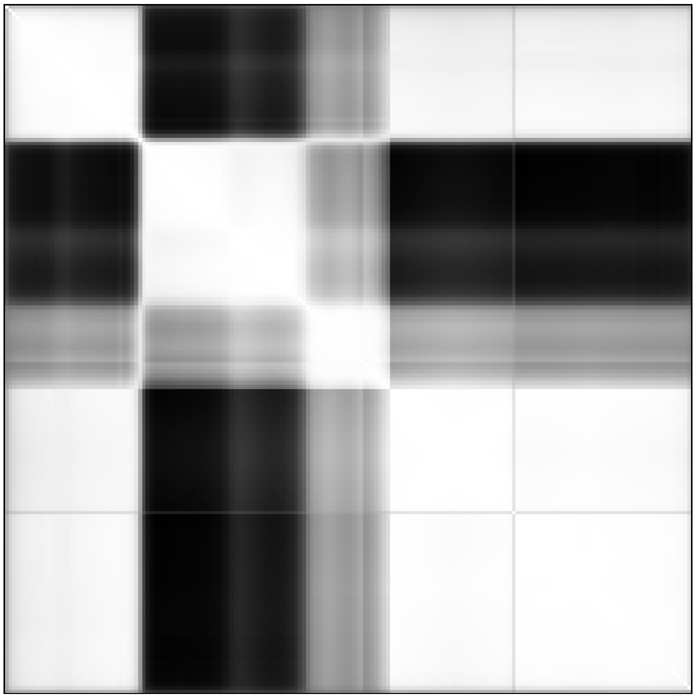


Figure 4.4: Image of a typical correlation matrix of dataset II (AVIRIS Indiana Pine with 182 bands which has been used in this research)

Once the image of correlation matrix has been obtained, an edge detection technique may be performed on the correlation matrix to find the exact boundary between the blocks. Let the entire image be divided into k number of subgroups, 1, 2, …k, based upon the edges got from edge detection technique. Now each subgroup has b_1, b_2, …b_K number of bands, respectively. The PCA is then applied on each group. Here, the sum of variances or the eigen vectors is considered to be 100% and based upon the requirement of number of bands or threshold value from each subgroup the PCs are taken for further processing.

Since PCA has been applied onto less number of bands, the computational load may be reduced significantly. But on the other hand, in SPCA, the segmentation of spectral bands has been performed by the correlation matrix but if it is with respect to the available classes then the accuracy may improve. Hence, a segmentation of spectral bands has been performed based on the properties of spectral curves of the available classes, which has been explained in the subsequent sections.

4.4.2 Spectrally Segmented Principal Component Analysis (SSPCA)

One technique to divide the spectral group into various groups is by finding edges on a correlation matrix and there are few other techniques too. Bell and Baranoski (2004) developed a piecewise PCA (PPCA) algorithm to study field collected spectral datasets of vegetation. The algorithm has been developed based on the concepts of how plants interact with solar radiation in absorption, reflection and transmittance throughout the light spectral wavelength regions due to the structure and constituents of the leaf. The collected spectrum is divided into several groups of different wavelength regions and the PCA is then applied to each group independently. The experiments of Bell and Baranoski indicated that PCA worked effectively in visible to near-infrared regions while PPCA further improved the efficiency and accuracy–cost ratio in data compression and restoration of plant spectra.

Tsai *et al.*, (2007) proposed a spectrally segmented PCA based on known characteristics of plant leaves over different regions of wavelength. Here, the aim is to extract information about particular types of plant species. Several experiments have been conducted to define various spectral groups. A number of PCs in each segment have been taken in order to find the aptness of the technique.

Now, a detailed description on how feature extraction has been performed by the two techniques, wavelet based feature extraction technique and PCA based feature extraction technique.

4.5 Implementation of Feature Extraction Techniques

The two feature extraction techniques, wavelet based and PCA based have been implemented. The transformation by wavelets has been applied on hyperspectral vector while PCA takes groups of bands for feature extraction. Detailed explanation about the implementation of the two techniques has been given in the following sub-sections.

4.5.1 Implementation of Wavelet Based Feature Extraction Technique of Hyperspectral Data

In order to fulfill the objectives of this study, set of experiments have been planned. The general methodology has been given in Figure 4.5.

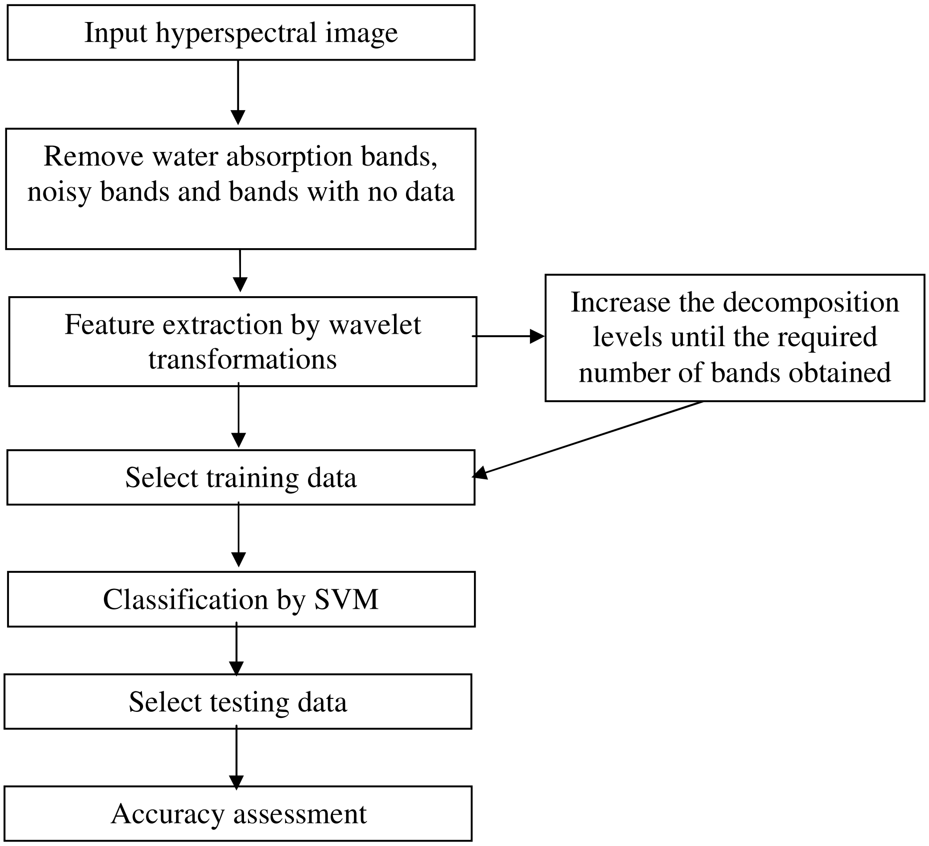

Input hyperspectral image
Remove water absorption bands, noisy bands and bands with no data
Feature extraction by wavelet transformations
Increase the decomposition levels until the required number of bands obtained
Select training data
Classification by SVM
Select testing data
Accuracy assessment

Figure 4.5: Flowchart of general methodology for feature reduction using wavelet transformation.

4.5.1.1 Decomposition of a Pixel Vector of Hyperspectral Data

Typically, the hyperspectral data is represented as a 3-dimension matrix with its rows and columns representing the spatial coordinates and the 3rd dimension for spectral details. Thus, the data pertaining to every pixel may be represented as a pixel vector, in which the length is same as that of the number of spectral bands or dimensions. In general, the wavelet transformation may be applied to 1-D signal (a pixel vector) or 2-D image (a matrix). In the context of hyperspectral data, the data reduction is not along its spatial dimension but along its spectral dimension. The wavelet transform is applied on each of the pixel vectors (here the pixel vectors are 1-D signals) of the hyperspectral data. In this study, the wavelet families from Haar, Daubechies and Coiflets have been used to extract features.

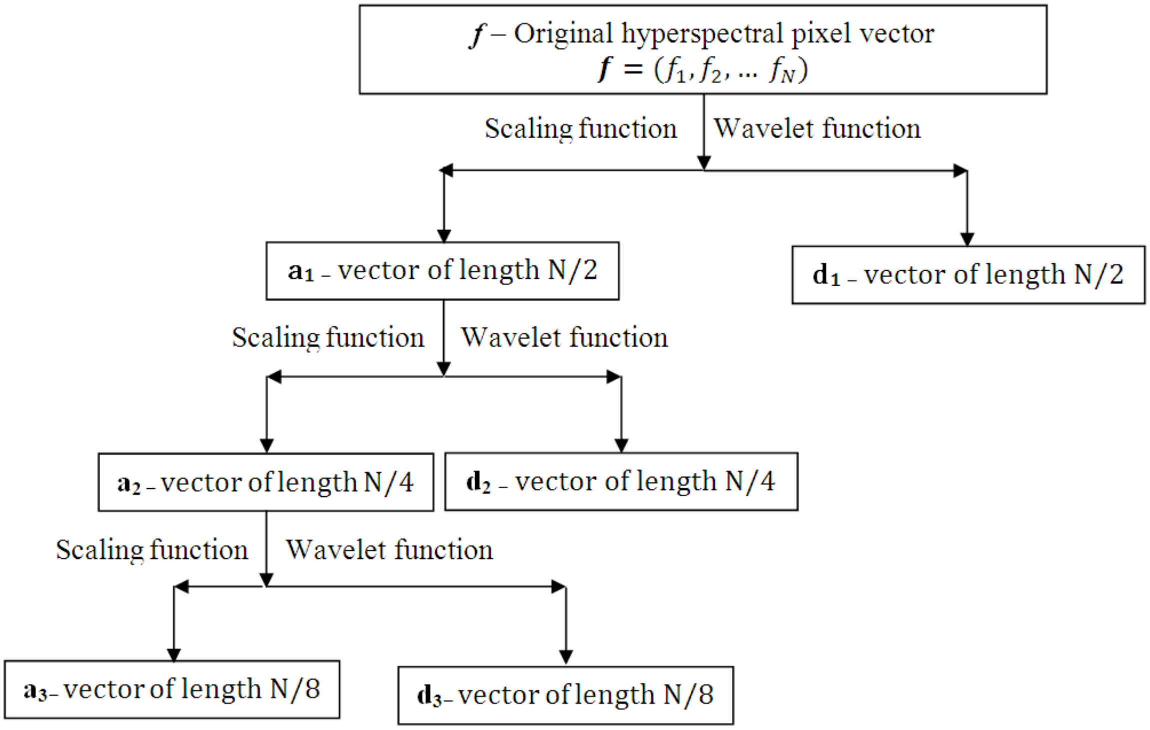

f – Original hyperspectral pixel vector
$f = (f_1, f_2, \dots f_N)$
Scaling function
Wavelet function
a_1 - vector of length N/2
d_1 - vector of length N/2
Scaling function
Wavelet function
a_2 - vector of length N/4
d_2 - vector of length N/4
Scaling function
Wavelet function
a_3- vector of length N/8
d_3- vector of length N/8

Figure 4.6: The process of decomposition of a hyperspectral signal when wavelet transformation is applied to 1-D signal vector.

The notations used in the Figure 4.6 are explained as follows:

f is the original pixel vector of the given hyperspectral data,

$\mathbf{a}_1$ and $\mathbf{a}_2$ are the first and second level scaling functions (scaling signals) respectively obtained by decomposing *f*,

$\mathbf{d}_1$ is the first level wavelet function such that $f = \mathbf{a}_1 + \mathbf{d}_1$,

$\mathbf{d_2}$ is the second level of wavelet function such that $\mathbf{a_1} = \mathbf{a_2} + \mathbf{d_2}$, i.e., $\boldsymbol{f} = \mathbf{a_2} + \mathbf{d_2} + \mathbf{d_1}$. Here, the binary operation '+' has been used to append two vectors.

Every wavelet family depends upon a 'mother wavelet'. Haar wavelet is the simplest among all the wavelets. The wavelet multi-resolution analysis is used to sub-sample the spectral domain by a factor 2 on each pixel vector of hyperspectral data considered as discrete signal. A discrete signal is an 1-D array of elements at discrete instants. Generally, a discrete signal is expressed as:

$$\boldsymbol{f} = (f_1, f_2, \dots f_N), \tag{4.12}$$

where N is a positive even integer, referred to as the length of $\boldsymbol{f}$. The values of $\boldsymbol{f}$ are the N real numbers $f_1, f_2, \dots f_N$. In hyperspectral data, $\boldsymbol{f}$ is a pixel vector, where the values f_i, i = 1 to N is the digital number of that pixel at i$^{\text{th}}$ spectral band and N is the total number of spectral bands of the hyperspectral data. Now, by applying the transformation on $\boldsymbol{f}$, it decomposes $\boldsymbol{f}$ into two parts. The first part is the scaling function and the second part is the wavelet function. The decomposition stages may be explained with the help of Figure 4.6.

Like all wavelet transforms, the Haar transform decomposes a discrete signal into two sub-signals of half its length. One sub-signal is a running average or approximation vector or trend while the other sub-signal is the running difference or detail or fluctuation. In general, the fluctuation is used to reconstruct the original signal with the help of trend. The trend is used for further processing of the signal. The first approximation vector $\boldsymbol{a_1} = (a_1, a_2, \dots a_{N/2})$, , for the signal $\boldsymbol{f}$ is computed by taking a running average in the following way.

Its first value, a_1, is computed by taking the average of the first pair of values $(f_1 + f_2)/2$; and then normalization by multiplying it by $\sqrt{2}$. Thus, the first value of approximation vector is given as:

$$a_1 = (f_1 + f_2)/\sqrt{2} \tag{4.13}$$

Similarly, its next value is computed by taking the average of next pair of values of $\boldsymbol{f}$: $(f_3 + f_4)/2$; and then multiplying it by $\sqrt{2}$. Hence, the second term in the sequence is given by:

$$a_2 = (f_3 + f_4)/\sqrt{2} \tag{4.14}$$

Continuing this way, all the values of approximation vector $\mathbf{a_1}$ are produced by taking averages of successive pairs of values of $\boldsymbol{f}$, and then multiplying these averages by $\sqrt{2}$. A formula for the values of $\mathbf{a_1}$ is given as,:

$$a_m = \frac{f_{2m-1} + f_{2m}}{\sqrt{2}} \text{, m = 1, 2, 3, …, N/2} \tag{4.14}$$

The other sub-signal is called the fluctuation or detail part. The first fluctuation of the signal $\boldsymbol{f}$, denoted by $\boldsymbol{d_1} = (d_1, d_2, \dots\ d_{N/2})$, is computed by taking a running difference in the following way.

Its first value, d_1, is found by taking half the difference of the first pair of values of $\mathbf{f} = (f_1 - f_2)/2$; and multiplying it by $\sqrt{2}$. That is,

$$d_1 = (f_1 - f_2)/\sqrt{2} \tag{4.15}$$

Likewise, its next value d_2 is found by taking half the difference of the next pair of values of $\mathbf{f} = (f_3 - f_4)/2$; and multiplying it by $\sqrt{2}$In other words,

$$d_2 = (f_3 - f_4)/\sqrt{2} \tag{4.16}$$

Continuing this way, all the values of d_l are produced according to equation (4.17):

$$d_m = \frac{f_{2m-1} - f_{2m}}{\sqrt{2}} \text{, m = 1, 2, 3, …, N/2.} \tag{4.17}$$

Here, $\mathbf{a_1}$ is a vector of length N/2. Now, decompose each pixel vector of the hyperspectral data using the above procedure and collect them to produce a reduced data.

Several experiments have been planned for feature extraction of hyperspectral data using eight wavelets, namely, Haar, Daubechies 2, Daubechies 3, Daubechies 4, Coiflets 1, Coiflets 2, Coiflets 3 and Coiflets 4 wavelets each for first four levels of decomposition. Thus, a total of 32 reduced datasets have been produced. Each of these reduced data is then classified using the most appropriate classifier, namely, SVM and their classification accuracy assessed to evaluate the quality of each wavelet transformation. The original hyperspectral datasets (after pre-processing) have also been classified. Thus, a total of 33 classified images for each dataset (for dataset II, dataset III and dataset IV) have been generated. The training and testing samples have been kept same in all the classifications. The number of training and testing samples used in supervised classification has been given in Table 3.2, Table 3.3 and Table 3.4 respectively, from Chapter 3.

4.5.2 Implementation of PCA Based Techniques in Feature Extraction of Hyperspectral Data

Two types of PCA based feature extraction have been made. The implementation details of both the techniques have been given in this section.

4.5.2.1 Implementation of Segmented PCA

The spectral region is divided into few number of groups based on the image obtained from correlation matrix of the pre-processed hyperspectral data. First, a correlation matrix has been generated and the same is converted into an image. Then, Canny edge detection technique has been applied on the image obtained from correlation matrix and the sharp variation in tone in the image has been identified. Based on this, the image has been segmented into 4 to 8 segments. If a segment contains less number of bands (say around less than 4), then they are grouped with the neighbouring segment. Figure 4.7 shows a typical image obtained from correlation matrix of dataset II (AVIRIS Indiana Pine dataset) and the edges detected by Canny edge detection technique. Out of 224 bands, after pre-processing, only 182 bands have been identified and segmented. A total of 8 segments have been obtained from the correlation image, which are 1 – 3, 4 – 36, 37 – 79, 80, 81 – 100, 101, 102 – 180, 181 – 182. Since few segments have only one band, such bands are combined with the neighbouring bands. The resulting segments are 0 – 36, 37 – 79, 80 – 101, 102 – 182. Now, PCA has been applied on each segment and from each group, few components have been chosen according to the required variance.

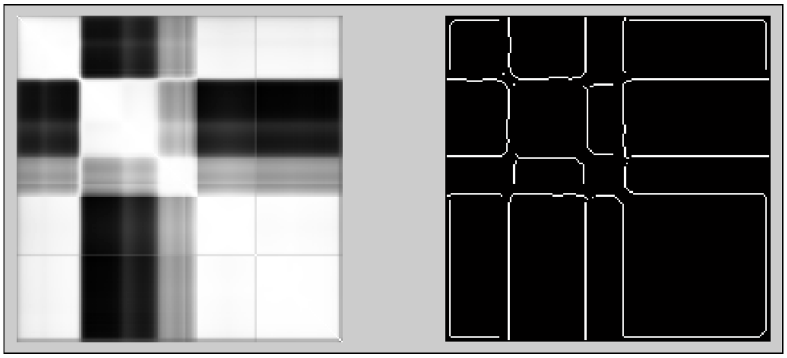


Figure 4.7: The image of correlation matrix. The edges of correlation image obtained after applying Canny edge detection technique.

4.5.2.2 Implementation of Spectrally Segmented PCA

In SSPCA, the spectral region is divided into 4 groups based on the spectral response of classes. The wavelength range for each class is different and hence the number of bands in each group is also different. The wavelength groups taken are VIS (visible), NIR (near infra-red), SWIR-1 (shortwave infra-red-1) and SWIR-2 (shortwave infra-red-2) and the corresponding ranges are 400-660nm, 670-900nm, 970-1350nm and 1530-2240nm respectively (Tsai *et al*., 2007).

After segmentation according to spectral region, PCA has been independently applied on each group and the predominant PCs from each group has been selected and used for further processing.

The quality of each feature reduction method has been assessed in terms of overall classification accuracy, users' and producers' accuracies based on the classifications obtained by applying SVM classifier on reduced datasets.

4.6 Results and Discussion

Application of these two feature extraction techniques, wavelet based feature extraction and segmented based PCA feature extraction techniques differs in the base mathematical formulation itself. The WBFE is multi-resolution analysis and has various levels of decomposition while in case of SPCA and SSPCA, the data has been segmented based on the spectral behavior of classes. For wavelet based feature extraction, the two parameters used in choosing the reduced datasets are the time required to carry over feature extraction and the overall classification accuracy. Priority has been given to classification accuracy and then to time required for feature extraction. For SPCA and SSPCA, the performance has been assessed based on the classification accuracy. The accuracy of classification produced from reduced number of bands in each case has been analysed. Since the land cover of the three datasets are different and also the spatial – spectral resolutions too, the analysis has been performed data wise.

4.6.1 Assessment of Feature Extraction Techniques for Dataset II (AVIRIS Indian Pine Dataset)

The performance of each of the feature extraction techniques has been assessed based on the overall accuracy and computational efficiency in reducing the dataset. Table 4.2 shows the number of features after reduction, duration of feature extraction (in seconds), duration of classification, overall accuracy and kappa coefficient for data reduced using different wavelets. Further, a detailed analysis on the basis of users' and producers' accuracies has been given on Chapter 5.

Table 4.2: The time duration for feature extraction and classification, overall accuracy and kappa coefficient for classifications obtained by considering features obtained at various levels from different wavelets for dataset II.

Sl. No.	**Name of the Wavelet**	**Number of Features**	**Duration of Feature Extraction (in seconds)**	**Duration of Classificatio n by SVM (in seconds)**	**Overall Accuracy**	**Kappa Coefficient**
1	Full Image	182	-	435.59	90.85	0.90
2	Haar_1	91	175.51	386.22	92.06	0.91
3	Haar_2	46	97.02	354.86	87.43	0.86
4	Haar_3	23	47.84	315.93	89.11	0.88
5	Haar_4	12	32.92	298.65	86.00	0.83
6	Daub2_1	92	195.83	387.62	100.00	1
7	Daub2_2	47	103.01	355.73	93.34	0.93
8	Daub2_3	25	55.96	316.39	91.57	0.91
9	Daub2_4	14	39.83	301.31	88.00	0.87
10	Daub3_1	93	191.71	389.24	92.52	0.92
11	Daub3_2	49	112.85	357.21	93.34	0.93
12	Daub3_3	27	60.76	322.56	92.91	0.92
13	Daub3_4	16	43.02	304.6	87.47	0.86
14	Daub4_1	94	194.07	390.38	92.32	0.91
15	Daub4_2	50	111.26	357.86	93.37	0.93
16	Daub4_3	28	63.09	325.41	74.34	0.72
17	Daub4_4	17	44.30	305.12	66.67	0.63
18	Coif1_1	93	197.05	386.67	88.52	0.87
19	Coif1_2	49	108.61	357.26	93.18	0.92
20	Coif1_3	27	56.26	321.29	92.82	0.92
21	Coif1_4	16	42.11	302.54	88.64	0.87
22	Coif2_1	96	197.45	392.53	88.71	0.87
23	Coif2_2	53	117.58	358.79	93.27	0.93
24	Coif2_3	32	70.84	337.98	91.67	0.91
25	Coif2_4	21	51.03	312.74	88.91	0.88
26	Coif3_1	99	203.51	394.68	92.13	0.91
27	Coif3_2	58	138.52	362.77	93.96	0.93
28	Coif3_3	37	81.20	344.56	66.83	0.64
29	Coif3_4	27	65.04	320.45	56.07	0.52
30	Coif4_1	102	223.74	396.45	86.25	0.85
31	Coif4_2	62	144.53	370.95	72.96	0.70
32	Coif4_3	42	99.00	350.48	74.34	0.72
33	Coif4_4	32	75.42	336.94	51.80	0.49
34	SPCA	48	120.43	352.94	90.25	0.87
35	SSPCA	50	118.57	353.86	93.80	0.91

From Table 4.2, it is evident that as the decomposition level increases, the time duration required to extract information for hyperspectral data decreases, and the number of features (bands become features after transformation) get decreased. The time required for classification is around 350 seconds for each of the datasets. It should not be misunderstood that 'if the number of features is reduced to half the time required to classify is also half '. Because the training stage depends upon various parameters, which includes number of training pixels, number of classes and finally number of features involved in classification duration varies.

But from Table 4.2, it is observed that among the 34 feature extraction techniques for dataset II, level-1 decomposition gives better classification accuracy for 3 out of 8 feature extractions while at level-2 decomposition 5 out of 8 feature extraction techniques give better accuracy with-in the sub-class of wavelets. Visual interpretation of Figure 4.9 shows that the classification from Daub2_1 and Coif4_4 are having significant difference between them. The overall accuracy for these two dataset also confirms the same. The overall accuracy for Daub2_1 is 100% while that of Coif4_4 is 51.804 and hence the information extraction by Daub2_1 is considered to be better than Coif4_4. Some classes are completely mis-classified into other classes which lead to low accuracy value in Coif 4_4. Even though the number of features has been reduced significantly, from 91 to 32, most of the useful information has been lost. So reduction time reduces from above 175 seconds (for Haar_1) to 75 seconds (for Coif4_4) also took away useful information from the dataset.

An insight study of reason for min-classification has been identified. Mis-classification between grass/pasture and wheat occurs while using Coif4, level-4 feature extraction. The spectral signatures of these two classes after decomposition by Daub2, level-1 and Coif4, level-4 feature extractions has been given in Figure 4.8 which shows that in Coif4_4, the discrimination between the two spectral signatures are less compared to that of Daub4_1. Similarly, mis-classification between corn-min and soy-min till, corn-min and soy-clean, woods and wheat (Table 4.3) is due to the level-4 decomposition by Coiflets. When applying Daub2, level-1 feature extraction technique, the 182 band data has been reduced to 92 features and from the reduced data, feature 26 has been given in Figure 4.9. Similarly, feature 17 from Coif4, level-4 reduced has been given in Figure 4.9. The correlation coefficient between the two features got from feature extraction by using Daub2_1 and Coif4_4 is 0.657. For calculating coefficient correlation between the two features, the last feature has been taken where in Daub2_1, the last feature is feature number 92 and in Coif4_4 it is 32. The correlation coefficient for the first feature from both the feature extraction techniques for dataset IV is 0.786.

In general both SPCA and SSPCA perform better in information extraction but more precise, SSPCA performs better than SPCA.

Table 4.3: Error matrix for dataset II reduced by Coiflet 4, level-4 decomposition.

Reference **Classified**	**Alfalfa**	**Corn-notill**	**Corn-min**	**Corn**	**Grass/pasture**	**Grass/trees**	**Grass/pasture-mowed**	**Hay-windrowed**	**Oats**	**Soy-notill**	**Soy-min till**	**Soy-clean**	**Wheat**	**Woods**	**Bldg-grass-trees-drives**	**Stone-steel towers**	**Total**
Alfalfa	26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**26**
Corn-notill	0	218	1	3	0	0	0	0	0	34	35	0	0	0	0	0	**291**
Corn-min	0	16	137	5	0	0	0	0	0	13	333	119	0	0	0	0	**623**
Corn	0	0	6	124	0	0	0	0	0	0	5	4	0	0	0	0	**139**
Grass/pasture	0	0	0	0	18	0	0	0	0	0	0	0	0	1	0	0	**19**
Grass/trees	0	0	0	0	0	184	1	0	0	0	4	1	0	51	39	0	**280**
Grass/pasture-mowed	0	0	0	0	0	0	21	0	0	0	0	0	0	0	2	0	**23**
Hay-windrowed	0	0	0	0	0	0	2	315	0	0	0	0	0	0	9	0	**326**
Oats	0	0	0	0	33	0	0	0	18	0	4	0	0	2	16	0	**73**
Soy-notill	0	0	0	0	0	0	0	0	0	283	17	15	0	0	0	0	**315**
Soy-min till	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	2	**4**
Soy-clean	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**0**
Wheat	0	0	0	0	247	0	0	0	0	0	9	0	126	425	12	0	**819**
Woods	0	0	0	0	0	0	0	0	0	0	0	0	0	46	0	0	**46**
Bldg-grass-trees-drives	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	0	**4**
Stone-steel towers	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	**60**
Total	**26**	**234**	**144**	**132**	**298**	**184**	**24**	**315**	**18**	**330**	**408**	**140**	**126**	**527**	**80**	**62**	**3048**

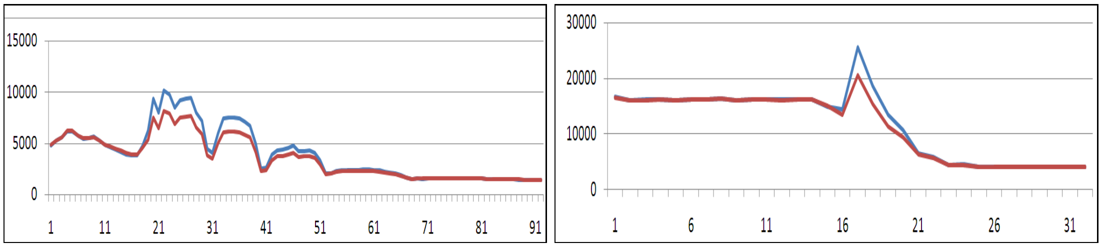


Figure 4.8: The spectral signatures for two classes (grass/pasture in blue colour and wheat in red colour) of dataset II after Daub2_1 and Coif4_4 wavelet feature extractions respectively.

Figure 4.9: Feature 26 after Daub2, level-1 reduction vs band 17 of Coif4, level-4 reduced datasets of dataset II.

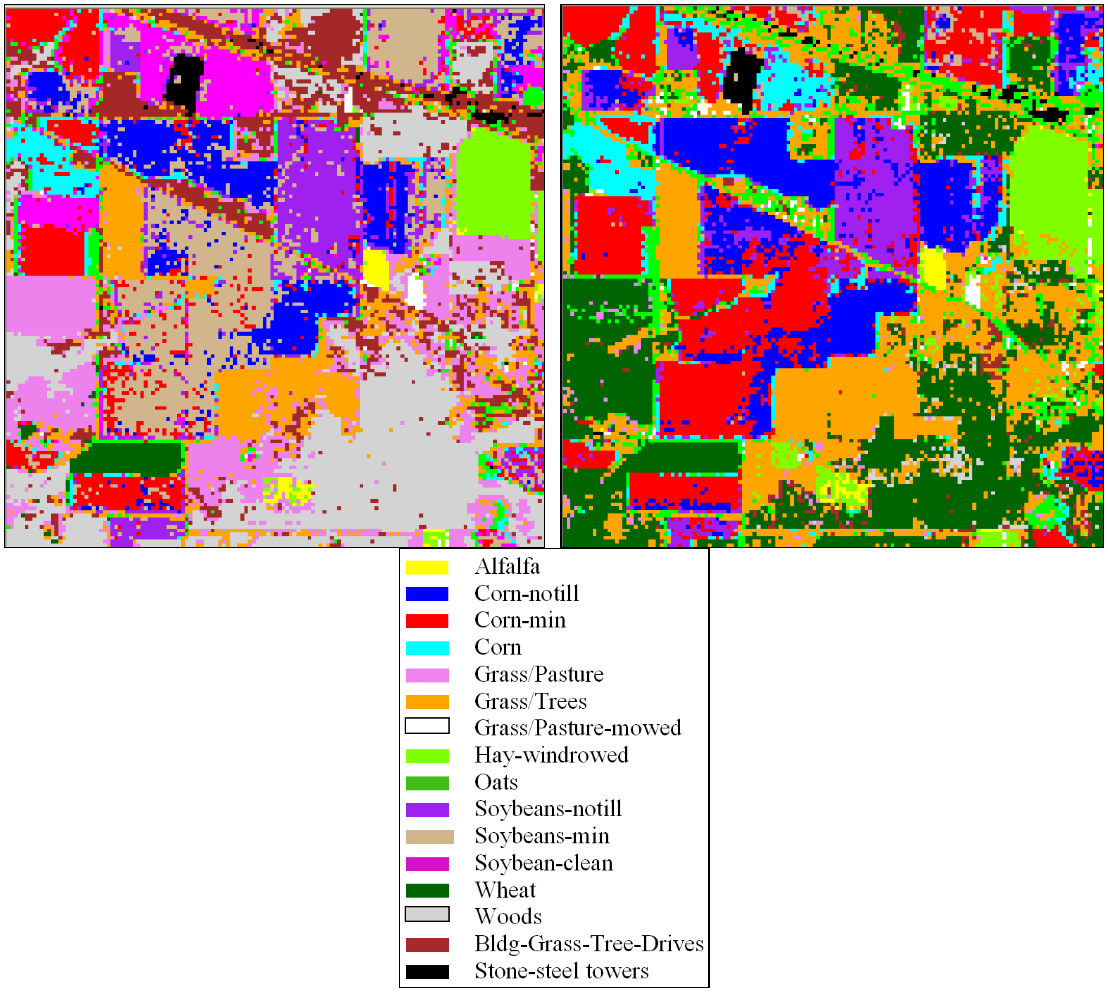


Figure 4.10: The pseudo coloured thematic map of dataset II classified by SVM which are reduced by Daub2, level-1 and Coif4, level-4 transformations along with legend.

4.6.2 Assessment of Feature Extraction Techniques for Dataset III (HYPERION Roorkee and its Surroundings Dataset)

The experimental dataset III is from HYPERION sensor which covers Roorkee and its surroundings. This dataset consists of trees and urban area as major classes while few other classes like barren land and sand are having less number of pixels. The classification accuracy wise comparison along with duration required to reduce the features has been given in this section.

Table 4.4: The time duration for feature extraction and classification, overall accuracy and kappa coefficient for classifications obtained by considering features obtained at various levels from different wavelets for dataset III.

Sl. No.	Name of the Wavelet	Number of Features	Duration of Feature Extraction (in seconds)	Duration of Classification by SVM (in seconds)	Overall Accuracy	Kappa Coefficient
1	Full Image	90	-	100.11	89.2	0.84
2	Haar_1	45	852.33	49.28	99.3	0.99
3	Haar_2	23	444.52	44.56	96.1	0.94
4	Haar_3	12	221.71	40.37	97.00	0.96
5	Haar_4	6	123.62	36.69	95.70	0.94
6	Daub2_1	46	821.94	50.34	98.10	0.97
7	Daub2_2	24	485.88	44.97	98.60	0.98
8	Daub2_3	13	247.34	41.43	39.90	0.27
9	Daub2_4	8	164.6	37.12	46.20	0.35
10	Daub3_1	47	754.38	51.52	97.10	0.96
11	Daub3_2	26	482.62	45.14	96.50	0.95
12	Daub3_3	15	288.38	42.32	47.40	0.37
13	Daub3_4	10	199.27	39.12	46.30	0.35
14	Daub4_1	48	784.01	52.47	97.70	0.97
15	Daub4_2	27	457.65	45.23	96.50	0.95
16	Daub4_3	17	335.5	43.27	46.90	0.36
17	Daub4_4	12	236.77	40.19	46.60	0.36
18	Coif1_1	47	762.78	51.96	97.50	0.96
19	Coif1_2	26	463.59	45.89	97.10	0.95
20	Coif1_3	15	283.72	41.48	46.90	0.36
21	Coif1_4	10	182.95	39.39	46.10	0.35
22	Coif2_1	50	861.71	56.43	97.30	0.96
23	Coif2_2	30	566.44	46.31	96.60	0.95
24	Coif2_3	20	387.79	43.45	46.40	0.35

25	Coif2_4	15	296.35	41.83	46.20	0.35
26	Coif3_1	53	896.72	57.58	98.30	0.98
27	Coif3_2	35	622.91	47.64	96.80	0.95
28	Coif3_3	26	494.7	45.73	45.70	0.34
29	Coif3_4	21	421.55	43.95	45.00	0.34
30	Coif4_1	56	977.36	58.8	89.30	0.96
31	Coif4_2	39	686.95	47.67	82.40	0.96
32	Coif4_3	31	583.86	46.37	48.20	0.37
33	Coif4_4	27	518.94	45.27	47.20	0.36
34	SPCA	27	465.73	45.92	92.50	0.91
35	SSPCA	30	556.80	43.82	96.40	0.93

Mis-classification happens between trees and urban, trees and water (Table 4.5) while decomposing the data by using Daub2, level-3 decomposition. Since the class urban and class trees occupy most of the region of dataset III, the mis-classification between them leads to significant decrease in overall accuracy (39.9 percent, from Table 4.4). Here also the visual interpretation of the two classified images shows significant difference. But the overall accuracy comes from Haar_1 is better among all reductions. Here also when the duration required for classification gradually increases (Table 4.4) with increase in number of features involved in classification. There is a sudden flow in classification accuracy when the feature extraction exceeds 2nd level decomposition, but a marginal change in duration of classification. The classification accuracy of such level algorithms is very poor.

Table 4.5: Error matrix for dataset III reduced by Daub 2, level-3 decomposition.

Sl. No.	**Reference Classified**	**Trees**	**Vegetation**	**Barren Land**	**Sand**	**Urban**	**Water**	**Total**
1	**Trees**	179	2	6	0	427	66	**680**
2	**Vegetation**	2	57	1	0	0	0	**60**
3	**Barren Land**	0	0	90	0	8	0	**98**
4	**Sand**	0	0	5	28	25	0	**58**
5	**Urban**	0	0	2	0	7	0	**9**
6	**Water**	0	0	0	0	0	0	**0**
	Total	**181**	**59**	**104**	**28**	**467**	**66**	**905**

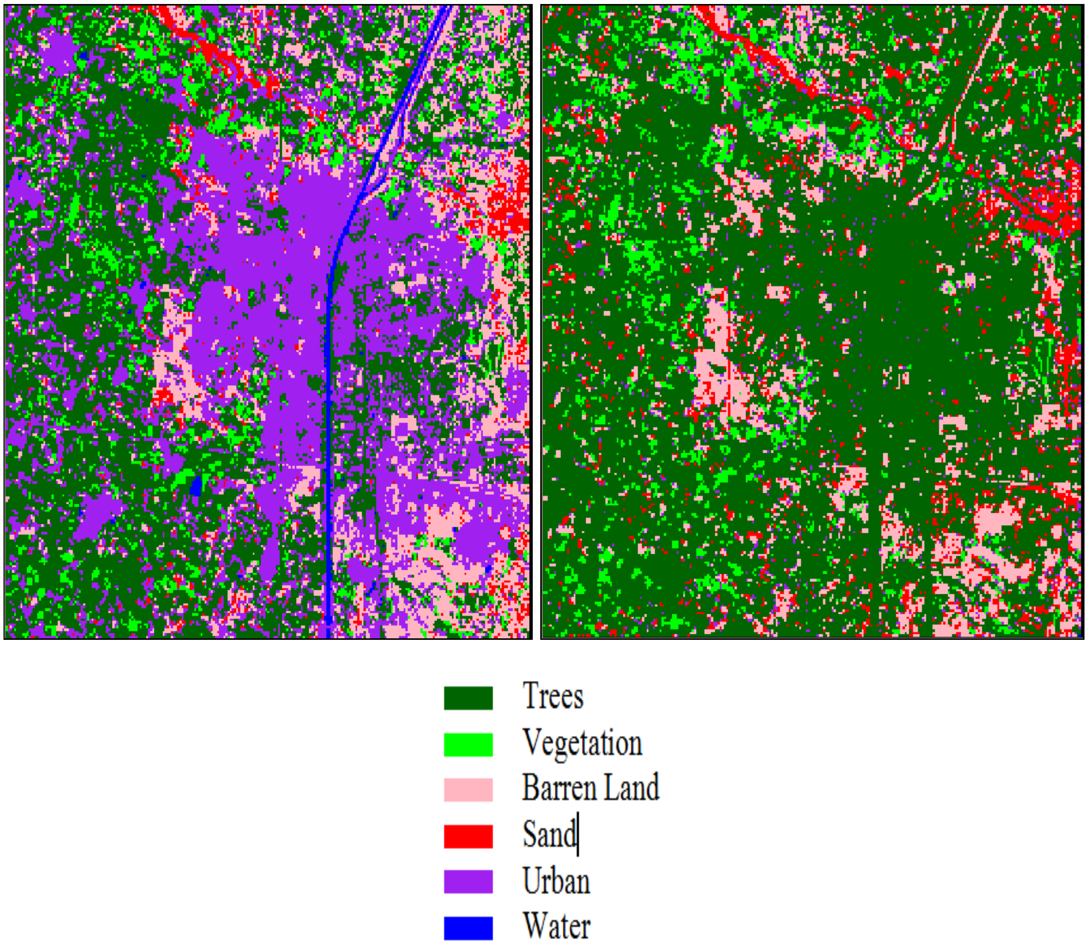


Figure 4.11: The pseudo coloured thematic map of dataset III classified by SVM which are reduced by Haar, level-1 and Daub2, level 3 transformations along with legend.

4.6.3 Assessment of Feature Extraction Techniques for Dataset IV (AVIRIS San Diego Naval Air Station Dataset)

Various types of concrete structures and various types of buildings are to be classified is considered to be the challenging part in classification of dataset IV. The feature reduction has been analysed by classification of the reduced datasets by SVM classifier has been given in this section.

Table 4.6: The time duration for feature extraction and classification, overall, accuracy and kappa coefficient for classifications obtained by considering features obtained at various levels from different wavelets for dataset IV.

Sl. No	Name of the Wavelet	Number of Features	Duration of Feature Extraction (in seconds)	Duration of Classification by SVM (in seconds)	Overall Accuracy	Kappa Coeff.
1	Full Image	189	-	45.72	96.00	0.95
2	Haar_1	95	188.81	37.23	97.20	0.97
3	Haar_2	48	101.59	25.16	99.30	0.99
4	Haar_3	24	49.62	17.84	34.50	0.25
5	Haar_4	12	34.73	17.64	34.00	0.24
6	Daub2_1	96	197.97	37.59	97.70	0.97
7	Daub2_2	49	106.96	25.31	98.60	0.98
8	Daub2_3	26	57.27	18.21	33.80	0.24
9	Daub2_4	14	39.38	17.21	33.30	0.24
10	Daub3_1	97	199.64	38.8	96.70	0.96
11	Daub3_2	51	114.97	25.56	98.80	0.98
12	Daub3_3	28	61.64	18.78	34.30	0.25
13	Daub3_4	16	42.7	17.49	36.20	0.27
14	Daub4_1	98	201.15	38.91	97.20	0.97
15	Daub4_2	52	113.52	25.83	96.20	0.96
16	Daub4_3	29	61.26	19.13	35.90	0.27
17	Daub4_4	18	42.88	17.93	34.70	0.25
18	Coif1_1	97	195.92	38.41	96.50	0.96
19	Coif1_2	51	105.44	25.97	98.40	0.98
20	Coif1_3	28	58.72	18.93	36.40	0.27
21	Coif1_4	16	39.67	17.42	35.00	0.25
22	Coif2_1	100	193.87	39.21	97.40	0.97
23	Coif2_2	55	115.2	25.94	98.80	0.98
24	Coif2_3	33	68.51	21.21	36.60	0.28
25	Coif2_4	22	49.2	17.01	36.60	0.27
26	Coif3_1	103	212.14	39.72	96.90	0.96
27	Coif3_2	60	134.07	26.3	90.60	0.89
28	Coif3_3	38	85.62	23.38	39.40	0.31
29	Coif3_4	27	62.81	18.67	36.90	0.28
30	Coif4_1	106	224.05	40.28	96.70	0.96
31	Coif4_2	64	148.1	27.49	98.80	0.98
32	Coif4_3	43	99.7	24.73	35.70	0.27
33	Coif4_4	33	79.13	20.28	35.50	0.26
34	SPCA	47	102.41	25.52	92.30	0.91
35	SSPCA	50	113.37	25.29	96.20	0.95

Table 4.7: Error matrix for dataset IV reduced by Daub 2, level-4 decomposition.

Reference Classified	Vegetation	Road Type 1	Road Type 2	Concrete Type 1	Concrete Type 2	Concrete Type 3	Building Type 1	Building Type 2	Building Type 3	Total
Vegetation	0	0	0	0	25	0	0	0	0	25
Road Type 1	4	7	2	0	40	6	0	0	9	68
Road Type 2	0	16	29	0	0	0	0	0	0	45
Concrete Type 1	0	13	2	33	1	5	0	0	0	54
Concrete Type 2	0	11	4	32	17	22	0	0	0	86
Concrete Type 3	0	0	1	0	1	3	0	0	27	32
Building Type 1	0	2	4	0	0	0	59	0	0	65
Building Type 2	0	0	0	0	5	0	0	0	0	5
Building Type 3	14	0	0	0	0	0	0	32	0	46
Total	**18**	**49**	**42**	**65**	**89**	**36**	**59**	**32**	**36**	**426**

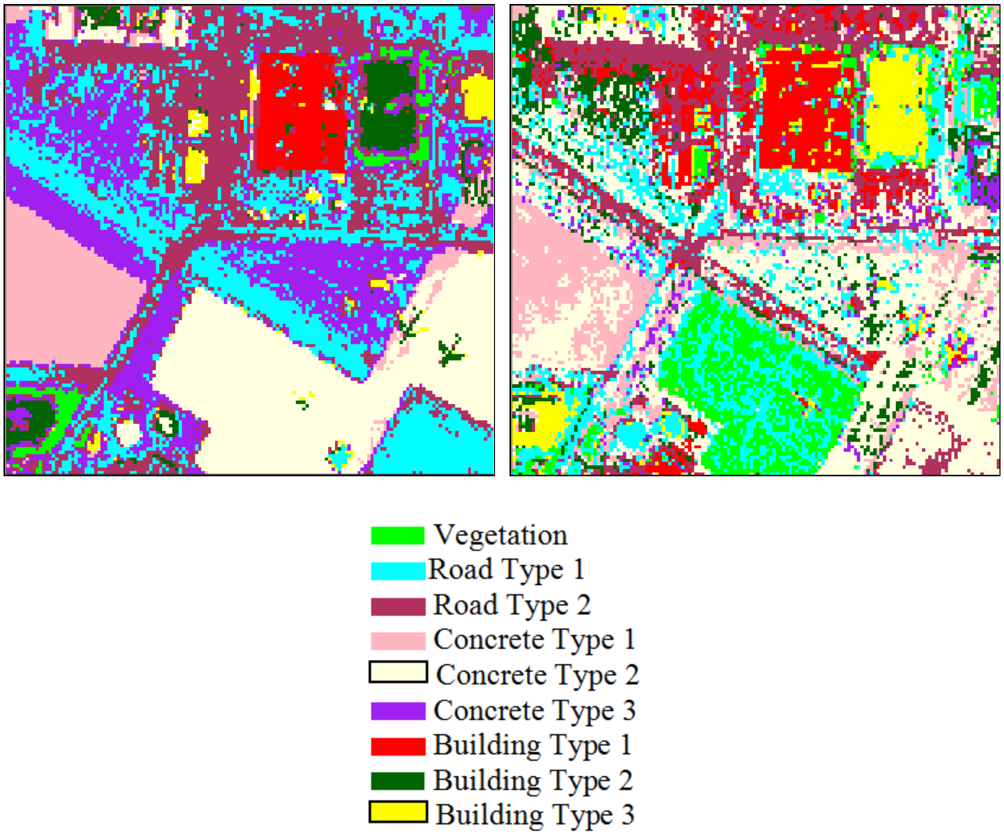


Figure 4.12: The pseudo coloured thematic map of dataset IV classified by SVM which are reduced by Haar, level-2 and Daub 2, level-4 transformations along with legend.

Here also the visual interpretation of the two classified images comes from Haar, level-2 decomposition and Daub, level-4 decomposition are having significant difference. The Table 4.7 above confirms the same which gives the error matrix for dataset IV reduced by Daub 2, level-4 decomposition. No pixel from the reference has been classified as vegetation, building type 2 and building type 3. Also mis-classification between every classes leads to poor overall accuracy of 36.6 %. Here also when the duration required for classification gradually increases (Table 4.6) with increase in number of features involved in classification.

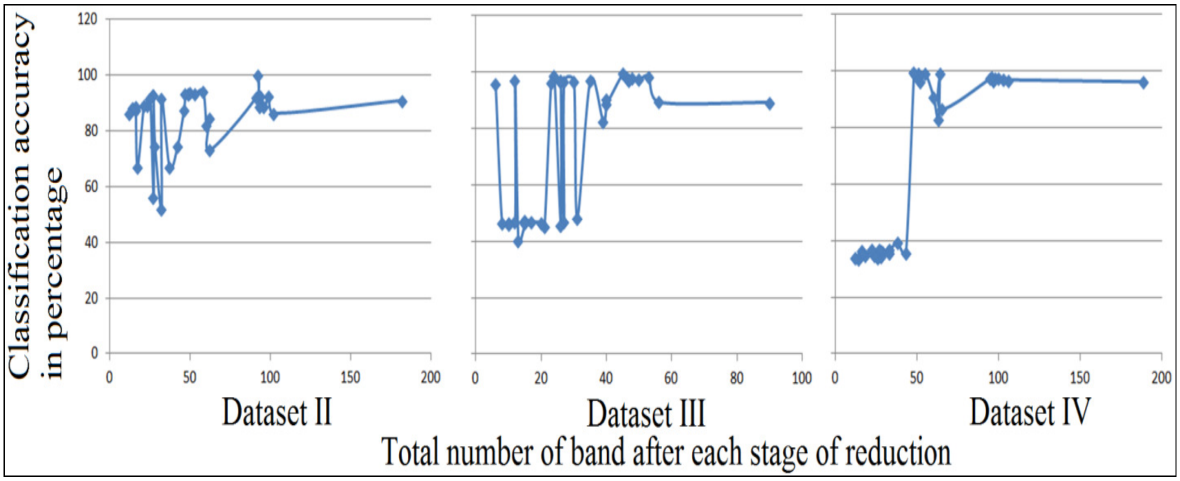


Figure 4.13: The band vs overall accuracy comparison for feature extraction at various levels of decomposition from different wavelets and PCA based feature extraction of dataset II, dataset III and dataset IV respectively.

From Figure 4.13 it is evident that for datasets III and IV, the classification accuracies have been increased when using 1st and 2nd level decompositions compared to original datasets. But when the decomposition level is increasing from 3rd level, the classification accuracy gets decreased for all the datasets. Hence for further analysis only the first 2 decomposition levels have been taken. Not only for 2D images but also for hyperspectral data of various spatial resolutions, Haar and Daubechies wavelets perform better than the rest and they are carried out for further experiments. For dataset II, since the classes are crisp the feature reduction performs better even when decomposition goes beyond 2nd level.

But from Table 4.2, it is observed that among the 34 feature extraction techniques for dataset II, level-1 decomposition gives better classification accuracy for 3 out of 8 feature extractions while at level-2 decomposition 5 out of 8 feature extraction techniques give better accuracy with-in the sub-class of wavelets.

The same observation has been made for the other two hyperspectral datasets too. From Table 4.4 for dataset III, it is clear that the first two levels of decomposition perform better in extracting useful information than the higher level decomposition. In fact, for this dataset, the level-1decomposition gives better classification accuracy for 6 out of 8 feature extractions while at level-2 decomposition 2 out of 8 feature extraction techniques give better accuracy.

For dataset IV, the situation is reversed even though the lower level decomposition performs better. Here, 6 out of 8 feature extractions comes from 2nd level decomposition produces better classification accuracy while only 2 out of 8 for 1st level decomposition produces better results obtained from Table 4.6.

In all the cases, there is no significant difference in classification accuracies by level-1 and level-2 decompositions but the number of bands used in 2nd level decomposition is almost half of that used in level-1 decomposition. This reduces time for feature extraction and time for classification. Hence level-2 decomposition from Daubechies is preferred both in accuracy wise as well as duration wise for feature extraction of hyperspectral data.

4.7 Summary

In this chapter, the mathematical background of wavelets and PCA has been described and the explanation about how they are useful in extraction of features from hyperspectral data has been given. The implementation of both the techniques has also been given in this chapter. There is a drastic reduction in the hyperspectral data by doing wavelet based feature extraction. When the level of decomposition increases the number of extracted bands (components) becomes almost half of the data on which reduction applied.

At the same time, the time duration required to extract the features decreases when the level of decomposition increases. The Haar wavelets are quite simple to handle while the Coiflets are complicated. In general, Daubechies wavelets perform better for 2D images and here also it is proved that they perform better than other wavelets and also than SPCA and SSPCA. All these wavelets take information from all the bands while PCA based extraction overlook the local variances that are helpful for the detection of useful information from hyperspectral data. In particular, the second level decomposition by wavelets gives better classification accuracy and also it reduces the hyperspectral data to 1/4th of the original size. The time duration for performing 2nd level decomposition is more than for performing 1st level decomposition but the classification accuracy wise 2nd level decomposition is better. Hence only the first two levels of decomposition based feature extraction by Haar and Daubechies sub classes have been taken for further information extraction.

Chapter 5

Mathematical Background, Implementation and Results of Per Pixel and Sub Pixel Classification of Hyperspectral Data

5.1 Introduction

Classification of remote sensing images is the process of labeling each pixel vector into various classes based on some mathematical matching logic. Classification is thus a key task to derive information from hyperspectral images for various applications like military applications (to distinguish between actual vegetation and camouflage (van der Meer and de Jong, 2001)), atmospheric characterization (Hsiao *et al.*, 2001) and climate research, snow and ice hydrology, monitoring coastal environments (Lucas *et al.*, 2002), understanding the structure and functioning of ecosystems, mineral exploration and land use land cover and vegetation mapping. The classification may be performed on per pixel, sub pixel or object basis depending upon the spatial resolution of the sensor viz medium, coarse and fine respectively.

The spatial resolution plays a vital role in classification. If the spatial resolution is finer for an image it may have classes or targets composed of several pixels as objects whereas if the spatial resolution is coarser, a pixel may contain many classes and is regarded as a mixed pixel. Thus, suitable classification algorithms must be used depending upon the spatial resolution of the dataset.

In object based classification, the image is segmented into groups of pixels to form objects (Pathak and Dikshit, 2003; Pathak and Dikshit, 2010). Each object is then assigned one class. In per pixel classification, a pixel is allotted to one and only class whereas in sub pixel classification, the proportion of classes present in a mixed pixel are obtained. These proportions are also called as fraction or abundance. The per-pixel, object based and sub-pixel classifications can be performed in supervised or unsupervised way depending upon whether the training data are available or not. To perform supervised or unsupervised classification, both parametric and non-parametric mathematical algorithms may be used. In this research three different types of algorithms have been used. Per pixel classification has been performed by support vector machines (SVMs) which requires less number of training pixels. In the case study, some of the classes of some experimental datasets are having less number of pixels leads to collection of less number of training pixels. SVMs are large margin classifiers and supervised and non-parametric in nature. The next two classifiers are sub pixel classifiers which produces fractional abundance of each pixel. Among the two classifiers the first one is supervised non-parametric (LMM) classifier. This is a supervised classifier and requires training data but the classification does not depend upon the statistics of the training data. Finally, an unsupervised parametric classifier by making use of ICAMM technique has been used to classify small classes. Classes which may not contain pure pixels and which may not be classified by using LMM may also be classified using ICAMM.

5.2 Brief Description of Algorithms Used for Classification of Hyperspectral Data

Every classification algorithm has its own mathematical background and hence the classification process differs in each case. A brief description about the classification algorithms used in classifying the hyperspectral data in this work has been given in this section.

5.2.1 Per Pixel Classification Algorithm

Per pixel classification algorithm aims in mapping each pixel to one and only one class. Here one machine learning algorithm (SVM) has been used and the description of this algorithm is given in this section.

5.2.1.1 Support Vector Machines (SVM)

It is one of the important many machine learning algorithms, introduced by Boser *et al.* (1992) and discussed in more detail by Vapnik (Vapnik, 1995, Vapnik 1998) has its root in statistical learning theory (Vapnik 1999) which intends to create a mathematical framework for learning from input training samples with known identity and produces the outcome of data points with unknown identity. The main advantage of using this algorithm is the capability of classification by taking less number of training pixels.

The classification of hyperspectral data having more than two classes may be performed by SVM by considering it to be a binary classification problem. First, a binary classification problem needs to be explained for clear understanding of SVM. Consider a set of data having two groups. This set needs to be partitioned into two groups such that the boundary between the two groups is as more as possible. Collect a set of training samples from the set. Let there be k number of samples which are represented by ($\mathbf{x_1}$, y_1), ($\mathbf{x_2}$, y_2), … ($\mathbf{x_k}$, y_k), where $\mathbf{x_i} \in \Re^N$ is an N-dimensional vector such that the samples belongs two any one of the two groups labeled as $y_i \in \{-1, +1\}$. The aim of the support vector classifier is to construct an optimal hyperplane from the set of k pixels that separates the two classes in such a way that the distance from the hyperplane to the closest training data points in each of the classes is as large as possible. Figure 5.1 show a set of training pixels consisting of two classes and few hyperplanes which divides the dataset into two groups. This distance is called the margin. The hyperplane can be represented by the following decision function:

$$\mathbf{w^T.x} + \mathrm{b} = 0, \tag{5.1}$$

where $\mathbf{x}$ is a point lying on the hyperplane, $\mathbf{w} \in \Re^N$ is normal to the hyperplane which determines the orientation of the discriminating hyperplane, and $\mathrm{b} \in \Re^N$ is a bias. The perpendicular distance from the hyperplane to the origin is $|b|/||\mathbf{w}||$, where $||\mathbf{w}||$ is the Euclidean norm of $\mathbf{w}$. Suppose that all the training data satisfy the following constraints (Bazi and Melgani, 2006):

$$\mathbf{w}^T\mathbf{x}_i + b \geq +1, \text{for } y_i = +1 \tag{5.2}$$

$$\mathbf{w}^T\mathbf{x}_i + b \leq -1, \text{for } y_i = -1 \tag{5.3}$$

These two equations can be further combined to give,

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 \geq 0 \tag{5.4}$$

One can then implicitly define a scale for ($\mathbf{w}$ x b) to generate two canonical hyperplanes (i.e., $\mathrm{P_1}$ and $\mathrm{P_2}$ in Figure 5.2), namely, $\mathbf{w}^T \times \mathbf{x}_i + b = 1$ for the closest training points lying on one side of the hyperplane with normal $\mathbf{w}$ and perpendicular distance from the origin $|1 - b|/||\mathbf{w}||$, and second $\mathbf{w}^T \times \mathbf{x}_i + b = -1$ for the closest training points on the other side with perpendicular distance from the origin $|-1 - b|/||\mathbf{w}||$. These training pixels are referred as support vectors. In other words, those vectors are central to the establishment of the optimal separating hyperplane. Accordingly, the margin between these two hyperplanes is $2/||\mathbf{w}||$. The maximization of this margin (Melgani and Bruzzone, 2004), in turn, leads to,

$$min\left\{\frac{||\mathrm{w}||^2}{2}\right\} \tag{5.5}$$

subject to equality constraints shown equation 5.4.

Equation (5.4) becomes easier to handle, if a primal Lagrangian formulation is used:

$$L_{primal} = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{n} \alpha_i(y_i(\mathbf{w} \times \mathbf{x_i} + b) - 1) + \sum_{i=1}^{n} \alpha_i, \quad (5.6)$$

where α_i are positive lagrangian multipliers. Now minimize L_{primal} with respect to **w** and b.

From Wolfe's theorem (Fletcher, 1987), take the derivatives of L_{primal} with respect to b and **w** to obtain,

$$\frac{\partial L_{primal}}{\partial b} = 0 \Rightarrow \sum_{\mathrm{i=1}}^{\mathrm{n}} \mathrm{\alpha_i y_i} = 0 \quad (5.7)$$

$$\frac{\partial L_{primal}}{\partial \mathbf{w}} = 0 \Rightarrow \quad \mathbf{w} = \sum_{\mathrm{i=1}}^{\mathrm{n}} \mathrm{\alpha_i y_i}\mathbf{x}_\mathrm{i} \quad (5.8)$$

and substituting them to the primal formulation (5.6) to give the Wolfe dual Lagrangian:

$$L_{dual} = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \times \mathbf{x}_j \quad (5.9)$$

subject to the constraints

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \quad (5.10)$$

and,

$$0 \leq \alpha_j \leq C, \qquad \text{for i = 1, 2, …,n} \quad (5.11)$$

Solution to equation (5.9) by satisfying the constraints given in equations (5.10) and (5.11) gives the separating plane. Hence this separating plane divides the entire set of points into two groups. While classification there are two types of technique may be followed, namely, one-against-one classification strategy Knerr *et al.* (1990) and one-against-all classification strategy. Also few kernel functions exist for performing SVM like, linear, polynomial, radial basis function (RBF) and Sigmoid kernel functions. In this research, the linear kernel function has been used and it is defined as the dot product of $\mathbf{x}_\mathrm{i}$ and $\mathbf{x}_\mathrm{j}$ ($\mathbf{x}_\mathrm{i}.\mathbf{x}_\mathrm{j}$).

In this research, one-against-one classification strategy has been used. If there are n number of classes, then there are ${}_{n}C_{2}$ hyperplanes created. By taking training samples for two classes, say class 1 and class 2, hyperplanes may be created and find the optimum hyperplane. Now pixels fall on one side are mapped as class 1 and the remaining pixels as class 2. Now take training samples for two classes, say class 1 and class 3. Here also each pixel is classified as either class 1 or class 3. Hence by performing various other combinations of training samples we get ${}_{n}C_{2}$ hyperplanes and based on these hyperplanes each pixel with unknown identity has ${}_{n}C_{2}$ values. These values represent the class it belong to by taking maximum appearance of class mapped to particular pixel. The Table 5.1 and Figure 5.3 give the hyperplanes and the classes of each pixel allocation to single class.

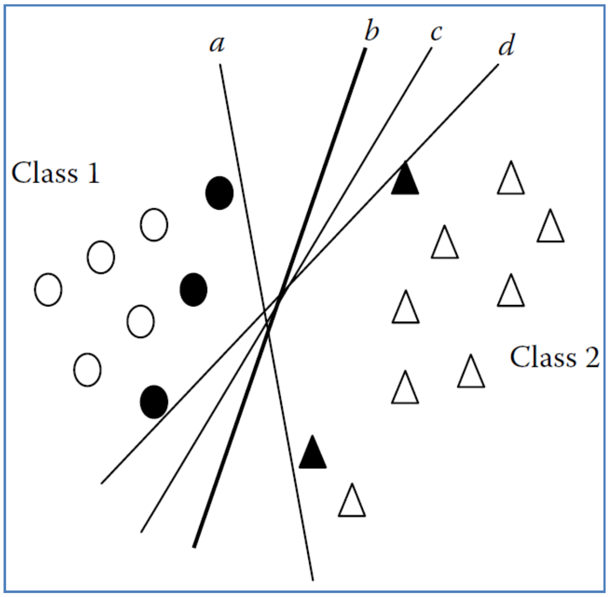


Figure 5.1: Hyperplane *b* separates the two classes with the maximal margin.

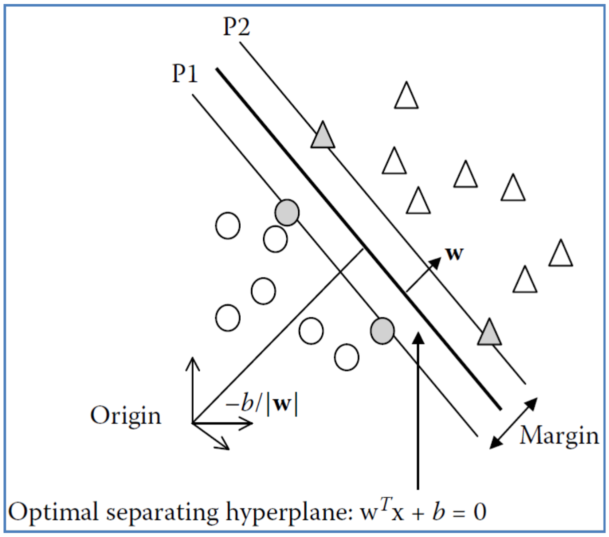


Figure 5.2: Linearly separable case

Table 5.1: Pixel allocation by each hyperplanes and final allocation

	Class 1vs Class 2	Class 1vs Class 3	Class 2vs Class 3	Final Allocation
Pixel 1	1	1	2	1
Pixel 2	1	1	2	1
Pixel 3	1	1	1	1
Pixel 4	1	1	1	1
Pixel 5	2	3	3	3
Pixel 6	2	3	3	3
Pixel 7	3	3	3	3
Pixel 8	3	3	3	3

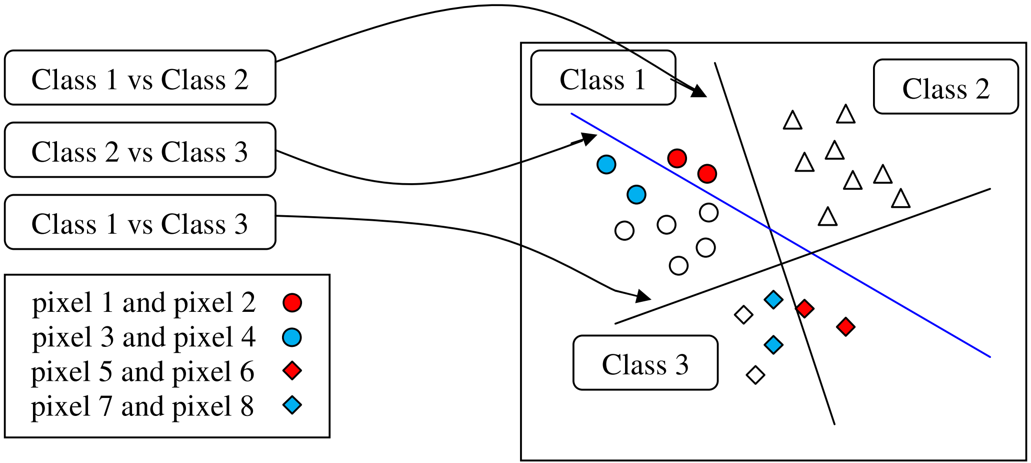


Figure 5.3: Allocation of class of each pixel by the hyperplanes

5.2.2 Sub Pixel Classification

Here two classification algorithms have been used while they both belong to different group, supervised classification and unsupervised classification. Information extraction by transforming the data into independent components has to be analysed and compared with that of the method of least square approximation technique. The supervised classification has been done by making use of Linear Mixture Model (LMM) while the unsupervised classification is by Independent Component Analysis Mixture Model (ICAMM). A brief description of these two classification algorithms have been discussed in the following sub-sections.

5.2.2.1 Linear Mixture Model (LMM)

The LMM assumes each pixel which is a spectral curve, a linear combination of the spectra of all classes within the pixel (Adams *et al*., 1995; Roberts *et al*., 1998; Ustin *et al*., 1998). The linear mixing model has been used due to its strong tie between the mathematical foundations of the model and the physical processes of mixing that result in much of the variance seen in hyperspectral imagery.

Common observations would support the case that there is a natural variation to almost all materials. Therefore, one would expect a certain degree of variability to exist for any materials that would be selected as pure pixel for a particular hyperspectral image. Few materials in a scene will actually be pure, and the collected training samples usually represent classes of materials that are spectrally very similar. Given the great diversity of materials of which a typical hyperspectral image is likely to be composed, the variation in spectral curve characteristics is further increased because of the practical limit to how many spectral curves can be used to represent a scene. When the intra-class variance (within class) is very small relative to the inter-class variance (between classes), the deterministic assumption upon which the linear mixing model is based may remain valid. However, when the intra-class variance becomes appreciable relative to the inter-class variance, the deterministic assumption becomes invalid and the training samples themselves should be treated as random vectors (Chang, 2007).

Depending upon whether the constraints are imposed or not, there are unconstrained and constrained approaches. The constraints applied to get an optimum solution in finding the fractional abundance are non-negativity constraint and sum-to-one constraint. The training pixels collected are generally assumed to be pure. The distribution followed by all the training pixels of each class should be unimodal normal distribution for each band. Various algorithms exist in extracting pure pixels from the data and the validation of pure pixels may be performed by pixel purity index (PPI). However, they are very difficult to obtain in practice. This is particularly true when the spectral curves are extracted directly from a hyperspectral data scene with no precise knowledge of ground truth.

Since the information extraction is from visible through mid-infrared region in 100s of bands, say n bands, each pixel vector is of length n. Now collect enough number of pure pixels for each of the available classes from the data. Let there be c number of classes. Now construct matrix M by collecting average pixel vectors from the collected pixel vectors of each class. We get c number of pixel vectors.

$$x = Mf + e, \tag{5.10}$$

Let M be the end member spectra which consists of pure pixels of each class. Each class has been arranged column wise and the band details for each class have been given row wise. Therefore, M is a matrix of order n x c. Now since each pixel is mixed by classes, the proportion of each class in that pixel is given by the vector, say f of dimension c x 1. Now x is the observed pixel vector of dimension n x 1. The last term e is an error vector of dimension n x 1. Here M and x are known and the aim is to find f, which is the fractional abundance of each class present in each pixel. But based upon two conditions that the fraction abundance could not be negative and the summation of each fraction abundance vector f to be one, the following two constraints have been included in estimation of f.

The non-negativity constraint is given by,

$$f_j \geq 1, \text{for all j land cover classes} \quad (5.11)$$

The sum-to-one constraint tells that the class proportions summation cannot exceed hundred percent is given by,

$$\sum_{j=1}^{c} f_j = 1 \quad (5.12)$$

Now solving equation (5.10), for f gives the fraction abundance of each class present in each pixel. Thus, we get solution of equation (5.10) by including the two constraints given in equation (5.11) and equation (5.12), we get a solution named as constrained solution while without the constraints, the obtained solution is called as unconstrained solution. The solution of equation (5.10) is unique if c = n+1. For c < n+1, there are infinite number of solutions while for c > n+1, there will be no exact solution.

In the second case (c < n+1), the number of unknowns is less than that of the number of equations and hence f can be found by making the error term in equation (5.10) to be zero. Hence method of least squares may be performed to find the class proportion f by keeping the sum of squares of the error or noise minimum (Settle and Drake, 1993).

$$(x - Mf)^T(x - Mf) = e^T e = e_1^2 + e_2^2 + e_3^2 + \ldots\ldots + e_c^2 \quad (5.13)$$

To solve equation (5.13), it is required to minimize the quadratic form of error, $Q(x, f)$,

$$Q(x, f) = (x - Mf)^T \sum_p^{-1}(x - Mf), \quad (5.14)$$

where $\sum_p$ is the pooled variance covariance matrix (Foody and Arora, 1996).

The unconstrained solution of equation (5.10) without imposing any condition is given by,

$$f = \left(M^T \sum_p^{-1} M\right)^T M^T \sum_p^{-1} x, \quad (5.15)$$

By imposing the constrains given in equation (5.11) and (5.12), the solution to (5.10) is given by,

$$f = \alpha U b + (U - \alpha U B U) M^{T} \textstyle\sum_{p}^{-1} x, \qquad (5.16)$$

where b is the unit matrix, α is a constant value equal to $(B^{T} U B)^{-1}$, where $B = (b\, b^{T})$, and $U = (M^{T} M)^{-1}$. Equation (5.16) is the constrained solution of equation (5.10)

The fraction values, f, of each vector are scaled between and 0 and 1 and further these values are normalized such that the summation of all the values in each vector comes to be one.

Pseudo code of LMM

```
Input the hyperspectral data
Identify the classes and collect training samples for each class
Find the average of training samples for each class band wise
Collect such vectors for each class got from the above step and put them in matrix M
For each pixel vector x
    Solve equation (5.10) for fractional abundance f
        Unconstrained solution: Equation (5.15) is the unconstrained solution
        Constrained solution: Equation (5.11) and equation (5.12) are the constraints and equation (5.16) is the constrained solution
    Normalize the values in the vector f
Display fractional image
```

5.2.2.2 Independent Component Analysis Mixture Model (ICAMM)

The origin of independent component analysis (ICA) is from signal processing. Imagine that in a room two persons are talking simultaneously. It was recorded by two microphones placed at two different locations in the room, but the positions of microphones are fixed throughout the recording process. These two microphones record the speech by both the persons but with different intensities. Let $x_1(t)$ and $x_2(t)$ be the intensities of the recorded signals at the two microphones at time t. Now this situation may be modeled as a linear combination of source signals weighed by parameters which depends upon the distance between microphones and the speakers. This situation has been expressed as, (Hyvärinen and Oja, 2000)

$$\begin{aligned} x_1(t) &= a_{11}s_1 + a_{12}s_2 \\ x_2(t) &= a_{21}s_1 + a_{22}s_2 \qquad (5.17) \end{aligned}$$

where a_{11}, a_{12}, a_{21}, a_{22} are the mixing parameters in which a_{ij} corresponds to distance between microphone i and person j and s_1 and s_2 are intensities of the original speech signals by the two persons respectively. In its simple form, the system of equations can be written as x = As. Recovering the original signal from the obtained sensor signal without having the knowledge about the mixing process is called as blind source separation (BSS) (Hyvärinen and Oja, 2000, Common, 1994).

Based upon the above analogy, the hyperspectral image may also be modeled as,

$$x_t = A_j s_{j,t} + b_j, \tag{5.18}$$

for deriving class proportions to produce a sub-pixel classification. Here, there are *t* number of mixed pixels are taken for finding fractional abundance, x_t is the intensity values of the pixels of hyperspectral data in various bands, A_j is the mixing matrix for class j which is unknown, $s_{j,t}$ is the source signal and b_j is the bias. Thus, in the present context, the solution of BSS problem leads to finding fractions, given the observed hyperspectral image.

The solution for ICA exists when the following restrictions are implicitly executed by the algorithm:

(i) The number of bands should be greater than or equal to the number of classes to be extracted.

(ii) Among the statistical distribution of all the classes for each band, at most one distribution can be Gaussian. Remaining distributions should be non-Gaussian.

(iii) For each class, either no sensor noise or only low additive noise signals are permitted.

The matrix A is the linear combination that transforms the data **x** into independent components **s**. Every class is modeled as a linear combination of spectral response values weighted by the class-component densities.

Let the number of spectral bands of a hyperspectral data be N. Then each pixel vector x_t, denoted as $\mathbf{x_t} = (x_1, x_2, \ldots x_N)^T$, is an N dimensional vector. Assuming that there are T number of pixels given by $X = \{\mathbf{x_1, x_2, \ldots x_T}\}$ that are to be unmixed, the likelihood of the data (Lee *et al.*, 1999) is given by the joint density,

$$p(X \mid \Theta) = \prod_{t=1}^{T} p(\mathbf{x_t} \mid \Theta), \tag{5.19}$$

where $\Theta = (\theta_1, \theta_2, \dots \theta_K)$ are the K class parameters, which are to be adapted.

The maximum likelihood estimate of Θ is the value of Θ that maximizes the value $p(X \mid \Theta)$ (Shah *et al.*, 2004).

Let the pixel vectors from the hyperspectral data be obtained probabilistically from the set of spectral classes$\{C_1, C_2, \dots C_K\}$. The probability of selecting the spectral class C_j is

$P(C_j)$ and hence the probability of a pixel x_t having the parameter Θ is given by the probability density function (pdf) of x_t, which may be expressed by the mixture density model (Lee *et al.*, 1999),

$$p(\mathbf{x_t} \mid \Theta) = \sum_{j=1}^{K} p(\mathbf{x_t} \mid \mathrm{C_j}, \theta_j), \tag{5.20}$$

The conditional density for class C_j is given by (Lee *et al.*, 2000),

$$p(\mathbf{x_t} \mid \mathrm{C_j}, \theta_j) = \frac{p(s_{j,t})}{|\det(A_j)|} \tag{5.21}$$

Normalize $p(s_{j,t})$, so that the fraction abundance becomes non-negative and less than one (Plumbley, 2003).

The classes are assumed to follow multivariate non-Gaussian distribution and the ICAMM algorithm is capable of finding the non-orthogonal directions of the data distributed. Since the distribution of the entire data is modeled as a weighted sum of the class-component densities of all the classes, it is represented as follows:

$$\mathbf{x_t} = A_j s_{j,t} + b_j \tag{5.22}$$

where A_j is a full-rank matrix assumed as the basis matrix or mixing matrix which specifies the linear combination of independent sources (Lee and Lewicki, 2002). By knowing only the observed mixed signals $\mathbf{x_t}$, the aim of ICAMM is to find the independent components, the mixing matrix A_j for each class and the bias b_j and hence the class member probability for each pixel.

Each A_j consists of M rows such that each data is represented by these M hyperplanes. In general, N ≥ M. For simplicity, let the number of bands be the same as number of components (N = M). Such assumption can be met by transforming the hyperspectral data in reduced feature space. The bias b_j are the translation parameters which gives the position of the center of cluster (or spectral mean of class) C_j. From equation (5.22) we get,

$$s_{j,t} = A_j^{-1} (\mathbf{x_t} - b_j) \tag{5.23}$$

For each pixel vector x_t, calculate the class-component density, given by equation (5.22).

Adapt A_j 's and b_j's by using gradient ascent for learning the parameters (Lee *et al.*, 1999). Learning the parameters is a process in which iteratively the parameters A_j and b_j are updated until they fit into the data as perfect as possible. They are adapted as follows:

$$\Delta A_j \propto p(x_t \mid \mathrm{C_j}, \theta_j) A_j \, [I - \Phi_j \tanh(s_{j,t})\, s_{j,t}^T - s_{j,t}. s_{j,t}^T] \tag{5.24}$$

where I is an identity matrix, and Φ_j is an N-dimensional diagonal matrix corresponding to class j and is composed of $\phi_{i,j}$ (Shah and Varshney, 2004) defined by:

$$\phi_{i,j} = \frac{m_4(s_{j,i,t})}{m_2^4(s_{j,i,t})} - 3, \tag{5.25}$$

where, m_p corresponds to the pth moment of the independent source. The value of kurtosis $\phi_{i,j}$ is positive for super-Gaussian, negative for sub-Gaussian and zero for Gaussian $s_{j,i}$.

For each class j, the bias is updated by,

$$b_j = \frac{\sum_{t=1}^{T} x_t P(C_j \mid x_t,\ \Theta)}{\sum_{t=1}^{T} P(C_j \mid x_t,\ \Theta)}, \tag{5.26}$$

In general, the proportionality constant is kept in-between 0 and 1 and here in this research we have taken it as 0.7. The kurtosis for ith independent component over class j, is used to find the class-component density and is given by equation (5.22). The independency of the component is checked from the joint probability density given as in equation (5.19).The kurtosis is used to find the class-component density. Once the A_j 's and b_j's are adapted, the class member probability for each pixel is calculated from Baye's theorem given as follows:

$$p(C_j \mid \mathrm{x_t}, \Theta) = \frac{p(x_t \mid \mathrm{C_j}, \theta_j) P(C_j)}{\sum_{j=1}^{K} p(x_t \mid \mathrm{C_j}, \theta_j) P(C_j)}, \tag{5.27}$$

where all $P(C_j)$s' are assumed to be equal.

Now the pixel x_t is allotted to class C_j, which is having highest probability and is given by:

$$x_t \in C_i \; if \; p(C_i \mid \mathrm{x_t}, \Theta) > p(C_j \mid \mathrm{x_t}, \Theta), for \; i \neq j \tag{5.28}$$

Pseudo code of ICAMM (Lee and Lewiciki, 2002)

```
Initialize the mixing matrices and bias
For each pixel vector $x_t$,
Repeat
    For each class
        Calculate $s_{j,t}$
        Calculate $p(x_t \mid \Theta)$
        Calculate $p(C_j \mid \mathrm{x_t}, \Theta)$
        Check for independence
        Adapt $A_j$
        Adapt $b_j$
Repeat the entire process for all t = 1, 2, …T.
Convert into hard classification output, if required.
```

5.3 Accuracy Assessment of Classification of Hyperspectral Data

The assessment of classification accuracy is an essential step in any classification process. Generally, for per pixel classification, error matrix based accuracy assessment has been performed while for a sub pixel classification technique, fuzzy error matrix based accuracy assessment has been found suitable.

5.3.1 Error Matrix Based Accuracy Assessment

Classification is not complete unless assessment of accuracy has been performed (Jensen, 2002). An excellent review on the status of classification accuracy assessment till 2002 has been provided in Foody (2002). One of the most common ways of representing the classification accuracy is by defining it with the help of an error matrix. An error matrix is a cross tabulation of the classes on the classified remotely sensed image and the reference data. It is represented by a c x c matrix (where *c* is the number of classes). The columns of the matrix generally define the reference data while the rows define the classified remotely sensed data albeit both are inter-changeable. The error matrix has been referred to in the literature by different names such as confusion matrix, contingency table, evaluation matrix and mis-classification matrix. The elements of this matrix indicate the number of samples in the testing data. A typical error matrix is shown in Table 5.2 (Varshney and Arora, 2004).

The error matrix is used to derive several classification accuracy measures. Overall accuracy (also referred to as percentage correct allocation) is the ratio of number of correctly classified samples (i.e., the sum of diagonal elements of the error matrix) to the total number of samples in the testing dataset. Overall accuracy is the measure of the classification as a whole. Even though to calculate the percentage of correctly classified/mapped pixels or overall accuracy, it requires the total number of correctly classified pixels and total number of pixels. There are other two simple measures that determine accuracy of an individual class; producer's accuracy and user's accuracy. The producer's accuracy is so called because the producer (or originator) of the classified map is typically interested in how well the samples from the reference data can be mapped using remotely sensed data. In contrast, the user's accuracy is an indication of the probability or reliability that a sample from the classified map actually represents that class on the reference data. From the error matrix, the producer's accuracy may be determined by dividing the number of correctly classified samples of a class by the column total (column marginal). The user's accuracy may be determined by dividing the number correctly classified sample of a class by the row total (row marginal) (Varshney and Arora, 2004).

If each row of error matrix denotes the classified data, the columns by reference data and let the confusion matrix be denoted by E then E is defined as,

$$E = \sum_{j=1}^{c} \sum_{i=1}^{c} n_{ij} \qquad (5.29)$$

The same equation may be represented as matrix form too as given in Table 5.2:

From this table, various accuracy measures like overall accuracy, users' accuracies, producers' accuracies, kappa coefficient, etc., may be calculated. The formulae to calculate each of these accuracy measures are given as follows:

The overall accuracy is the ratio of number of correctly classified samples to total number of samples. From the table, the number of correctly classified samples may get from the trace of the matrix.

$$Overall\ accuracy = \sum_{i=1}^{c} n_{ii} \qquad (5.30)$$

Table 5.2: Conventional error matrix for number of classes is c.

Classified data	Reference data					Row Total
	Class 1	Class 2	Class 3		Class c	
Class 1	n_{11}	n_{12}	n_{13}	.	n_{1c}	$N_1.$
Class 2	n_{21}	n_{22}	n_{23}	.	n_{2c}	$N_2.$
Class 3	n_{31}	n_{32}	n_{33}	.	n_{3c}	$N_3.$
	.	.	.	.	.	.
Class c	n_{c1}	n_{c2}	n_{c3}	.	n_{cc}	$N_c.$
	$N._1$	$N._2$	$N._3$		$N._c$	$N = \sum_{i=1}^{c} N_i. = \sum_{i=1}^{c} N._i$

Definition of terms used
1. N is the total number of testing samples
2. n_{ii} is the number of correctly classified samples
3. n_{ij} is the number of wrongly classified samples for class j in the reference data and class i in the classified data
4. $N_i.$ is the sum of ith row of the matrix and by taking all the columns
5. $N._j$ is the sum of jth column of the matrix and by taking all the rows

Even though we get an overall view of the classified map from overall accuracy that does not give about accuracy of how individual classes are mapped. So here two simple accuracy measures users' accuracies and producers' accuracies may be helpful in such studies. The producer of the map wishes to check how well the algorithm is capable to map the reference data correctly. The producer's accuracy for class i may is the ratio of number of samples from class i correctly classified to total number of samples of class i from the reference and hence it is defined as:

$$For\ class\ i, producer's\ accuracy = \frac{n_{ii}}{N._i}, for\ all\ i = 1, 2, \dots c \quad (5.31)$$

On the other side, the user of the map is willing to know that how well the sample from the classified map represents that class on the reference data. From the table it can be calculated by taking the ratio of number of samples from class i correctly classified to total number of samples of class i from the classified and hence defined as:

$$For\ class\ i, user's\ accuracy = \frac{n_{ii}}{N_{i\cdot}}, for\ all\ i = 1, 2, \dots c \quad (5.32)$$

The overall accuracy takes only the diagonal elements of the error matrix and hence if one class is having significantly more number of samples then the classification accuracy result may dominate by that particular class. On the contrary, a class with significant amount of less number of samples may not be projected well. Hence users' and producers' accuracies have to be done to focus how individual classes have been mapped. But even then the use of unclassified samples has not been properly used. These accuracy measures are not capable of comparing between classified and reference data. Such measures usually overestimate the accuracy. The kappa coefficient of agreement has the capability to account for class agreement. The kappa coefficient is denoted by κ and defined as follows:

$$\kappa = \frac{(n)(\sum_{i=1}^{c} n_{ii}) - \sum_{i=1}^{c}(N_{i}.)(N._{i})}{n^2 - \sum_{i=1}^{c}(N_{i}.)(N._{i})} \qquad (5.33)$$

Hence there exist many measures of a classified hyperspectral data based on different assumptions about the dataset. All the accuracy measures have been derived from the error matrix which gives full picture about the pixels mapped correctly, even wrongly classified pixels to which class it has been mapped. But in error matrix, only non-negative integer values are entertained because both the classified data as well as the reference data taken are per pixels. This may underestimate or overestimate the accuracy when there are more mixed pixels in the data. So a modification to error matrix is required by giving fuzzy input values both at classified data as well as reference data.

5.3.2 Fuzzy Error Matrix (FERM) Based Accuracy Assessment

Unlike in traditional error matrix based accuracy assessment, here instead of one error matrix to be produced, for each pixel of the testing sample, error matrix may be produced. Hence, there is an equal number of error matrices equal to the size of the set containing testing samples. Then these matrices are added–up to form fuzzy error matrix. From the obtained fuzzy error matrix we may able to calculate the overall accuracy, users' accuracies for each class, producers accuracies for each class and the kappa coefficient, in the same way how these have been calculated from traditional matrix. The way how error matrix may be obtained for each pixel from the training sample set may be defined as follows:

Let R_i and C_i (for i = 1, 2, …c) be set of vectors representing the proportion of each class in a pixel from a reference pixel and a classified pixel, respectively. Both the vectors R_i and C_i are of equal in length whos length is same as that of the number of class in the reference data. The elements in the two vectors are arranged by the corresponding classes in equally (i.e., class 1, class 2, … class c). Here R_i and C_i contains values in fraction whos summation equals 1. It means R_i and C_i are fuzzy sets. The membership function of R_i and C_i may be given as:

$$\mathfrak{I}_{R_i}: X \rightarrow [0,1], for\ i = 1, 2, \dots c$$

and

$$\mathfrak{I}_{C_i}: X \rightarrow [0,1], for\ i = 1, 2, \dots c \qquad (5.34)$$

where, [] represents closed interval in which both the end points 0 and 1 are included in the set. Since, these membership functions represent the proportions of classes in each pixel vector, the summation of each vector is 1.

i.e., $\sum_{i=1}^{c} \mathfrak{I}_{R_i} = 1$

and $\sum_{i=1}^{c} \mathfrak{I}_{C_i} = 1 \qquad (5.35)$

Hence all the accuracy measures may be calculated by the above formulae and the quality of classified images are analysed.

5.4 Experimental Set-up to Extract Information from Hyperspectral Data from Per-pixel and Sub-pixel Classification

From the feature extraction techniques discussed in Chapter 4 and section 4.6 we have concluded that only Haar and Daubechies wavelets performed better that Coiflets wavelet transform and PCA based feature extraction techniques for all the three datasets. The comparison has been made by considering few parameters like classification accuracy, time duration required to do feature extraction, time required to do classification, levels of decomposition and number of components/features extracted after feature extraction. Also, only the first two levels of decomposition have been performed well in extraction of information from hyperspectral data.

SVM classification output **LMM classification output**

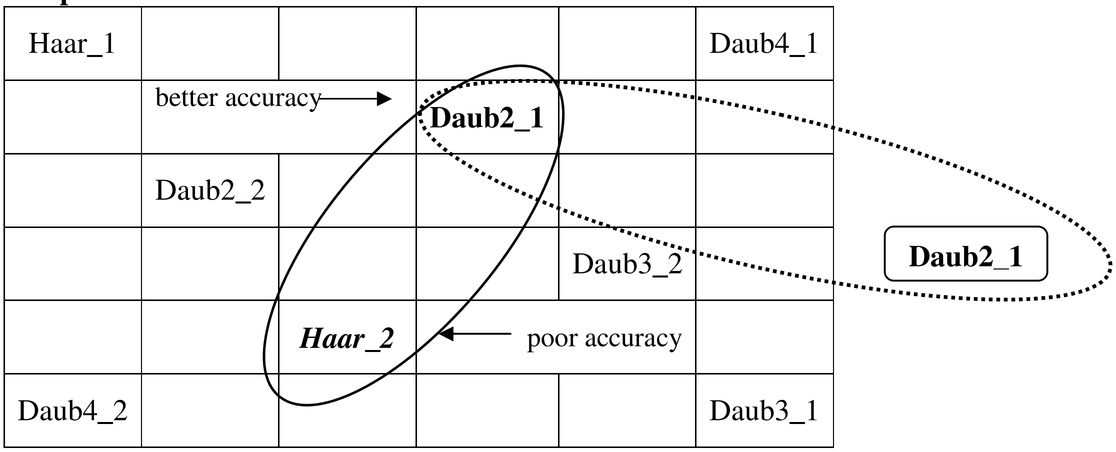


Figure 5.4: Comparison between SVM and LMM classified images

──── Comparison between better and poor overall accuracy from SVM classifier

·········· Comparison between SVM classified and LMM classified outputs of same feature extraction techniques

In this chapter, an in-depth analysis of information extraction has been made by classifying the reduced datasets (reduced by the above said techniques) by three classifiers. For SVM, comparison of best classification with the worst classification has been done for all the datasets. Also, the best classification of SVM is compared with the corresponding LMM classification because of the SVM classification produces good accuracy and hence it has been taken as reference image for validating LMM output. For example, for dataset II, the SVM classification for Daub2_1 performs better than other feature extraction techniques, while Haar_2 performs poor when compared to other first and second level decomposition techniques. So here Daub2_1 has been compared with Haar_2. Also, since Daub2_1 performs well in SVM classification it has been kept as a bench mark and the LMM classification of Daub2_1 has been compared with it. For ICAMM, the extractions of classes which have significant less number of pixels have been analysed. The experimental set-up, training data and testing data have been given in Chapter 3. The process of comparison of SVM classified and LMM classified images are given in Figure 5.4.

5.4.1 Observations and Result for Experimental Dataset II (AVIRIS – Indiana Pine)

Few observations based on classification by SVM, LMM and ICAMM has been made and they are measured by image to image comparison, error matrix accuracy assessment, fuzzy error matrix accuracy assessment, etc. Also, the number of pixels mentioned here is for pixels taken from reference data.

5.4.1.1 Error Matrix Based Accuracy Assessment for SVM

1. The SVM classification of Daub2_1 (Figure 5.5(a)) gives better classification accuracy while that for Haar_2 (Figure 5.5(b)) gives poor classification accuracy. Hence a comparison has been made between these two classified images.
2. For Haar_2 feature extraction and followed by SVM classification, 322 pixels (Table 5.3) are classified as corn-min. Among this, 101 pixels which have to be classified as corn and 52 pixels have been classified as soy-min till instead of classified as corn-min, leads to poor user's accuracy for corn-min and poor producer's accuracy for corn and soy-min till (corn-min vs corn, corn-min vs soy-min till).
3. Similarly, among the 366 pixels classified as soy-min till, 29 pixels from reference data for corn-notill in SVM classification of Haar_1 feature extraction. This leads

to poor user’s accuracy for soy-min till and poor producer’s accuracy for corn-notill (soy-min till vs corn-notill). Similarly, among the 366 pixels classified as soy-min till, 21 pixels from reference data for soy-notill. This lead to poor user’s accuracy for soy-min till and poor producer’s accuracy for soy-notill (soy-min till vs soy-notill).

4. On the contrary, among the 126 pixels which are to be classified as wheat, 16 pixels have been classified as grass/pasture and 37 pixels have been classified as grass/trees. This leads to poor producer’s accuracy in wheat and poor users accuracies in grass/pasture and grass/trees (wheat vs (grass/pasture and grass/trees)).

Table 5.3: Error matrix for Haar_2

Reference **Classified**	**Alfalfa**	**Corn-notill**	**Corn-min**	**Corn**	**Grass/pasture**	**Grass/trees**	**Grass/pasture-mowed**	**Hay-windrowed**	**Oats**	**Soy-notill**	**Soy-min till**	**Soy-clean**	**Wheat**	**Woods**	**Bldg-grass-trees-drives**	**Stone-steel towers**	**Total**
Alfalfa	26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**26**
Corn-notill	0	187	0	11	0	0	0	0	0	5	11	0	0	0	0	0	**214**
Corn-min	0	18	138	101	0	0	0	0	0	1	52	12	0	0	0	0	**322**
Corn	0	0	6	19	0	0	0	0	0	0	2	0	0	0	0	0	**27**
Grass/pasture	0	0	0	0	298	0	0	0	0	0	8	0	16	0	2	0	**324**
Grass/trees	0	0	0	0	0	184	0	0	0	0	3	0	37	0	3	0	**227**
Grass/pasture-mowed	0	0	0	0	0	0	23	0	0	0	0	0	0	0	0	0	**23**
Hay-windrowed	0	0	0	0	0	0	1	315	0	0	0	0	0	0	0	0	**316**
Oats	0	0	0	0	0	0	0	0	18	0	7	0	6	0	1	0	**32**
Soy-notill	0	0	0	0	0	0	0	0	0	303	12	3	0	0	0	0	**318**
Soy-min till	0	29	0	0	0	0	0	0	0	21	310	4	0	0	0	2	**366**
Soy-clean	0	0	0	1	0	0	0	0	0	0	1	121	0	0	0	1	**124**
Wheat	0	0	0	0	0	0	0	0	0	0	1	0	67	0	0	0	**68**
Woods	0	0	0	0	0	0	0	0	0	0	0	0	0	523	0	0	**523**
Bldg-grass-trees-drives	0	0	0	0	0	0	0	0	0	0	1	0	0	4	74	0	**79**
Stone-steel towers	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	**59**
Total	**26**	**234**	**144**	**132**	**298**	**184**	**24**	**315**	**18**	**330**	**408**	**140**	**126**	**527**	**80**	**62**	**3048**

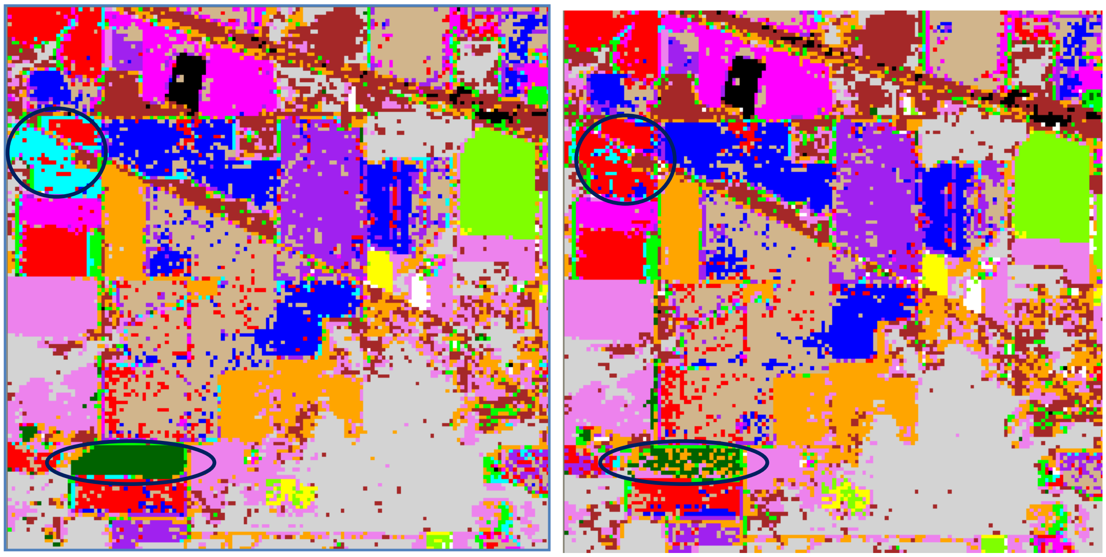


Figure 5.5 (a) SVM classified pseudo image got from Daub2_1 feature extraction of AVIRIS dataset of Indiana Pine, (b) SVM classified pseudo image got from Haar_2 feature extraction of AVIRIS dataset of Indiana Pine.

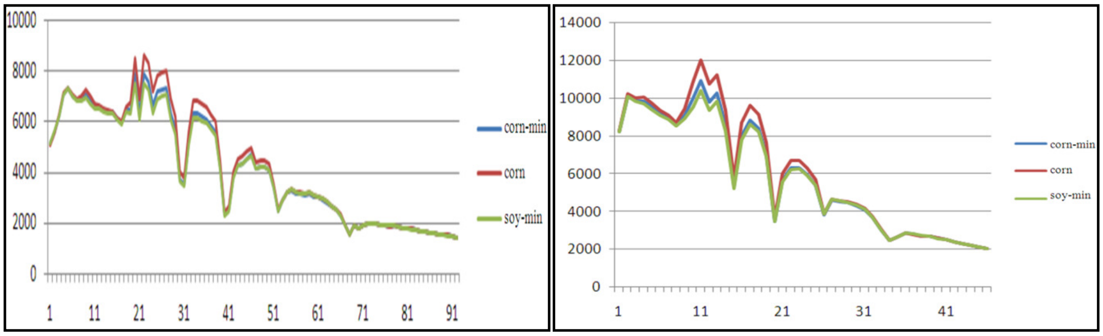


Figure 5.6:- Spectral curves for the three classes corn-min, corn and soy-min till after decomposition using (a) Daub2_1 and (b) Haar_2.

The spectral curves for the three classes corn-min, corn and soy-mintill comes from Daub2_1 feature extraction and Haar_2 feature extraction wavelets have been given in the Figure 5.6. For the second level decomposition the number of bands reduced to half that of the previous decomposition level and also the peaks and valleys also have been reduced (as seen from figure 5.6). The band values at 41-51 for Daub2_1 transformation is compared with the equivalent region between 21 and 25 from Haar_2. Here the class corn-min is having less discrimination with the others leads to mis-classification. In overall performance Daub2_1 feature extraction retrieves information better than other feature extraction techniques.

5.4.1.2 FERM Based Accuracy Assessment for LMM

The sub pixel classification accuracy assessment for LMM has been compared with the corresponding SVM classification. It means, Daub2_1 feature extraction performs better for per pixel classification, hence the LMM has been applied on the same transformed dataset by Daub2_1 feature extraction. Also the two solutions by LMM, constrained and unconstrained solutions are almost equal in extracting information, only the constrained solution has been taken. The fraction images for each class have been shown in Figure 5.7. Also the user's and the producer's accuracies for SVM and LMM classified comes from Daub2_1 reduced have been given in Table 5.5. Few observations have been made on the output from LMM.

1. In all the fraction images the domination of granules/noise is more. Since there is a natural variation to almost all materials, a certain degree of variability to exist for any materials that would be selected as end-members for a particular hyperspectral image it effects in classification by LMM. Since in LMM the solution has been obtained by solving linear equations, for each variable (proportion of class) certain fraction is allotted to every class leads to granules.
2. Classes which are small or having less number of pixels, like alfalfa, grass/pasture-mowed, and oats are completely hidden because of the presence of granules.
3. There are no separate classes for railway track, metallic road and non-metallic road due to the non-availability of pure pixels for them. Hence they are classified to the existing class. Since the image is covered by vegetation and among the available 16 classes, there are 15 classes of vegetation and only one class is non-vegetation, stone-steel towers, whose area cover is very less. Now the spectral curve of railway track, metallic road and non-metallic road are not resembled with vegetation while, they mostly resemble with stone-steel tower, they are classified as stone-steel tower. Also, because of the spatial resolution is 20m, those three classes are mixed with vegetation, leads to mixture of granules.
4. The areawise comparison of SVM classified and LMM constrained solution of dataset II has been given in Figure 5.8 and Table 5.4. The classes 3, 4, 5, 6, 14 (corn-min, corn, grass/pasture, grass/trees and woods, respectively) show much deviation in mapping between SVM classification and LMM constrained solution. Since the class 14 – woods cover large area and have boundary with many other classes, the per pixel classification forced the boundary pixels to classify into neighbouring classes leads to reduction in the area. But since the LMM gives fractional abundance, such mixed pixels give fractional proportion and add-up to increase area of woods. Also since the pair corn-min and corn are having with-in

class variation there is a big classification difference between the two above said classifiers. Also the same explanation follows for grass/pasture and grass trees.

Table 5.4: Areawise comparison of dataset II classified by LMM (constrained solution) and SVM, where the area is scaled to number of classes.

Classifier / Class	Alfalfa	Corn-notill	Corn-min	Corn	Grass/pasture	Grass/trees	Grass/pasture-mowed	Hay-windrowed	Oats	Soy-notill	Soy-min till	Soy-clean	Wheat	Woods	Bldg-grass-trees-drives	Stone-steel towers
LMM	0.36	0.69	1.1	0.82	0.33	0.33	0.34	0.55	0.34	1	1.01	0.56	0.16	0.39	0.31	0.1
SVM	0.05	0.58	0.53	0.21	1	0.89	0.03	0.25	0.2	0.59	1.07	0.39	0.11	1.55	0.87	0.01

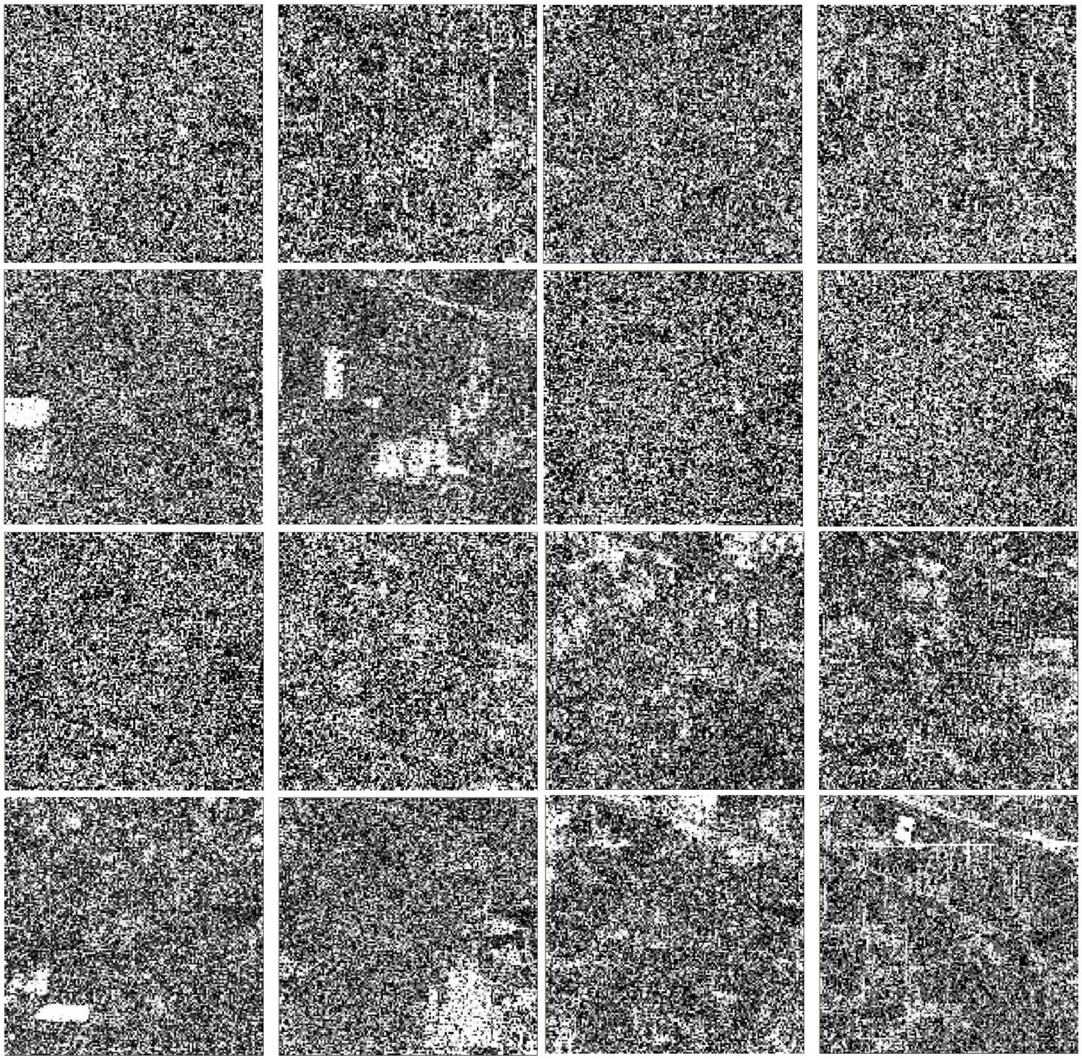


Figure 5.7: Fraction images for all the 16 classes of dataset AVIRIS over Indiana Pine.

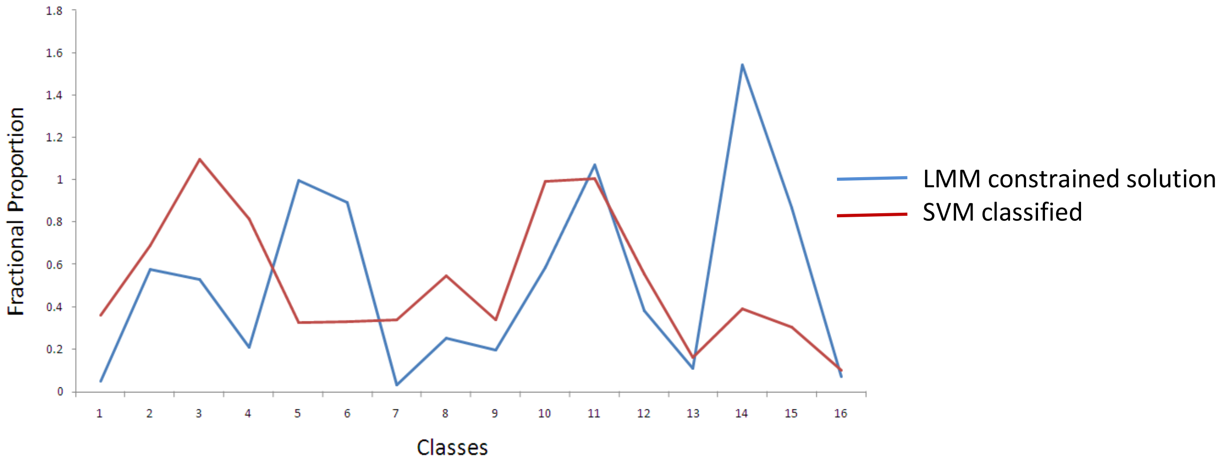


Figure 5.8: Area wise comparison of LMM classified constrained solution of feature extracted dataset II along with SVM classification solution.

Table 5.5: User's and producer's accuracies for classification by SVM and LMM (constrained solution) of dataset II comes from Daub2_1 feature extraction technique.

Sl. No.	**Class**	**Daub2_1 - SVM**		**Daub2_1 – LMM Constrained**	
		User's Accuracy	**Producer's Accuracy**	**User's Accuracy**	**Producer's Accuracy**
1	Alfalfa	100	100	75.4	67.6
2	Corn-notill	100	100	63.1	71.7
3	Corn-min	100	100	70	69.8
4	Corn	100	100	66.6	72.5
5	Grass/pasture	100	100	68.6	60.8
6	Grass/trees	100	100	63.4	63.6
7	Grass/pasture-mowed	100	100	71.3	61.9
8	Hay-windrowed	100	100	71.2	68.5
9	Oats	100	100	63.6	64.7
10	Soy-notill	100	100	70	68.6
11	Soy-min till	100	100	66.4	69.3
12	Soy-clean	100	100	60.7	68.2
13	Wheat	100	100	71.1	55.1
14	Woods	100	100	70.8	63.8
15	Bldg-grass-trees-drives	100	100	64.3	67.4
16	Stone-steel towers	100	100	67.9	55.9
	Average	100	100	67.78	65.59

5.4.1.3 Extraction of Small Classes via ICAMM

To extract information for which training pixels are not available or less number of training pixels are available or the class is covered by small area, unsupervised classification may be performed. Here, few observations regarding classification by

ICAMM have been made. The main aim of using ICAMM is to extract information from data which are independent from other classes.

1. The class stone-steel tower is the only non-vegetation class present in the dataset. The same class is available at few other places (circled in red) which are not perfectly classified by LMM (Figure 5.9) due to granules but has been classified well by ICAMM (Figure 5.9).
2. By applying Daub2_1 transform on dataset II and then classified by ICAMM, the objects metallic and non-metallic roads have been extracted (Figure 5.10).
3. By applying Daub2_2 transform on dataset II and then classified by ICAMM, the objects metallic and non-metallic roads have been classified as class stone-steel towers (Figure 5.11). Since, all the other classes are vegetation the above said objects are mapped as non-vegetation class.
4. There is no or less number of granules present in ICAMM classified images (Figure 5.9) compared to LMM classified images (Figure 5.7).

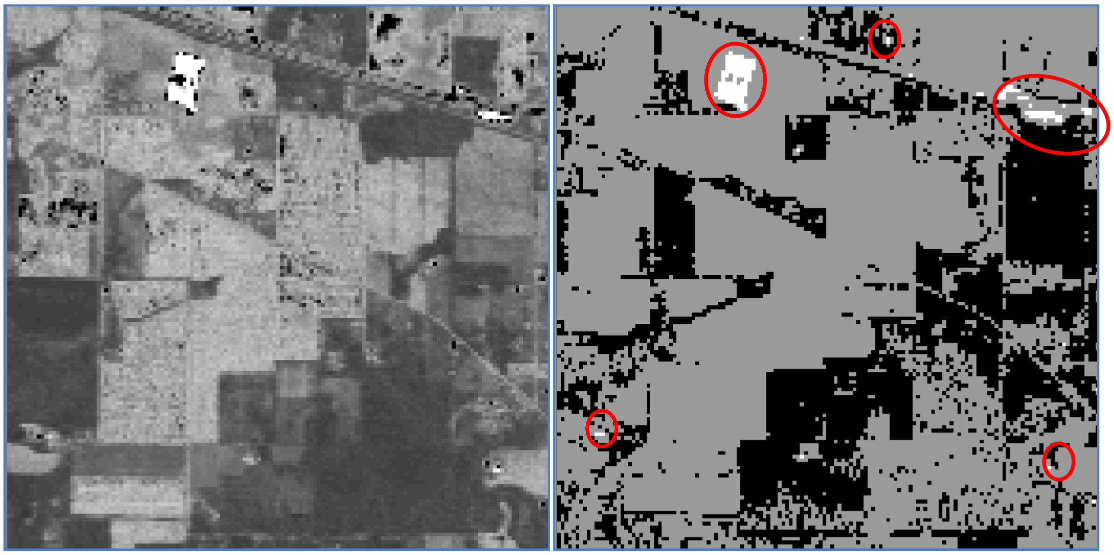


Figure 5.9: The extraction of class stone-steel towers by ICAMM two different iterations applied on Haar_1 feature extraction.

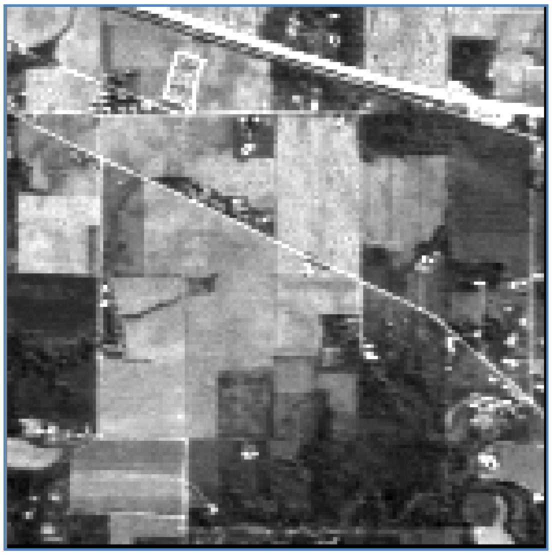


Figure 5.10: The extraction of metallic and non-metallic roads by ICAMM applied on Daub2_1 feature extraction.

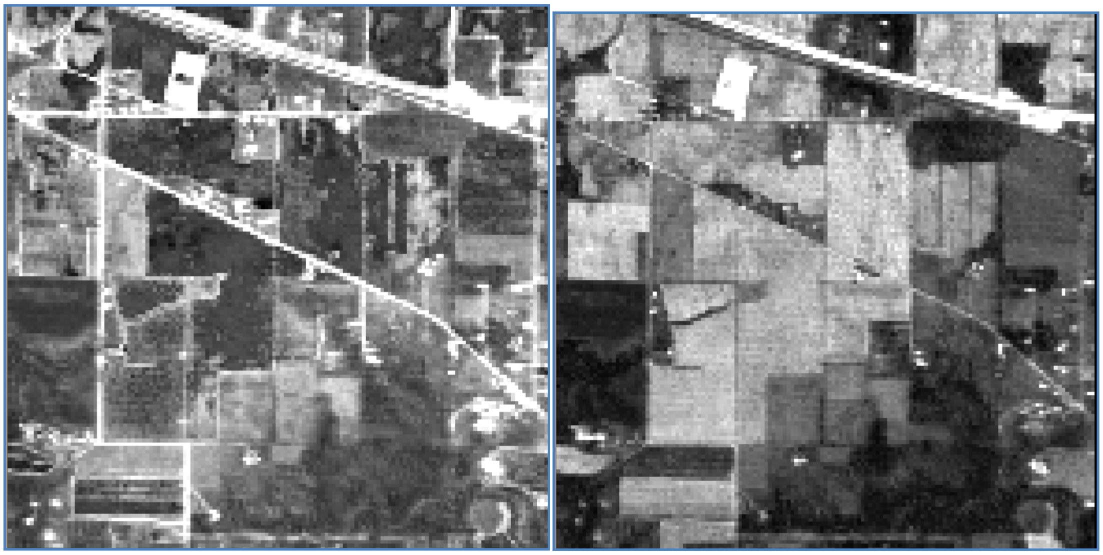


Figure 5.11: The extraction of metallic and non-metallic roads which are classified as stone-steel towers by ICAMM applied on Daub2_2 feature extraction.

5.4.2 Observations and Result for Experimental Dataset III (HYPERION – Roorkee and its Surroundings Dataset)

Few observations based on classification by SVM, LMM and ICAMM has been made and they are measured by various measuring techniques described as follows:

5.4.2.1 Error Matrix Based Accuracy Assessment for SVM

1. The overall classification accuracy for Haar level-1 decomposition followed by SVM classification is better than the other feature extraction techniques. For comparison of the Haar_1 feature extraction, we used Daub4_2 feature extraction

which gives poor classification accuracy. Also we have compared with the classification of original dataset (Figure 5.13).

2. For Haar_1 feature extraction followed by SVM classification of extracted dataset, 33 pixels have been classified as sand. Among this, only 28 pixels need to be classified as sand and it has been done. Hence the producer's accuracy is 100 percent (Table 5.6, Figure 5.12) while among the remaining 5 pixels which are classified as sand, 2 pixels are from barren land and 3 pixels are from urban leads to poor user's accuracy for sand. Since the class is small and hence the testing samples are small in size, only the user's accuracy has been affected but not the overall classification accuracy.
3. The pre-processed dataset also has been classified by SVM and compared with the extracted result. Here, 52 pixels have been classified as sand instead of 28 pixels which leads to poor user's accuracy for sand. Also 18 pixels for barren land which are at reference data have been classified as sand leads to poor producer's accuracy for barren land. Also among the 66 pixels from reference for water, 9 pixels have been classified as urban leads to only 84.8% of producer's accuracy for water. These 9 pixels together with 20 more pixels from barren land in the reference lead to poor user's accuracy for urban.
4. In Figure 5.14, the spectral curves for the three classes barren land, sand and urban after pre-processing, Haar_1 decomposition and Daub4_2 decomposition have been given and the discrimination between the curves may be compared.

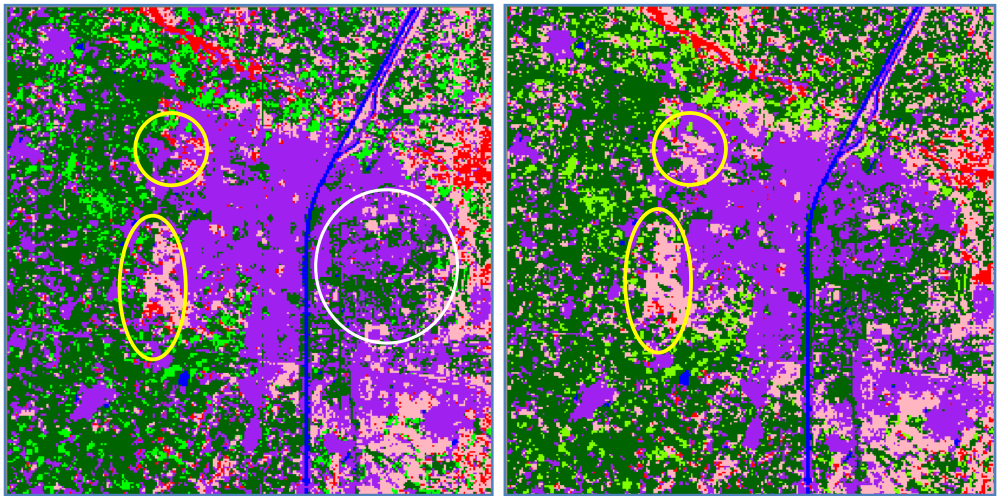


Figure 5.12: (a) SVM classified pseudo image got from Haar_1 feature extraction of HYPERION dataset of Roorkee and its surroundings, (b) SVM classified pseudo image got from Daub4_2 feature extraction of HYPERION dataset of Roorkee and its surroundings.

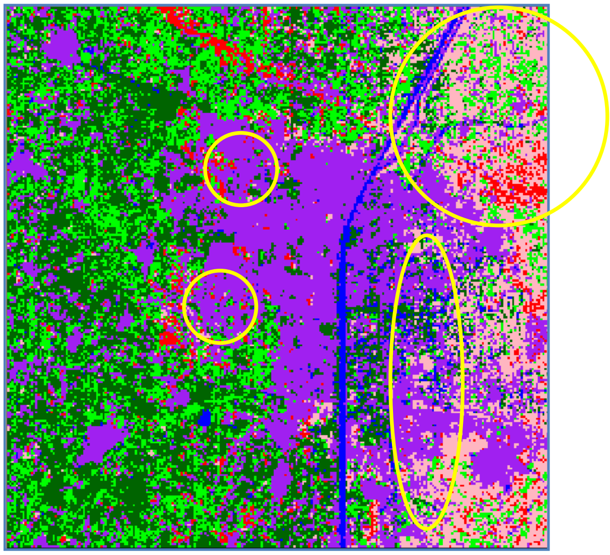
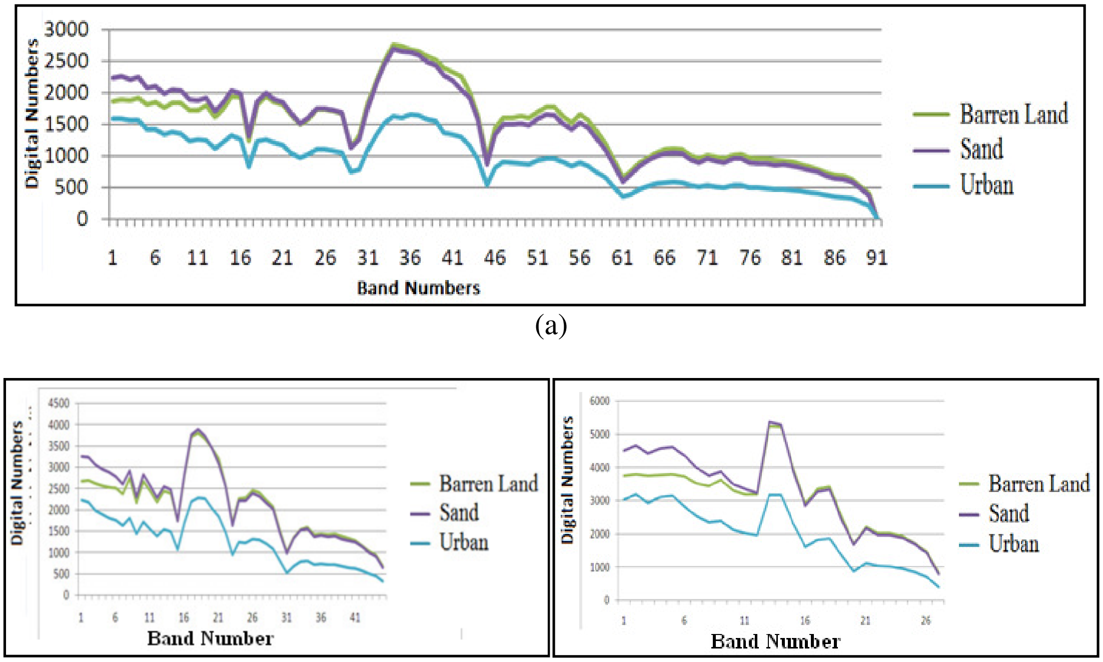


Figure 5.13: SVM classified pseudo image got from pre-processed HYPERION dataset of Roorkee and its surroundings.



(a)

(b)

(c)

Figure 5.14: Spectral curves for three classes barren land, sand and urban after (a) pre-processing, (b) Haar_1 decomposition and (c) Daub4_2 decomposition.

Table 5.6: User's and producer's accuracies for classification by SVM of dataset II comes from Haar_1, Daub4_2 feature extraction techniques and for original dataset.

Class	Haar_1		Daub4_2		Original Image	
	User's Accuracy	**Producer's Accuracy**	**User's Accuracy**	**Producer's Accuracy**	**User's Accuracy**	**Producer's Accuracy**
Trees	98.37	100	92.188	97.79	94.253	90.608
Vegetation	98.305	98.305	94.915	94.915	61.364	91.525
Barren Land	100	99.038	92.157	90.385	100	50
Sand	96.552	100	84.848	100	51.923	96.429
Urban	99.785	99.358	99.78	97.002	93.996	97.216
Water	100	98.485	100	98.485	100	84.848
Average	**98.84**	**99.2**	**93.98**	**96.43**	**83.59**	**85.1**

5.4.2.2 FERM Based Accuracy Assessment for LMM

The sub pixel classification accuracy assessment for LMM has been compared with the corresponding SVM classification. It means, here Haar_1 performs better for per pixel classification, hence the LMM has been applied on the same transformed dataset by Haar_1. Also the two solutions by LMM, constrained and unconstrained solutions are almost equal in extracting information, only the constrained solution has been taken. The fraction images for each class have been shown in Figure 5.15. Few observations have been made on the output from LMM given as follows:

1. In all the fraction images the domination of granules/noise is more. Since there is a natural variation to almost all materials, a certain degree of variability exist for any materials that would be selected as end-members for a particular hyperspectral image it effects in classification by LMM. Since in LMM the solution has been obtained by solving linear equations, for each variable (proportion of class) certain fraction is allotted to every class leads to granules.
2. The two classes trees and vegetation have spectral curves more similar to each other while the classes barren land, sand and urban have spectral curves more similar to each other.
3. The spatial resolution of the dataset is 30 m and hence a large number of mixed pixels are there. This shows effect in classification, particularly a huge variance may be expected between per pixel and sub pixel classification.
4. Generally, misclassification occurs when spectral signatures of two classes are nearly similar, but while comparing sub pixel classification output with per pixel classification output for dataset III, the area occupied by the classes in each of the

pixels could affect. In Figure 5.17, the circular portion indicates the region where urban and trees are mixed together leads to the pixels are mixed. In SVM, these pixels are classified as urban (represented by white coloured circle in Figure 5.12(a)) hence it leads to increase in area cover of urban when compared to LMM output (Table 5.8 and Figure 5.16). Also, class sand is mapped as class urban in SVM classification. So the area cover of urban increases while for sand it decreases.

5. Table 5.7 gives the users' and the producers' accuracies for classification by SVM and LMM (constrained solution) of dataset II comes Haar_1feature extraction technique.

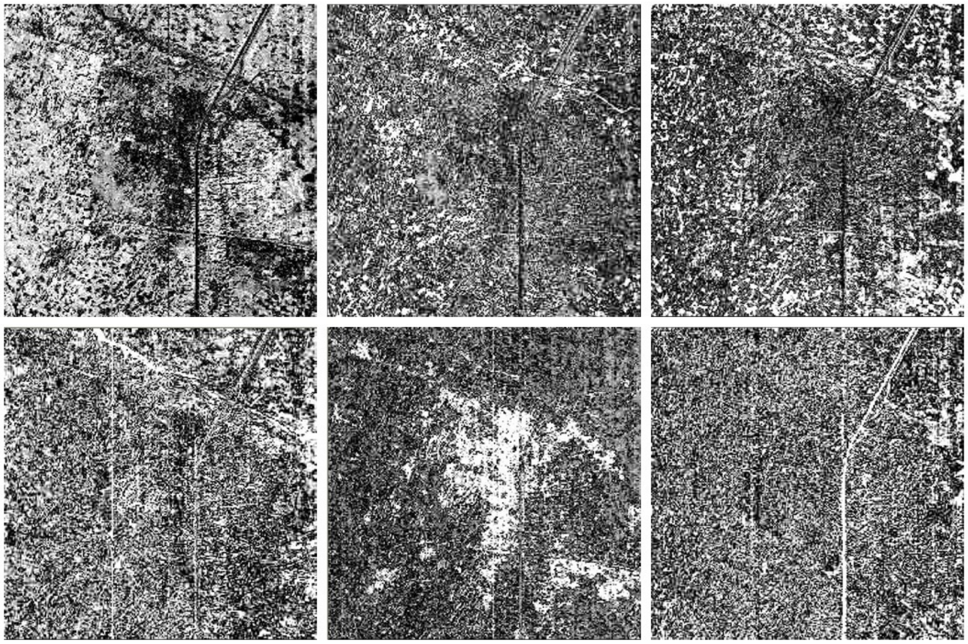


Figure 5.15: Fraction images for all the 6 classes of dataset HYPERION over Roorkee and its surroundings comes from constrained solution of LMM from Haar_1 feature extraction.

Table 5.7: User's and producer's accuracies for classification by SVM and LMM (constrained solution) of dataset II comes Haar_1feature extraction technique.

Class	**Haar_1 – SVM**		**Haar_1 – LMM Constrained**	
	User's Accuracy	**Producer's Accuracy**	**User's Accuracy**	**Producer's Accuracy**
Trees	98.37	100	56.7	63.5
Vegetation	98.305	98.305	69.7	57.9
Barren Land	100	99.038	54.2	73.1
Sand	96.552	100	70	59.7
Urban	99.785	99.358	65.2	62.4
Water	100	98.485	70.6	57
Average	**98.84**	**99.2**	**64.4**	**62.27**

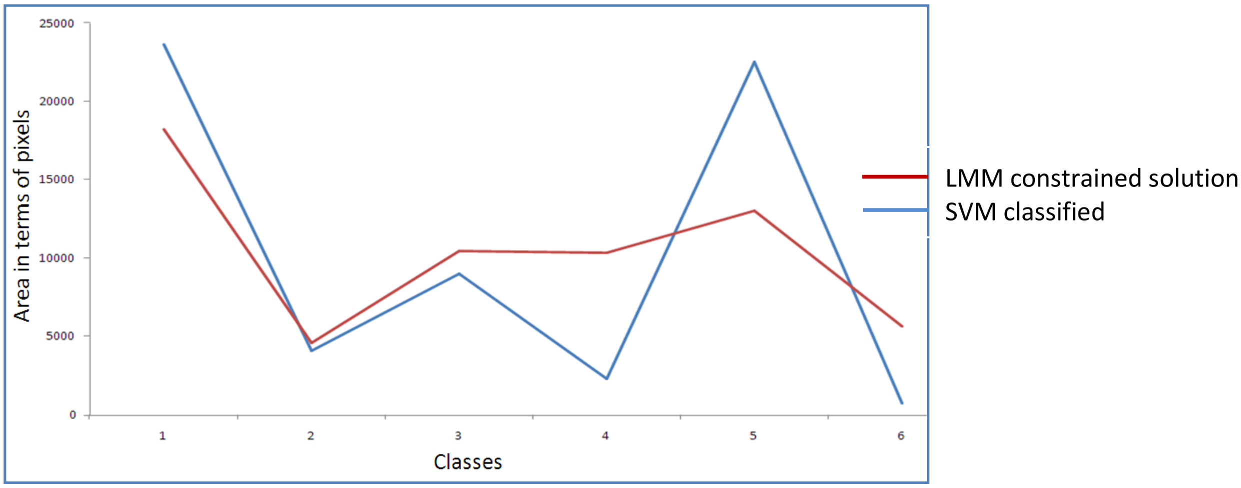


Figure 5.16: Area wise comparison of LMM classified constrained solution of Haar_1 feature extracted dataset III along with SVM classification.

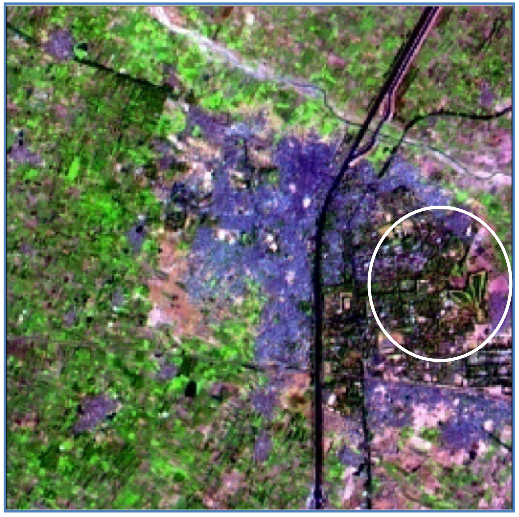


Figure 5.17: The FCC of Roorkee and its surrounding area by Hyperion sensorp (Red: Band 100, Green: Band 30, Blue: Band 20).

Table 5.8: Areawise comparison of dataset III classified by LMM (constrained solution) and SVM, where the area is scaled to number of classes.

Classifier Class	Trees	Vegetation	Barren Land	Sand	Urban	Water
LMM	18266.62	4614.2	10493.47	10366.52	13065.97	5688.62
SVM	23659	4149	9056	2324	22569	743

5.4.2.3 Extraction of Small Classes via ICAMM

As mentioned in the result analysis of dataset II in section 5.4.1.3, here for dataset III also the granules are not there in extraction of features from hyperspectral data from ICAMM. Few observations and analysis have been made for this dataset are given as follows:

1. The class water has been mapped clearly compared to LMM constrained solution. Particularly, the canal which is a linear structure has been mapped perfectly by ICAMM for which Haar_1 feature extraction technique has been used.
2. ICAMM is capable for extraction of other water bodies like small lakes (highlighted by red circles) in Figure 5.18(a).
3. The Golf course made up of grass land which is a small object has also been classified (Figure 5.18(c), highlighted by red coloured circle).
4. Even though the class vegetation covers the entire image, it has distinguished from class trees (Figure 5.18(d)).

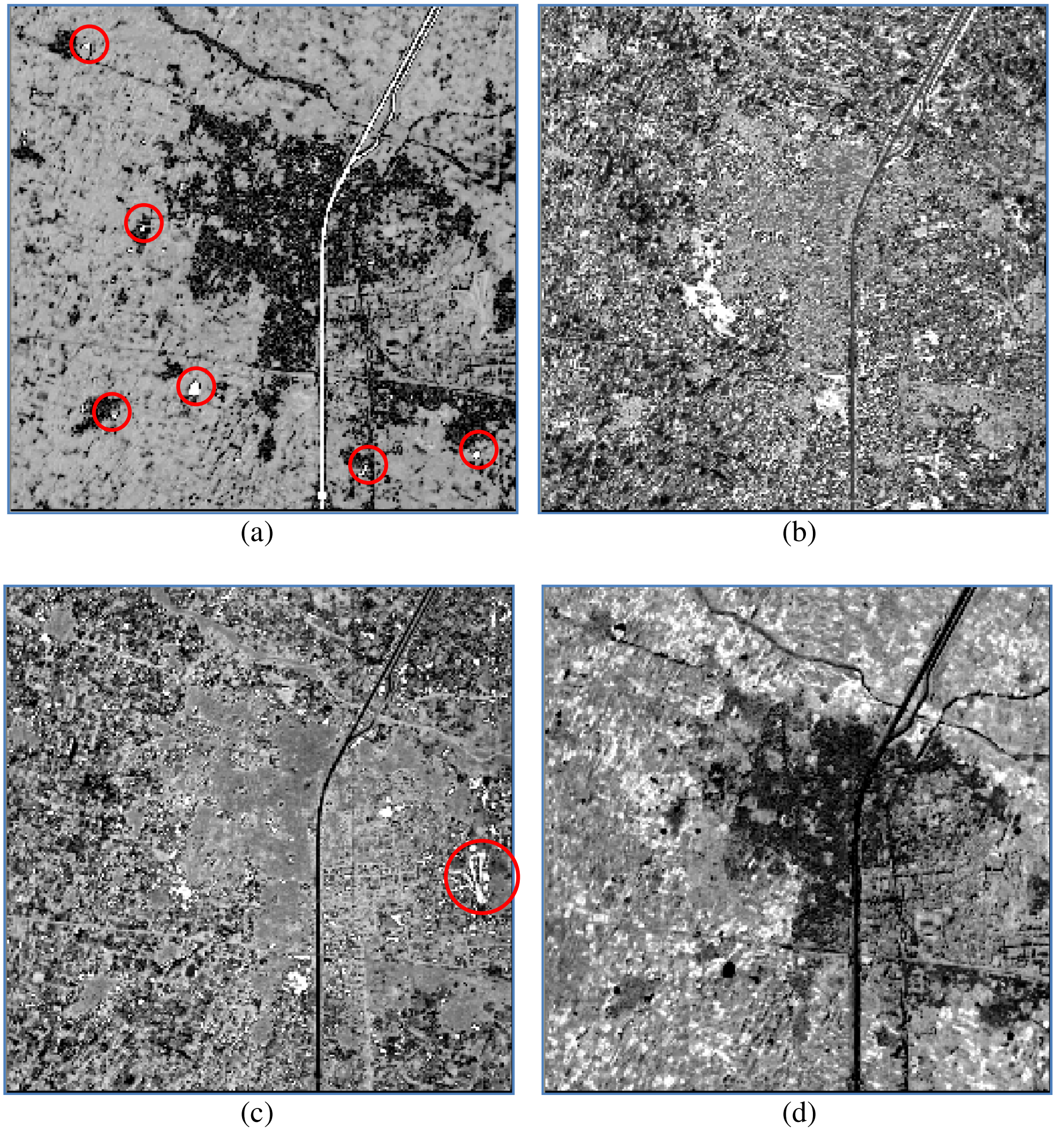

(a) (b) (c) (d)

Figure 5.18: Fraction images obtained by classifying Haar_1 feature extraction of dataset III by ICAMM (a) water, (b) barren land, (c) a new object golf course (made up of grass) and (d) vegetation.

5.4.3 Observations and Result for Experimental Dataset IV (AVIRIS – San Diego Naval Dataset)

Few observations based on classification by SVM, LMM and ICAMM has been made and they are measured by various measuring techniques described as follows:

5.4.3.1 Error Matrix Based Accuracy Assessment for SVM

1. The overall classification comes from Haar 2nd level decomposition followed by SVM classification is better than other feature extraction techniques. Comparatively, Daub4_2 feature extraction is poor than Haar_2 feature extraction. Hence an assessment comparing Haar_2 and Daub4_2 has been made.
2. In Haar_2, one of the three aircrafts has been mapped clearly, but unfortunately, due to the non-availability of pure pixels for this object it is not considered as a separate class. Even though, by this observation we may not conclude which feature extraction is better, we are interested in analysing the mapping of aircraft. Here, the aircraft has been classified as building type 2 while the same has been classified as building type 3 (few pixels) and building type 2 (remaining pixels) in Daub 4, 2nd level decomposition.
3. Also, in Daub4_2, few pixels from the class building type 3, have been misclassified as building type 2.
4. Few other small objects like cylindrical drum, small aircrafts, etc., have also been classified to different classes while using different feature extraction techniques.
5. In the classification of original data, one of the aircrafts has been classified into building type 3 and more granules appears for the class road type 1 (Figure 5.19) (high).
6. In the classification of original data, 10 pixels have been misclassified into road type 1, instead of road type 2 leads to poor user's accuracy of road 2 and poor producer's accuracy of road type 1.

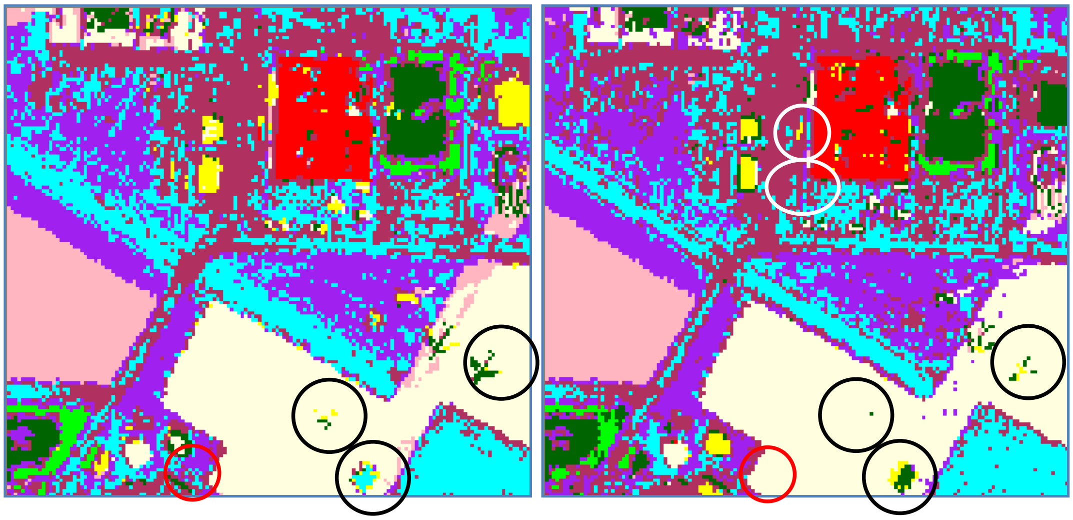


Figure 5.19: (a) SVM classified pseudo image got from Haar_2 feature extraction of AVIRIS dataset of San Diego region, (b) SVM classified pseudo image got from Daub4_2 feature extraction of AVIRIS dataset of San Diego region.

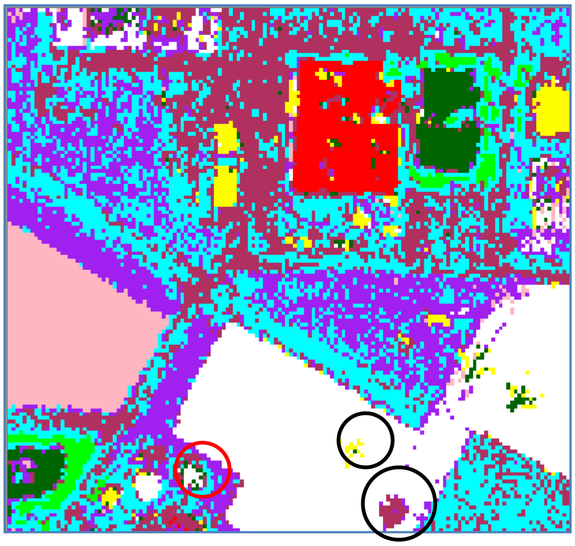


Figure 5.20: SVM classified pseudo image got from classification of original dataset IV, AVIRIS dataset of San Diego region.

5.4.3.2 FERM Based Accuracy Assessment for LMM

The sub pixel classification accuracy assessment for LMM has been compared with the corresponding SVM classification. It means, here Haar_2 feature extraction followed by SVM classification performs better for per pixel classification, hence the LMM has been applied on the same transformed dataset by Haar_2 feature extraction. Also the two solutions by LMM, constrained and unconstrained solutions are almost equal in extracting information, only the constrained solution has been taken. The fraction images for each class have been shown in figure 5.21. Few observations have been made on the output from LMM given as follows:

1. Unlike other fraction images of dataset II and dataset III, here the LMM classification is able to produce better fraction images. This is due to few factors which involves spatial resolution (here it is 4 m), crispy nature of classes, etc. Among the nine classes, class vegetation is completely differs from the other classes and we have two types of roads, three types of concrete surfaces and three types of buildings. It is expected to have with-in class variation.
2. None of the fraction images contain the object aircraft. Because of the solution of LMM has been obtained by solving linear equations, for each pixel of object aircraft a proportion has been goes to each class, because of the non-availability of pure pixels for aircraft.
3. The areawise comparison of SVM classified and LMM classified outputs (Table 5.10) shows that almost all the corresponding classes are classified well, except for three classes, road type 1, concrete 2 and concrete 3.
4. Mixed pixels are mapped as mixed pixels by using LMM while they are mapped as pure pixels by SVM. It is observed in the fraction images of concrete 1 and concrete 2, where actually they are mixed pixels and mapped as mixed pixels (Figure 5.21) (highlighted as red coloured circles) whereas these pixels are mapped as pure pixels in SVM classification (Figure 5.19(a) and Figure 5.20).
5. Table 5.9 shows the users' and the producers' accuracies for classification by SVM and LMM (constrained solution) of dataset III comes Haar_2 feature extraction technique in which most of the classes are classified well by SVM as the two accuracies are 100 percent in those cases.

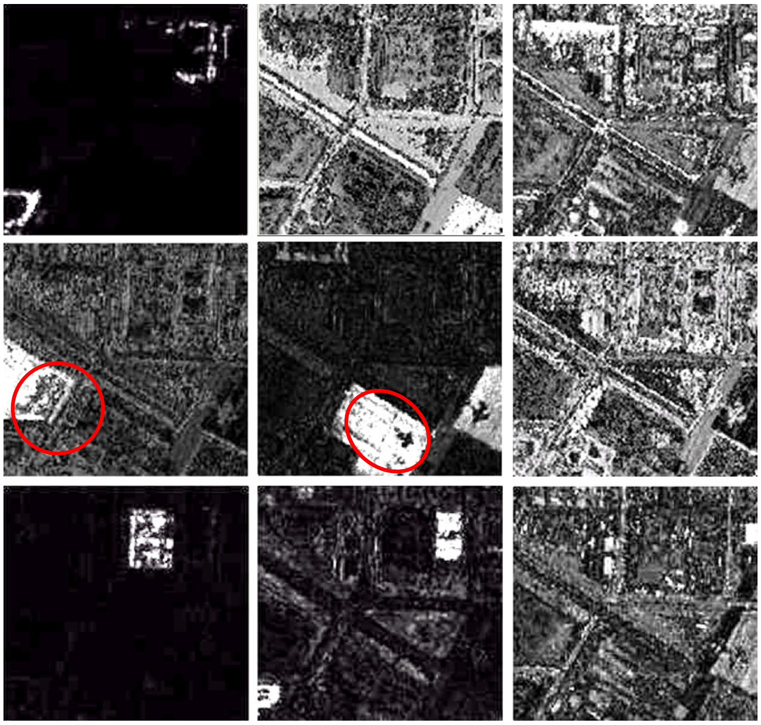


Figure 5.21: Fraction images for all the 9 classes of dataset IV AVIRIS over San Diego region, comes from constrained solution of LMM from Haar_2 feature extraction.

Table 5.9: User's and producer's accuracies for classification by SVM and LMM (constrained solution) of dataset III comes Haar_2 feature extraction technique.

Sl. No.	**Class**	**Haar_2 – SVM**		**Haar_2 – LMM Constrained**	
		User's Accuracy	**Producer's Accuracy**	**User's Accuracy**	**Producer's Accuracy**
1	Vegetation	100	100	27.6	7.2
2	Road Type 1	97.917	97.917	63	74.1
3	Road Type 2	95.349	95.349	62	65.7
4	Concrete Type 1	100	100	56.6	57.8
5	Concrete Type 2	100	100	49.1	38.2
6	Concrete Type 3	100	100	61.4	75.6
7	Building Type 1	100	100	66.6	12.5
8	Building Type 2	100	100	56.8	37.2
9	Building Type 3	100	100	66.3	47.7
	Average	**99.25**	**99.25**	**56.6**	**46.22**

Table 5.10: Areawise comparison of dataset IV classified by LMM (constrained solution) and SVM, where the area is measured in number of pixels.

Classifier / Class	Vegetation	Road Type 1	Road Type 2	Concrete Type 1	Concrete Type 2	Concrete Type 3	Building Type 1	Building Type 2	Building Type 3
LMM	349	4472	4448	1815	3950	4000	808	774	384
SVM	75.21	6068.33	4079.31	1919.66	1447.97	5161.34	168.47	638.93	1393.7

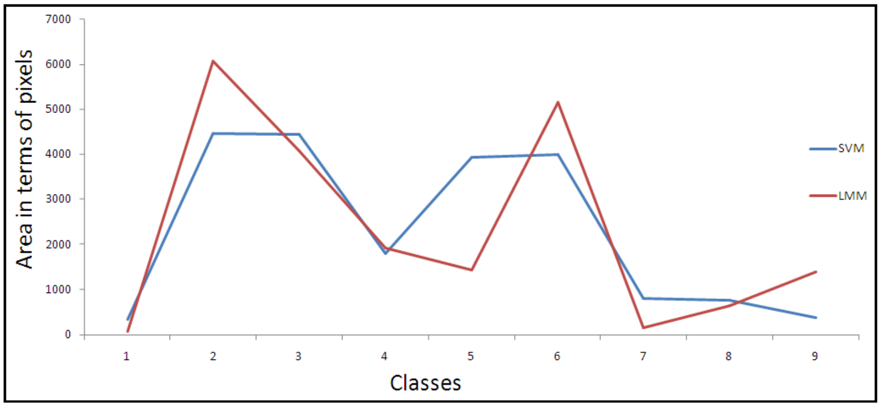


Figure 5.22: Area wise comparison of LMM classified constrained solution of Haar_1 feature extracted dataset III along with SVM classification.

5.4.3.3 Extraction of Small Classes via ICAMM

Some small classes for which either pure pixels may not be available or less number of pure pixels are available, but such classes are useful in applications, may not be classified using SVM and LMM. ICAMM, an unsupervised classifier detects such useful information from dataset IV. Few of the observations and result are given as follows:

1. In dataset IV, the object aircraft has not at all classified by the other two above mentioned classifiers, but ICAMM is able to map well (Figure 5.23). Not only aircraft but also few vehicles which occupy in sub pixels have also been classified (highlighted by red coloured circle in Figure 5.23).
2. No or less number of granules present in ICAMM classified image.
3. Since the aircrafts, the vehicles and the cylindrical drums may be made up of metals these forms a cluster in ICAMM classification. The two small rectangular buildings (highlighted as yellow coloured circle in Figure 5.23) also made up of metals and hence all these objects are classified as single class.

4. Area wise comparison (Figure 5.22) of the SVM and LMM constrained solution comes from Haar_1 shows a significant difference in classification of the two classes road type 1 and concrete 2.

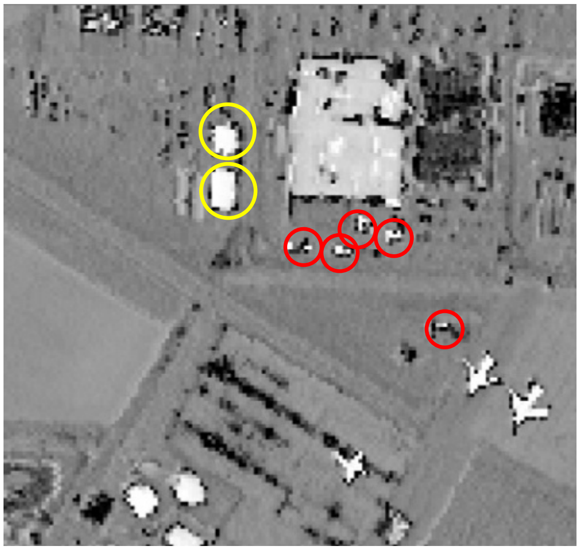


Figure 5.23: Extraction of small objects by ICAMM for dataset IV by AVIRIS sensor over San Diego region.

5.6 Summary

In this chapter, a brief description of the per pixel classification algorithm SVM and sub pixel classification algorithms LMM and ICAMM have been given. The mathematical concepts and the derivation of these algorithms give an insight to the information extraction for hyperspectral data in the name of classification. The SVM is supervised, per pixel classifier which requires no statistical details of the data classified every pixel into single class. Then sub pixel classification has been performed by one supervised classifier (LMM) and another by unsupervised classifier (ICAMM). Based on the classifiers we obtain the following results:

(i) SVM classification has been performed on all the three datasets which comes from feature extraction. Here only the first and second level decomposition of the original data has been made by Haar and Daubechies transforms.

(ii) Haar and Daubechies sub class 2 wavelets produce better classification accuracy for all the datasets.

(iii) Accuracy assessment for SVM classification has been performed by error matrix based accuracy assessment while that for LMM classification, fuzzy error matrix (FERM) has been performed.

(iv) Due to the non-availability of mixed pixels for reference, the reference data has been taken from best result of SVM classification for each dataset.

(v) The noise present in fraction images by LMM solution is due to the mathematical phenomenon of LMM.

(vi) The ICAMM is capable to produce noise less classes and objects/classes which are having no or minimum number of pure pixels.

Chapter 6

Spatial Enhancement of Information Extraction via Super Resolution Mapping Techniques

6.1 Introduction to the Problem

In the previous chapters, results from experimental investigations on two major tasks related to hyperspectral imaging, namely, the feature extraction and the classification for extraction of information have been reported, analysed and discussed. The feature extraction has been performed by two methods, wavelet based feature extraction and PCA based feature extraction. Both per pixel and sub pixel classifications have been examined for extraction of accurate information of hyperspectral via support vector machines, LMM and ICAMM. Although, sub pixel classification extracted better information than per pixel classification, the spatial arrangement of the sub pixels inside a pixel was lacking. Thus, while the areal extent of the classes within a pixel may be accurately estimated through sub pixel classification, their spatial location on ground may not be inferred. This problem can be overcome by applying super resolution mapping, which, in the literature few definitions exist which are given as follows:

"Mapping land cover at a spatial resolution finer than the size of the pixel of the image is called super resolution mapping" – (Kasetkasem *et al.*, 2005)

"Super resolution mapping is a set of techniques to increase the spatial resolution of a land cover map obtained by soft classification methods" (Genitha and Vani, 2010)

Hence super resolution mapping may be defined as "the technique of producing a fine spatial resolution data from the coarse spatial resolution data by dividing the pixels into smaller size".

The super resolution may be achieved by dividing each of the pixels of a hyperspectral data into equal number of sub pixels and filling each sub pixel by one and only one class. Generally, the output from a sub pixel classification (e.g., class proportion or fraction abundance) within a pixel is considered to define the number of sub pixels to be mapped in that pixel based on a pre-defined zoom factor to enhance the spatial resolution.

Thus, with the help of information content in the neighbouring pixels and the fraction abundance, the super resolution mapping may be achieved via one of the many optimization techniques.

6.2 Need for Study

Super resolution mapping is an emerging area in remote sensing, particularly in hyperspectral imaging. Often, targets and classes fall within a pixel and the exact location of these may be desired in many applications such as military target detection, crop type identification, coastal boundary extraction (Niedermeier *et al.*, 2000), road and other linear feature extraction etc. Thus, finding the exact location of a target or the boundary of a linear feature from remote sensing data has been a problem to be investigated. It is anticipated that super resolution mapping may be quite useful in solving this problem, which is the main objective of this chapter. Current super-resolution mapping methods include the Hopfield neural network (HNN) (Tatem *et al.*, 2001), linear optimization (Verhoeye and Wulf, 2002), genetic algorithm (Mertens *et al.*, 2003; Siedlecki and Sklansky, 1989), feed-forward neural network (Mertens *et al.*, 2004), Markov random field (Kasetkasem *et al.*, 2005), pixel swapping (Atkinson 2005, Thornton *et al.*, 2007), simulated annealing (Makido *et al.*, 2007), inverse Euclidean algorithm (Tiwari *et al.*, 2007) and geostatistical methods (Villa *et al.*, 2011). The geostatistical approach described in (Villa *et al.*, 2011) has been studied for mixed pixels mixes with only two classes. Here the algorithm has been applied on a region where sea meets the shore. So this is almost considered as linear boundary classification problem. So a general super resolution mapping algorithm is required which maps mixed pixels mixed with two or more classes. The main limitation of spatial regularization by simulated annealing (Thornton, 2006) is it is incapable of mapping linear features. Since the objective function used in this algorithm is the cost function to be minimized, where the cost function here is the perimeter of the areas belonging to the same class. So, error in one pixel will be carried over to the other pixels also. So an algorithm in which the super resolution mapping of one pixel does not affect the super resolution mapping of the other is to be considered. The drawback of the spatial pixel swapping algorithm (Shen *et al.*, 2009) is it is also incapable of super resolving linear pixels. In case of target detection using pixel swapping algorithm (Arora and Tiwari, 2013) the binary class problem may be extended to multi-class problem. Collectively, by considering the limitations of the above said super resolution mapping algorithms, we propose an algorithm, based on inverse Euclidean distance, which works better for pixels having linear boundary and multi-class classification problem.

In this research, a novel algorithm has been proposed by fulfilling few of the limitations exist in other algorithms. The proposed algorithm works for any number of classes is the big advantage over few other super resolution mapping algorithms. The mapping of small classes has also been taken with much care to avoid confusion between small classes and noise. Generally, in other super resolution mapping algorithms, classes having few pixels may be considered as noise and may be removed, but in the proposed algorithm every class is taken as equal importance. By finding weights to the sub pixel locations the filling process has been done. So introduction of new noise was highly impossible. Since the super resolution mapping algorithm is new, a dataset with known pixel values is required and hence a synthetic data of size 45x60 has been created.

6.3 Super Resolution Mapping via Pixel Swapping Algorithm

Atkinson (2001) proposed a super resolution mapping by calculating membership values of each sub pixel locations within a pixel of the hyperspectral data. Based on the membership value of two sub pixels, the class label on it may be interchanged or swapped and hence it got the name Pixel Swapping Algorithm. Super resolution mapping takes information about fraction abundance of all neighbouring pixels of the pixel to be super resolved (PTS) and maps them according to a pre defined zoom factor (defined by user, generally an odd number like, 3, 5, etc.,) which gives a spatial resolution finer than the pixel resolution of the image.

The literature suggests that initially, the pixel swapping algorithm (Atkinson, 2001; Atkinson, 2005) has been implemented on simulated data and on some satellite images for extraction of only binary land cover classes or single target detection. Later, Thorton (2006) used this algorithm for mapping land cover proportions for each pixel obtained from a soft classification and mapped to fewer classes for real satellite datasets.

In this algorithm, each pixel is divided into a pre-defined fixed number of sub pixels by a suitable zoom factor. Zoom factor governs the spatial resolution of the super resolved image. For instance, a 5 x 5 pixels zoom factor implies that soft output of land cover proportion for each single pixel needs to be mapped into 5 rows of sub pixels with 5 columns. Thus, a total of 25 sub pixels within each pixel are created. Similarly, a 7 x 7 pixels zoom implies a total of 49 sub pixels with 7 rows of sub pixels with 7 columns. Moreover, a soft proportion of 0.6 implies that 60% of the total zoomed pixels *i.e* 25 x 0.6=15 pixels (in case of 5 x 5 pixels zoom) and 49 x 0.6=29.4 = 29 or 30 pixels (approximately, in the case of 7 x 7 pixels zoom). The zoom factor for each pixel remains fixed throughout the pixel swapping procedure. During this process, each sub pixel is allocated to a single land cover class such that the original class proportions in each pixel, (i.e., output from the soft classification) are maintained throughout the procedure. The implementation steps in brief are given as,

1. *Randomly allocate subpixels to binary classes based on pixel proportions.*

2. *For each iteration –*

(a) *For each pixel ;*

(i) *For each subpixel within the pixel –*

(aa) *Calculate attractiveness for each neighbouring subpixel within a window. The order of clique considered is 2 x 2 (see Figure 6.1). The attractiveness A_i of a pixel i is predicted as a distance weighted function of its j =1,2….j neighbours and is given by,*

$$A_i = \sum_{j=1}^{j} \lambda_{ij} z(x_j), \qquad (6.1)$$

where $z(x_j)$ is the (binary) class of the pixel at location x_j and λ_{ij} is a distance dependent weight predicted as,

$$\lambda_{ij} = exp\left(\frac{-h_{ij}}{a}\right), \qquad (6.2)$$

where, $-h_{ij}$ is the distance between the location x_j of a pixel i for which the attractiveness is desired, location x_j of a neighbouring pixel j, and a is non-linear parameter.

(ab) *Find minimum attractiveness A_i for all sub pixels currently allocated to 1(i.e., $A_i = min(A_i) \mid z(x_j) = 1$).*

(ac) *Find maximum attractiveness A_j for all sub pixels currently allocated to 0 (i.e., $A_j = min(A_j) \mid z(x_j) = 0$).*

(ii) *if $A_i < A_j$, then swap the single pair of sub pixel allocations.*

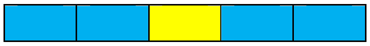


Figure 6.1: Two-pixel neighbourhood

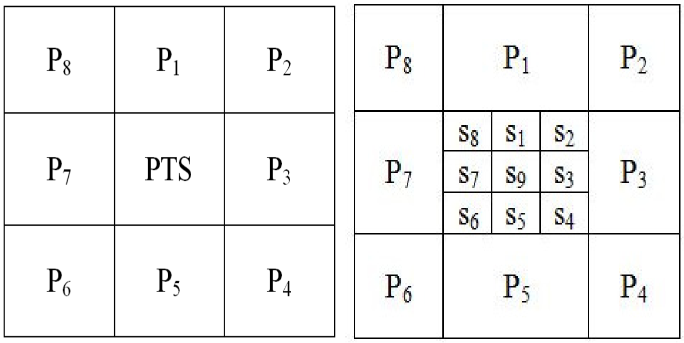
Two limitations have been stated by the proposer of the algorithm as well as the authors who used this algorithm (Atkinson, 2005; Thornton, 2006). First, the algorithm has been written to map linear features (such as hedgerows, paths, walls and fences) and hence this will work well for the case of linear features and that too when the width of the class occupies more than one pixel. Also, the algorithm fails to map features which are at sub pixel locations. The second limitation reported by the authors was, whenever there is mis-allocation of certain sub pixels, it carries the error till the end of the algorithm and hence cluster the misclassified pixels.

Apart from these limitations reported by the authors, there are few other issues, which are worth mentioning. The algorithm commences with random allocation of sub pixels depending upon the fraction abundance of each pixel. For example, if the fraction abundance of a typical pixel is 60% and if the zoom factor is 5, then the pixel is divided into 5^2 = 25 sub pixels, among these, $\frac{60}{100} \times 25 = 15$, sub pixels belong to that class. These 15 sub pixels are randomly arranged in 25 sub pixel locations. There are ${}_{25}C_{15}$ ways of arranging these pixels in sub pixel locations. Not all the arrangements are going to produce same result at the end of the algorithm, and hence the solution to the problem of super resolution by pixel swapping may not be unique. In a binary classification problem, since the expected result in the final output is that, all the sub pixels of same class in each pixel should be clustered together, this random arrangement increases unnecessary complication. The requirement here is to seek the spatial distribution of all the sub pixels determined using their respective abundance fraction. Therefore, it may be appropriate to treat all sub pixels of a given fraction as one unit and iterated to get their correct spatial distribution.

Similarly, the algorithm uses a non-linear parameter and a variable number of iterations, neither of which can be determined deterministically for different zoom factors and may vary for different applications. Also for multi-class super resolution mapping, this method is not effective because of the swapping has been done between two classes finds optimum and when for another class if the swapping has been introduced then the optimality between the first two classes may be disturbed. Therefore, a new method based on filling the pixels at the required location has been proposed in this thesis, which may overcome the limitations of the pixel swapping algorithm.

6.4 Super Resolution Mapping via Pixel Filling Algorithm

Once the fraction abundance of each pixel or the number of sub-pixels of each class belonging to each pixel is known, the super resolution mapping is carried out. To achieve super resolution mapping, the number sub-pixels of each class in the neighbouring pixels should also known. Let the pixel to be super resolved is denoted by PTS and the 8-neighbourhood pixels be named from top and arranged in clockwise direction as top, top right, right, bottom right, bottom, bottom left, left and top left are denoted as P_1, P_2, … P_8, (Figure 6.2(a)). The description given here is for a zoom factor 3 but the algorithm can be appropriately modified for any odd order zoom factors. The PTS is divided into 9 sub pixels denoted as s_1, s_2,…s_9, (Figure 6.2(b)) taken in clockwise order from top sub-pixel. Here, the PTS which consists of 9 sub-pixels has been grouped into 3 unequal parts, (i) corner sub pixels, (ii) center sub pixel, (iii) remaining sub pixels other than those in (i) and (ii). For zoom factor 3, there will be 4 sub pixels of case (i), only one center sub pixel and 4 sub pixels of case (iii). The weights at each sub pixel location of PTS are determined.



(a) (b)

Figure 6.2: (a) The arrangement of PTS and its neighbouring pixels, (b) the sub pixels of PTS.

Case (i) Weights of corner sub-pixels

The corner sub pixels directly depend upon the 3 neighbours. For instance, s_8 depends upon P_7, P_8 and P_1. At the same time P_1 and P_8 are not in equal distance to s_8. Since the spatial proximity is considered, the influence of P_1 and P_7 is more than P_8 for location s_8 and therefore these are given different weights inversely proportion to the Euclidean distance. Thus, the weights for corner pixels will be half that of the middle pixel. The weights 8 and 17 have been calculated from the nature of the neighbouring pixels, and these are taken and applied to sub pixel s_8. So the weight of sub pixel s_8 will be,

$$s_8 = P_7 x17 + P_8 x8 + P_1 x17, \qquad (6.3)$$

Similarly for s_2, it depends upon P_1, P_2 and P_3 and the respective weights are 17, 8 and 17. Hence s_2 may be calculated by,

$$s_2 = P_1x17 + P_2x8 + P_3x17, \tag{6.4}$$

The other two corner sub-pixels s_4 and s_6 are also filled in the same way. Hence

$$s_4 = P_3x17 + P_4x8 + P_5x17, \tag{6.5}$$

$$s_6 = P_5x17 + P_6x8 + P_7x17, \tag{6.6}$$

Case (ii) Weights of center sub-pixel

The centermost sub pixel s_9 depends upon all the 8 neighbours of PTS. Here also the contribution of corner pixels P_2, P_4, P_6 and P_8 are less compared to that of the remaining adjacent pixels P_1, P_3, P_5, and P_7. So, for the centre sub-pixel, the s_9 value for a particular class could be the summation of product of the number of sub pixels for that class at pixel location P_8, P_1, P_2, P_3, P_4, P_5, P_6, P_7 and 8, 17, 8, 17, 8, 17, 8, 17. So the value s_9 is given as,

$$s_9 = P_8x8 + P_1x17 + P_2x8 + P_3x17 + P_4x8 + P_5x17 + P_6x8 + P_7x17 \tag{6.7}$$

Case (iii) Weights of remaining sub-pixels

The center row/column sub pixels except the center most sub pixel, they depend upon 3 neighbouring pixels which falls in their respective side. There are four such sub pixels and they are s_1, s_3, s_5 and s_7. For example, consider s_1, it is at the top middle position and it depends upon the corresponding pixels which are adjacent to it and lying on the same side of the matrix. Here s_1 depends upon P_8, P_1 and P_2. The number of sub pixels at these locations for a particular class, is multiplied by the weights 8, 17 and 8. The value s_1 is given as,

$$s_1 = P_8x8 + P_1x17 + P_2x8 \tag{6.8}$$

Similarly, the other values for s_3, s_5 and s_7 may be calculated by the following formulae:

$$s_3 = P_2x8 + P_3x17 + P_4x8 \tag{6.9}$$

$$s_5 = P_4x8 + P_5x17 + P_6x8 \tag{6.10}$$

$$s_7 = P_6x8 + P_7x17 + P_8x8 \tag{6.11}$$

Now collect all the values of s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8 and s_9 and put them in matrix form given in figure 6.2(b). We obtain this matrix called as weight matrix for one class. Similarly, for the remaining classes also we have to find such matrices. The number of weight matrix is same as that of the number of classes present in the image. A general consequence we obtain is that always the middle sub pixel value is larger than other sub pixel values because the middle sub pixel value depends upon all the surrounding eight pixels while the remaining sub pixels values depend upon only three of the neighbouring pixels. Hence the weight matrices are normalized by dividing each of the values got from equation (6.3) to equation (6.11) by the summation of respective weights. Now for each weight matrix we get normalized weight matrix. The pixel filling process completely depends on these normalized weight matrices.

Pixel Filling Process

Since each normalized weight matrix is related to each class, there is an equal number of matrix named as filled matrix for class i, of order zf x zf (zf means zoom factor) has been created. The filling process has been explained in step-by-step procedure.

Step 1: Find the number of sub pixels to be mapped as class 1.

Step 2: Sort the values in normalized weight matrix for class 1 in descending order.

Step 3: The number of values to be mapped (got from step 1) has been filled in the place where the normalized values are larger (from step 2).

Here, if only 3 sub pixels are to be mapped as class 1, then only those three places may be filled by number 1 and the remaining places of the filled matrix i are kept as empty or zero.

Step 4: Repeat step 1, step 2 and step 3 for next classes.

At this stage we get the number of filled matrices equal to number of classes.

Step 5: Now combine all the filled matrices by simple matrix addition. At this stage we get three possibilities which should be taken care. The first possibility is there may be no overlapping in the arrangement of numbers. The second possibility is there may be overlapping between classes. The third possibility is there may be some sub pixel locations which are unoccupied. The third case occurs only if the second case occurs. But the pixel filling processes for these last two cases are different. If no overlapping is there then the sub pixel locations are filled by mere addition of the filled matrices and proceed to next pixel. Otherwise the procedure continues to step 6.

Step 6: Overlapping of sub pixels after addition of filled matrices.

Case (i): Find the place of overlapping and from the filled matrices for each class find the maximum value which is among all the values of same location in different matrices and fill it by the class value. Do this for all the overlapping sub pixel locations.

Case (ii): If any sub pixel location is unoccupied, it means some pixel locations might be overlapped. So once the case (i) of step 6 has been performed the unoccupied location needs to be filled. Go to the unoccupied location and for that location, find the maximum value of the same location from the pixel filled matrices and fill it by the class value. Since there is one-to-one correspondence between the pixels to be occupied and the number of elements in sub matrix, no place will be unoccupied.

Step 7: Repeat the process for each pixel.

Pseudo-code of Pixel Filling (PF) algorithm

```
1) Input fractional output
2) Decide zoom factor
3) For each subpixel within the pixel of consideration
    a) Calculate the weights for each subpixel using the neighbouring pixels
        i) For corner subpixels use equations (6.3), (6.4), (6.5) and (6.6)
        ii) For middle subpixels which are at the outer boundary use equation (6.7)
        iii) For middle subpixel use equations (6.8), (6.9), (6.10) and (6.11)
    b) Normalize the matrices name it as normalized matrices.
4) Fill the subpixels in a new matrix (named as pixel filled matrix), whose dimension is
   zf x zf times bigger than the original data, where zf is zoom factor, whose weight is
   more.
5) If unoccupied pixels are there or pixels with two or more classes trying to occupy at
   one subpixel, then the maximum value at the normalized matrices' class will be filled.
```

For each sub-pixel position which is unoccupied in the final pixel-matrix, select the candidate classes and fill with the class having maximum weight for corresponding pixels from all the candidate classes. The same algorithm may be extended for any zoom factor by scaling the weights between 8 and 17.

The coding for pixel swapping algorithm written for binary classes in IDL language has been downloaded from website http:\\www.globalchange.msu.edu/gomsu/models.php. However, the same coding has been implemented in MATLAB for the current experiments. Here for implementation for multi-class problem, a one-against-all strategy has been followed to obtain super resolved image. Extract each class versus all the other classes and collect the information where overlapping is not there. All the extracted pixels are put it in a matrix to obtain a super resolution map.

6.5 Methodology

After pre-processing of hyperspectral data, the feature extraction techniques, as discussed in earlier chapters, have been applied to obtain reduced hyperspectral data, which then has been classified to produce both per pixel and sub pixel classifications. The per pixel classification has been performed by SVM to produce hard classification output. Generally, a sub pixel classification output is required to do sub pixel mapping, but due to non-availability of soft reference data for validation purpose, the super resolution mapping algorithms have been applied on low pass filter applied on output of SVM classification. The flowchart given in Figure 6.3 gives a complete overview of the methodology followed in this section.

The experiment has been done on four datasets, one synthetic dataset and three hyperspectral datasets. The result analysis has been made according to the three types of assessment techniques described in section 6.7. The super resolved images are compared with best classified SVM output for each dataset, since SVM output acts as reference data. The novel pixel filling algorithm has been compared with pixel swapping algorithm (Atkinson, 2005).

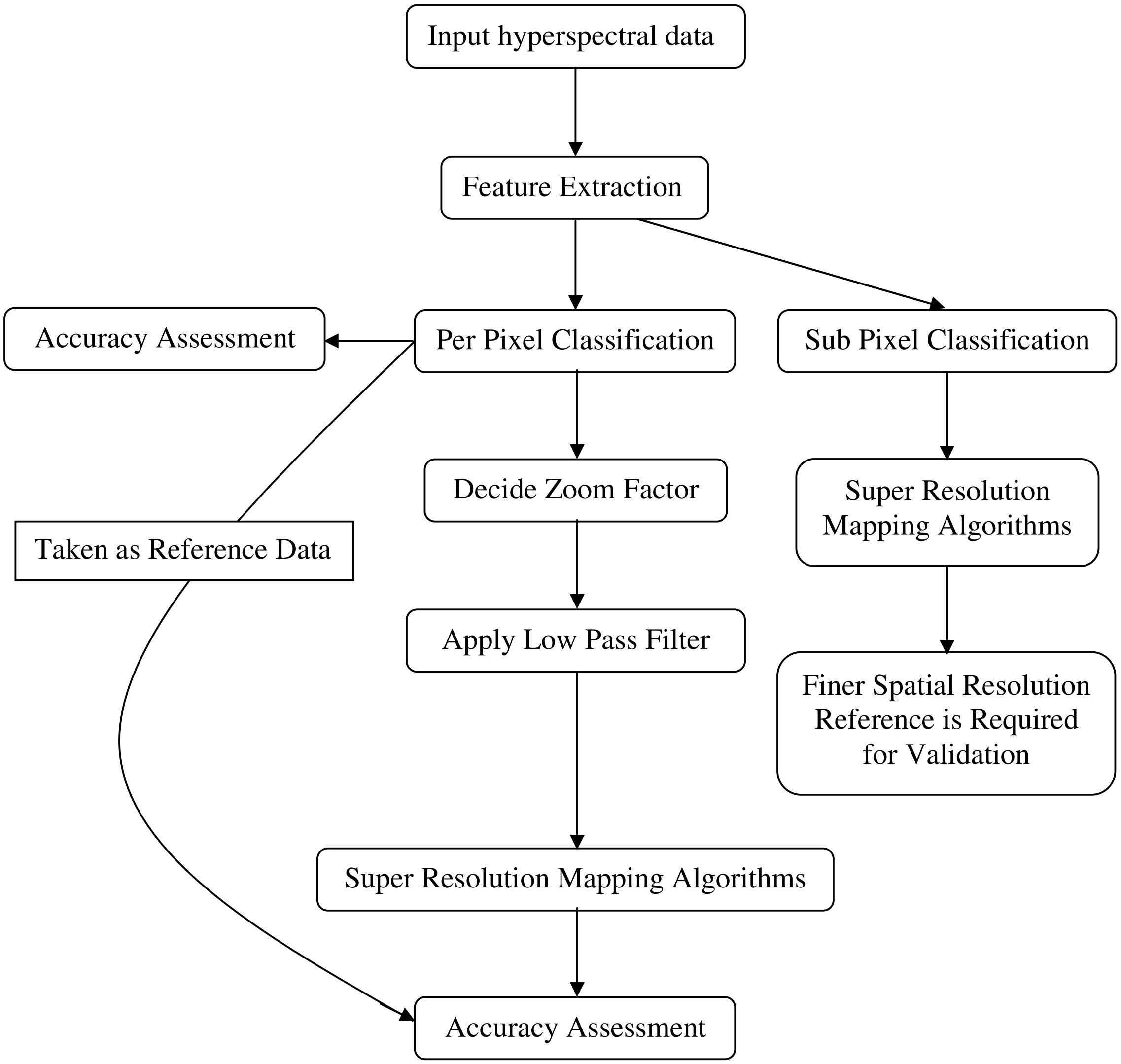


Figure 6.3: Flowchart of the methodology followed to obtain a super resolution map

All the implementation steps from feature extraction of hyperspectral to super resolution mapping have now been described as follows:

(i) feature extraction of hyperspectral data,

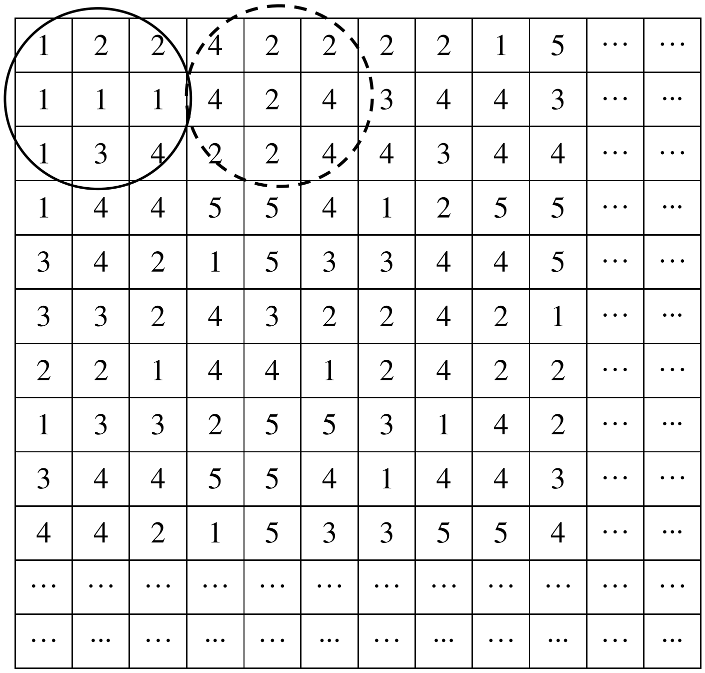
(ii) per pixel classification of hyperspectral data,

(iii) application of low pass filters 3 x 3 and 5 x 5 on classified data to coarsen the data by factor 3 and 5 respectively, and finally,

(iv) application of super resolution mapping algorithms on the coarsen data obtained at step (iii).

Let the hyperspectral data, after pre-processing, is of dimension M x N x B, where M and N denotes the spatial locations (of x and y axes respectively) and B is the total number of bands. The feature extraction of hyperspectral data has been performed by wavelet based techniques and PCA based techniques. For super-resolution mapping, features extracted from only 3 feature reduction techniques, Haar_1, Haar_2 and Daub2_1 wavelets, have been considered.

Each of these feature reduced datasets has been classified using SVM classifier to produce a thematic map of size M x N pixels, and its accuracy has been determined. The thematic map with highest classification accuracy has been considered as reference data for super resolution mapping. A low-pass filter of size zf x zf (zf may be 3 or 5 or 7, etc.,) has been applied to the thematic map got from the hard output of SVM classification. By doing this, the spatial resolution of each pixel has been degraded by zf times along x-direction and z times along y-direction. So, the SVM classified (which will act as reference data for validation of super resolution mapping algorithms) data of dimension M x N has been degraded to size $\frac{M}{zf} \times \frac{N}{zf}$. At this level, every zf x zf square window (or sub-matrix) takes information from SVM classified output about the number of classes present in this sub matrix as a vector of length equals the total number of classes.



1	2	2	4	2	2	2	2	1	5	…	…
1	1	1	4	2	4	3	4	4	3	…	...
1	3	4	2	2	4	4	3	4	4	…	…
1	4	4	5	5	4	1	2	5	5	…	...
3	4	2	1	5	3	3	4	4	5	…	…
3	3	2	4	3	2	2	4	2	1	…	...
2	2	1	4	4	1	2	4	2	2	…	…
1	3	3	2	5	5	3	1	4	2	…	...
3	4	4	5	5	4	1	4	4	3	…	…
4	4	2	1	5	3	3	5	5	4	…	...
…	…	…	…	…	…	…	…	…	…	…	…
…	...	…	...	…	...	…	...	…	...	…	...

Figure 6.4: Class labels of a thematic map

Let us take a zoom factor of 3 and first apply corresponding low-pass filter on the thematic map (Figure 6.4) and then collect the number of pixels belonging to each of the class for every 9 (3x3 sub-matrix) pixels. Since for zoom factor 3, each pixel is divided into 9 sub pixels, 3 along x-direction and 3 along y-direction, the 3x3 filter is being applied. Now we collect only the class labels for each of the 3x3 window and the number of class label is noted for each class.

From the Figure 6.4, a thematic map is given and the numbers represent class labels. Assume that there are five classes. The number of pixels covered by solid border circle is 9. Hence the values in that circle may be represented as the vector (5 2 1 1 0). It means there are 5 pixels having class label 1, while 2 pixels are having class label 2 and so on, and finally no pixel belonging to class 5. Similarly, for the dotted border circle, class values vector is (1 2 2 2 2). To emphasize the meaning of the vector notation the same may be represented in Table 6.1. Now this class values vector is converted into fraction values by dividing the numbers by 9 (since the zoom factor is 3x3 = 9). This is the fraction abundance of each pixel and used as input for producing super resolution map.

Table 6.1: Pixel details and the corresponding fraction abundance for each pixel

	Number of pixels with class label					**Total**	**Fraction abundance for each pixel**					**Total**
Class label	**1**	**2**	**3**	**4**	**5**		**1**	**2**	**3**	**4**	**5**	
Pixel 1	5	2	1	1	0	9	0.6	0.2	0.1	0.1	0	1
Pixel 2	0	5	0	4	0	9	0	0.6	0	0.4	0	1
…	…	…	…	…	…	…						

Now the thematic map of size MxN is degraded / reduced to size $\frac{M}{3} \times \frac{N}{3}$ and each pixel has been divided into 9 sub-pixels. These 9 sub-pixels have to be filled by class labels given according to the Table 6.1. To achieve this, the proposed super resolution mapping based on neighbouring pixels label is discussed in the next section.

Now on this reduced datasets which is in the form of matrix representation given in Table 6.1, super resolution mapping algorithms described on section 6.3 and section 6.4 have been applied. Sometimes the number of pixels along row or column or both, in the dataset on which super resolution mapping algorithm is applied may not be divisible by the zoom factor. In that case, few rows or columns or both may be reduced to make it divisible by zoom factor. So a maximum of z–1 rows or z–1 columns or both may be removed.

6.6 Accuracy Assessment for Super Resolution Mapping Algorithms

The performances of two super resolution mapping algorithms have been evaluated in different ways,

1. conventional error matrix construction based on some testing samples
2. conventional error matrix construction based on pixel-to-pixel comparison of the whole classified image
3. conventional error matrix construction for patches of mixed pixels (mixed by 2 classes or 3 classes or 4 classes) respectively.

6.6.1 Conventional Error Matrix Construction Based on Some Testing Samples

The first type of accuracy assessment has been done by exactly how the same has been done for per pixel classification. The reference samples / testing samples used here are exactly used from the same. Since the output of a super resolution mapping is a thematic map such accuracy assessment is possible. Also there is no need of calculating user's and producer's accuracies. Since in classification problems, we are interested in finding how much variation is there in with-in class species, for instance, how much pixels belonging to class alfalfa has been mapped to class oats, because there spectral characteristics plays a big role, while in super resolution mapping the spatial characteristic is important. Hence overall accuracy and kappa coefficient alone has been calculated in analysing the super resolution mapping. The reference data taken here is same as that of used for evaluating per pixel classification algorithm which has been given in Chapter 3.

6.6.2 Conventional Error Matrix Construction Based on Pixel-to-Pixel Comparison of the Whole Classified Image

Since SVM output has been degraded by applying low pass filters, the SVM output has been taken as reference. Now comparing the SVM output as well as the super resolved output and by using the same contingency table / error matrix accuracy assessment has been performed. Also, if a pixel is pure, the application of any super resolution mapping algorithm on that pixel gives 100% accuracy for that pixel. So, the overall accuracy for a dataset having more number of pure pixels will be generally high that a dataset having less number of pure pixels. Hence from the degraded dataset by applying low pass filter, has been bifurcated into two sets, set of pure pixels only and the other set containing only mixed pixels. Definitely these two sets are mutually exclusive and collectively exhaustive and hence such partition is possible. Now for the set having only pure pixel, the super resolution accuracy would be 100 percent. For the remaining pixels, the accuracy has been calculated by the following formula given by:p-to-p

$$\begin{aligned} &Accuracy\ of \\ &mixed\ pixels \\ &= \frac{[\{n(PP) +\ n(MP)\} \times a(OA\ by\ p - to - p) - n(PP) \times 100]}{n(MP)} \end{aligned} \qquad (6.12)$$

where,

PP – set of all pure pixels
MP – set of all mixed pixels
n(PP) – denotes the number of elements belonging to the set PP, i.e., the number of pure pixels
n(MP) – denotes the number of elements belonging to the set MP, i.e., the number of mixed pixels
$a(OA\ by\ p - to - p)$ – denotes overall accuracy obtained by pixel-to-pixel measurement

By this accuracy assessment we get how mixed pixels are resolved but an in-depth view of this type of accuracy is needed, since the mixed pixels may be mixed by two classes or more classes. A modification in testing data requires to do further analysis.

6.6.3 Conventional Error Matrix Construction for Patches of Mixed Pixels (mixed by 2 classes or 3 classes or 4 classes)

Now the focus of accuracy assessment tends to how each of the mixed pixels has been mapped. To some extent this could be achieved by modifying the testing data. Generally, for classification purpose, testing samples have been taken at middle of class. On the contrary, here testing samples are specially taken where it is mixed by two classes, three classes or four classes, etc., separately. Now the usual error matrix based accuracy assessment has been conducted for these types of mixed pixels separately.

6.7 Results and Discussions

The experiment has been done for four datasets, one synthetic dataset and three hyperspectral datasets. The result analysis has been made according to the three types of assessment techniques described in section 6.6. Results according to datawise analysis have been presented in this section.

6.7.1 Results of Super Resolution Mapping for Dataset I – Synthetic Data

By taking reference data described in section 6.6.1, the accuracy assessment for the super resolved data which comes from pixel filling algorithm has been calculated. Here the overall accuracy of super resolved image by pixel filling algorithm gives 90.7% and 72.7% for zoom factors 3 and 5 respectively. This includes pure pixels also for which always the super resolved accuracy is 100%. So only for mixed pixels the accuracy has been calculated and it comes to be 65.5% and 46.3% for zoom factors 3 and 5 respectively. Since this is the result for synthetic data, a hypothesis may be set that these are the overall accuracies for pixel filling algorithm by taking zoom factor 3 and 5 respectively.

By taking reference data described in section 6.6.2, in the synthetic data when low-pass filters have been applied to the SVM classified map, the pure pixel are more (219 pure pixels out of 300 pixels) when the low-pass filter is 3, while for the low-pass filter 5, it is near to 50 percent (53 pure pixels out of 108 pixels) (refer Table 6.2). The accuracy for only mixed pixels for zoom factor 3 is relatively higher than that of zoom factor 5. Even though the overall accuracy has been 72.67% for zoom factor 5, due to the removal of pure pixels there is a deep difference in accuracy.

Table 6.2: Accuracy assessment for super resolution mapping algorithm by Pixel Filling Algorithm for synthetic dataset

Sl. No.	**Zoom level**	**Total no. of pixels**	**Accuracy**		**No. of pure pixels**	**Accuracy (100%)**		**No. of mixed pixels**	**Intermediate value**		**Accuracy of mixed pixels**
1	3x3	300	90.70		219	100		81	5309.4		65.5
2	5x5	108	72.67		53	100		55	2548.79		46.3

6.7.2 Results of Super Resolution Mapping for Dataset II – AVIRIS Indiana Pine Dataset

The accuracy assessment has been performed by using conventional error matrix and by taking the same reference data which has been used for evaluating per pixel classification.

1. The Indiana Pine dataset is full of vegetation cover. Each of the vegetation class is available on patches and almost they are crisp in nature and hence possibility of mixed pixels inside the patches is less. But many patches are small in size

and the boundaries are mixed pixels. So in Figure 6.5(b) and Figure 6.6(b), the noise present inside the patches of classes is due to classification error. Such types of errors are carried over for super resolution mapping.

2. Here also the boundary pixels are well classified for zoom factor 3 when compared to zoom factor 5.

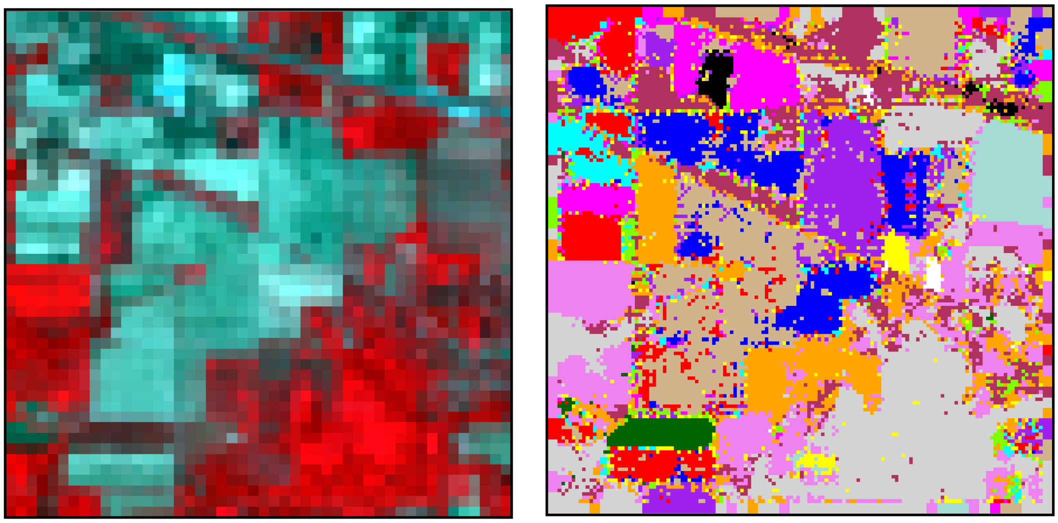


Figure 6.5: (a) FCC (Red : 40, Green : 30, Blue : 20) of subset of Indiana Pine dataset subsampled by 3. (b) Super resolution map by taking zoom factor 3 of 3x3 sub-sampled Indiana Pine dataset.

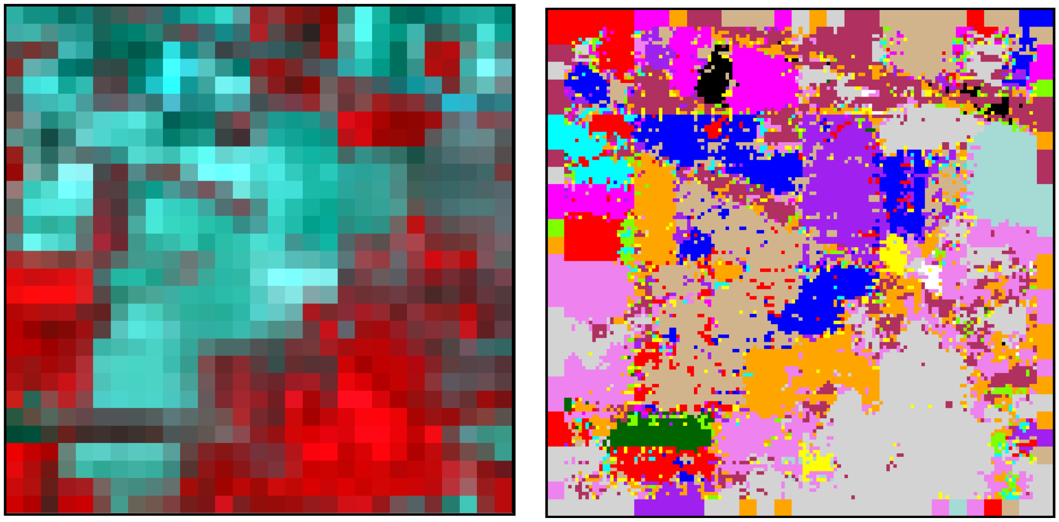


Figure 6.6: (a) FCC (Red : 40, Green : 30, Blue : 20) of subset of Indiana Pine dataset subsampled by 5. (b) Super resolution map by taking zoom factor 5 of 5x5 subsampled Indiana Pine dataset.

Table 6.3: Comparison of the proposed Pixel Filling algorithm with Pixel Swapping algorithm for zoom factor 3.

	Dataset II	
	Pixel Filling	**Pixel Swapping**
Haar_1	92.4	98.7
Haar_2	87.7	98.4
Daub2_1	93	98.2

Table 6.4: Comparison of the proposed Pixel Filling algorithm with Pixel Swapping algorithm for zoom factor 5.

	Dataset II	
	Pixel Filling	**Pixel Swapping**
Haar_1	91.9	94.8
Haar_2	87.2	94
Daub2_1	92.3	93.8

The comparison of two algorithms for zoom factor 3 and 5 has been displayed in Table 6.3 and Table 6.4 respectively. Moreover from these two tables we may compare how the two algorithms performed for various datasets. Also from both the Tables 6.3 and 6.4, it is observed that pixel swapping algorithm performs better than pixel filling algorithm. Its due to the classes of this dataset are crisp in nature and pixel swapping algorithm works for such datasets.

Now the analysis has been based on how well the mixed pixels are super resolved which was described in section 6.6.2. The AVIRIS Indiana Pine dataset has classes dominated by vegetation and few classes occupies less area. Except the class wood which is having irregular shape boundary almost all other classes are having linear boundaries except for those classes which shares boundary with class woods. The classes are also crisp in nature and possibility of mixed pixels within the region of class is less. Hence for zoom factor 3, the accuracy of mixed pixels is around 60 percent. Even though the overall accuracy is around 70 percent, the algorithm performs well for super resolving mixed pixels. For zoom factor 3, the results are given in Table 6.5:

Table 6.5: Accuracy assessment for super resolution mapping algorithm by Pixel Filling Algorithm for Indiana Pine Dataset

		Total no. of pixels	**Accuracy**		**No. of pure pixels**	**Accuracy (100%)**		**No. of mixed pixels**			**Accuracy of mixed pixels**
1	Haar_1	2304	70.814		625	100		1679	100655.5		59.9
2	Haar_2	2304	72.627		692	100		1612	98132.61		60.9
3	Daub2_1	2304	71.677		651	100		1653	100043.8		60.5

Now analysis has been done by taking patches of mixed pixel alone which was described in section 6.6.3.

1. From Table 6.6, it is evident that the accuracy of mixed pixels when mixed by 2 classes is 80.9% while the worst super resolved is for pixels with 4 classes mixed which comes to be more than 51%.
2. For zoom factor 5, the accuracy of super resolved mixed pixels mixed by 2 classes is around 70% while the same for mixed for 4 classes is above 40% (Table 6.7).

Table 6.6: Accuracy assessment for super resolution mapping algorithm by Pixel Filling Algorithm for Indiana Pine dataset for 2-3-4- mixed pixels by taking zoom factor 3

	2 Classes			**3 Classes**			**4 Classes**	
Haar_1	75.5	0.73		61.1	0.57		51.9	0.47
Haar_2	71.8	0.69		59.2	0.55		52.7	0.48
Daub2_1	80.9	0.79		63.9	0.61		56.5	0.52

Table 6.7: Accuracy assessment for super resolution mapping algorithm by Pixel Filling Algorithm for Indiana Pine dataset for 2-3-4- mixed pixels by taking zoom factor 5

	2 Classes			**3 Classes**			**4 Classes**	
Haar_1	68.9	0.65		50.6	0.46		44.4	0.39
Haar_2	65.4	0.61		49.7	0.45		43.9	0.38
Daub2_1	71.3	0.68		51.7	0.47		45.8	0.4

6.7.3 Results of Super Resolution Mapping for Dataset III – HYPERION Roorkee and its Surroundings Dataset

The Roorkee dataset is a challenging dataset because of two reasons. The first one is that the region is textured with the classes are not in regular shape and the second reason is the spatial resolution is 30 m which leads to more pixels mixed.The canal which runs across the image is part of the class water (dark blue colour) and this is the only portion having linear boundary. When the zoom factor is 3x3 the class water is super resolved well (Figure 6.7) while when the zoom factor is 5, the 5x5 window consists of 3 classes, namely, water, urban (purple colour) and trees (dark green). The poor super resolution is due to the class water is not having any pixels in its left and right neighbouring pixels (Figure 6.8) but only along its top and bottom pixels, hence the sub pixels accumulate towards top and bottom of the pixel. So in every 5x5 pixel in super resolved image the center row never contains the class water.

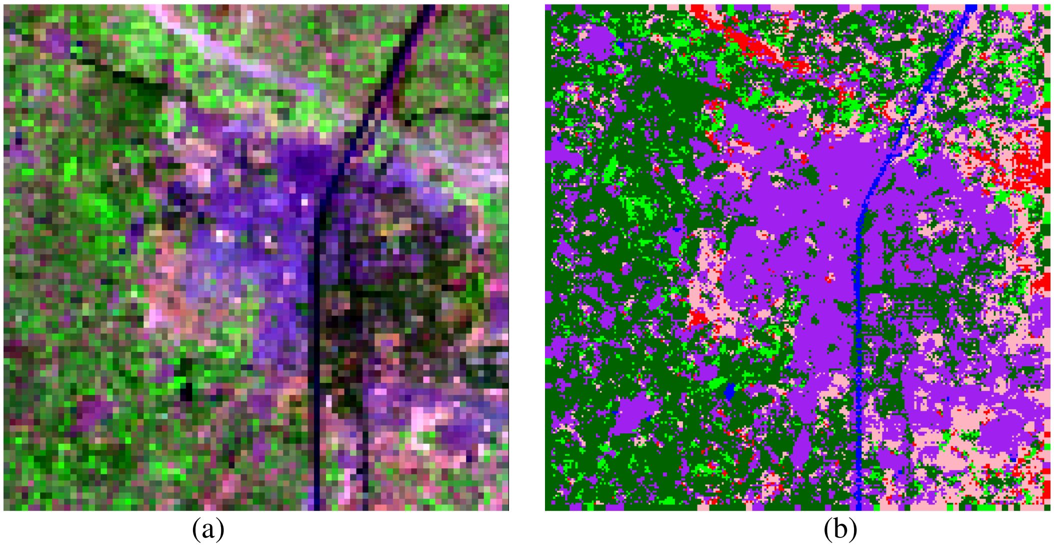


Figure 6.7: (a) FCC (Red : 70, Green : 20, Blue : 2) of subset of Roorkee dataset subsampled by 3. (b) Super resolution map by taking zoom factor 3 of 3x3 subsampled Roorkee dataset.

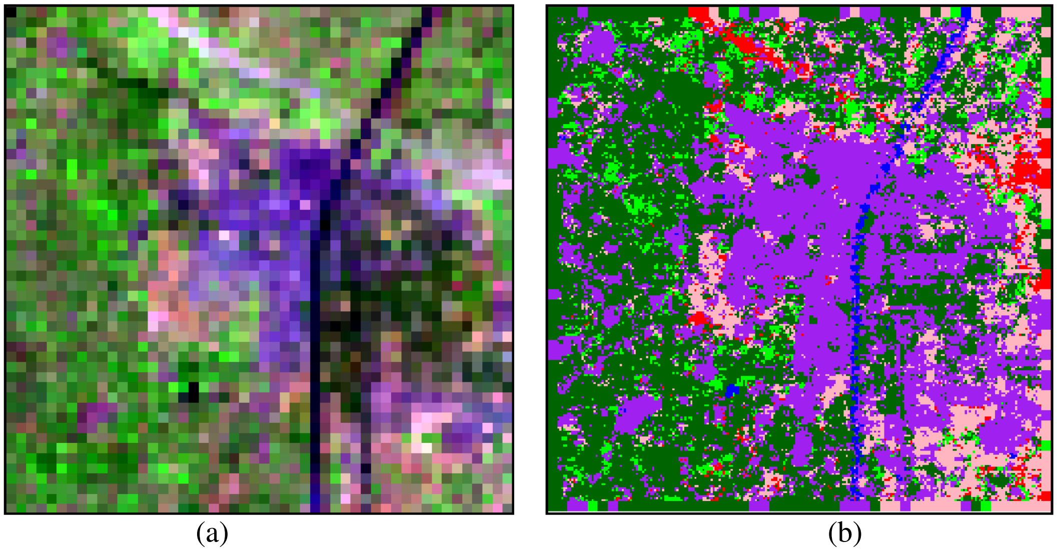


Figure 6.8: (a) FCC (Red : 70, Green : 20, Blue : 2) of subset of Roorkee dataset subsampled by 5. (b) Super resolution map by taking zoom factor 5 of 5x5 subsampled Roorkee dataset.

Table 6.8: Comparison of the proposed Pixel Filling algorithm with Pixel Swapping algorithm for zoom factor 3.

	Dataset III	
	Pixel Filling	**Pixel Swapping**
Haar_1	96.6	93.3
Haar_2	94.8	87.9
Daub2_1	95.7	90.4

Table 6.9: Comparison of the proposed Pixel Filling algorithm with Pixel Swapping algorithm for zoom factor 5.

	Dataset III	
	Pixel Filling	**Pixel Swapping**
Haar_1	93.3	77.1
Haar_2	89	69.8
Daub2_1	90	73.1

Contrary to dataset II, for this dataset, the pixel filling algorithm performs better than the pixel swapping algorithm (Table 6.8 and Table 6.9) when zoom factors 3 and 5 are used. Here the classes are meshed together and here efficacy of the pixel filling algorithm is shown good.

By taking reference data described in section 6.6.2, the accuracy assessment for the super resolved data which comes from pixel filling algorithm has been calculated. The spatial resolution of Roorkee dataset is 30 m and hence more mixed pixels are there. Also canal is the only linear class available and the other classes have irregular boundary. So the effect of super resolution algorithm can be viewed well for this dataset. Here the mixed pixels are resolved much better compared to Indiana Pine dataset. Here it comes to be around 62 percent while the overall classification is still around 70 percent (refer Table 6.10).

Table 6.10: Accuracy assessment for super resolution mapping algorithm by Pixel Filling Algorithm for Roorkee and its surroundings dataset

	Total no. of pixels	**Accuracy**		**No. of pure pixels**	**Accuracy (100%)**		**No. of mixed pixels**	**Intermediate value**		**Accuracy of mixed pixels**
Haar_1	6889	69.289		1274	100		5615	349931.9		62.3
Haar_2	6889	70.849		1373	100		5516	350778.8		63.6
Daub2_1	6889	70.015		1347	100		5542	347633.3		62.7

By taking reference data described in section 6.6.3, the accuracy assessment for the super resolved data which comes from pixel filling algorithm has been calculated. There is a significant difference in accuracies when taking mixed pixel mixed by 2 classes while using zoom factor 3 and 5 (Table 6.11 and Table 6.12). The reason is because of the classes have irregular boundaries, the mis-mapping increases when zoom factor increases. In general, the super resolved pixels mixed by 2 classes and mixed by 3 classes are better for Roorkee dataset when compared to the other 2 datasets.

Table 6.11: Accuracy assessment for super resolution mapping algorithm by Pixel Filling Algorithm for Roorkee and its surroundings dataset for 2-3- mixed pixels by taking zoom factor 3

	2 Classes			**3 Classes**	
Haar_1	98.3	0.97		88.6	0.83
Haar_2	89.9	0.86		78.4	0.73
Daub2_1	93.6	0.91		87.9	0.85

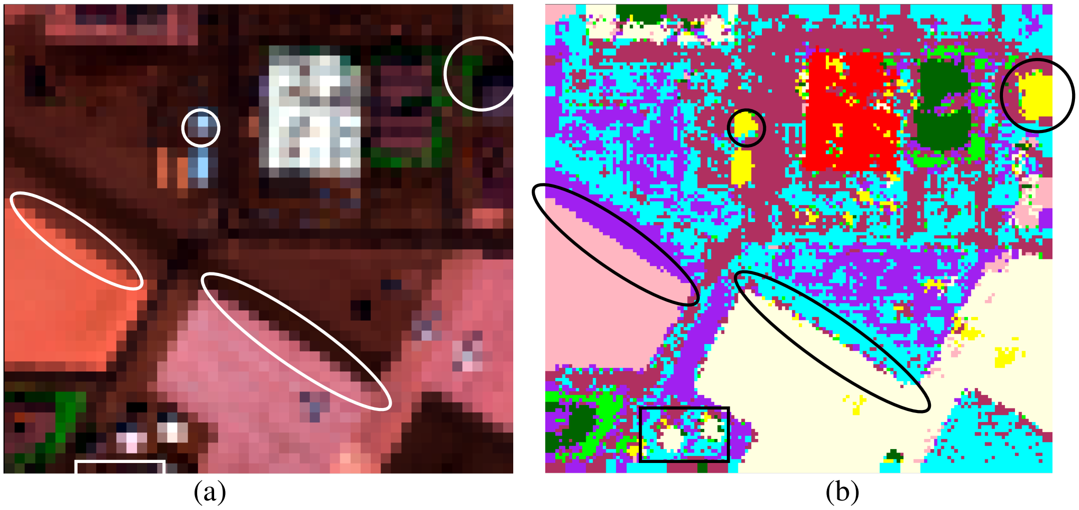
Table 6.12: Accuracy assessment for super resolution mapping algorithm by Pixel Filling Algorithm for Roorkee and its surroundings dataset for 2-3- mixed pixels by taking zoom factor 5

	2 Classes			**3 Classes**	
Haar_1	70.3	0.59		44.6	0.32
Haar_2	69.3	0.58		45.1	0.33
Daub2_1	69.5	0.58		47.1	0.35

6.7.4 Results of super resolution mapping for dataset IV – AVIRIS San Diego Dataset

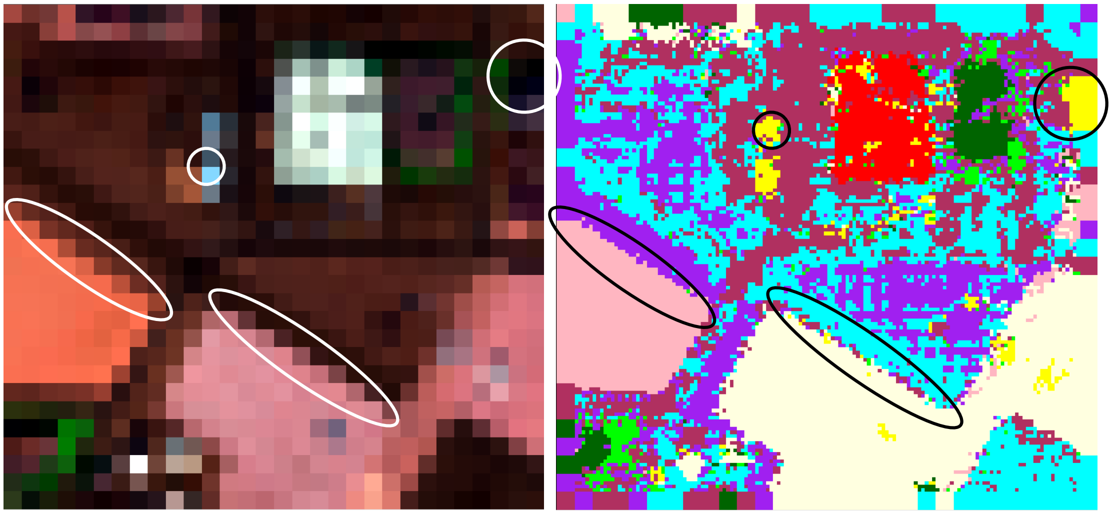
By taking reference data described in section 6.6.1, the accuracy assessment for the super resolved data which comes from pixel filling algorithm has been calculated.

1. The classes having linear borders are super resolved well by using both the zoom factors 3 and 5, as it can be seen from the elliptical region in Figures 6.9 (a) and 6.9(b) and Figures 6.10 (a) and 6.10(b).
2. The algorithm is strong enough to map pixels surrounding by mixed classes. The class building having all its border pixels mixed with classes has also super resolved well, as highlighted by circular region from Figure 6.9 and Figure 6.10.
3. Not only the linear boundaries, but also for curved boundaries between classes may be super resolved. The two circular region demarcated in rectangular boxes have also super resolved well, as it can be seen from Figure 6.9 and Figure 6.10.


(a)

(b)

Figure 6.9: (a) FCC (Red : 140, Green : 80, Blue : 20) of subset of San Diego dataset subsampled by 3. (b) Super resolution map by taking zoom factor 3 of 3x3 subsampled San Diego dataset.



(a) (b)

Figure 6.10: (a) FCC (Red : 140, Green : 80, Blue : 20) of subset of San Diego dataset subsampled by 5. (b) Super resolution map by taking zoom factor 5 of 5x5 subsampled San Diego dataset.

Table 6.13: Comparison of the proposed Pixel Filling algorithm with Pixel Swapping algorithm for zoom factor 3.

	Dataset III	
	Pixel Filling	**Pixel Swapping**
Haar_1	97.4	91
Haar_2	98.4	92.6
Daub2_1	97.4	87.4

Table 6.14: Comparison of the proposed Pixel Filling algorithm with Pixel Swapping algorithm for zoom factor 5.

	Dataset IV	
	Pixel Filling	**Pixel Swapping**
Haar_1	88.5	68.3
Haar_2	97.4	70.3
Daub2_1	92.7	66

There is a significant difference in accuracies (Table 6.13 and Table 6.14) when using pixel filling algorithm and pixel swapping algorithm for dataset IV. Pixel filling algorithm outperforms pixel swapping in accuracy wise.

By taking reference data described in section 6.6.2, the accuracy assessment for the super resolved data which comes from pixel filling algorithm has been calculated. The classes of this dataset look like crisp but it didn't so. Class concrete of one type is mixed with class concrete of other type. Even though this effect more in classification but not in super resolution mapping but for few 3x3 window class of one type will have less number of pixel while its neighbouring pixels will don't have this class. So this effects in super resolution of mixed pixels. Even though it is so the accuracy of super resolved mixed pixels is around 64 percent while the overall accuracy is around 76 percent (refer Table 6.15).

Table 6.15: Accuracy assessment for super resolution mapping algorithm by Pixel Filling Algorithm for San Diego dataset

		Total no. of pixels	**Accuracy**		**No. of pure pixels**	**Accuracy (100%)**		**No. of mixed pixels**	**Intermediate value**		**Accuracy of mixed pixels**
1	Haar_1	2300	75.986		757	100		1543	99067.8		64.2
2	Haar_2	2300	78.647		893	100		1407	91588.1		65.1
3	Daub2_1	2300	76.256		779	100		1521	97488.8		64.1

By taking reference data described in section 6.6.3, the accuracy assessment for the super resolved data which comes from pixel filling algorithm has been calculated. The algorithm works almost uniform for mixed pixels mixed by two classes for both the zoom factors 3 and 5 while there is a significant fall in accuracy when zoom factor is 5 for mixed pixels mixed by 3 classes. Such types of pixels are very less and have less impact on the overall accuracy.

Table 6.16: Accuracy assessment for super resolution mapping algorithm by Pixel Filling Algorithm for San Diego dataset for 2-3- mixed pixels by taking zoom factor 3

	2 Classes			**3 Classes**	
Haar_1	69.4	0.53		71.4	0.62
Haar_2	69.4	0.55		68.3	0.59
Daub2_1	69.4	0.53		71.4	0.62

Table 6.17: Accuracy assessment for super resolution mapping algorithm by Pixel Filling Algorithm for San Diego dataset for 2-3- mixed pixels by taking zoom factor 5

	2 Classes			**3 Classes**	
Haar_1	64	0.46		54	0.41
Haar_2	66.7	0.51		52.4	0.39
Daub2_1	64.9	0.47		57.1	0.44

In Table 6.18 and Table 6.19, a collective information about the accuracy assessment for super resolution mapping of all the 3 hyperspectral datasets have been given for zoom factors 3 and 5 respectively, by taking mixed pixels separately. Among all the datasets the Pixel Filling Algorithm performs better for Roorkee dataset.

Table 6.18: The super resolved accuracy for pixels mixed by 2, 3 and 4 classes of all the 3 datasets by taking zoom factor 3.

	Indiana Pine						**Roorkee**				**San Diego**			
Mixed by	**2 classes**		**3 classes**		**4 classes**		**2 classes**		**3 classes**		**2 classes**		**3 classes**	
Haar_1	75.5	0.73	61.1	0.57	51.9	0.47	98.3	.97	88.6	.83	69.4	0.53	71.4	0.62
Haar_2	71.8	0.69	59.2	0.55	52.7	0.48	89.9	0.86	78.4	0.73	69.4	0.55	68.3	0.59
Daub2_1	80.9	0.79	63.9	0.61	56.5	0.52	93.6	0.91	87.9	0.85	69.4	0.53	71.4	0.62

Table 6.19: The super resolved accuracy for pixels mixed by 2, 3 and 4 classes of all the 3 datasets by taking zoom factor 5.

	Indiana Pine						**Roorkee**				**San Diego**			
	2 classes		**3 classes**		**4 classes**		**2 classes**		**3 classes**		**2 classes**		**3 classes**	
Haar_1	68.9	0.65	50.6	0.46	44.4	0.39	70.3	0.59	44.6	0.32	64	0.46	54	0.41
Haar_2	65.4	0.61	49.7	0.45	43.9	0.38	69.3	0.58	45.1	0.33	66.7	0.51	52.4	0.39
Daub2_1	71.3	0.68	51.7	0.47	45.8	0.4	69.5	0.58	47.1	0.35	64.9	0.47	57.1	0.44

6.8 Summary

Two algorithms for super resolution mapping have been presented in this chapter and the results have been compared. One algorithm is the existing pixel swapping algorithm which requires lots of computation time to perform super resolution map while the other is novel pixel filling algorithm which outperforms the earlier algorithm for complicated land cover classes. The pixel filling algorithm explained here will work for classes with less number of pixels, any number of classes, for any odd order zoom factor and if boundary between classes are linear or non-linear. Three types of accuracy assessments have been performed and in almost all analysis the pixel filling algorithm shows high accuracy that the other one.

Chapter 7

Conclusion and Future Research

7.1 Introduction

To study about the earth surface hyperspectral data are very useful in extracting useful information from them. Based on the application, the hyperspectral spatial resolution varies from few meters to 10s of meters. In this research, three original hyperspectral data have been taken for study whose spatial resolutions are 4m, 20m and 30m. The advantage of using hyperspectral data over multispectral data is the availability of rich spectral details present in hyperspectral data which are further useful in discrimination of various inter-class and intra-class variance. Sometimes, a multispectral data of 2m spatial resolution may not be useful in study about soil types or vegetation species but a hyperspectral data of 20m spatial resolution could be useful.

Due to the large data size and redundant information the hyperspectral data suffers from dimensionality problem. A solution to this problem is to reduce the number of bands by some mathematical transformations. Even though there are two major techniques available to reduce the number of bands, namely feature selection and feature extraction, the literature suggests that feature selection technique takes more time to select optimum number of bands and also loss of information from the hyperspectral data could be more. So in this research, only feature extraction techniques have been performed.

Once the feature extraction has been performed, now to identify the percentage of class present in the hyperspectral data, it should be classified. There are mainly two types of classifications, per pixel and sub pixel classification. With the help of training samples the classification may be done, named as supervised classification and when no training samples are used in unsupervised classification. Moreover, the statistical properties may be used in any type of classification called as parametric and the one which does not depend on statistical parameters are known as non-parametric classification. In this research, all these type of classifications have been executed. For supervised per pixel classification SVM has been used and for supervised but sub pixel classification LMM has been used. ICAMM is unsupervised sub pixel parametric classifier. Hence the efficacy of all these algorithms for different spatial and spectral resolutions has been studied.

A modified type of mapping has been performed to improve the spatial resolution of the hyperspectral data by means of super resolution mapping. When literature is having no or less work on multi-class super resolution mapping, a novel algorithm is required at this stage to tackle such situation. Also, if the boundaries between classes are irregular or atleast non-linear, the literature is having very algorithm to do spatial enhancement.

Without validation the research may not end. So at every stage of the information extraction accuracy assessment has been performed and analysed. For feature extraction, the analysis is mainly based on classification accuracy, time required to extract the features and duration of classification. For per pixel classification, conventional error matrix based accuracy assessment has been performed while for sub pixel classification, fuzzy error matrix based has been performed. For super resolution mapping technique, three types of accuracy has been performed in have an indepth study of that technique.

7.2 Summary of the Study

Hyperspectral data has been used largely by remote sensing community to extract useful information from earth atmosphere. But due to large information content in a hyperspectral data due to redundancy of information, further processing becomes complicated. This creates a necessity to reduce the hyperspectral data by some means without loss of much information. Hence, by two feature extraction techniques namely wavelet based feature extraction and PCA based feature extraction the hyperspectral data has been reduced. Both these are having different mathematical structures in which transformation based on multi-resolution and statistical transformation techniques have been employed, respectively. In each of these two techniques a couple of special sub classes have been studied. In wavelet transformation, Haar, Daubechies and Coiflet wavelets have been used along with their sub classes while in PCA based feature extraction technique, SPCA and SSPCA techniques have been employed. Among all these feature extractions, Daubechies wavelets second sub class first and second level decomposition levels (denoted as Daub2_1 and Daub2_2) performs better than the rest.

A collective study about classification of various spatial resolution hyperspectral data was missing from the literature and hence it triggered to do such an analysis for hyperspectral data. Hence, after feature extraction, the extracted features are classified by various categories of classifiers. Here three classification algorithms have been implemented which are SVM (per pixel, supervised), LMM (sub pixel and supervised in nature) and finally ICAMM (sub pixel but unsupervised classification). Here ICAMM alone is parametric but the other two classifications are non-parametric. Each of these algorithms performs best in extracting information within its limitations.

Once the classification is performed, the classified output goes to super resolution mapping technique. In the literature, only for binary class super resolution is mainly available or extraction of linear features is available. Hence a super resolution algorithm has been developed for multi-class super resolution and analysis has been made how well this algorithm super resolves mixed pixels. Since the three datasets, described in Chapter 3, varies not only in spatial and spectral resolutions but also the arrangement of classes the efficacy of the proposed super resolution mapping algorithm named as pixel filling algorithm has been analysed. This novel algorithm performs better for all the three hyperspectral datasets and one synthetic data when compared to pixel swapping algorithm.

7.3 Conclusions

Following broad conclusions have been drawn from the research carried out in this study,

i. The feature reduction technique using wavelet based reduces the data drastically and retrieves more information while decomposing the hyperspectral data upto two levels.

ii. The Haar wavelets and Daubechies wavelets are better for any type of hyperspectral dataset than Coiflets and PCA based techniques.

iii. When the decomposition level is getting increased by wavelet transformation, then the loss of information is also increasing in hyperspectral data.

iv. SSPCA based extraction technique is better than SPCA for all the datasets as the segmentation of bands collects information from the few necessary bands.

v. The classification accuracy of SVM is remarkable for all the three datasets even though few classes having less number of training samples.

vi. A small change in spectral curve with other curve of same class effects fractional change in proportion of class by using LMM.

vii. ICAMM has been able to detected new objects in case of all datasets, which was not possible using LMM. This proves the utility of ICAMM as a potential sub-pixel technique for hyperspectral datasets.

viii. Adaptation parameter value has been fixed to 0.7 in estimation of fractional abundance of hyperspectral data using ICAMM.

ix. ICAMM extracts two classes namely railway track and non-metallic road which could not be extracted by SVM and LMM due to the non-availability of pure pixels of the two classes.

x. In general, the super resolution mapping by pixel filling algorithm performs better for both zoom factors 3 and 5 than the pixel swapping algorithm. Particularly, the classes having linear and circular boundaries have been accurately mapped.

xi. The pixel filling algorithm works better for complicated dataset of HYPERION Roorkee and its surroundings dataset which has classes rolled together. The efficacy of the algorithm has been shown for this dataset.

xii. The pixel filling algorithm takes few seconds to produce super resolution map while the other algorithm pixel swapping takes more than a minute to produce a finer map.

7.4 Major Research Contributions

The major research contributions from the work carried out in this research are as follows:

i) A novel algorithm for super resolution mapping to map the features has been proposed for hyperspectral data.

ii) The applicability of wavelet based transformation using different wavelets for as a potential feature reduction technique for hyperspectral data has been demonstrated.

iii) The efficacy of ICAMM to produce sub-pixel outputs from hyperspectral data has been proven. It has further been shown that ICAMM can be effectively used to extract certain unidentfied features in the datasets.

7.5 Future Research

The research work presented in this thesis can be considered as contribution in the area of super resolution mapping. Although this research focus on feature extraction, classification and super resolution mapping some area of research needs further attention which have been identified and narrated as follows:

(i) The need of a new feature extraction technique will always be there in hyperspectral data.

(ii) Evaluation of feature extraction technique without finding classification is required.

(iii) LMM produces noisy fraction images which should be modified to produce noiseless fraction images.

(iv) The execution and derivation of ICAMM algorithm is very complicated and requires much attention to reduce the complication.

(v) Super resolution mapping may be extended for target/anomaly detection from hyperspectral data.

Bibliography

AKGUN, T., ALTUNBASAK, Y. and MERSEREAU, R.M., 2005, Super-resolution reconstruction of hyperspectral images, *IEEE Transactions on Image Processing*, **14(11)**, pp. 1860-1875.

APAN, A., HELD, A., PHINN, S. and MARKLEY, J., 2004, Detecting sugarcane 'orange rust' disease using EO-1 Hyperion hyperspectral imagery, *International Journal of Remote Sensing*, **25**, pp. 489–498.

ARCHIBALD, R. and FANN, G., 2007, Feature selection and classification of hyperspectral images with support vector machines, *IEEE Geoscience and Remote Sensing Letters*, **4(4)**, pp. 674–677.

ARORA, M. K. and TIWARI, K. C., 2013, Subpixel target enhancement in hyperspectral images, *Defence Science Journal*, **63(1)**, pp. 63-68.

ASNER, G.P. and HEIDEBRECHT, K.B., 2002, Spectral unmixing of vegetation, soil and dry carbon cover in arid regions: comparing multispectral and hyperspectral observations, *International Journal of Remote Sensing*, **23**, pp. 3939–3958.

ATKINSON, P.M., 2005, Super-resolution target mapping from soft classified remotely sensed imagery, *Photogrammetric Engineering and Remote Sensing*, **71**, pp. 839–846.

BAKOS, K. L. and GAMBA, P., 2011, Hierarchical hybrid decision tree fusion of multiple hyperspectral data processing chains, *IEEE Transactions on Geoscience and Remote Sensing*, **49(1)**, pp. 388- 394.

BAUDAT, G. and ANOUAR, F., 2000, Generalized discriminant analysis using a kernel approach, *Neural Computation*, **12**, pp. 2385–2404.

BAZI, Y. and MELGANI, F., 2006, Toward an optimal SVM classification system for hyperspectral remote sensing images, *IEEE Transactions on Geoscience and Remote Sensing*, **44(11)**, pp. 3374–3385.

BELLUCO, E., CAMUFFO M., FERRARI S., MODENESE L., SILVESTRI S., MARANI A. and MARANI M., 2006, Mapping salt-marsh vegetation by multispectral and hyperspectral remote sensing, *Remote Sensing of Environment*, **105**, pp. 54-67.

BENEDIKTSSON, J.A. and KANELLOPOULOS, I., 1999, Classification of multisource and hyperspectral data based on decision fusion, *IEEE Transactions on Geoscience and Remote Sensing*, **37(3)**, pp. 1367-1377.

BENEDIKTSSON, J.A., PALMASON, J.A. and SVEINSSON, J.R, 2005, Classification of hyperspectral data from urban areas based on extended morphological profiles, *IEEE Transactions on Geoscience and Remote Sensing*, **43(3)**, pp. 480-491.

DIAS, B.J.M., PLAZA, A., DOBIGEON, N., PARENTE, M., QIANDU, GADER, P. and CHANUSSOT, J., 2012, Hyperspectral Unmixing Overview: Geometrical, Statistical, and Sparse Regression-Based Approaches, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, **5(2)**, pp. 354-379.

BLACKBURN. G. A., 2007, Wavelet decomposition of hyperspectral data: a novel approach to quantifying pigment concentrations in vegetation, *International Journal of Remote Sensing*, **28(12)**, pp. 2831–2855.

BOSER. B.E., GUYON. I.M. and VAPNIK. V.N., 1992, A training algorithm for optimal margin classifiers, *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*, 27-29 July 1992, Pittsburgh, PA, USA.

BOUCHER, A.A. and KYRIAKIDIS, P.C., 2006, Super-resolution land cover mapping with indicator geostatistics, *Remote Sensing of the Environment*, **104**, pp. 264–282.

BOUCHER, A.A. and KYRIAKIDIS, P.C., 2007, Integrating fine scale information in super resolution land-cover mapping, *Photogrammetric Engineering and Remote Sensing*, **73**, pp. 913–921.

BRODLEY, C.E., FRIEDL, M.A. and STRAHLER, A.H., 1996, New approaches to classification in remote sensing using homogeneous and hybrid decision trees to map land cover, *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, **1**, pp. 532-534.

BROWN, M., LEWIS, H. and GUNN, S., 2000, Linear spectral mixture models and support vector machines for remote sensing, *IEEE Transactions on Geoscience and Remote Sensing*, **38(5)**, pp. 2346–2360.

BRUCE, L.M., Koger, C.H. and Li, J., 2002, Dimensionality Reduction of Hyperspectral Data Using Discrete Wavelet Transform Feature Extraction, *IEEE Transactions on Geoscience and Remote Sensing*, **40(10)**, pp. 2331–2338.

BRUZZONE, L. and SERPICO, S. B., 2000, A technique for feature selection in multiclass problems, *International Journal of Remote Sensing*, **21(3)**, pp. 549–563.

CAMPS-VALLS, G., GOMEZ-CHOVA, L., MUNOZ-MARI, J., VILA-FRANCES, J. and CALPE-MARAVILLA, J., 2004, Composite kernels for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, **3(1)**, pp. 93–97.

CHANUSSOT, J., BENEDIKTSSON, J. A. and FAUVEL, M., 2006, Classification of remote sensing images from urban areas using a fuzzy probabilistic model, *IEEE Geoscience and Remote Sens. Letters*, **3**(1), pp. 40-44.

CHENG, Q., VARSHNEY, P. K. and ARORA, M. K., 2006, Logistic regression for feature selection and soft classification of remote sensing data, *IEEE Transactions on Geosciences and Remote Sensing Letters*, **3(4)**, pp. 491-494.

CHERIYADAT. A. and BRUCE, L.M., 2003, Why principal component analysis is not an appropriate feature extraction method for hyperspectral data, *IEEE Proceeding of Geoscience and Remote Sensing Symposium, IGARSS' 03*, **6**, pp. 3420–3422.

CHI, M., FENG, R. and BRUZZONE, L., 2008, Classification of hyperspectral remote-sensing data with primal SVM for small-sized training dataset problem, *Advances in Space Research*, **41(11)**, pp. 1793–1799.

CLEVERS, J.G.P.W., VAN DER HEIJDEN, G.W.A.M., VERZAKOV, S. and SCHAEPMAN, M.E., 2007, Estimating grassland biomass using SVM band shaving of hyperspectral data, *Photogrammetric Engineering and Remote Sensing*, **73(10)**, pp. 1141–1148.

COMMON, P., 1994, Independent component analysis, A new concept? *Signal processing*, **36**, pp. 287-314.

CONGALTON, R.G., 1991, A review of assessing the accuracy of classifications of remotely sensed data, *Remote Sensing of Environment*, **46**, pp. 35-46.

DALPONTE, M., BRUZZONE, L., VESCOVO, L. and GIANELLE, D., 2009, The role of spectral resolution and classifier complexity in the analysis of hyperspectral images of forest areas, *Remote Sensing of Environment*, **113(11)**, pp. 2345–2355.

DEMIR, B. and ERTÜRK, S., 2007, Hyperspectral image classification using relevance vector machines, *IEEE Geoscience and Remote Sensing Letters*, **4(4)**, pp. 586–590.

DEMIR, B. and ERTÜRK, S., 2009, Clustering-based extraction of border training patterns for accurate SVM classification of hyperspectral images, *IEEE Geoscience and Remote Sensing Letters*, **6(4)**, pp. 840–844.

DU, Q. and YANG, H., 2008, Similarity-based unsupervised band selection for hyperspectral image analysis," *IEEE Geosciences and Remote Sensing Letters,* **5(4)**, pp. 564–568.

DUDA, R.O., HART, P. E. and STORK, D. G., 2001, Pattern Classification, 2nd edition, New York: John Wiley & Sons.

FAUVEL, M., BENEDIKTSSON, J.A., CHANUSSOT, J. and SVEINSSON, J.R., 2008, Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles, *IEEE Transactions on Geoscience and Remote Sensing*, **46(11)**, pp. 3804–3814.

FOODY, G. M., and ARORA, M. K., 1996, Incorporating mixed pixels in the training, allocation and testing stages of supervised classifications, *Pattern Recognition Letters*, **17(13)**, 1389-1398.

FOODY, G. M., LUCAS, R. M., CURRAN, P.J. and HONZAK, M., 1997, Non-linear mixture modeling without end-members using an artificial neural network, *International Journal of Remote Sensing,* **18**, pp. 937–953.

FOODY, G.M., MUSLIM, A.M. and ATKINSON, P.M., 2005, Super-resolution mapping of the waterline from remotely sensed data, *International Journal of Remote Sensing*, **26(24)**, pp. 5381–5392.

FUKUNAGA, K., 1990, Introduction to Statistical Pattern Recognition, 2nd edition, New York: Academic Press.

GENITHA, C.H. and VANI, K., 2010, Super resolution mapping of satellite images using Hopfield neural networks, *IEEE Conference on Recent Advances in Space Technology Services and Climate Change (RSTSCC),* pp. 114-118, 13-15 Nov 2010, Chennai.

GOEL, P.K., PRASHER, S.O., PATEL, R.M., LANDRY, J.A., BONNELL, R.B. and VIAU, A.A., 2003, Classification of hyperspectral data by decision trees and artificial neural networks to identify weed stress and nitrogen status of corn, *Computers and Electronics in Agriculture*, **39**, pp. 67–93.

GREEN, A., BERMAN, M., SWITZER, P. and CRAIG, M. D., 1988, A transformation for ordering multispectral data in terms of image quality with implications for noise removal, *IEEE Transactions on Geoscience and Remote Sensing*, 26(1), pp. 65–74.

GUO, B., GUNN, S., DAMPER, R. and NELSON, J., 2006, Band selection for hyperspectral image classification using mutual information, *IEEE Geosciences and Remote Sensing Letters*, 3(4), pp. 522–526.

HARSANYI J.C. and CHANG, C.I., 1994, Hyperspectral Image Classification and Dimensionality Reduction: An Orthogonal subspace projection approach, *IEEE Transactions on Geosciences and Remote Sensing,* ***32(4)***, pp.779-785.

HASTIE, T., BUJA, A. and TIBSHIRANI, R., 1995, Penalized discriminant analysis. *Annals of Statistics*, 23(1), pp. 73–102.

HEINZ, D.C. and CHANG, C.I., 2001, Fully constrained least squares linear mixture analysis for material quantification in hyperspectral imagery, *IEEE Transactions on Geoscience and Remote Sensing*, **39(3)**, pp. 529–545.

HUANG, C., DAVIS, L.S. and TOWNSHEND, J.R.G., 2002, An assessment of support vector machines for land cover classification, *International Journal of Remote Sensing*, **23(4)**, pp. 725–749.

HUGHES, G.F., 1968, On the mean accuracy of statistical pattern recognizers, *IEEE Transactions on Information Theory*, **14(1)**, pp. 55-63.

HYVÄRINEN, A. and OJA, E., 2000, Independent component analysis: Algorithms and applications, *Neural Networks: the Official Journal of the International Neural Network Society*, **13(4-5)**, pp. 411-430.

JAIN, A. and ZONGKER, D., 1997, Feature selection: Evaluation, application, and small sample performance, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19(2)**, pp. 153–158.

JENSEN. S. K and WALTS. F. A, 1997, Principal component analysis and canonical analysis in remote sensing, *Proceedings of American Society, Photogramm. 45th Annual Meeting*, pp. 337–348.

JIA, X. and RICHARDS, J. A., 1994, Efficient maximum likelihood classification for imaging spectrometer data sets, *IEEE Transactions on Geoscience and Remote Sensing*, **32(2)**, pp. 274-281.

JIA, X. and RICHARDS, J.A., 1999, Segmented principal components transformation for efficient hyperspectral remote-sensing image display and classification, *IEEE Transactions on Geoscience and Remote Sensing*, **37**, pp. 538–542.

JIMENEZ, L. O. and LANDGREBE, D. A, 1999, Hyperspectral data analysis and feature reduction via projection pursuit, *IEEE Transactions on Geoscience and Remote Sensing*, **37(6)**, pp. 2653–2667.

JIN, J., WANG, B. and ZHANG, L.M., 2010, A novel approach based on fisher discriminant null space for decomposition of mixed pixels in hyperspectral imagery, *IEEE Geoscience and Remote Sensing Letters*, **7**, pp. 699–703.

KAEWPIJIT, S., MOIGNE, J. L. and EL-GHAZAWI, T., 2003, Automatic reduction of hyperspectral imagery using wavelet spectral analysis, *IEEE Transactions on Geoscience and Remote Sensing*, **41(4)**, 863-871.

KASETKASEM, T., ARORA, M. K., VARSHNEY, P. K. and AREEKUL, V., 2011, Improving subpixel classification by incorporating prior information in linear mixture models, *IEEE Transactions on Geoscience and Remote Sensing*, **49(3)**, 1001–1013.

KASETKASEM, T., ARORA, M.K. and VARSHNEY, P.K., 2005, Super-resolution land cover mapping using a Markov random field based approach, *Remote Sensing of Environment*, **96**, pp. 302–314.

KAVZOGLU, T. and COLKESEN, I., 2009, A kernel functions analysis for support vector machines for land cover classification. *International Journal of Applied Earth Observation and Geoinformation*, **11(5)**, pp. 352–359.

KESHAVA, N. and MUSTARD, J.F., 2002, Spectral unmixing, *IEEE Signal Processing Magazine*, **19(1)**, pp. 44-57.

KESHAVA, N., 2003, A survey of spectral unmixing algorithms, *Lincoln Laboratory Journal*, **14(1)**, pp. 55-78.

KESHAVA, N., 2004, Distance metrics and band selection in hyperspectral processing with applications to material identification and spectral libraries, *IEEE Transactions on Geoscience and Remote Sensing*, 42(7), pp. 1552–1565.

KNERR, S., PERSONNAZ, L. and DREYFUS. G., 1990. Single-layer learning revisited: A stepwise procedure for building and training neural network. In *Neurocomputing: Algorithms, architectures and applications*, Fogelman, J. (ed), NATO ASI, 41–50. Berlin: Springer-Verlag.

KORYCINSKI, D, CRAWFORD, M. M. and BARNES, J. W., 2003a, Adaptive feature selection for hyperspectral data analysis, *Proceedings of the SPIE Conference on Image and Signal Processing for Remote Sensing IX*, Barcelona, Spain, pp. 213–225.

KORYCINSKI, D., CRAWFORD, M. M., BARNES, J. W. and GHOSH, J., 2003b, Adaptive feature selection for hyperspectral data analysis using a binary hierarchical classifier and tabu search, *Proceedings of the 2003 IEEE International Geoscience and Remote Sensing Symposium*, **1**, pp. 297–299, Toulouse, France, 21–25 July.

KUMAR, S., GHOSH, J. and CRAWFORD, M. M., 2001, Best-bases feature extraction algorithms for classification of hyperspectral data, *IEEE Transactions on Geoscience and Remote Sensing*, **39(7)**, pp. 1368–1379.

KWON, H. and NASRABADI, N.M., 2005, Kernel RX-algorithm: a nonlinear anomaly detector for hyperspectral imagery, *IEEE Transactions on Geoscience and Remote Sensing,* **43(2)**, pp. 388-397.

LANDGREBE, D. A., 2003, Signal Theory Methods in Multispectral Remote Sensing, Wiley-Interscience, New York.

LEE, C. and LANDGREBE, D. A., 1993, Feature extraction based on decision boundaries, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15(4)**, pp. 388–400.

LEE, T. W., GIROLAMI, M. and SEJNOWSKI, T. J., 1999, Independent component analysis using an extended informax algorithm for mixed sub-Gaussian and super-Gaussian sources, *Neural Computation*, **11**(2), pp. 417-441.

LEE, T. W., LEWICKI, M. S. and SEJNOWSKI, T. J., 2000, ICA Mixture Models for Unsupervised Classification of Non-Gaussian Classes and Automatic Context Switching in Blind Signal Separation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(10), 1078–1089.

LEE, T.W. and LEWICKI, M.S., 2002, Unsupervised image classification, segmentation and enhancement using ICA mixture models, *IEEE Transactions on Image Processing: a publication of the IEEE Signal Processing Society*, **11(3)**, 270–279.

LING, F. and FU, B., 2009, Super-resolution mapping of urban buildings with remotely sensed imagery based on prior shape information, *IEEE Conference on Urban Remote Sensing Joint Event.*

LING, F., DU, Y., XIAO, F., XUE, H. and WU, S., 2010, Super resolution land cover mapping using multiple sub-pixel shifted remotely sensed images, *International Journal of Remote Sensing*, **31(19)**, pp. 5023-5040.

LIU, H. and MOTODA, H., 1998, Feature extraction, construction and selection: A data mining perspective. Norwell, MA:Kluwer.

LIU, M., LIU, X., WU, L., DUAN, L. and ZHONG, B, 2011, Wavelet based detection of crop zinc stress assessment using hyperspectral reflectance, *Computers & Geosciences*, **37(9)**, pp. 1254–1263.

LU, D., BATISTELLA, M. and MORAN, E., 2004, Multitemporal spectral mixture analysis for Amazonian land-cover change detection, *Canadian Journal of Remote Sensing*, **30(1)**, pp. 87–100.

LUCAS, N.S., SHANMUGAM, S. and BARNSLEY, M.J., 2002, Sub-pixel habitat mapping of a coastal dune ecosystem, *Applied Geography*, **22**, pp. 253-270.

MAKIDO, Y., SHORTRIDGE, A. and MESSINA, J.P., 2007, Assessing alternatives for modeling the spatial distribution of multiple land-cover classes at sub-pixel scales. *Photogrammetric Engineering and Remote Sensing*, **73**, pp. 935–943.

MANOLAKIS, D., SIRACUSA, C. and SHAW, G., 2001, Hyperspectral subpixel target detection using the linear mixing model, *IEEE Transactions on Geoscience and Remote Sensing*, **39(7),** pp. 1392–1409.

MELGANI, F. and BRUZZONE, L., 2002, Support vector machines for classification of hyperspectral remote-sensing images, *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, **1**, pp. 506-508.

MELGANI, F. and BRUZZONE, L., 2004, Classification of hyperspectral remote sensing images with support vector machines, *IEEE Transactions on Geoscience and Remote Sensing*, **42(8)**, 1778–1790.

MERTENS, K.C., VERBEKE, L.P.C., DUCHEYNE, E.I. and DE WULF, R.R., 2003, Using genetic algorithms in sub-pixel mapping, *International Journal of Geographical Information Science*, **24**, pp. 4241–4247.

MERTENS, K.C., VERBEKE, L.P.C., WESTRA, T. and DE WULF, R. R., 2004, Sub-pixel mapping and sub-pixel sharpening using neural network predicted wavelet coefficients. *Remote Sensing of Environment*, **91**, pp. 225–236.

MIANJI, F. A., ZHANG, Y. and BABAKHANI, A., 2009a, Super resolution of hyperspectral images using backpropagation neural network, *2nd International Workshop on Nonlinear Dynamics and Synchronization*, INDS '09.

MIANJI, F.A., ZHANG, Y. and BABAKHANI, A., 2009b, Optimum method selection for resolution enhancement of hyperspectral imagery, *Information Technology Journal*, **8(3)**, pp. 263-274.

MIANJI, F.A., ZHANG, Y. and BABAKHANI, A., 2010, Key information retrieval in hyperspectral imagery through spatial-spectral data fusion*, Radio Engineering* **19(4)**, pp. 734–744.

MIKA, S., RATSCH, G., SCHOLKOPF, B., SMOLA, A., WESTON, J. and MULLER, K.R., 1999 Invariant feature extraction and classification in kernel spaces, *Advances in Neural Information Processing Systems*,**12**, MIT Press, Cambridge, MA.

MOHAN, B.K., 2000, Classification of Remotely Sensed Images using Artificial Neural Networks , *IETE Journal of Research*, **46(5)**, pp. 401-410.

MULLER, K. R., MIKA, S., RATSCH, G., TSUDA, K. and SCHOLKOPF, B., 2001, An introduction to kernel-based learning algorithms, *IEEE Transactions on Neural Networks*, 12(2), pp. 181–201.

NEVILLE, R.A., LEVESQUE, J., STAENE, K., NADEAU, C., HAUFF, P. and BORSTAD, G.A., 2003, Spectral unmixing of hyperspectral imagery for mineral exploration: comparison of results from SFSI and AVIRIS, *Canadian Journal of Remote Sensing*, **29**, pp. 99–110.

NIEDERMEIER, A., ROMANEEßEN, E. and LENHER, S., 2000, Detection of coastlines in SAR images using wavelet methods, *IEEE Transactions on Geoscience & Remote Sensing*, **38**, pp. 2270–2281.

OKIN, G.S., ROBERTS, D.A., MURRAY, B. and OKIN, W.J., 2001, Practical limits on hyperspectral vegetation discrimination in arid and semiarid environments, *Remote Sensing of Environment*, **77**, pp. 212–225.

PAL, M. and MATHER P.M., 2005, Support vector machines for classification in remote sensing, *International Journal of Remote Sensing*, **26(5)**, pp. 1007–1011.

PAL, M., 2006, Support vector machine-based feature selection for land cover classification: A case study with DAIS hyperspectral data. *International Journal of Remote Sensing*, **27 (14)**, pp. 2877–2894.

PATHAK, V. and DIKSHIT, O., 2003, Segment based classification using IRS-1C, LISS-III data, *IEEE Geoscience and Remote Sensing Symposium*, *IGARSS' 03* **6**, pp. 3513-3515.

PATHAK, V. and DIKSHIT, O., 2004a, Per-field classification of Indian urban environment using IRS-1C satellite data, *3rd International Symposium on New Technologies for Urban Safety of Mega Cities in Asia, October 18-19,* Agra.

PATHAK, V. and DIKSHIT, O., 2004b, Segment based classification of Indian urban environment, *3rd International Symposium on New Technologies for Urban Safety of Mega Cities in Asia, October 18-19,* Agra.

PATHAK, V. and DIKSHIT, O., 2010, A new approach for finding an appropriate combination of texture parameters for classification, *Geocarto International*, **25(4)**, pp. 295-313.

PENG, J., SHEN, H., HE, S., WU, J., 2013, Soil moisture retrieving using hyperspectral data with the application of wavelet analysis, *Environmental Earth Sciences,* 69(1), pp. 279-288.

PLATT, J. C., 1999, Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In A.Smola(ed.), Advances in Large Margin Classifiers, MIT Press, Cambridge, MA.

PLAZA, A., BENEDIKTSSON, J.A., BOARDMAN, J.W., BRAZILE, J., BRUZZONE, L., CAMPS-VALLS, G., CHANUSSOT, J., FAUVEL, M., GAMBA, P., GUALTIERI, A., MARCONCINI, M., TILTON, J.C. and TRIANNI, G., 2009, Recent advances in techniques for hyperspectral image processing, *Remote Sensing of Environment*, **113(1)**, pp. 110–122.

PLUMBLEY, M.D., 2003, Algorithms for nonnegative independent component analysis. *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, **14**(3), pp. 534-43.

PUDIL, P., NOVOVICOVA, J. and KITTLER, J., 1994, Floating search methods in feature selection, *Pattern Recognition Letters*, **15**, pp. 1119–1125.

REDDY, G.M. and MOHAN, B.K., 2005, **Classification of Remotely Sensed Images Using Neural-Network Ensemble and Fuzzy Integration,** *Pattern Recognition and Machine Intelligence,* S.K. Pal, S. Bandyopadhyay and S. Biswas (eds.), LNCS 3776, Springer, pp. 350-355.

ROBILA, S.A. and VARSHNEY, P.K., 2002, Target detection in hyperspectral images based on independent component analysis, *Proceedings of SPIE Automatic Target Recognition XII*, **4726**, pp.173-182

SAHOO, B.C., OOMMEN, T., MISRA, D. and NEWBY, G., 2007, Using the one-dimensional s-transform as a discrimination tool in classification of hyperspectral images, *Canadian Journal of Remote Sensing*, **33(6)**, pp. 551–560.

SAWANT, N., CHANDRAN, S. and MOHAN, B.K., 2006, Retrieval of Images for Remote Sensing Applications**,** *Computer Vision, Graphics and Image Processing*, Prem Kalra and Shmuel Peleg (ed.), LNCS 4338, pp. 849-860.

SCHMIDT, K.S., SKIDMORE, A.K., KLOOSTERMAN, E.H., VAN OOSTEN, H., KUMAR, L. and JANSSEN, J.A.M., 2004, Mapping coastal vegetation using an expert system and hyperspectral imagery, *Photogrammetric Engineering and Remote Sensing*, **70**, pp. 703–715.

SERPICO, S. and MOSER, G., 2007, Extraction of spectral channels from hyperspectral images for classification purposes, *IEEE Transactions on Geosciences and Remote Sensing*, 45(2), pp. 484–495.

SERPICO, S. B., D'INCA, M., MELGANI, F. and MOSER, G., 2003, A comparison of feature reduction techniques for classification of hyperspectral remote sensing data, in Proceedings SPIE 4885, *Image and Signal Processing Remote Sensing VIII*, pp. 347–358.

SETTLE, J.J. and DRAKE, N.A., 1993, Linear mixing and the estimation of ground cover proportions, *International Journal of Remote Sensing*, **14**, pp. 1159–1177.

SHAH, C. A., VARSHNEY, P. K., 2004, A higher order statistical approach to spectral unmixing of remote sensing imagery, *Proceedings of Geoscience and Remote Sensing Symposium*, IGARSS' 2, pp. 1065-1068.

SHAH, C., ARORA, M. and VARSHNEY, P., 2004, Unsupervised classification of hyperspectral data: an ICA mixture model based approach, *International Journal of Remote Sens.*, ***25***(2), 481-87.

SHAH, C.A., VARSHNEY, P.K. and ARORA, M.K., 2007, ICA mixture model algorithm for unsupervised classification of remote sensing imagery, *International Journal of Remote Sensing*, **28(8)**, pp.1711–1731.

SHANMUGAM, P., AHN, Y.H. and SHANMUGAM, S., 2006, A comparison of the classification of wetland characteristics by linear spectral mixture modelling and traditional hard classifiers on multispectral remotely sensed imagery in southern India, *Ecological Modelling*, **194(4)**, pp.379–394.

SHANMUGAM, S. and ABHISHEKH, P.V., 2006, Spectral unmixing of hyperspectral data to map bauxite deposits, *Proceedings of SPIE 6405, Multispectral, Hyperspectral, and Ultraspectral Remote Sensing Technology, Techniques, and Applications*.

SHIPPERT, P., 2004, Introduction to Hyperspectral Image Analysis, IDL Training Manual, Research Systems, Inc.

SHEN, Z., QI, J. and WANG, K., 2009, Modification of pixel-swapping algorithm with initialization from a sub-pixel/pixel spatial attraction model, *Photogrammetric Engineering and Remote Sensing*, **75(5)**, pp. 557-561.

SIEDLECKI. W. and SKLANSKY. J., 1989, A note on genetic algorithms for large-scale feature selection, *Pattern Recognition Letters*, **10**, pp.335–347.

SOHN, Y. and REBELLO, N.S., 2002, Supervised and unsupervised spectral angle classifiers, *Photogrammetric Engineering and Remote Sensing*, **68**, pp.1271–1281.

SOHN, Y., MORAN, E. and GURRI, F., 1999, Deforestation in north-central Yucatan (1985–1995): mapping secondary succession of forest and agricultural land use in Sotuta using the cosine of the angle concept, *Photogrammetric Engineering and Remote Sensing*, **65**, pp.947–958.

SOMOL, P., PUDIL, P., NOVOVICOVA, J. and PACLIK, P., 1999, Adaptive floating search methods in feature selection, *Pattern Recognition Letters*, 20, pp. 1157–1163.

STORY, M. and CONGALTON, R., 1986, Accuracy assessment: a user's perspective, *Photogrammetry Engineering and Remote Sensing*, **52(3)**, pp.397-99.

TATEM, A.J., LEWIS, H.G., ATKINSON, P.M. and NIXON, M.S., 2003, Increasing the spatial resolution of agricultural land cover maps using a hopfield neural network, *International Journal of Geographical Information Science*, **17(7)**, pp. 647–672.

THORNTON, M. W., ATKINSON, P. M. and HOLLAND, D. A., 2006, Sub-pixel mapping of rural land cover objects from fine spatial resolution satellite sensor imagery using super-resolution pixel-swapping, *International Journal of Remote Sensing,* **27(3)**, pp. 473-491.

TIWARI, K.C., ARORA, M.K. and SINGH, D., 2007, Subpixel target enhancement using super resolution technique, *Photogrammetric Engineering and Remote Sensing*, 63(1), pp.63-68.

TSAI, F., LIN, E.K. and YOSHINO, K., 2007, Spectrally segmented principal component analysis of hyperspectral imagery for mapping invasive plant species, *International Journal of Remote Sensing*, **28(5)**, pp. 1023–1039.

VAN DER MEER, F.D. and JONG D.S.M., 2001, Introduction. In: van der Meer FD, de Jong SM (eds), Imaging Spectrometry: Basic Principles and Prospective Applications, Kluwer Academic Publishers, Dordrecht, The Netherlands, pp. 21-23.

VAPNIK. V.N, 1995, The nature of statistical learning theory, New York: Springer.

VAPNIK. V.N, 1998, Statistical learning theory, New York: Wiley.

VAPNIK. V.N, 1999, An overview of statistical learning theory, *IEEE Transactions on Neural Networks*, **10**, pp. 988-99.

VERHOEYE, J. and WULF D.G., 2002, Land cover mapping at sub-pixel scales using linear optimization techniques, *Remote Sensing of Environment*, **79(1)**, pp.96–104.

VILLA, A., CHANUSSOT, J., BENEDIKTSSON, J. A. and JUTTEN, C., 2011, Spectral unmixing for the classification of hyperspectral images at a finer spatial resolution, *IEEE Journal of Selected Topics in Signal Process*, **5(3)**, pp.521–533.

WALKER, J. S., 2008, A Primer on Wavelets and Their Scientific Applications, Second Edition, Chapman & Hall/CRC, Taylor & Francis Group.

WANG J. and CHANG C., 2006, Independent Component Analysis based dimensionality reduction with applications in Hyperspectral image analysis, *IEEE Transactions on Geosciences and Remote Sensing,* **44(3),** pp.1586-1686.

WANG, C., JINCHEN and WANG R., 2012, Improving hyperspectral data classification by fusing band and wavelet features, *International Journal of Remote Sensing,* **33(10)**, pp. 3268-3285.

WATANACHATURAPORN, P. and ARORA, M.A., 2004, Support vector machines for classification of multi- and hyperspectral data, *Advanced Image Processing Techniques for Remotely Sensed Hyperspectral Data,* P.K. Varshney and M.K. Arora (eds), Springer-Verlag, pp. 237–255.

WATANACHATURAPORN, P., ARORA, M.K. and VARSHNEY, P.K., 2008, Multisource classification using support vector machines: an empirical comparison with decision tree and neural network classifiers, *Photogrammetric Engineering & Remote Sensing*, **74(2)**, pp. 239–246.

XU, M., WATANACHATURAPORN, P., VARSHNEY, P. K., and ARORA. M. K., 2005, Decision tree regression for soft classification of remote sensing data, *Remote Sensing of Environment*, **97(3)**, pp. 322-36.

YU, B., OSTLAND, I. M., GONG, P. and PU, R., 1999, Penalized discriminant analysis of in situ hyperspectral data for conifer species recognition, *IEEE Transactions on Geoscience and Remote Sensing*, **37(5)** pp. 2569–2577.

ZHANG, J., YUAN, L., PU, R., LORAAMM, R.W., YANG, G. and WANG, J, 2014, Comparison between wavelet spectral features and conventional spectral features in detecting yellow rust for winter wheat, *Computers and Electronics in Agriculture*, **100**, pp. 79–87.

ZORTEA, M. and HAERTEL, V., 2004, Experiments on feature extraction in remotely sensed hyperspectral image data, *Proceedings of IGARSS*, Anchorage, AK, pp. 964–967.

Research Publications

Journal Publication:

Prabhu. N and Manoj K. Arora, 2014, “Comparison of fractional abundance estimation of hyperspectral data by using LMM and ICAMM”, *IEEE Journal: Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS)* (Communicated).

Prabhu. N, Manoj K. Arora and Balasubramanian. R, “Exploring the utility of wavelet techniques in feature extraction of hyperspectral data”, (To be communicated)

Prabhu. N, Manoj K. Arora and Balasubramanian. R, “Segmenting spectral signatures for feature extraction of hyperspectral data”, (To be communicated)

Prabhu. N, Manoj K. Arora, Balasubramanian. R, and Kapil Gupta, “Spatial resolution enhancement mapping of hyperspectral image via pixel filling algorithm”, (To be communicated)

Conference Publication:

Prabhu. N, Manoj K. Arora, Balasubramanian. R and Kapil Gupta, 2013, “An ICA mixture model based approach for sub-pixel classification of hyperspectral data”, 3rd International Conference on Soft Computing for Problem Solving (SocProS–2013), Dec 26 – 28, 2013. Greater Noida Extension Centre of IIT Roorkee, India.

Prabhu. N, Manoj K. Arora, Balasubramanian. R and Kapil Gupta, 2013, Binary Weights to Neighbouring Pixels Based Resolution Enhancement Mapping of Hyperspectral Image. IEEE 2nd International Conference on Image Information Processing (ICIIP–2013), Dec 09 – 11, 2013. Waknaghat, Shimla, India.