CROP MONITORING AND RECOMMENDATION SYSTEM USING MACHINE LEARNING TECHNIQUES

A PROJECT REPORT

Submitted by

S KRISHNA PRASAD (2013503514)
B SIVA SREEDHARAN (2013503525)
S JAISHANTH (2013503565)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

MADRAS INSTITUTE OF TECHNOLOGY, CHENNAI

ANNA UNIVERSITY: CHENNAI 600 025

April 2017

BONAFIDE CERTIFICATE

Certified that this project report titled "**CROP MONITORING AND RECOMMENDATION SYSTEM USING MACHINE LEARNING TECHNIQUES**" is a bonafide work done by "**S.KRISHNA PRASAD (2013503514), B.SIVASREEDHARAN (2013503525) and S.JAISHANTH (2013503565)**" under my supervision, in partial fulfilment for the award of the degree of Bachelor of Engineering in Computer Science and Engineering. Certified further, that to the best of my knowledge the work reported here in does not form part or full of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion to this or any other candidate.

SIGNATURE	**SIGNATURE**
Dr. P. ANANDHAKUMAR	**Dr. P.VARALAKSHMI**
HEAD OF THE DEPARTMENT	**SUPERVISOR**
Department of Computer Technology,	Associate Professor,
Madras Institute of Technology,	Department of Computer Technology,
Anna University,	Madras Institute of Technology,
Chennai-600 044.	Anna University,
	Chennai-600 044.

Date:

Place

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	**iii**
	LIST OF ABBREVIATIONS	**vii**
	LIST OF FIGURES	**viii**
	LIST OF TABLES	**ix**
1	**INTRODUCTION**	1
	1.1 MACHINE LEARNING	1
	1.1.1 Types of Problems and tasks	2
	1.1.2 Support Vector Machine	2
	1.1.2.1 Advantages	3
	1.1.3 Spatial Spectral classification	4
	1.1.4 Simple Linear Iterative Clustering	4
	1.2 ADABOOST MODEL	6
	1.3 MULTIPLE MORPHOLOGICAL COMPONENT ANALYSIS	7
	1.4 OBJECTIVE OF PROPOSED WORK	7
	1.5 SCOPE	8
	1.8 THESIS OVERVIEW	8
2	**LITERATURE SURVEY**	9
	2.1 IMAGE DENOISING	9
	2.2 IMAGE CLASSIFICATION	10
	2.3 CROP CLUSTERING	11
	2.4 WAVELET TRANSFORMS	13
	2.5 PREDICTION ALGORITHM	14
	2.6 OVERVIEW	16

3 PROPOSED WORK 17

3.1 PROPOSED SYSTEM ARCHITECTURE 17

3.2 MODULE DESIGN 18

3.2.1 Image pre-processor 18

3.2.2 Image Segmentation – SLIC 18

3.2.3 SVM prefixed by SSSE 18

3.2.4 Image Classifier Evaluator 18

3.2.5 Recommendation 19

3.2.6 Calculating NDVI 19

3.2.7 NDVI VS LAI Relationship 19

3.2.8 Yield Calculation based on NDVI 19

3.3 SYSTEM DEVELOPMENT 19

3.4 ALGORITHM AND WORKING OF EACH MODULE 21

3.4.1 Support Vector Machine 21

3.4.2 Image pre-processor 22

` 3.4.2.1 Algorithm Used: Multiple morphological component analysis 21

3.4.3 Image Classifier Evaluator 25

3.4.4 Recommendation Model Evaluator 32

4 IMPLEMENTATION AND RESULTS 34

4.1 TOOLS 34

4.1.1 MATLAB 34

4.1.1.1 Features 34

4.1.2 PyCharm 35

4.1.2.1 Features 35

4.1.3 Matplotlib 36

4.2 DEPLOYMENT DETAILS 37

	4.3 DATASET	37
	4.4 PRPROCESSOR	39
	4.5 IMAGE CLASSIFIER	41
	4.6 CROP YIELD PREDICTION	43
	4.7 RELEVANCE PREDICTION	44
	4.8 PERFORMANCE MEASURES	46
	4.9 YIELD PREDICTION	47
	4.10 CLASSFICATION	49
5	**CONCLUSIONS AND FUTURE WORK**	53
6	**REFERENCES**	54

LIST OF ABBREVIATIONS

Abbreviation	**Expansion**
CNN	Convolutional Neural Network
SVM	Support Vector Machine
NDVI	Normalized difference vegetation index
LAI	Leaf Area Index
MMCA	Multiple morphological component analysis
FBS	Feature Band Extraction
KNN	K Nearest neighbour
MODIS	Moderate-resolution Imaging Spectroradiometer
RBF	Radial Basis Function
SSSE	Spatial Spectral Schrodinger Eigenmaps
LE	Laplacian Eigenmaps
SLIC	Simple Linear iterative clustering
OOC	Object oriented classification
CPS	Class Pair separability
DMP	Differential morphological profiles
SAR	Synthetic aperture radar
SLIC	Simple Linear Iterative Clustering
MMCA	Multiple Morphological Component Analysis
SE	Schrodinger Eigen maps
SM	Shi – Malik
GB	Gilles – Bowles
HZYZ	Hou-Zhang- Ye-Zheng
BE	Benedetto et al

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
3.1	Architecture Diagram	17
3.2	Flow Diagram	20
4.1	Pre-processor Module	39
4.2	Original image	40
4.3	Mid-Stretch Filter	40
4.4`	Noise removal using grey	41
4.5	Classification Image Results	42
4.6	Classification Accuracy Measures	43
4.7	Training data execution results	44
4.8	Crop Yield Result	44
4.9	Crop soil data set	45
4.10	Relevance prediction for red soil	45
4.11	Training set regression analysis	47
4.12	Testing Regression analysis	47
4.13	True vs Predicted Yield Analysis	48
4.14	Overall Accuracy	50
4.15	Average Specificity	50
4.16	Average Precision	51
4.17	Average Accuracy	51
4.18	Average Sensitivity	52

LIST OF TABLES

TABLE NO.	TITLE	PAGE NO.
4.1	DATASET	38
4.2	CLASSIFICATION ACCURACIES	49
4.3	ANALYSIS MATHODS	49

ABSTRACT

This document proposes a crop recommendation system using Spectral spatial classification and Support Vector Machine (SVM).The farmer provides the crop field image as an input to the application. In the pre-processing stage, Denoising is done using Multiple morphological component analysis (MMCA) and as a result, filtering the image retaining its necessary portions. SVM prefixed by Spatial Spectral Schrodinger Eigen Maps (SSSE) is used as a classification method wherein partial knowledge propagation is leveraged to improve the classification accuracy.

The classified image along with the Ground truth statistical data containing the weather, crop yield, state & county wise crops are used to predict the yield of a particular crop under a particular weather condition. This predictive model used AdaBoost classifier. Crop recommendation is facilitated then by collaborative filtering. Further scope of the project would extend to predictive analytics on the commodity market of the goods grown in the agricultural fields to predict its waxing and waning.

ACKNOWLEDGEMENT

We are highly indebted to our respectable Dean, **Dr. A. RAJADURAI** and to our reputable Head of the Department **Dr. P. ANANDHAKUMAR**, Department of Computer Technology, MIT Campus, Anna University for providing us with sufficient facilities that contributed to success in this endeavor.

We would like to express my sincere thanks and deep sense of gratitude to our Supervisor, **Dr. P. VARALAKSHMI** for her valuable guidance, suggestions and constant encouragement which paved way for the successful completion of this phase of project work.

We would be failing in our duty, if we forget to thank all the teaching and non-teaching staff of our department, for their constant support throughout the course of our project work.

S.KRISHNA PRASAD
B.SIVA SREEDHARAN
S. JAISHANTH

Chapter 1
Introduction

This chapter gives a description about machine learning, tasks involved, models, advantages, disadvantages, objective of the proposed work and thesis overview.

This document describes a crop yield prediction and recommendation system using spatial spectral classification and AdaBoost classifier. The classified crop field images along with the historical weather and yield data are modelled to obtain the predicted crop yield and recommend suitable crops for a particular field. This in turn involves machine learning and image processing for classification and prediction.

1.1 Machine Learning

Machine Learning is a field of Computer Science, where new developments evolve at recent times, and also helps in automating the evaluation and processing done by the mankind, thus reducing the burden on the manual human power. According to techtarget, Machine learning is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed.

Machine learning focuses on the development of computer programs that can change when exposed to new data. Finding out the suitable crops based on the soil's appearance becomes tedious for novice farmers. There also exists a need to prevent the agricultural decay.

Effective utilization of agricultural land is crucial for ensuring food security of a country. In this document, we propose a crop recommendation system using spectral spatial classification and AdaBoost meta-algorithm.

1.1.1 Types of Problems and tasks

Machine learning tasks are typically classified into three broad categories, depending on the nature of the learning "signal" or "feedback" available to a learning system. These are

- **Supervised learning**: The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.

- **Unsupervised learning**: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).

- **Reinforcement learning:** A computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle), without a teacher explicitly telling it whether it has come close to its goal. Another example is learning to play a game by playing against an opponent

1.1.2 Support Vector Machine (SVM)

SVM constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class, since in general the larger the margin the lower the generalization error of the classifier.

The computational load should be sensible, the mappings are used by the SVM scheme to ensure the dot products will be computed in terms of the variable in the original scope, for that a kernel function **k(x,y)** selected to get the optimal computational time.

The higher-dimensional space in the hyper planes is distinct as the set of points whose dot product with a vector in that space is constant. These vectors in the hyper planes defining the hyper planes can be chosen to be linear combinations with parameters of images of feature vectors that occur in the data base. With this choice of a hyperplane, the points in the feature space that are mapped into the hyperplane are defined by the relation: The equation of the output from a linear SVM is

$$\mathbf{u = w . x - b}$$

Where w is the normal vector of the hyperplane, and x is the input vector.

1.1.2.1 Advantages

Support vector machine is one of the most widely used classification algorithms due to the advantages it enjoys which are as follows:

- SVMs are helpful in text and hypertext categorization as their application can significantly reduce the need for labeled training instances in both the standard inductive and transductive settings.
- Classification of images can also be performed using SVMs. Experimental results show that SVMs achieve significantly higher search accuracy than traditional query refinement schemes after just three to four rounds of

relevance feedback. This is also true of image segmentation systems, including those using a modified version SVM.

1.1.3 Spatial Spectral Classification (SSSE)

Schrodinger Eigenmaps (SE) has recently arose as a powerful graph-based system for semi-supervised manifold learning and recovery. The spatial spectral is extended the Laplacian of a graph which is constructed from hyper spectral imagery to include barrier or cluster potentials, Schrodinger Eigenmaps will enable machine learning techniques that employ the expert or label the information provided at a subset of pixels.

The Non - diagonal potentials can be used within the Schrodinger Eigenmaps framework in a way that allows for the incorporation of spatial and spectral information in unsupervised manifold learning and recovery. The non - diagonal potentials scramble spatial proximity then the output is combined with the spectral proximity information in the original graph which will yields a framework that is competitive with state-of-the-art spectral and spatial fusion approaches for clustering and subsequent classification of hyperspectral image data.

1.1.4 Simple Linear Iterative Clustering (SLIC)

Simple Linear Iterative Clustering is a simple and efficient method to decompose an image in visually homogeneous regions. It is based on a spatially localized version of k-means clustering. SLIC is similar to mean shift or quick shift, where each pixel is associated to a feature vector and then k-means clustering is run on those.

$$\Psi(x,y) = \begin{bmatrix} \lambda x \\ \lambda y \\ I(x,y) \end{bmatrix}$$

The spatial and appearance components of the feature vectors are imposed a degree of spatial regularization to the extracted regions are balanced by the coefficient λ.

The nominal size of the region is regionSize and the strength of the spatial regularization is regularizer. The image is first divided into a grid with step regionSize. The center of each grid tile is used to initialize a corresponding k-means up to a small shift to avoid image edges. By using the Lloyd algorithm, the k-means centers and clusters are refined which will yield the segmented image

For the restriction and simplification in Simple Linear Iterative Clustering, during the k-means iterations each pixel can be assigned to only the 2 x 2 centers corresponding to grid tiles adjacent to the pixel.

The trade-off between clustering appearance and spatial regularization is set by the parameter regularizer. The coefficient λ is obtained by using

$$\lambda = \frac{\mathtt{regularizer}}{\mathtt{regionSize}}$$

Simple Linear Iterative Clustering optionally removes any segment after the k-means step whose area is smaller than a threshold minRegionSize by merging them into larger ones.

K-means uses the standard LLoyd algorithm alternating assigning pixels to the closest centers a re-estimating the centers as the average of the corresponding feature vectors of the pixel assigned to them. The only difference compared to standard k-means is that each pixel can be assigned only to the center originated from the neighbour tiles. This guarantees that there are exactly four pixel-to-center comparisons at each round of minimization.

SLIC eliminates any connected region after k-means has converged whose area is less than minRegionSize pixels. This is done by avidly merging regions to neighbour ones: the pixels by pixels are scanned in lexicographical order and the corresponding connected components are visited. If a region has already been visited, it is skipped; if not, its area is computed and if this is less than minRegionSize its label is changed to the one of a neighbour region at p that has already been visited.

1.2 AdaBoost Model

Boosting can be defined as a means for creating a strong classifier using a set of weaker classifiers. It can be explained as a method in which a second model is use for rectifying the errors in the first model i.e., the training model. AdaBoost is used in this context for binary classification. The first model constituting the training dataset is weighted. The primary weight is modelled to:

$$\mathbf{weight\ (y_i) = 1/n}$$

Where y_i is the i^{th} training instance and n is the total number of instances.

Weak models are sequentially added and trained using the training data as calculated by the above formula. The process is continued until a predefined number of weak learning instances have been created or no further enhancement can be made on the training data set.

Predictions are made by calculation of the weighted averages if the weak classifiers. The predicted values are weighted by each weak learners stage value. The prediction for the ensemble model is calculate as a the sum of the weighted prediction values. If the sum is +ve, then the first class is predicted, else the second class is predicted.

1.3 Multiple Morphological Component Analysis (MMCA)

Remote sensing data has multiple issues related to the dimensionality of the original set data. This happens due to the large spectral resolution in the hyperspectral images. This dimensionality reduction uses the minimum noise fraction (MNF) to reduce the total existing components to a small number of bands. MMCA aims at decomposing the remote sensing image into texture and smoothness tuples using feature extraction as a prefix method. These tuples convey the essence of the original image as a linear combination as shown below:

$$\mathbf{Y = y_a + y_b + n}$$

Where n is the approximation image residue value. a and b denote the smoothness and texture components respectively. Content parameters, coarseness directional vectors and contrast are the additional parameters to be taken into account with respect to MMCA.

1.4 Objective of proposed work

This project aims at predicting the crop yield at a particular weather condition and thereby recommending suitable crop for that field. It involves the following steps.

- Collect satellite images for agricultural crop monitoring.

- Classify the image based on Soil type, moisture content, weather conditions, pH value, organic nitrogen etc.
- Perform satellite image processing with respect to tex tural and spatial features.
- Analyze crop patterns with the help of past records and map them with calculated data.
- Monitor crop yield and find ways for increasing it.
- Recommend profitable crops for each land type.

1.5 Scope

Agriculture is the backbone for a developing economy like India and there is an enormous need to maintain the agricultural sustainability. Hence it is a significant contribution towards the economic and agricultural welfare of the countries across the world.

1.6 Thesis overview

The rest of the thesis is organized as follows. Chapter 2 gives an overview of related work. Chapter 3 describes the proposed architecture and details of the machine learning & image processing models used for evaluation. The implementation and experimental results are given in Chapter 4. Chapter 5 summarizes the overall work and future work.

Chapter 2
Literature Survey

2.1 Image denoising

Xiang Xu et.al. (2016) present a new technique based on multiple morphological component analysis (MMCA) that exploits multiple textural features for decomposition of remote sensing images. The proposed MMCA framework separates a given image into multiple pairs of morphological components (MCs) based on different textural features, with the ultimate goal of improving the signal-to-noise level and the data separability. A distinguishing feature of our proposed approach is the possibility to retrieve detailed image texture information, rather than using a single spatial characteristic of the texture. In this paper, four textural features: content, coarseness, contrast, and directionality (including horizontal and vertical), are considered for generating the MCs.

Hemant Kumar et. al. (2016) state how Hyperspectral unmixing is the process of estimating constituent endmembers and their fractional abundances present at each pixel in a hyperspectral image. A hyperspectral image is often corrupted by several kinds of noise. This work addresses the hyperspectral unmixing problem in a general scenario that considers the presence of mixed noise. The unmixing model explicitly takes into account both Gaussian noise and sparse noise. The unmixing problem has been formulated to exploit joint-sparsity of abundance maps. A total-variation-based regularization has also been utilized for modeling smoothness of abundance maps. The split-Bregman technique has been utilized to derive an algorithm for solving resulting optimization problem. Detailed experimental results on both synthetic and real hyperspectral images demonstrate the advantages of proposed technique.

Gabriela Ghimpe teanu et.al. (2016) explain the image decomposition model that provides a novel framework for image denoising. The model computes the components of the image to be processed in a moving frame that encodes its local geometry (directions of gradients and level lines). Then, the strategy we develop is to denoise the components of the image in the moving frame in order to preserve its local geometry, which would have been more affected if processing the image directly. Experiments on a whole image database tested with several denoising methods show that this framework can provide better results than denoising the image directly, the index metrics is similar in terms of peak signal-to-noise ratio.

2.2 Image classification

Xia Zhang et.al.(2016) introduced the new crop classification method which involves the construction of the vegetation feature band set (FBS) and optimization object-oriented classification (OOC). The model also computes additional 20 spectral indices sensitive as parameters for feature band set to distinguish specific vegetation. For reducing the data redundancy Class pairs separability (CPS) to improve the separability between class pairs. The proposed algorithm shows that the crop classification accuracy will improve significantly, reduce edge effects and textural features combined with spectral indices that give sensitive to the chlorophyll, carotenoid and Anthocyanin indicators which also contribute significantly to improve the crop classification. Therefore, it is an effective approach for classifying crop species, monitoring invasive species, as well as precision agriculture related applications.

Xin Huang et. al. (2016) lists the differential morphological profiles (DMPs) that are widely used for the spatial/structural feature extraction and classification of remote sensing images. The DMPs shows the response of the image structures that are related to different scales and sizes of the structural elements (SEs).Traditional DMPs will ignore the discriminative information for features that are across the scales in the profiles. The proposed model will scale span with differential profiles to obtain the entire differential profiles. GDMPs used to obtain the complete shape spectrum and measure the difference between arbitrary scales. Since the random forest is used to interpret GDMPs for high dimensionality data and ability of evaluating the importance of variables. The random forest errors are used to quantify the importance of each channel of GDMPs and also discriminative information for the entire profiles.

Kunshan Huang et.al.(2016) proposed a novel spectral–spatial hyper spectral image classification method based on K nearest neighbor (KNN). The proposed method consists of support vector machine which is used to obtain the initial classification probability maps. The obtained pixel-wise probability maps are refined with the proposed KNN filtering algorithm that is based on matching and averaging non local neighborhoods. The model does not need any particular segmentation and optimization strategies which use the nonlocal principle of real images by using KNN and gives the competitive classification with fast computation. Several experiments performed on the hyper spectral data sets in which the result show that the classification results obtained by the proposed method are comparable to several recently proposed hyper spectral image classification methods.

2.3 Crop clustering

Wei Yao et.al. applies evaluates a modified Gaussian-test-based hierarchical clustering method for high resolution satellite images. The purpose of the model to obtain homogeneous clusters within each hierarchy level which later allow the classification and annotation of image data ranging from single scenes up to large satellite data archives. After cutting a given image into small patches and feature extraction from each patch, k-means are used to split sets of extracted image feature vectors to create a hierarchical structure. As image feature vectors usually fall into a high-dimensional feature space, we test different distance metrics, to tackle the "curse of dimensionality" problem. By using three different synthetic aperture radar (SAR) and optical image datasets, Gabor texture and Bag-of-Words features are extracted, and the clustering results are analyzed via visual and quantitative evaluations.

Michael D. Johnson et.al developed crop yield forecast models for barley, canola and spring wheat grown on the Canadian Prairies for vegetation indices which is derived from satellite data and machine learning methods. The model use hierarchical clustering to group the crop yield data from Census Agricultural Regions (CARs) into several larger regions for building the forecast models. The Moderate-resolution Imaging Spectro radiometer (MODIS) derived Enhanced Vegetation Index (EVI), Advanced Very High Resolution Radiometer (AVHRR) derived Normalized Difference Vegetation Index (NDVI) and these model are considered as predictors for crop yields. Multiple linear regression(MLR) and machine learning models – Bayesian neural networks (BNN) and model-based recursive partitioning (MOB) – were used to forecast crop yields, with various combinations of MODIS-NDVI, MODIS-EVI and NOAA-NDVI as predictors.

R.B. Arango et.al. (2016) focuses on a methodology for the automatic delimitation of cultivable land by means of machine learning algorithms and satellite data. The method uses a partition clustering algorithm called Partitioning Around Medoids and considers the quality of the clusters obtained for each satellite band in order to evaluate which one better identifies cultivable land. The proposed method was tested with vineyards using as input the spectral and thermal bands of the Landsat 8 satellite. The experimental results show the great potential of this method for cultivable land monitoring from remote-sensed multispectral imagery.

2.4 Wavelet transforms

M. Krishna Satya Varma et.al.(2016) state and create thematic maps of the land wrap present in an image, the classified data thus obtained may then be used. Classification includes influential an appropriate classification system, selecting, training sample data, image pre-processing, extracting features, selecting appropriate categorization techniques, progression after categorization and precision validation. Aim of this study is to assess Support Vector Machine for efficiency and prediction for pixel-based image categorization as a contemporary reckoning intellectual technique. Support Vector Machine is a classification procedure estimated on core approaches that was demonstrated on very effectual in solving intricate classification issues in lots of dissimilar appropriated fields. The latest generation of remote Sensing data analyzes by the Support Vector Machines exposed to efficient classifiers which are having amid the most ample patterns.

Olivier Regniers et.al.(2016) explore the potentialities of using wavelet-based multivariate models for the classification of very high resolution optical images. A strategy is proposed to apply these models in a supervised classification framework. This strategy includes a content-based image retrieval analysis applied on a texture database prior to the classification in order to identify which multivariate model performs the best in the context of application. Once identified, the best models are further applied in a supervised classification procedure by extracting texture features from a learning database and from regions obtained by a pre segmentation of the image to classify. The classification is then operated according to the decision rules of the chosen classifier. The use of the proposed strategy is illustrated in two real case applications using Pléiades panchromatic images: the detection of vineyards and the detection of cultivated oyster fields. In both cases, at least one of the tested multivariate models displays higher classification accuracies than gray-level co-occurrence matrix descriptors.

N. Prabhu et.al. (2016 have presented a series of experiments to investigate the effectiveness of some wavelet based feature extraction of hyper spectral data. The model introduces the three types of wavelets have been used which are Haar, Daubechies and Coiflets wavelets and the quality of reduced hyper spectral data has been assessed by determining the accuracy of classification of reduced data using Support Vector Machines classifier. The hyper spectral data has been reduced up to four decomposition levels. Among the wavelets used for feature extraction Daubechies wavelet gives consistently better accuracy than that produced from Coiflets wavelet. The two -level decomposition is capable of preserving more useful information from the hyper spectral data. Furthermore, two-level decomposition takes less time to extract features from the hyper spectral data than one-level decomposition.

2.5 Prediction algorithm

Monali Paul et.al.(2016) analysis the Soil Behaviour and Predicted the Crop Yield using Data Mining approach. To predict the proper selection of crops various yield prediction algorithm used to help farmers. Traditional yield prediction was performed by considering the farmer's experience on a particular field and crop. The proposed system uses data mining techniques in order to predict the category of the analyzed soil datasets. The category which is predicted will indicate the yielding of crops. The problem of predicting the crop yield is formalized as a classification rule, where Naive Bayes and K-Nearest Neighbor methods are used.

Yvette Everingham et.al.(2016) predict the accurate prediction of sugarcane yield using a random forest algorithm. A data mining method like random forests can cope with generating a prediction model when the search space of predictor variables is large. Researcher investigated that the accuracy of random forests to explain annual variation in sugarcane productivity and the suitability of predictor variables generated from crop models coupled with observed climate and seasonal climate prediction indices is limited. Simulated biomass from the APSIM (Agricultural Production Systems sIMulator) sugar-cane crop model, seasonal climate prediction indices and observed rainfall, maximum and minimum temperature, and radiation were supplied as inputs to a random forest classifier and a random forest regression model to explain annual variation in regional sugarcane yields.

Benjamin Dumont et.al.(2016) predict the Assessing the potential of an algorithm based on mean climatic data to predict wheat yield .This paper presents a methodology that addresses the problem of unknown future weather by using a daily mean climatic database, based exclusively on available past measurements. It involves building climate matrix ensembles, combining different time ranges of projected mean climate data and real measured weather data originating from the historical database or from real-time measurements performed in the field. Used as an input for the STICS crop model, the datasets thus computed were used to perform statistical within-season biomass and yield prediction. This work demonstrated that a reliable predictive delay of 3–4 weeks could be obtained. In combination with a local micrometeorological station that monitors climate data in real-time, the approach also enabled us to (i) predict potential yield at the local level, (ii) detect stress occurrence and (iii) quantify yield loss (or gain) drawing on real monitored climatic conditions of the previous few days.

2.6 Overview

Hence, wavelet algorithms can be leveraged for denoising the image by means of dimensionality reduction. Crop classification involves calculating the NDVI and LAI as a means for training the model for existing greenery percentage. K Nearest neighbour is also of prime importance for graph edge construction in case of fusion between locational and intensity components. Classification is improved by means of taking into account class pair separability for object oriented classification. The above mentioned aspects are kept in mind for building our proposed architecture of the project thereby eradicating the limitations of algorithms mentioned above.

CHAPTER 3
PROPOSED WORK

This chapter gives a detailed description about the architecture and algorithms used in the proposed work.

3.1 Proposed system architecture

The overall block diagram of the system is shown in the figure 3.1. The image pre-processor and the image segmentation does the same task of segmenting the image to get a vivid soil portion from the image, as it may contain unwanted portions which may make the system to work with decreased efficiency. The SSSE module takes the pre-processed image as input and finds the type of the soil and returns the soil class, which is then given to SVM along with latitude and longitude values.

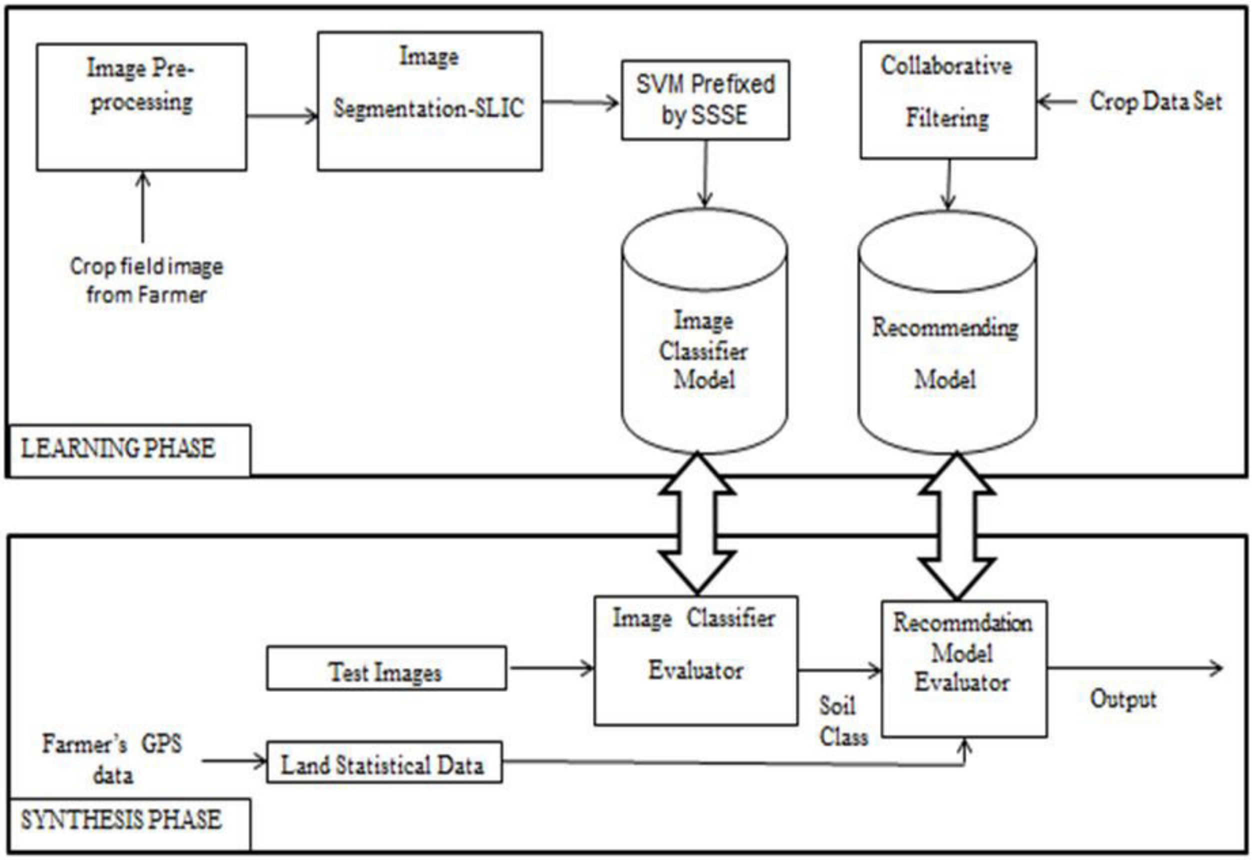


Figure 3.1 Architecture diagram

The system eventually aims at predicting the yield of crops based on the set of whether and yield data including geographical parameters.

3.2 Module design

The working of the recommendation system incorporates the following modules for orderly functioning.

3.2.1 Image preprocessing

The pre-processor uses multiple morphological component analysis (MMCA) for dimensionality reduction by means of content, coarseness, directional vectors and contrast features. This works by finding the possible contours with the specification to identify the soil areas.

3.2.2 Image Segmentation - SLIC

This module leverages the creation of superpixels using Simple linear iterative clustering (SLIC). In this method, the preprocessed images are taken as input and segmented crop field image is produced as the result.

3.2.3 SVM prefixed by SSSE

This module is the primary analysis module of the project. The SVM classifier is the backend classifier which takes the input of the Spectral spatial Schrodinger Eigen maps. This involves considering both spectral and spatial parameters in addition to textural features for partial knowledge and knowledge propagation.

3.2.4 Image classifier evaluator

This module takes in the pre-processed input image at the end and uses the pickled historical data sets of whether and crop yield parameters to predict the soil class and yield values.

3.2.5 Recommendation model evaluator

This module takes in the soil class and GPS parameters as input and also uses the Recommendation Model to recommend the crop suitable for the image and historical data set provided.

3.2.6 Calculating NDVI

The normalized difference vegetation index (NDVI) is a simple graphical indicator that can be used to analyze remote sensing measurements and assess whether the target being observed contains live green vegetation or not.

3.2.7 NDVI vs LAI relationship calculation

Accurate estimation of LAI is important for monitoring vegetation dynamics, and LAI information is essentially required for the prediction of microclimate and various biophysical processes within and below canopy.

$$\mathbf{(LAI = 0.128 * exp\ (NDVI/0.311))}$$

3.2.8 Yield calculation based on NDVI

The relationship between NDVI and yield of the data analyzed, indicates the possibility of considering agrometeorological conditions to obtain accuracy in yield estimation.

3.3 System development

The workflow of the system is shown in figure 3.2. The module descriptions, algorithms used with analysis methods and tools used to achieve the result are as follows

.

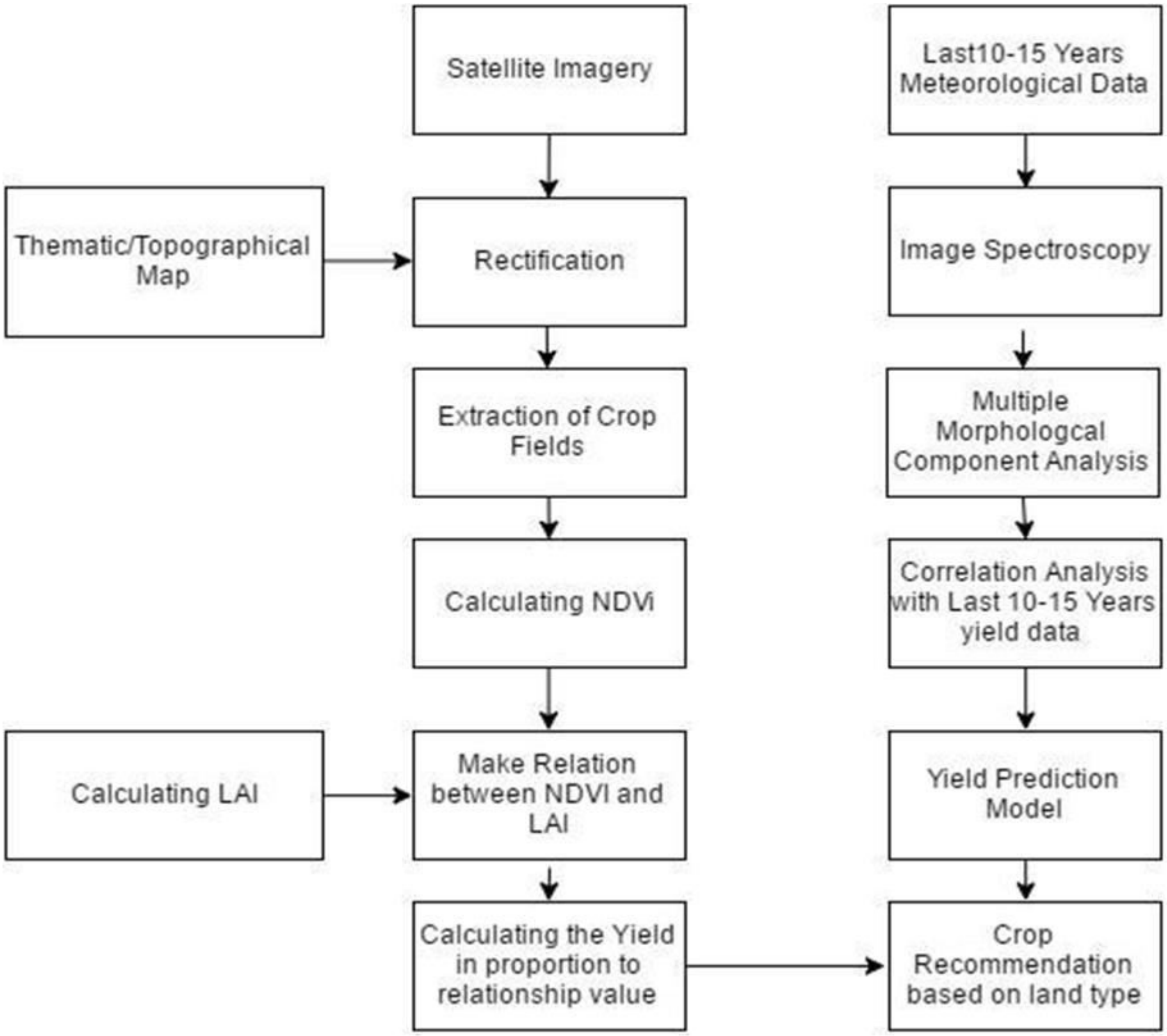

Satellite Imagery
Last10-15 Years Meteorological Data
Thematic/Topographical Map
Rectification
Image Spectroscopy
Extraction of Crop Fields
Multiple Morphologcal Component Analysis
Calculating NDVi
Correlation Analysis with Last 10-15 Years yield data
Calculating LAI
Make Relation between NDVI and LAI
Yield Prediction Model
Calculating the Yield in proportion to relationship value
Crop Recommendation based on land type

Figure 3.2 Flow diagram

This system has been developed using WinPython 2.7 which includes the packages such as: NumPy, SciPy, Pandas, Theano , Sklearn, Matplotlib and OpenCV.

The overview of the algorithm for the system is as given below.

Initially Img is assigned the input image, Lon is assigned with longitude and Lat is assigned with latitude. This is fed to SVM prefixed by SSSE classifier module which in turn invokes the AdaBoost and Collaborative filtering module modules to predict the crop yield and preferable crop respectively. Then the soil class together with Lat and Lon are used to predict the suitable crop. Algorithm 1 explains the functioning of the entire recommendation system.

```
Img ←Input Soil Image
Lon ←Longitude
Lat ←Latitude
PREDICT(Img, Lon, Lat):
1. soil class ←IMAGE CLASSIFIER EVALUATOR(Img)
2. suitable crop ←RECOMMENDATION MODEL EVALUATOR(soil class, Lon, Lat)
```

Algorithm 1 : Overall Working

3.4 Algorithms and working of each module

The algorithms used under each module are explained using pseudocode representations and performance analysis measures as follows.

3.4.1 Support vector machine

SVM finds its place in this work for training the Recommendation system with training set. It is additionally used after the classification using SSSE.

Input: Training set containing suitable crops for given soil class and Latitude and longitude parameter from SSSE module.

Output: Recommendation Model Evaluator.

The working algorithm of SVM is explained below as Algorithm - 2

```
DS ←dataset of soil images with soil classes as labels
SVM[i] ←one vs rest Support Vector Machine for class i
for row in D
SVM[row.class].train(row)
```

Algorithm 2 : SVM

3.4.2 Image pre – processer

Preprocessing plays a crucial role in denoising the crop field image by dimensionality reduction of components by means of wavelet algorithms. The proposed MMCA framework is elucidated in Algorithm 3.

Input: Crop field Image

Output: Processed soil image

```
I ←input image
Features[] ← Extract{content,coursness,direction,contrast}
Proc image ←MMCA (I, Fearures[])
Return Proc image
```

Algorithm 3: Preprocessing

3.4.2.1 Algorithm Used: Multiple morphological component analysis

Morphological component analysis allows to separate features contained in an image when those tuples represent different morphological aspects. Considering N as the number of pixels such that $y \in R^N$, the aim is to generate the sparsest solution for the noise removal problem as follows:

$$\mathbf{x = \arg\min \|x\|_i}$$

where $x \in R^K$ denotes the sparse coefficients of the MC. K represents the number of atoms in the dictionary of pixels (typically K>N). The resultant is the separation fo textural and content components as a result of decomposition by MCA.

Smoothness and texture component represent the original image under a linear combination as follows:

$$\mathbf{y= y_s + y_t + n}$$

The limitation of this representation is the ignorance in including the spatial components which are in abundance in the remote sensing images. MMCA aims at exploring the entire scope of spatial textural features to aid in noise removal ie denoising. Hence, as a result, 4 new textural features: content, coarseness, contrast and directionality are added in this linear combination. Algorithm 4 clearly expounds the proposed MMCA algorithm.

1. Randomly choose several portions from the image for initializing the smoothness and texture component dictionaries.
2. Build dictionaries for y_s and y_t based on some transitions on the image snippets chosen.
3. Perform sparse coding to learn the Morphological

Algorithm 4: MMCA

Issues in the existing MCA algorithm

The construction of dictionaries for the 2 components creates a large over heard in memory and time.

Enhancement made to accommodate the limitations without overhead

In addition to the existing algorithm components like smoothness and texture, the below mentioned 4 components make the entirety of the MMCA dimensionality reduction procedure.

Content tuple: Anisotropic structures, smooth curves and varied length edges are extracted using wavelet transforms and local ridgelet transforms. To build the dictionary from the randomly chosen mage snippets, cosine transform is made use of.

Coarseness tuple: It's a Gaussian distribution based dictionary construction mechanism wherein filtering replaces the intensity value at each pixel of an image with the weighted average values of intensity with respect to the neighbouring pixels. The radiometric differences between pixels are also taken into account which makes possible procedural looping and a s a result regulating the weights to the nearby pixels accordingly.

Contrast tuple: The variance of the grey scale distribution wherein high and low contrast values form the rapid and slow intensity anomalies form the crux f the contrast tuple. The dictionary construction mechanism involves achieving both high and low contrast dictionaries by means of anisotropic diffusion.

Directionality tuple: It involves the orientation of the textural features into horizontal and vertical vectors. Stationary wavelet transform is leverages to complete the task of decomposing the texture into different direction components.

Line likeliness, regularity, roughness and scattering ability are the other textural features which are in the future scope of the denoising algorithm.

Considering D_S and D_T to be the dictionaries for the smoothness and texture component and x_S and x_T be the sparse coefficients for the same. Therefore according to the linear combination of the dual components: in Equation 1

$$\mathbf{Y = y_S + y_T + n = D_S x_S + D_T x_T + n} \qquad \mathbf{(1)}$$

$\mathbf{y_S}$ and $\mathbf{y_T}$ are calculated as a result of the optimization problem as follows in Equation 2 :

$$< \boldsymbol{y_s}, \boldsymbol{y_t} > = \mathbf{argmin}_{\boldsymbol{y_s}, \boldsymbol{y_t},} \frac{\mathbf{1}}{\mathbf{2}} ||\boldsymbol{y} - \boldsymbol{y_s} - \boldsymbol{y_t}||_{\mathbf{2}} + \boldsymbol{\lambda_1} ||\boldsymbol{T_s y_s}||_{\mathbf{1}} + \boldsymbol{\lambda_2} ||\boldsymbol{T_T y_T}||_{\mathbf{1}} \qquad \mathbf{(2)}$$

where λ_1 and λ_2 are the regularization parameters. D_S and D_T are serially updated at each iteration using total variation and regularization threshold.

Issues in the proposed work

To fully retrieve the image texture information, the inability to incorporate varied textural features is a marked overhead since a single spatial feature is not always enough in case of different kinds of remote sensed imagery. Selection of an optimal feature among the multiple tuples for a particular image is a possibility to be analyzed in future.

3.4.3 Image classifier evaluator

This module evaluates the processed soil image to identify the crop class for the particular field.

Input: Processed Soil Image

Output: Soil class of the image

```
proc image ←preprocessed image
SSSE ←Spectral + Spatial barrier parameters
N ←number of potentials in the column of barriers
YiSSSE ←Classification of SSSE for the class i
Yi ← Prediction SSSE for the class i
Yi ←1/N (ΣSSSE YiLE)
return argmaxi(Yi)
```

Algorithm 5: SSSE

Existing algorithm: Laplacian Eignenmaps

It's a non-linear dimensionality reduction algorithm which preserves computational efficiency and locality preservation properties. The affinity to clustering algorithms is evident in the flow of the functioning of the algorithm.

Steps involved

1. Build an undirected graph G = (X, E) where the vertices are the set of piints in X and edges E defined based on the spatial proximity of the vertices. For this proximity calculation, ∈ - neighborhoods and K nearest neighbor search is made use of.

2. The weights for each edge are defined in E. The heat kernel based on the Euclidean distance between the pixels is the common method for defining weights as explained in Equation 3.

$$\boldsymbol{W_{i,j} = \exp\left(-\frac{||x_i - x_j||^2}{\sigma}\right)} \qquad \textbf{(3)}$$

if the edge exists or $W_{i,j} = 0$ otherwise.

3. The smallest m+1 eigenvalues an eigenvectors are calculated using the conventional eigenvector problem Lf = **λDf** where D is the diagonal weighted matrix by $\mathbf{D_{i,j}} = \Sigma\ _j\mathbf{W_{i,j}}$ and **L= D – W** is the laplacian matrix. The eigenvectors found $f_0,\ldots..,f_m$ are serially arranged in order so that

$$\mathbf{0 = \boldsymbol{\lambda}_0 \leq \boldsymbol{\lambda}_1 \leq . \ldots \ldots \leq \boldsymbol{\lambda}_m}$$

Then the points y_1^T , y_2^T , ..., y_3^T are defined to be the rows of

$$\boldsymbol{F} = [\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots \ldots, \boldsymbol{f}_m]$$

Schroedinger Eigenmaps

It can be elucidated as a variance of the LE algorithm in incorporate a potential matrix V. The eigenvector problem is modified as in Equation 4:

$$\mathbf{(L + \boldsymbol{\alpha} V)f = \boldsymbol{\lambda} Df} \tag{4}$$

Where **α** is the weight parameter to prioritize the contributions of the laplacian and potential matrices. Barrier and cluster potentials are both analyzed in this mechanism. A non-negative diagonal matrix is used as barrier potentials wherein the +ve values are forcing the points in y towards the origin. Cluster potentials are defined as a entirety of non-diagonal matrices using equation 5.

$$\boldsymbol{V^{(i,j)}} = \begin{cases} \mathbf{1} & , (\boldsymbol{k}, \boldsymbol{l}) \in \{(\boldsymbol{i}, \boldsymbol{i}), (\boldsymbol{i}, \boldsymbol{i})\} \\ \mathbf{-1} & , (\boldsymbol{k}, \boldsymbol{l}) \in \{(\boldsymbol{i}, \boldsymbol{j}), (\boldsymbol{j}, \boldsymbol{i})\} \\ \mathbf{0} & , \quad \boldsymbol{otherwise} \end{cases} \tag{5}$$

The crucial advantages of SE over LE is that semi supervised clustering is made possible by the potential matrix V. K means algorithm is invoked as the ending classifier model.

Spectral Spatial fusion in Schrodinger algorithm

A combination of the spatial (location) and spectral (intensity) is incorporated in this dimensionality reduction method. This happens by concatenating both the spatial location x_i^p and spectral x_i^f.

$$\mathbf{x_i^T = [\, x_i^f, x_i^p \,]}$$

Analysis of multiple fusion methods

The analysis differs in the nature of how edge weights are defined in each method.

Shi - Malik

1. The graph is constructed based on the spatial neighborhood

   ie €- neighborhood between edges in such a way that an edge is defined between x_i and x_j if $\|x_i^p - x_j^p\|2 < €$.

2. Define edge weights as calculated in Equation 6.

$$W_{i,j} = \begin{cases} \left\{ \left(\exp\left(-\frac{\left\| x_i^f - x_j^f \right\|^2}{\sigma_f^2} \right) - \frac{\left\| x_i^p - x_j^p \right\|^2}{\sigma_p^2} \right) \right. & ,(x_i, x_j) \in \varepsilon \\ \mathbf{0} & ,otherwise \end{cases} \quad \textbf{(6)}$$

3. Proceed with the final step in the LE algorithm.

Gilles – Bowles

This method extended the Shi Malik approach towards penalty on the differences in the directionality of the spectral information. The edgr weight is based on Equation 7.

$$\boldsymbol{W_{i,j}} = \begin{cases} \boldsymbol{exp}\left(-\boldsymbol{cos^{-1}}\left(\frac{<x_i^f,x_j^f>}{\left\|x_i^f\right\|.\left\|x_j^f\right\|}\right) - \frac{\left|\left|x_i^p - x_j^p\right|\right|^2}{\sigma_p^2}\right), (\boldsymbol{x_i}, \boldsymbol{x_j}) \,\boldsymbol{\epsilon}\, \boldsymbol{\varepsilon} \\ \boldsymbol{0} \qquad\qquad , \boldsymbol{otherwise} \end{cases} \quad \textbf{(7)}$$

Hou – Zhang –Ye - Zheng

This model proposes a slightly different approach to spectral spatial fusion. In the above mentioned methods, graph edges were based solely on

spatial information and weights were based on spectral –spatial fused information. On the other hand, this method uses the fused information in the step of defining the graph edges in addition to using binary weights**.**

1. Construct G so that edges E are defined based on the k nearest neighbor algorithm ie define an edge if it's in the k nearest neighbors of each other according to the measure mentioned in Equation 8.

$$\boldsymbol{d(x_{i,}x_j)} = \left(\boldsymbol{1 -} \ \textbf{exp}\left(-\frac{\left\|x_i^f - x_j^f\right\|^2}{2\sigma_f^2}\right)\right).\left(\boldsymbol{1 - exp}\left(-\frac{\left\|x_i^p - x_j^p\right\|^2}{2\sigma_p^2}\right)\right) \quad \textbf{(8)}$$

2. Define binary edge weights by calculating based on Equation 9.

$$\boldsymbol{W_{i,j}} = \begin{cases} \boldsymbol{1}, \ (\boldsymbol{x_{i\,,}x_j}) \in \boldsymbol{\varepsilon} \\ \boldsymbol{0}, \ \boldsymbol{otherwise} \end{cases} \quad \textbf{(9)}$$

Benedetto et al

This method proposes a variety of ways to fuse spectral and spatial information into the LE based algorithm that is used in integration with linear discriminant analysis (LDA) for classification. The metric used in Equation 10 is used for the same.

$$\boldsymbol{d_\beta(x_{i,}x_j) = \sqrt{\beta\left(\frac{||x_i^f - x_j^f||_2^2}{\sigma_f^2}\right) + (1-\beta)\left(\frac{||x_i^p - x_j^p||_2^2}{\sigma_p^2}\right)}} \qquad \textbf{(10)}$$

Where $0 \leq \beta \leq 1$. d_0 measures the scaled eucliedan distance based purely on the spatial components and d_1 measures the scaled euclidean distance vased purely on spectral components.

Further G_β is the graph constructed so that the edges $\varepsilon_{\ \beta}$ are defined based no the k – nearest neighbor algorithm. The weight matrix W_β is defined using equation 11.

$$\boldsymbol{W_{i,j}^{(\beta)} = \begin{cases} exp(-d_\beta(x_i\,, x_j)^2\,)\,, & (x_i\,, x_j)\ \ \epsilon\ \varepsilon_\beta \\ \qquad 0 \qquad\quad , & otherwise \end{cases}} \qquad \textbf{(11)}$$

Laplacian matrix $L_\beta = D_\beta - W_\beta$

Modified Spectral Spatial Schrodinger Eigenmaps

This involves spatial spectral fusion based on Schrodinger Eigenmaps using cluster potentials and spatial proximity.

Steps involved

1. Construct an undirected graph G= (X,E) with vertices in X and edges E defined based in spatial proximity.
2. Define weight for the edges between points in X as in equation 12.

$$\boldsymbol{exp}\left(\frac{-|\left|\boldsymbol{x_i^f}-\boldsymbol{x_j^f}\right||^2}{\boldsymbol{\sigma_f^2}}\right) \qquad \textbf{(12)}$$

3. Define the cluster potential matrix V representing the spatial proximity between the vertices as in Equation 3.

$$V = \sum_{i=1}^{k} \sum\nolimits_{x_j \epsilon\, N_\epsilon^p(x_i)} V^{(i,j)}.\,\gamma_{i,j}\,.\,exp\left(-\frac{||\boldsymbol{x_i^p}-\boldsymbol{x_j^p}}{\boldsymbol{\sigma_f^2}}\right) \qquad \textbf{(13)}$$

Where N_ϵ^p is the set of points in X whose spatial components are in the ϵ neighborhood of the spatial components of x_i

4. The smallest m+1 eigenvalues an eigenvectors are calculated using the conventional eigenvector problem Lf = **λDf** where D is the diagonal weighted matrix by $\mathbf{D_{i,j}}$ = **Σ** $_{\mathbf{j}}\mathbf{W_{i,j}}$ and **L= D – W** is the laplacian matrix. The eigenvectors found f_0,…..,f_m are serially arranged in order so that

$$\boldsymbol{0 = \lambda_0 \leq \lambda_1 \leq . \dots \dots \leq \lambda_m}$$

Then the points y_1^T , y_2^T , ..., y_3^T are defined to be the rows of

$$\boldsymbol{F = [f_1, f_2, \dots \dots, f_m]}$$

Knowledge propogation:

Situation arise when cluster potentials which are extracted from small set of manually provided labels may have a crucial impact on the dimensionality reduction method. This can be achieved by replacing the cluster potential matrix M as follows in Equation 14.

$$M = \sum_{\left(x_{i_k}, x_{j_k}\right) \epsilon\, M}^{n} \eta_{i_{k,j_k}} .\ WD^{-1}V^{(i_k,\, j_k)} \qquad \textbf{(14)}$$

3.4.4 Recommendation model evaluator

Evaluate the soil class obtained from the image classifier and data and use the historical data to recommend suitable crops for each crop field.

Input: Feature vector containing features like soil class, terrain,
slope, elevation, etc

Output: a list of recommended crops

```
1. X ←input feature vector
2. AdaBoost ←one vs rest Booster for class i
3. return $_{\text{argmaxi}}$(AdaBoost .predict(X))
```

Algorithm: 6

Technical description

Given the training data $(x_1, y_1), \ldots \ldots \ldots \ldots \quad (x_m, y_m)$

$y_i \ \in \{-1, +1\}, x_i \ \in \ X$ is the object or instance, y$_i$ is the classification.

For t=1,……,T

Form a distribution D_T on {1,…..,m}

Select weaker classifier with smallest error $\in_{\boldsymbol{t}}$ on D_t

$$\in_{\boldsymbol{t}} = PrD_t[h_t(x_i) \neq y_i]$$

$$h_t : X \rightarrow \{-1, +1\}$$

Output single classifier $\mathbf{H_{final}(x)}$

CHAPTER 4

IMPLEMENTATION AND RESULTS

This chapter gives a detailed description about the experimental setup, tools used for crop prediction, the implementation of proposed work, its analysis and output results.

4.1 Tools

This part gives a detailed introduction and feature analysis of th tools and libraries used to achieve the results.

4.1.1 MATLAB

MATLAB (matrix laboratory) is multi-paradigm mathematical computing background and fourth-generation programming language. A patented programming language established by MathWorks, MATLAB performs matrix operations, plotting of functions and data, execution of algorithms, formation of user interfaces, and interfacing with programs written in other languages, including C, C++, C#, Java, Fortran and Python. Although MATLAB is envisioned chiefly for numerical computing, an noncompulsory toolbox uses the MuPAD symbolic engine, allowing admission to symbolic computing capabilities. An added package, Simulink, adds graphical multi-domain simulation and model-based design for dynamic and embedded systems

4.1.1.1 Features

Matlab provides the following important characteristic features of usage.

- It offers functions for assimilating MATLAB based algorithms with external applications and languages such as Java, .NET, C and MS Excel.

- It provides implements for constructing applications with custom graphical interfaces.

- It provides built-in graphics for envisioning data and tools for creating custom plots.

- It is a high-level language for numerical computation, visualization and application development.

- MATLAB's programming interface gives development tools for refining code quality maintainability and maximizing performance.

- It also provides an interactive environment for iterative exploration, design and problem solving.

- It provides huge library of mathematical functions for linear algebra, statistics, Fourier analysis, solving ordinary differential equations, numerical integration, filtering and optimization.

4.1.2 PyCharm

PyCharm is an Integrated Development Environment (IDE) used for programming in Python. It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web development with Django.

4.1.2.1 Features

PyCharm provides the following important characteristic features of usage

- Coding Assistance and Analysis, with code completion, syntax and error highlighting, linter integration, and quick fixes

- Project and Code Navigation: specialized project views, file structure views and quick jumping between files, classes, methods and usages
- Python Refactoring: including rename, extract method, introduce variable, introduce constant, pull up, push down and others
- Support for web frameworks: Django, web2py and Flask
- Integrated Python Debugger
- Integrated Unit Testing, with line-by-line coverage
- Google App Engine Python Development
- Version Control Integration: unified user interface for Mercurial, Git, Subversion, Perforce and CVS with changelists and merge

4.1.3 Matplotlib

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. Matplotlib is a library for making 2D plots of arrays in Python. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like wxPython, Qt, or GTK+. The Matplotlib code is conceptually divided into three parts: the pylab interface is the set of functions provided by Matplotlib.

The Matplotlib frontend or Matplotlib API is the set of classes that do the heavy lifting, creating and managing figures, text, lines, plots and so on. This is an abstract interface that knows nothing about output. The backends are device-dependent drawing devices, aka renderers, that transform the frontend representation to hardcopy or a display device

4.2 Deployment details

The deployment of the system requires Windows 10 (or) 8.1 operating system. The system must also be installed with Python 2.7 (or) 3.4. Any IDE like Pycharm can be used to deploy the system successfully.

4.3 Dataset

The following are the datasets used for training and testing the prediction model built.

The weather data is extracted using Darksky API to get instantaneous data regarding weather , humidity details etc. Hyperspectral images are obtained from Hyperspectral Remote sensing scenes.

URL:

www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes

County and crop yield data sets are obtained from fao.org

URL:

www.fao.org/soils-portal/soil-survey/soil-maps-and-databases/harmonized-world-soil-database-v12/en/

Weather	
weather_all.csv	Instantaneous weather data
Yield	
yield_proc_data.csv	Yield forecast text data

Pickle Feed	
PickleFeed.csv	County wise crop data
Location	
county_lat_lng.csv	Locational data for each county
Images	
PaviaU.mat	Hyperspectral images
KSC.mat	Hyperspectral images
Salinas.mat	Hyperspectral images
Salinas_corrected.mat	Hyperspectral images
Botswana.mat	Hyperspectral images
Indian_pines_g.mat	Hyperspectral images

Table 4.1 Dataset

For Training the model, years 2010 – 2014 have been considered and the subsequent 2 years till 2016 are taken into account for testing. Training images are taken as Pavia, Salinas, Botswana region and testing: Indian Pines. The hyperspectral images extracted from the above mentioned URL are for the United States of America. The crop yield and crop recommendation modules are implemented for 4 main states in the USA with multiple counties within the state. Hence our area under consideration is Barley and it's variances of crop growth.

4.4 Preprocessor

The preprocessor module performs variety fo classifiers and segmentation mehods on the training data. Figure 4.1 shows the user interface for the preprocessor module.

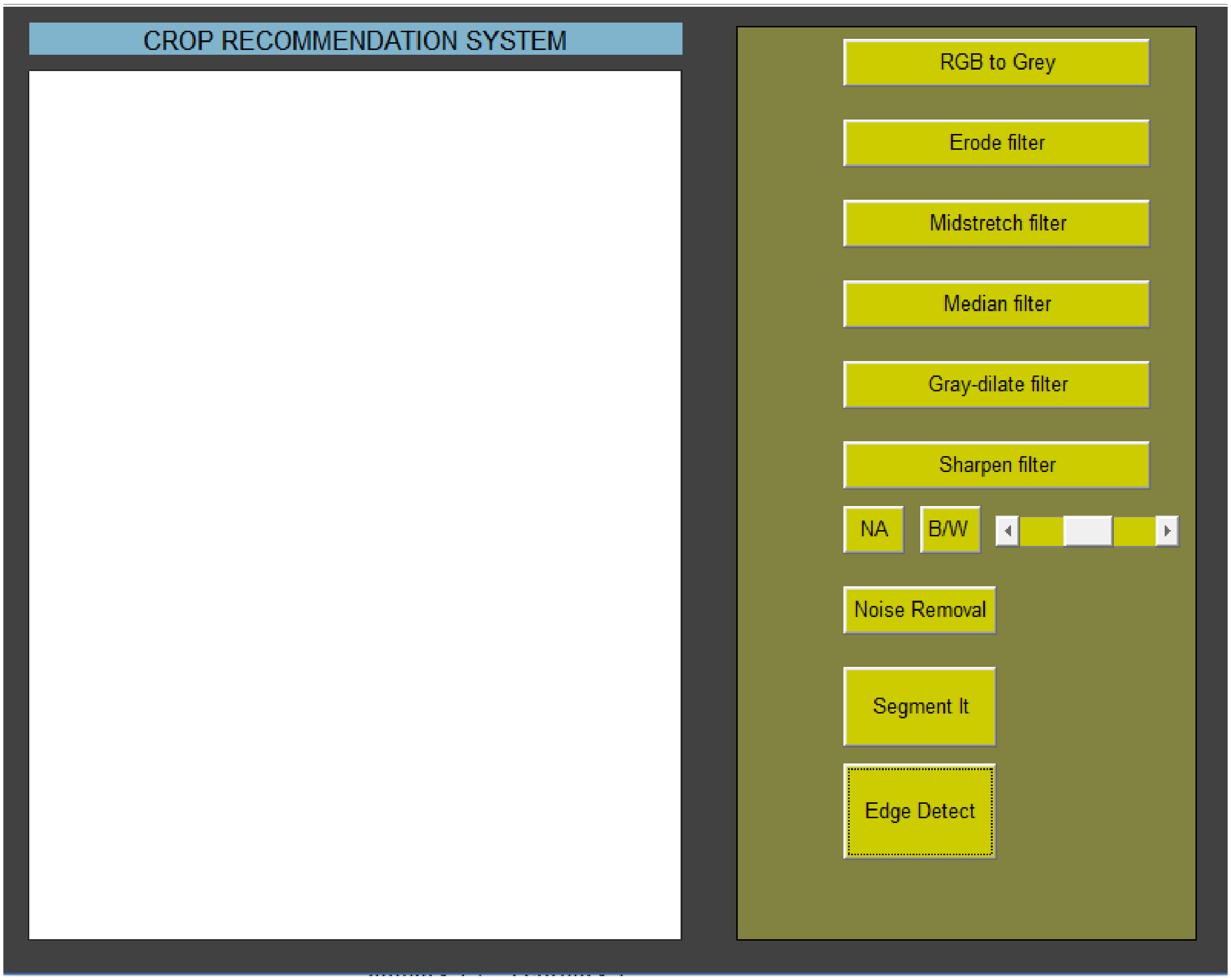
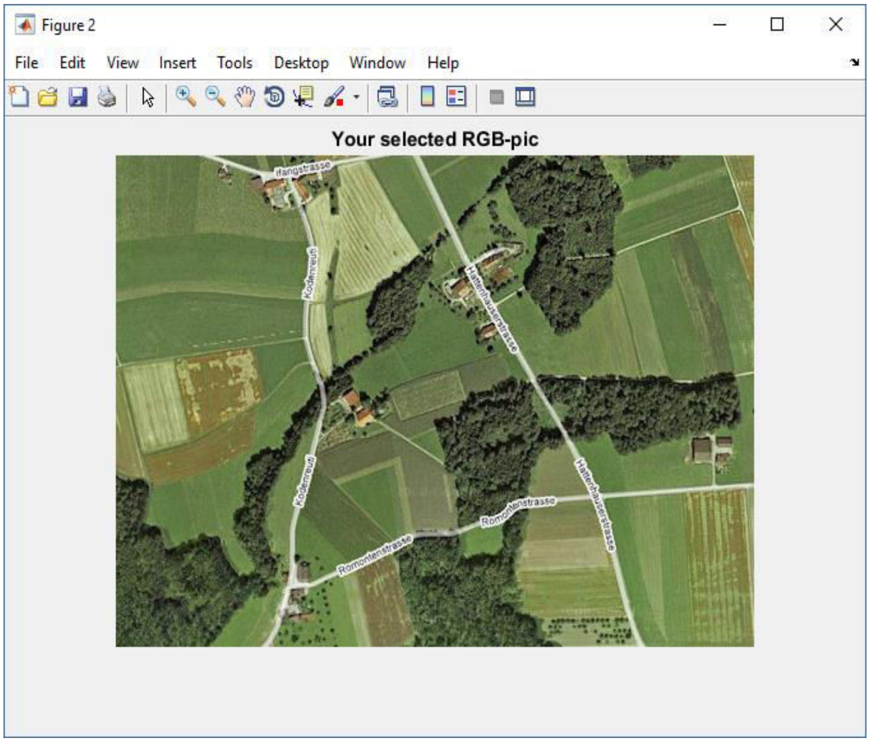


Figure 4.1 Preprocessor module



‘ ’

Figure 4.2 Original Image

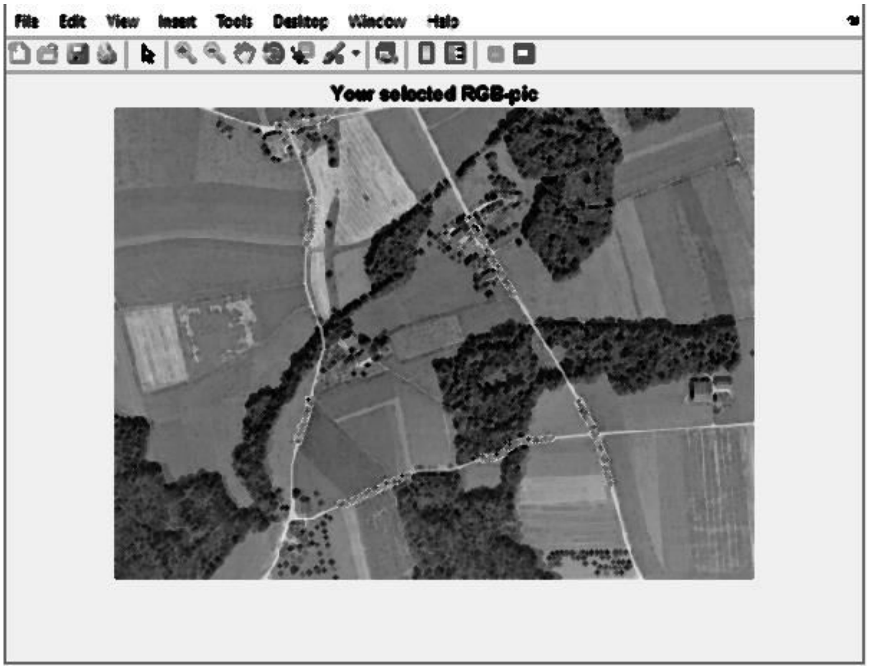


Figure 4.3 Mid-Stretch filter

As shown above, multiple filters like erode, midstretch, median, grey dialate and sharpen filter are applied to the training image for noise removal using Multiple morphological component analysis in figures 4.2 ad 4.3.

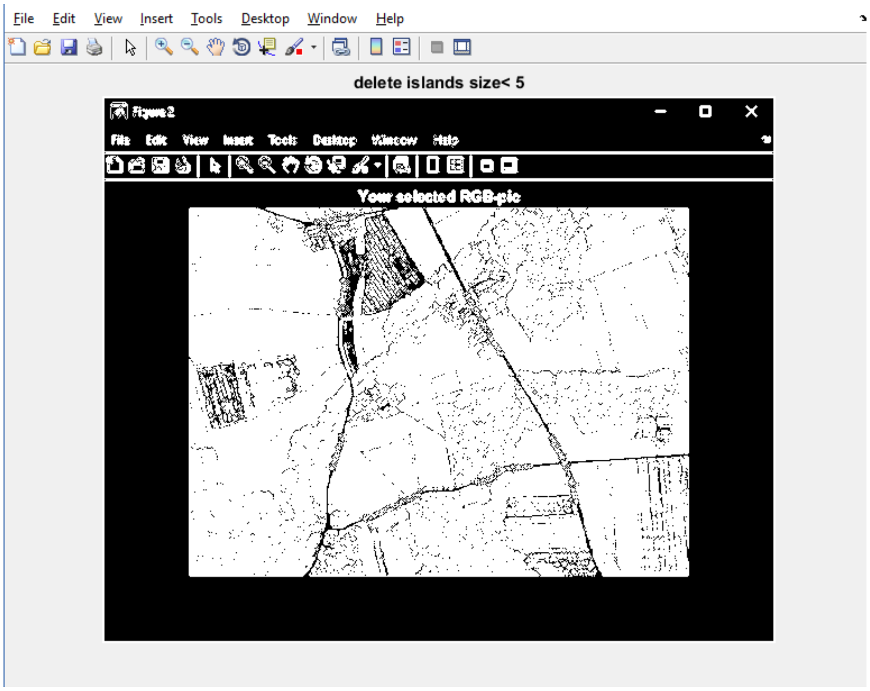

File Edit View Insert Tools Desktop Window Help
delete islands size< 5
Figure2
File Edit View Insert Tools Desktop Window Help
Your selected RGB-pic

Figure 4.4 Noise removal using grey- dialate filter

4.5 Image Classifier

For the image classifier, the test data consists of images belonging to different soil class collected manually from the internet. The input to the image classifier is a soil image. It predicts the soil class for a given image using a spatial spectral fusion methodology which is fed into the SVM along with latitude and longitude values The input to the SVM is a tuple containing soil class, latitude and longitude values using which it recommends a crop

The following figures represent the output of the classification method for Spatial Spectral Schrodinger Eigenmaps. The performance measures are simultaneously calculates using the standards like overall accuracy, average accuracy, average precision, average sensitivity and average specificity.

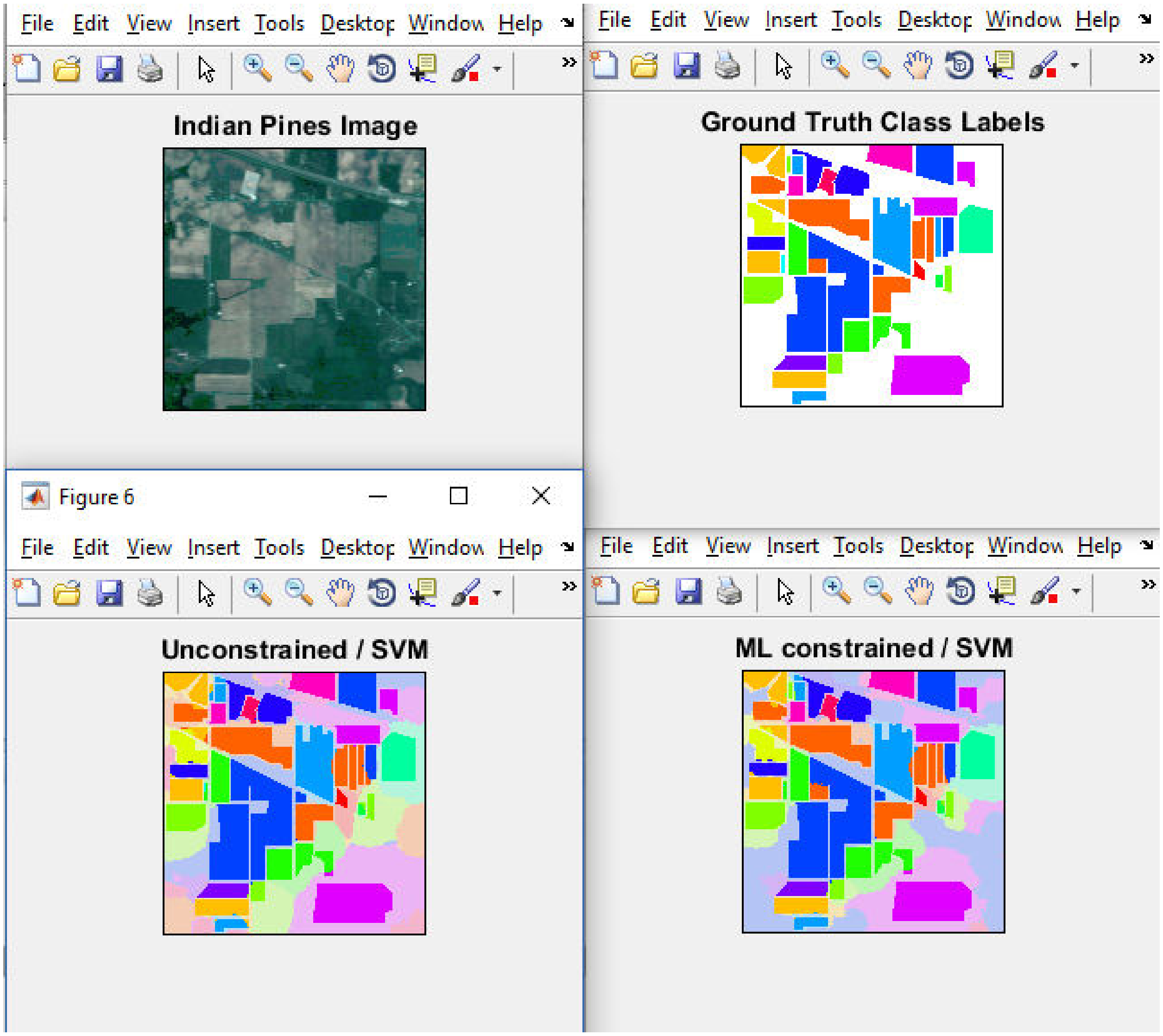
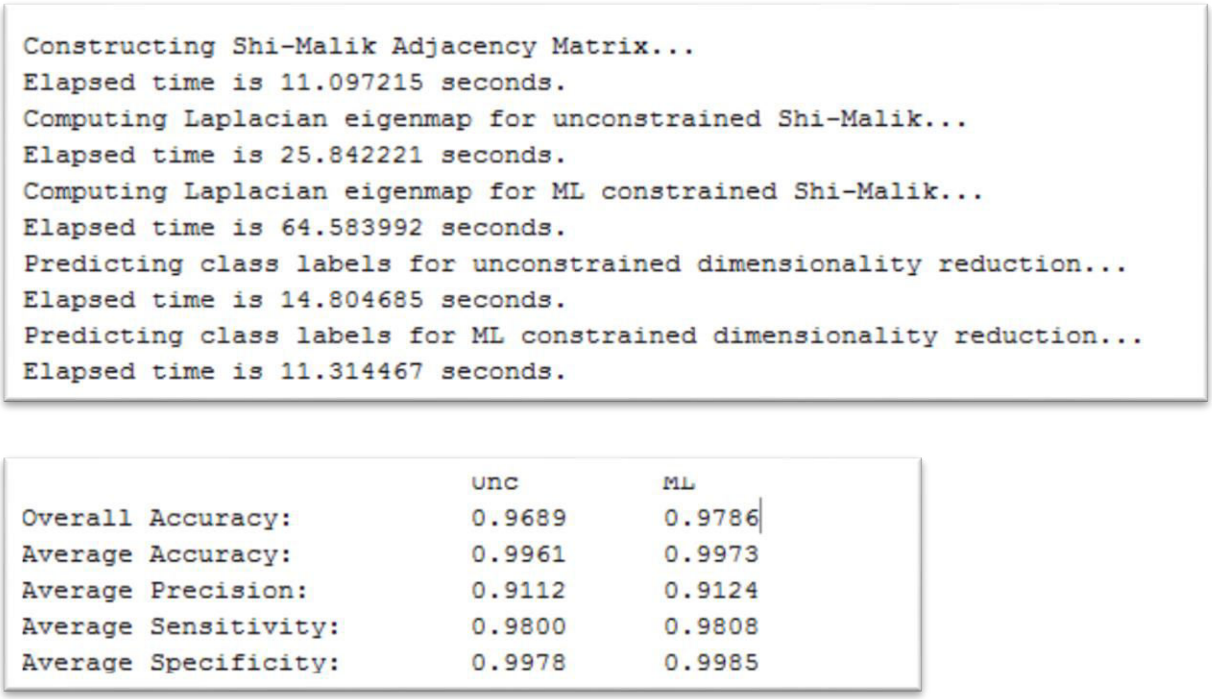

File Edit View Insert Tools Desktop Window Help
Indian Pines Image
File Edit View Insert Tools Desktop Window Help
Ground Truth Class Labels
Figure 6
File Edit View Insert Tools Desktop Window Help
Unconstrained / SVM
File Edit View Insert Tools Desktop Window Help
ML constrained / SVM

Figure 4.5 Classification image results

Figure 4.5 displays the image classification results and 4.6 shows the accuracy and time elapsed for the same process. This involves time to setup the unconstrained and ML constrained method of classification.



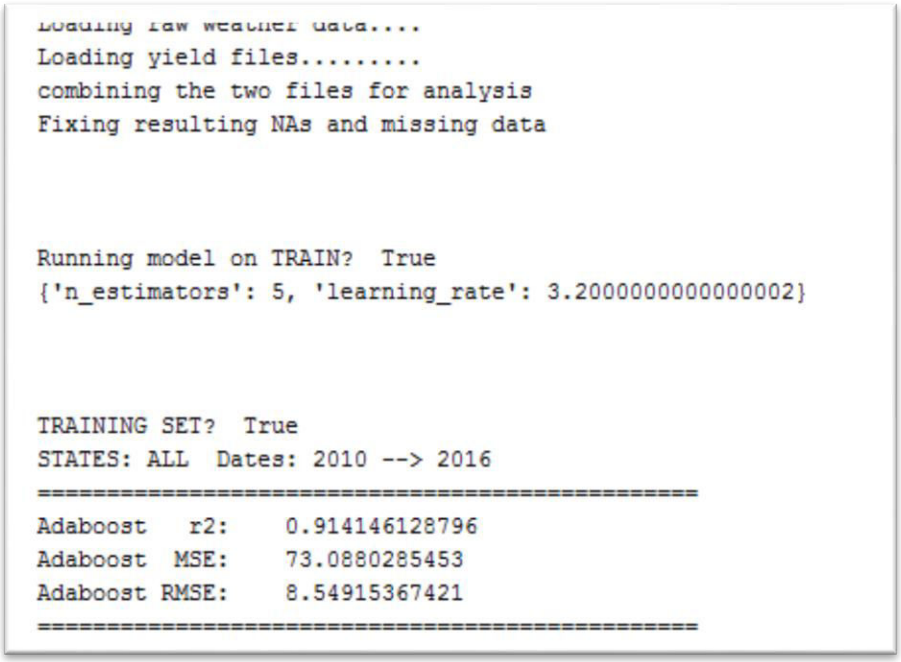
```
Constructing Shi-Malik Adjacency Matrix...
Elapsed time is 11.097215 seconds.
Computing Laplacian eigenmap for unconstrained Shi-Malik...
Elapsed time is 25.842221 seconds.
Computing Laplacian eigenmap for ML constrained Shi-Malik...
Elapsed time is 64.583992 seconds.
Predicting class labels for unconstrained dimensionality reduction...
Elapsed time is 14.804685 seconds.
Predicting class labels for ML constrained dimensionality reduction...
Elapsed time is 11.314467 seconds.
```

```
                             unc         ML
Overall Accuracy:            0.9689      0.9786
Average Accuracy:            0.9961      0.9973
Average Precision:           0.9112      0.9124
Average Sensitivity:         0.9800      0.9808
Average Specificity:         0.9978      0.9985
```

Figure 4.6 Classification Accuracy measures

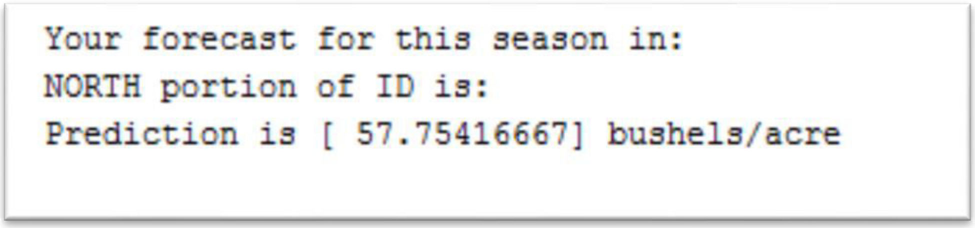
4.6 Crop yield prediction

The adaBoost classifier model is used to predict the crop yield per state or country for whichever weather condition that is available at the darksky server.



```
Loading raw weather data....
Loading yield files.........
combining the two files for analysis
Fixing resulting NAs and missing data



Running model on TRAIN?  True
{'n_estimators': 5, 'learning_rate': 3.2000000000000002}



TRAINING SET?  True
STATES: ALL  Dates: 2010 --> 2016
================================================
Adaboost   r2:    0.914146128796
Adaboost  MSE:    73.0880285453
Adaboost RMSE:    8.54915367421
================================================
```

Figure 4.7 Training data execution results



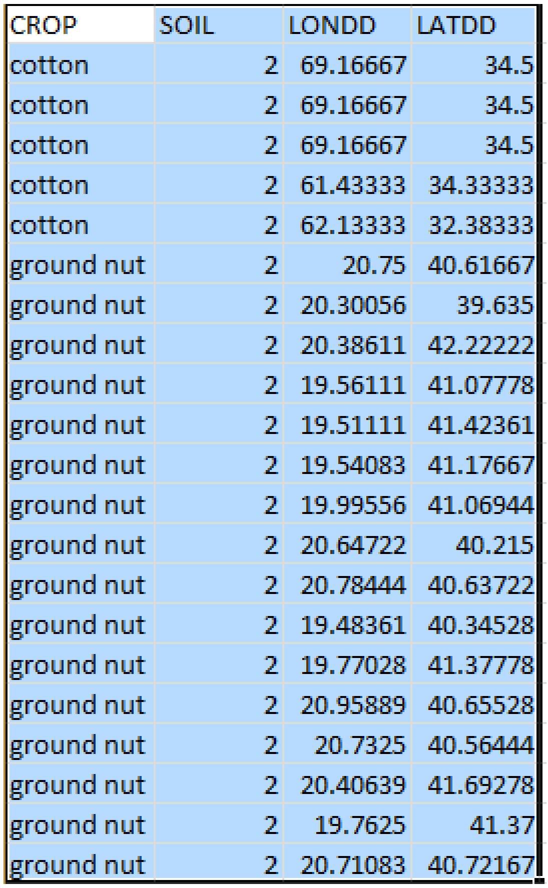
```
Your forecast for this season in:
NORTH portion of ID is:
Prediction is [ 57.75416667] bushels/acre
```

Figure 4.8 Crop yield result

The above figures 4.7 and 4.8 show the results of the crop yield prediction along with the training module root mean square error values.

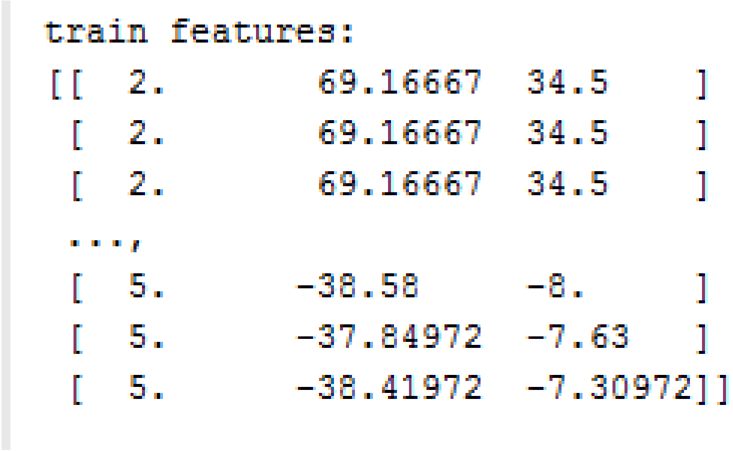
4.7 Relevance prediction of crops

This model involve the comparison between truth and predicted crop for that particular latitude and longitude as evident in Figure 4.10



CROP	SOIL	LONDD	LATDD
cotton	2	69.16667	34.5
cotton	2	69.16667	34.5
cotton	2	69.16667	34.5
cotton	2	61.43333	34.33333
cotton	2	62.13333	32.38333
ground nut	2	20.75	40.61667
ground nut	2	20.30056	39.635
ground nut	2	20.38611	42.22222
ground nut	2	19.56111	41.07778
ground nut	2	19.51111	41.42361
ground nut	2	19.54083	41.17667
ground nut	2	19.99556	41.06944
ground nut	2	20.64722	40.215
ground nut	2	20.78444	40.63722
ground nut	2	19.48361	40.34528
ground nut	2	19.77028	41.37778
ground nut	2	20.95889	40.65528
ground nut	2	20.7325	40.56444
ground nut	2	20.40639	41.69278
ground nut	2	19.7625	41.37
ground nut	2	20.71083	40.72167

Figure 4.9 Crop soil data set



```
train features:
[[  2.        69.16667  34.5     ]
 [  2.        69.16667  34.5     ]
 [  2.        69.16667  34.5     ]
 ...,
 [  5.       -38.58     -8.      ]
 [  5.       -37.84972  -7.63    ]
 [  5.       -38.41972  -7.30972]]
```

Figure 4.10 Relevance prediction for red soil

4.8 Performance measures

A binary classifier foresees all data instances of a testing information as +ve or -ve. This gives four main results – true positive, true negative, false positive and false negative.

True Positive (TP): correct positive prediction

False Positive (FP): incorrect positive prediction

True Negative (TN): correct negative prediction

False Negative (FN): incorrect negative prediction

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} = \frac{TP + TN}{P + \ N}$$

$$Sensitivity = \frac{TP}{TP + FN} = \frac{TP}{P}$$

$$Specificity = \frac{TN}{TN + FP} = \frac{TN}{N}$$

$$Precision = \frac{TP}{TP + FP}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - y_i^{\wedge}\right)^2}$$

Where n is the total number of tuples and y_i are the data points.

4.9 Yield Prediction

Figure 4.9 deals with training model analysis of regression, The geaph is set upon the relation between each data point and the yield in bushels/acre.

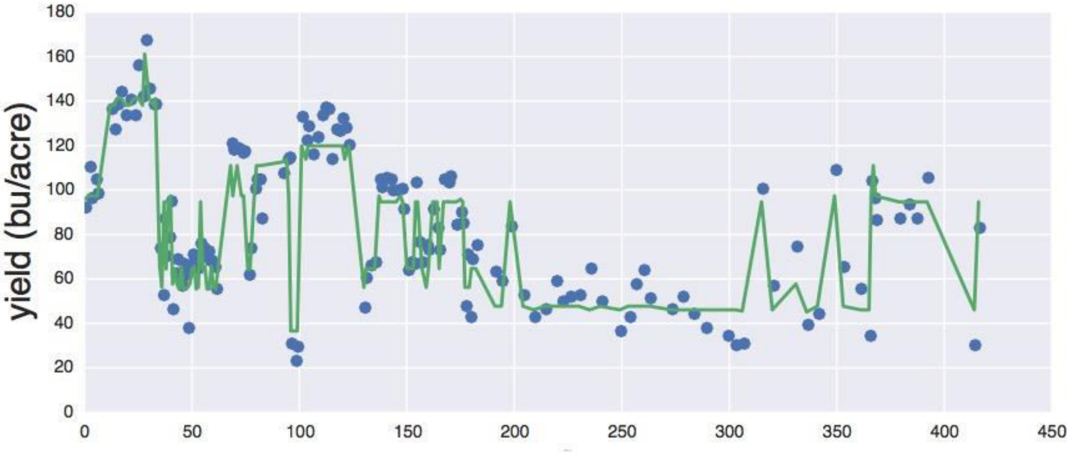


Figure 4.11 Training set regression analysis

After the training part of the model is completed for the years 2010 to 2014, the testing phase starts for the subsequent years till 2016.Regression analysis of the testing mechanism results in the following figure 4.10.

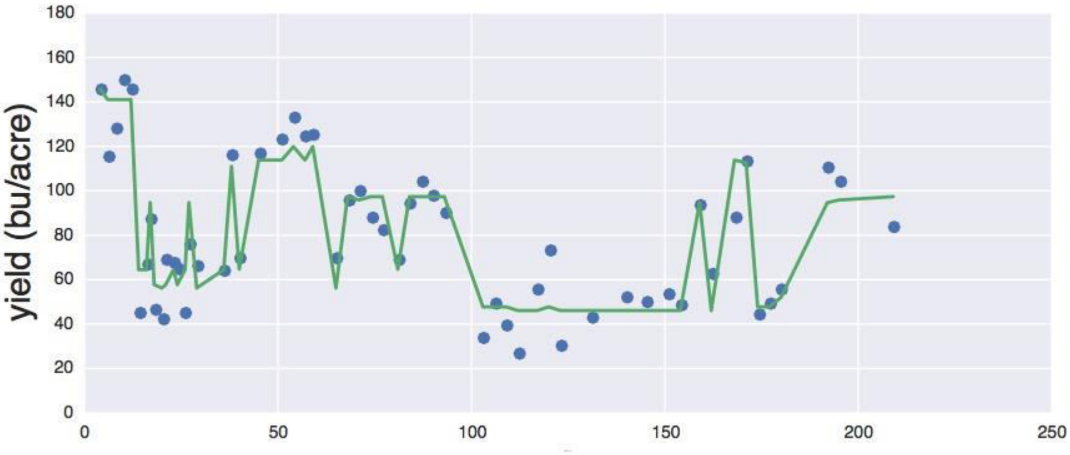


Figure 4.12 Testing regression analysis

Figure 4.11 shows the handoff between the existing yield values and the predicted one and how far they are correlated in context with the data sets..

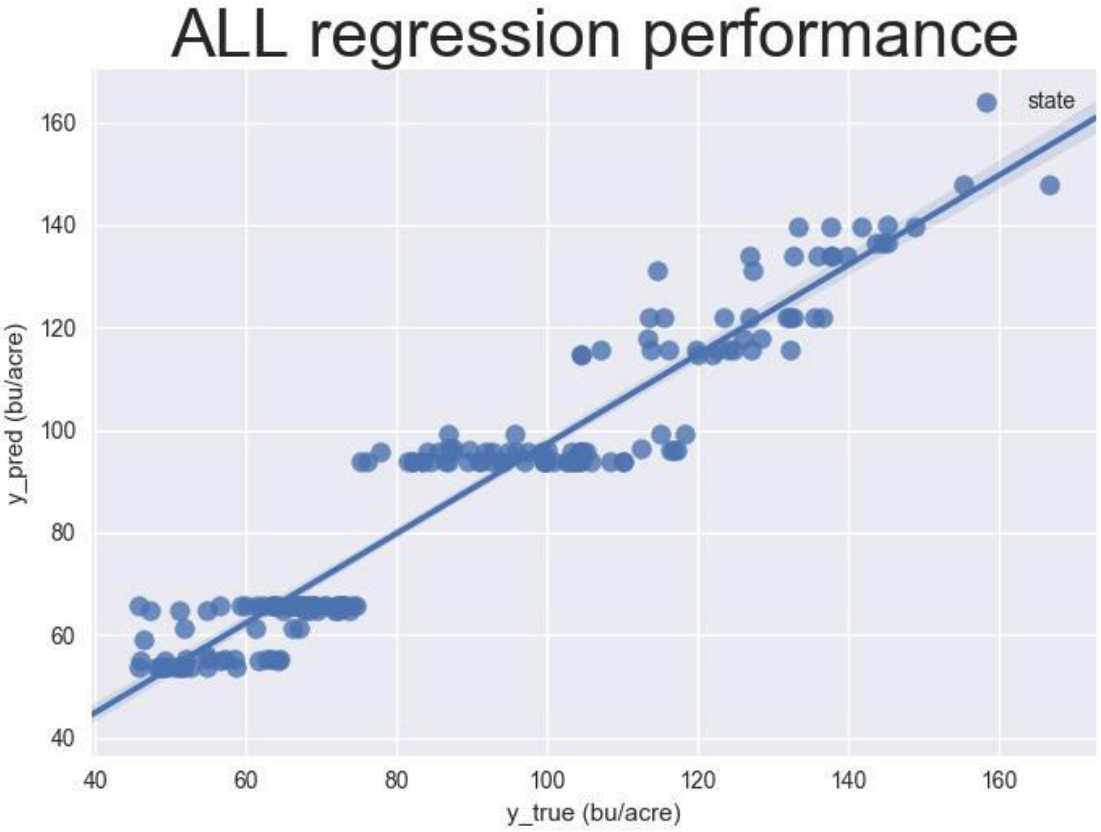


Figure 4.13 True vs Predicted yield analysis

The performance analysis measures for the comparison between different classification mechanism are as follows

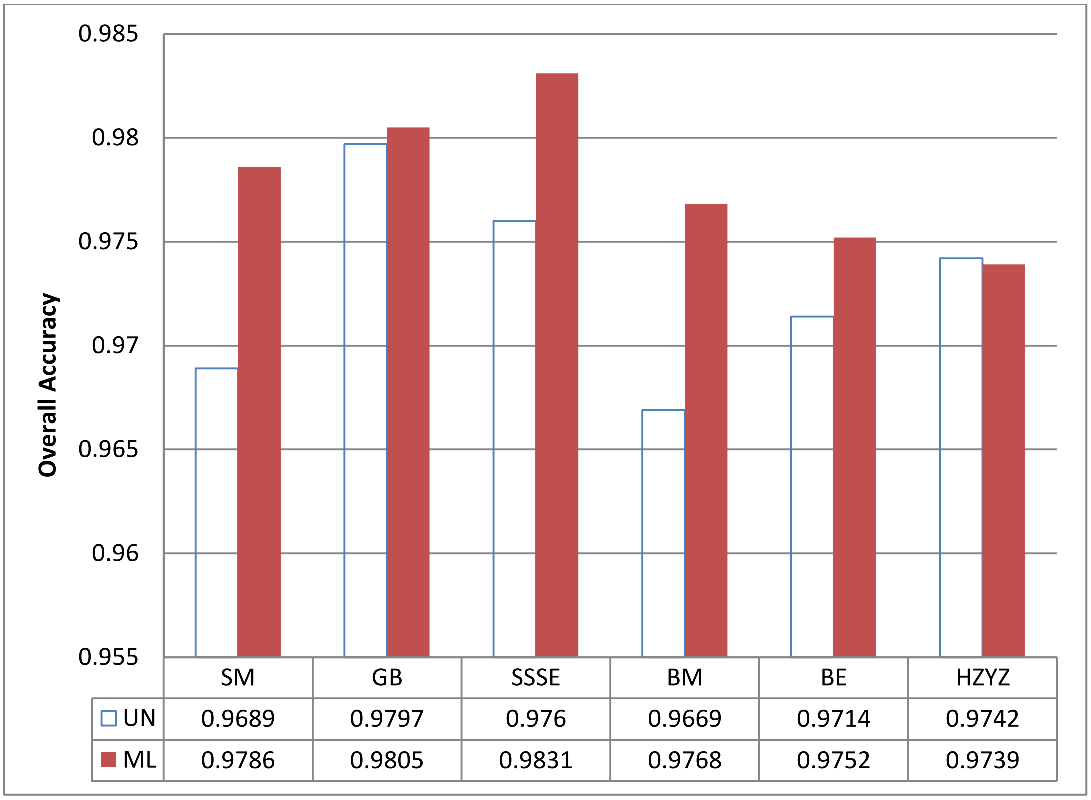
4.10 Classification

=>	SM		SSSE		GB		BM		BE		HZYZ	
=>	UN	ML	UN	ML	UN	ML	UN	ML	UN	ML	UN	ML
Overall Accuracy:	0.9689	0.9786	0.9760	0.9831	0.9799	0.9800	0.9669	0.9768	0.9714	0.9752	0.9742	0.9739
Average Accuracy:	0.9961	0.9973	0.9975	0.9979	0.9966	0.9975	0.9959	0.9971	0.9964	0.9969	0.9968	0.9967
Average Precision:	0.9112	0.9124	0.9812	0.9815	0.9341	0.9188	0.9051	0.9188	0.9329	0.9185	0.9767	0.9680
Average Sensitivity:	0.9800	0.9808	0.9851	0.9863	0.9842	0.9831	0.9760	0.9810	0.9818	0.9797	0.9767	0.9751
Average Specificity:	0.9978	0.9985	0.9981	0.9988	0.9986	0.9986	0.9977	0.9984	0.9980	0.9983	0.9982	0.9982

Table 4.2 Classification Accuracies

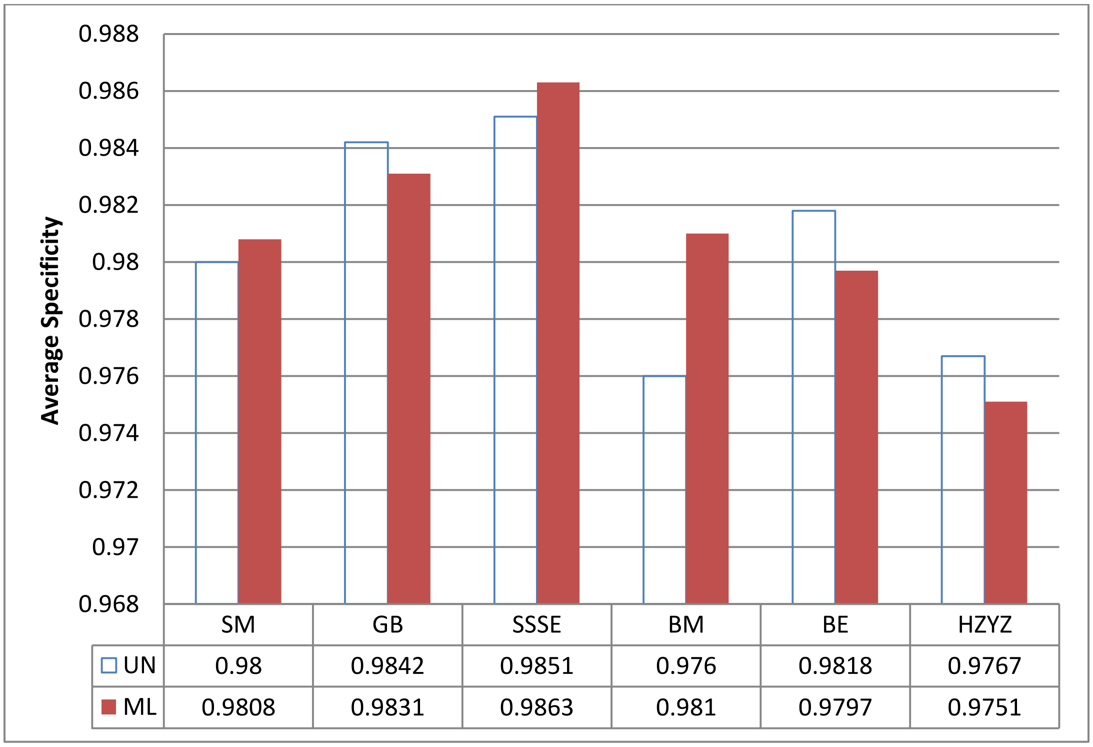
SM – Shi Malik
GB – Gillis Bowles
SSSE – Spatial Spectral Schrodinger Eigenmaps
BM - Benedetto-M: Fused Metric
BE - Benedetto-E: Fused Eigenvectors
HZYZ - Hou-Zhang-Ye-Zheng

Table 4.3 Analysis methods



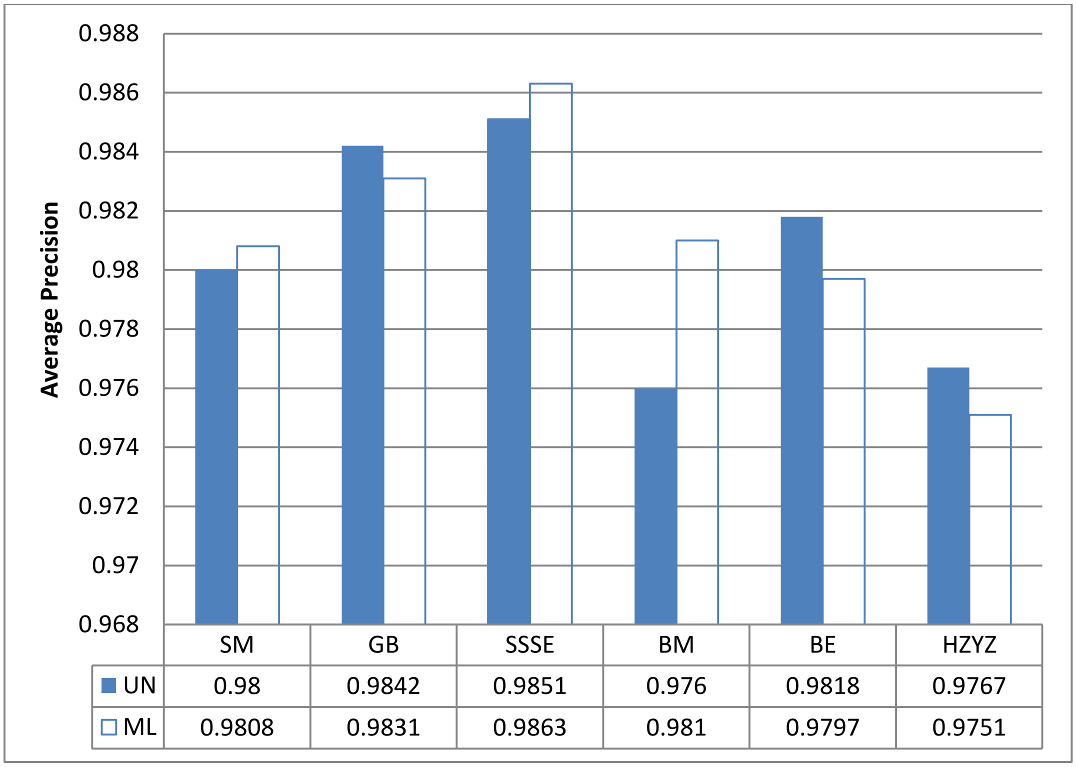
	SM	GB	SSSE	BM	BE	HZYZ
□ UN	0.9689	0.9797	0.976	0.9669	0.9714	0.9742
■ ML	0.9786	0.9805	0.9831	0.9768	0.9752	0.9739

Figure 4.14 Overall Accuracy



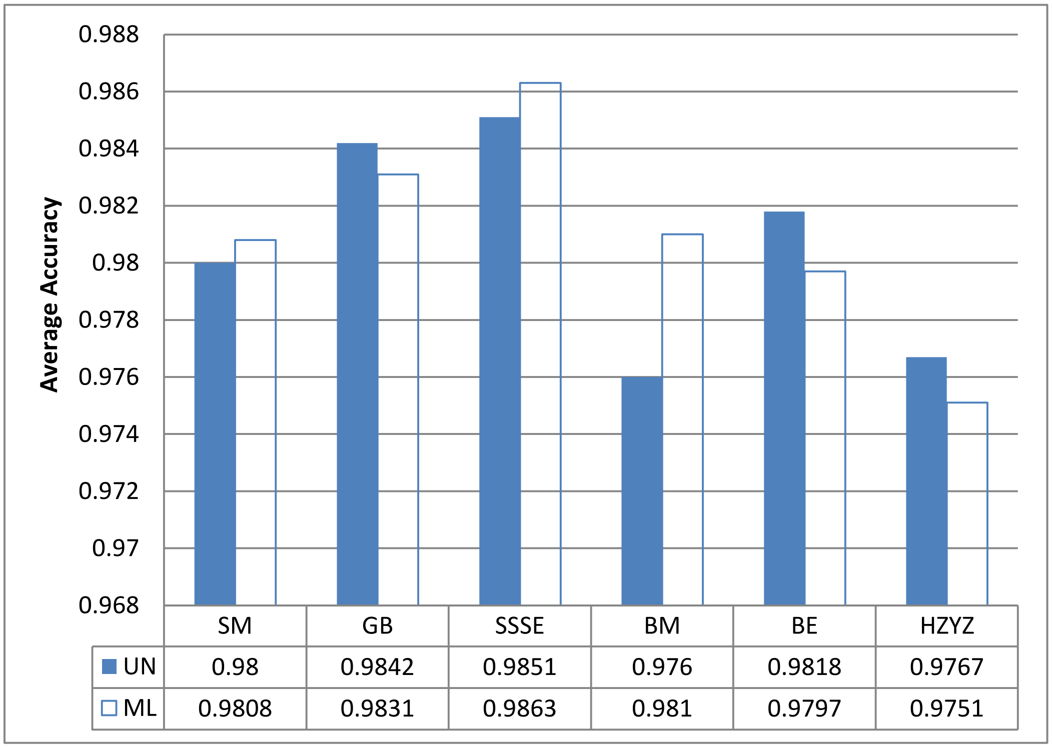
	SM	GB	SSSE	BM	BE	HZYZ
□ UN	0.98	0.9842	0.9851	0.976	0.9818	0.9767
■ ML	0.9808	0.9831	0.9863	0.981	0.9797	0.9751

Figure 4.15 Average Specificity



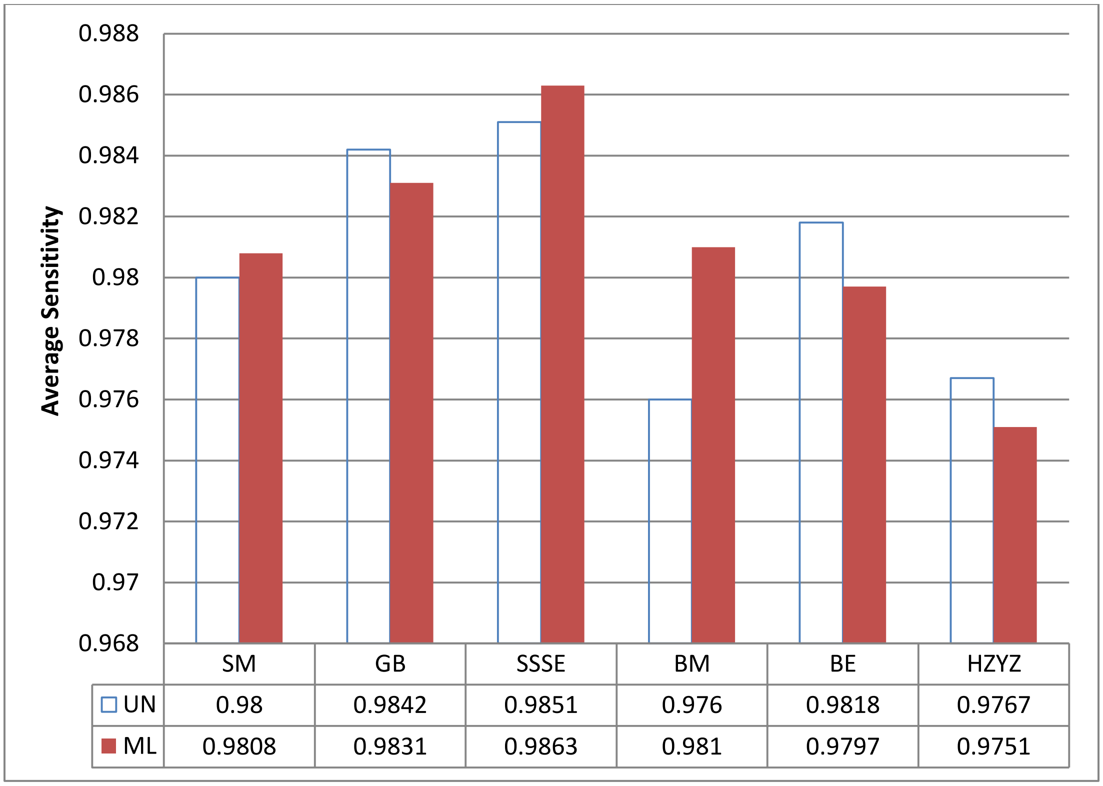
	SM	GB	SSSE	BM	BE	HZYZ
■ UN	0.98	0.9842	0.9851	0.976	0.9818	0.9767
□ ML	0.9808	0.9831	0.9863	0.981	0.9797	0.9751

Figure 4.16 Average Precision



	SM	GB	SSSE	BM	BE	HZYZ
■ UN	0.98	0.9842	0.9851	0.976	0.9818	0.9767
□ ML	0.9808	0.9831	0.9863	0.981	0.9797	0.9751

Figure 4.17 Average Accuracy



	SM	GB	SSSE	BM	BE	HZYZ
□ UN	0.98	0.9842	0.9851	0.976	0.9818	0.9767
■ ML	0.9808	0.9831	0.9863	0.981	0.9797	0.9751

Figure 4.18 Average Sensitivity

The above Figures from 4.11 to 4.15 depict the performance measures of the classification done by analysis of 6 methods mentioned in table 4.2. Hence the ML constrained performance is upgraded in every measure in case of Spectral spatial Schrodinger Eigenmaps.

Chapter 5

Conclusion and future work

Machine learning is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed. The proposed machine recommends the suitable crop given an image of the soil and the parameters like latitude and longitude, with classification of the soil class intermediate. The system builds up an Image Classifier Model, using SSSE and SVM, which acts as an image classifier builder. The Image Pre-processor used to remove the noise from the image (unwanted area). This could then ease the work of Image Classifier Evaluator, to predict the soil class with improved accuracy. This model also uses the super pixel (collection of pixel) instead of individual pixel.

The system can be extended to the mobile application to help the farmers by uploading the image of agriculture area. The efficiency of the pre-processing is limited by the amount of unwanted information (like leaves, grass and other stuffs) present in it. Due to this undesirable information present in the input image, both during training and classification, the pre-processor fails to identify the exact contours, thus failing to perform with improved efficiency. The parameter for the image like climatic factor, moisture and past dataset can be used to predict the yield of the crop. Collection of more valid details of soil class, latitude, longitude and suitable crop can greatly accelerate the efficiency of work. The pre-processing unit could hence be improved and a lot more features can be extended, thus significantly contributing towards the agricultural welfare worldwide.

Chapter 6

References

[1] Xiang Xu, Jun Li Mauro Dalla Mura- Multiple Morphological Component Analysis Based Decomposition for Remote Sensing Image Classification-IEEE Transactions on GeoScience and Remote Sensing. Pages 3083-3102,Jan-2015.

[2] X.E. Pantazi D. Moshou T. Alexandridis b , R.L. Wheton M Mouazen Wheat yield prediction using machine learning and advanced sensing techniques Computers and Electronics and Agriculture Pages 57–65,June-2016.

[3] Xia Zhang, Yanli Sun, Kun Shang, Lifu Zhang, Senior Member, IEEE, and Shudong Wang "IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING" Pages 01,Fevruary-2016.

[4] Wei Yao, Otmar Loffeld - Application and Evaluation of a Hierarchical Patch Clustering Method for Remote Sensing Images, VOL. 9, NO. 6, JUNE 2016 2279 – 2289.

[5] Michael Johnson, William Hsieha, AlexJ. Cannonb, Andrew Davidsonc, Frédéric Bédardd -Crop yield forecasting on the Canadian Prairies by remotely sensed vegetation indices and machine learning methods on Agricultural and Forest Meteorology 218–219 (2016).

[6] Z. Xue, J. Li, L. Cheng, and P. Du, "Spectral–spatial classification of hyperspectral data via morphological component analysis-based image separation," IEEE Transaction, vol. 53, no. 1, pp. 70–84, Jan. 2015.

[7] J. L. Starck, M. Elad, and D. L. Donoho, “Image decomposition via the combination of sparse representations and a variational approach,” IEEE Trans. Image Process., vol. 14, no. 10, pp. 1570–1582, Oct. 2015.

[8] X. Jia, B. C. Kuo, and M. Crawford, “Feature mining for hyperspectral image classification,” Proc. IEEE, vol. 101, no. 3, pp. 676–697, Mar. 2015.

[9] Imaging Spectroscopy: Earth and Planetary Remote Sensing with the USGS Tetracorder and Expert Systems Roger N. Clark, Gregg A. Swayze, K. Eric Livo, Raymond F. Kokaly, Steve J. Sutley, Journal of Geophysical Research, 2012.

[10]J. Li, J. M. Bioucas-Dias, and A. Plaza, “Semi-supervised discriminative random field for hyperspectral image classification,” in Proc. 4th WHISPERS, pp. 4/ 2015.

[11] J. L. Starck, M. Elad, and D. L. Donoho, “Image decomposition via the combination of sparse representations and a variational approach,” IEEE Trans. Image Process., vol. 14, no. 10, pp. 1570–1582, Oct. 2015.

[12] M. Fauvel, J. Chanussot, and J. A. Benediktsson, “A spatial–spectral kernel-based approach for the classification of remote-sensing images,” Pattern Recognit., vol. 45, no. 1, pp. 381–392, Jan. 2012.

[13] J. C. Nunes, Y. Bouaoune, E. Delechelle, O. Niang, and P. Bunel, “Image analysis by bidimensional empirical mode decomposition,” Image Vis. Comput., vol. 21, no. 12, pp. 1019–1026, Nov. 2013.

[14] B. Demir and S. Erturk, “Empirical mode decomposition of hyperspectral images for support vector machine classification,” IEEE Trans. Geosci. Remote Sens., vol. 48, no. 11, pp. 4071–4084, Nov. 2010.

[15] Y. Tang, Y. Lu, and H. Yuan, “Hyperspectral image classification based on three-dimensional scattering wavelet transform,” IEEE Trans. Geosci. Remote Sens., vol. 53, no. 5, pp. 2467–2480, May 2015.

[16] J. M. Bioucas-Dias et al., “Hyperspectral remote sensing data analysis and future challenges,” IEEE Geosci. Remote Sens. Mag., vol. 1, no. 2, pp. 6–36, Jun. 2013.

[17] M. Fauvel, Y. Tarabalka, J. A. Benediktsson, J. Chanussot, and J. C. Tilton, “Advances in spectral–spatial classification of hyperspectral images,” Proc. IEEE, vol. 101, no. 3, pp. 652–675, Mar. 2013.

[18] A. Plaza et al., “Recent advances in techniques for hyperspectral image processing,” Remote Sens. Environ., vol. 113, no. S1, pp. 110–122, Sep. 2014.

[19] Y. Tarabalka, J. A. Benediktsson, and J. Chanussot, “Spectral–spatial classificationof hyperspectral imagery based on partitional clustering techniques,” IEEE Trans. Geosci. Remote Sens., vol. 47, no. 8, pp. 2973–2987, Aug. 2009.

[20] Y. Tarabalka, J. Chanussot, and J. A. Benediktsson, “Segmentation and classification of hyperspectral images using watershed transformation,” Pattern Recognit., vol. 43, no. 7, pp. 2367–2379, Jul. 2010.

[21] Benedetto, J., Czaja, W., Dobrosotskaya, J, Doster, T, Duke, K, and Gillis, “Integration of heterogeneous data for classification in hyperspectral satellite imagery,” vol. 21, no. 12, pp. 1019–1026, Nov. 2

[22] Cahill, N. D, Czaja, W, and Messinger, “Schroedinger eigenmaps with nondiagonal potentials for spatial-spectral clustering of hyperspectral imagery”, vol. 48, no. 11, pp. 4071–4084, Nov. 2010.

[23] Belkin, M. and Niyogi, “Laplacian eigenmaps for dimensionality reduction and data representation,” vol. 45, no. 1, pp. 381–392, Jan. 2012.

[24] Czaja, W. and Ehler, “Schroedinger eigenmaps for the analysis of biomedical data,” IEEE Transactions on Pattern Analysis and Machine Intelligence vol 35,pp 1274–1280 (May 2013).

[25] Wagstaff, Cardie, Rogers, and Schrodl, “Constrained k-means clustering with background knowledge,” in Proceedings of the Eighteenth International Conference on Machine Learning, ICML ’01,pp 577–584 2015.

[26] Benedetto, Czaja, Dobrosotskaya, and Gillis, “Integration of heterogeneous data for classification in hyperspectral satellite imagery,” Vol. 8390, 839027–1–839027–12 June 2014.

[27] Bachmann, Ainsworth, and Fusina, “Exploiting manifold geometry in hyperspectral imagery,” Geoscience and Remote Sensing, IEEE Transactions on 43, 441–454 March 2015.

[28] Fauvel, Chanussot, and Benediktsson, “Kernel principal component analysis for the classification of hyperspectral remote sensing data of urban areas,” EURASIP Journal on Advances in Signal Processing ,vol 14 ,pp 1–14 2014.

[29] Prasad, S. and Bruce, “Limitations of principal components analysis for hyperspectral target recogntion,” Geoscience and Remote Sensing Letters, IEEE 5, 625–629., and Zheng 2013.