A Decision Tree Approach to Extract Knowledge for Improving Satellite Image Classification

Amit Bharti
December, 2004
A Decision Tree Approach to Extract Knowledge for Improving Satellite Image Classification

by

Amit Bharti

Thesis submitted to the International Institute for Geo-information Science and Earth Observation in partial fulfilment of the requirements for the degree of Master of Science in Geoinformatics

Thesis Assessment Board:

Chairman: Prof. Dr. Alfred Stein, ITC
External Examiner: Dr. P. K. Garg, IIT
IIRS Member: Mr. C. Jegananthan, IIRS
Supervisor: Dr. Sameer Saran, IIRS

Thesis Supervisors:

Dr. Sameer Saran (IIRS)
Dr. Valentyn Tolpekin (ITC)

INTERNATIONAL INSTITUTE FOR GEO-INFORMATION SCIENCE AND EARTH OBSERVATION
ENSECHEDE, THE NETHERLANDS

&

INDIAN INSTITUTE OF REMOTE SENSING, NATIONAL REMOTE SENSING AGENCY (NRSA),
DEPARTMENT OF SPACE, DEHRADUN, INDIA
I certify that although I may have conferred with others in preparing for this assignment, and drawn upon a range of sources cited in this work, the content of this thesis report is my original work.

Signed:

Disclaimer

This document describes work undertaken as part of a programme of study at the International Institute for Geo-information Science and Earth Observation. All views and opinions expressed therein remain the sole responsibility of the author, and do not necessarily represent those of the institute.
Dedicated to my mother…
Acknowledgements

I would like to express my gratitude to Dr. Sameer Saran, my first supervisor for all his guidance, comments, unconditional help and suggesting me this topic. My gratefulness to my second supervisor, Dr. Valentyn Tolpekin, his comments and valuable suggestions are spread in this work.

Thanks are extended to J.R. Quinlan, author of See5, for providing me the evaluation version of See5 software for ten days. Without that it could not be possible for me to do this research.

I want to express my sincere thanks to Dr. P. S. Roy, former Dean, IIRS, Dr. V. K. Dadhwal, Dean, IIRS, Mr. P. L. N. Raju, Head, Geoinformatics Division and Mr. C. Jeganathan, Program Coordinator, Msc Geoinformatics for their help to the students.

Special thanks to my colleagues in the Geoinformatics, 2003 Batch. It was a great experience to work and study with all of you. Thanks to Hitender Padalia for helping me during my field work. I am grateful to Harish Karnatak for encouraging me throughout the project. Thanks to Sanjeev, Ranbir, Nikhil, Mukesh, and Dilip for their help and valuable discussions.

Special thanks are reserved for H.S. Sudhira “Gubbi - The Urban Planner”, for helping me in making table of contents and formatting of my thesis. Thanks for making my thesis presentable.

My special thanks to my parents, bhaiya and bhabhi and to my sister for their love and moral support, due to their blessings I could get the opportunity to visit Holland for three months and pursue my dreams.

Lastly, my thanks go to Ms Lesley Edna Wintle for reading the thesis and offering succinct suggestions about improvements to its readability. Thanks for you encouragements, support and for your prayers.
Abstract

Supervised classification is one of important tasks in remote sensing image interpretation, in which the image pixels are classified to various predefined land use/land cover classes based on the spectral reflectance values at different bands. In reality some classes have very close spectral reflectance values and therefore they overlap in the feature space. This produces spectral confusion among the classes and results in inaccurate classified images. It is very difficult to classify such classes correctly using traditional parametric classifier like Maximum Likelihood Classifier. The other drawback of these traditional parametric classifiers is that they require training data to be normally distributed. Due to this fact it becomes difficult to add ancillary layers into classification procedures to improve accuracy of the classification. To remove such spectral confusion we require extra spectral and spatial knowledge. Extra knowledge can come from the ancillary information or from experience and knowledge of domain experts. Classification done using such knowledge is known as knowledge base classification. But such classification needs strong knowledge base, which sometimes become drawback of this process because of the knowledge acquisition process. Generally knowledge base is created with the help of knowledge acquired by interacting with the experts. The traditional way of knowledge acquisition is that the knowledge engineer interacts with the corresponding domain expert; write up his /her experience and knowledge in a interpretable form and then feed the entire acquired knowledge in the computer in a symbolic form such as if-then rules. This is usually long and repeated process, takes too much of time. It is also not always possible that expert is available all the time.

The question is, from where and how we can build knowledge base automatically? This research explores a non-parametric decision tree classifier to extract knowledge from the spatial data in the form of classification rules.

The results of the research show that the knowledge extracted from proposed approach can remove the problem of spectral confusion to some extent. The result of the classified image using the extracted knowledge was compared with the result of the maximum likelihood classification. It was found that the overall accuracy of the classification was improved by approximately 10 percent after using the extracted knowledge in the classification of satellite images.

Keywords: Knowledge Base, Decision Tree Classifier, Maximum Likelihood Classification, Accuracy.
# Table of Contents

Acknowledgements ........................................................................................................... I
Abstract............................................................................................................................ II
Table of Contents .............................................................................................................. III
List of Figures .................................................................................................................. V
List of Tables .................................................................................................................... VI
1. Introduction .................................................................................................................. 1
   1.1. Overview .................................................................................................................... 1
   1.2. Decision Tree ............................................................................................................ 2
      1.2.1. Decision Tree Algorithm .................................................................................... 2
   1.3. Previous Work ........................................................................................................... 3
   1.4. Problem Definition .................................................................................................... 4
   1.5. Motivation ................................................................................................................ 4
   1.6. Research Objectives ................................................................................................ 4
   1.7. Research Questions .................................................................................................. 5
   1.8. Proposed Method ...................................................................................................... 5
   1.9. Software Used .......................................................................................................... 6
   1.10. Thesis Structure ...................................................................................................... 6
2. Image Classification ....................................................................................................... 7
   2.1. Introduction ............................................................................................................... 7
   2.2. Supervised Classification ........................................................................................... 7
   2.3. Unsupervised Classification ....................................................................................... 8
   2.4. Parametric Classifier ................................................................................................ 8
      2.4.1. Maximum Likelihood Classification ..................................................................... 8
      2.4.2. Bayes Rule .......................................................................................................... 9
   2.5. Non-Parametric Classifier ....................................................................................... 10
      2.5.1. Decision Tree ..................................................................................................... 10
   2.6. Fuzzy Classifier ........................................................................................................ 10
   2.7. Accuracy Assessment ............................................................................................... 11
      2.7.1. Overall Accuracy ................................................................................................. 11
      2.7.2. User’s and Producer’s Accuracy ......................................................................... 11
      2.7.3. The Kappa Statistic ............................................................................................ 12
   2.8. Knowledge Base Classification ................................................................................. 12
      2.8.1. Automatic Extraction of Knowledge .................................................................... 12
      2.8.2. Knowledge Representation ............................................................................... 13
      2.8.3. Knowledge from Decision Tree Classifier ........................................................ 13
3. Decision Tree ............................................................................................................... 14
   3.1. Introduction ............................................................................................................... 14
   3.2. Decision Tree Classifier .......................................................................................... 15
   3.3. Criteria for Evaluating Splits – Information Theory ............................................... 16
A DECISION TREE APPROACH TO EXTRACT KNOWLEDGE FOR IMPROVING SATELLITE IMAGE CLASSIFICATION

3.3.1. Tests on Continuous Attributes ................................................................. 17
3.4. Pruning ........................................................................................................... 18
3.4.1. Error Based Pruning .................................................................................. 19
3.5. Fuzzy Decision Tree ...................................................................................... 19
3.6. Decision Trees to Rules ................................................................................ 21
3.7. Boosting ......................................................................................................... 22
3.8. Strengths of Decision Tree Methods ............................................................ 23
3.9. Weaknesses of Decision Tree Methods ....................................................... 23

4. Study Area and Data ......................................................................................... 24
4.1. Study Area ...................................................................................................... 24
4.2. Satellite Data .................................................................................................. 26
4.3. Reconnaissance Survey ................................................................................ 27
4.4. Geometric Image Correction ........................................................................... 27
4.5. Feature Selection ............................................................................................ 27
4.5.1. Transformed Divergence .......................................................................... 27
4.5.2. Feature Space Plot ..................................................................................... 29
4.6. Ancillary Layers ............................................................................................. 30
4.6.1. Digital Elevation Model ........................................................................... 30
4.6.2. Slope Map .................................................................................................. 30
4.6.3. Texture ....................................................................................................... 31

5. Methods ........................................................................................................... 32
5.1. Introduction .................................................................................................... 32
5.2. Conceptual Flow Chart .................................................................................. 33
5.3. Maximum Likelihood Classification (MLC) ................................................ 34
5.4. Decision Tree Classifier ................................................................................ 34
5.4.1. Data Preparation for Decision Tree ......................................................... 34
5.4.2. Training and Testing Dataset ................................................................... 35
5.4.3. Fuzzy Decision Tree ................................................................................ 38
5.4.4. Pruning ...................................................................................................... 38
5.4.5. Boosting .................................................................................................... 38
5.4.6. Classification Rules ................................................................................... 38
5.5. Knowledge Base Classification .................................................................... 40
5.5.1. Classification using Extracted Knowledge .............................................. 40
5.5.2. Prior Probabilities ..................................................................................... 40
5.5.3. Post-Classification Sorting ..................................................................... 41
5.6. Accuracy Assessment ................................................................................... 41

6. Results and Discussions .................................................................................. 42
6.1. Maximum Likelihood Classification ............................................................ 42
6.2. Training Set Statistics .................................................................................... 43
6.2.1. Pruning ...................................................................................................... 45
6.2.2. Boosting .................................................................................................... 46
6.2.3. Fuzzy Decision Tree ................................................................................ 46
6.3. Classification using Extracted Knowledge ................................................................. 46
   6.3.1. Classification Rules Generated by See5 Decision Tree Classifier .......................... 47
6.4. Maximum Likelihood Classification with Prior Probabilities ........................................ 49
6.5. Post Classification Sorting ......................................................................................... 51
6.6. Summary .................................................................................................................... 53
7. Conclusions and Recommendation ................................................................................. 55
   7.1. Conclusions .............................................................................................................. 55
      7.1.1. Training Set ....................................................................................................... 55
      7.1.2. Decision Tree .................................................................................................... 56
      7.1.3. Accuracy ........................................................................................................... 56
   7.2. Recommendations .................................................................................................. 57
8. Bibliography .................................................................................................................. 58
Annexure A ....................................................................................................................... 62
Annexure B ......................................................................................................................... 63
Annexure C ........................................................................................................................ 66

List of Figures

Figure 1.1: Example of Decision Tree (Source: Pal et al, 2001) ............................................. 3
Figure 3.1: Decision Tree .................................................................................................... 15
Figure 4.1: Location of Study Area .................................................................................... 25
Figure 4.2: LISS-III FCC Image of Study Area ................................................................. 26
Figure 4.3: Transformed Divergence for all Four Bands of LISS-III ................................. 28
Figure 4.4: Overlapping of Classes in Feature Space ......................................................... 29
Figure 5.1: Flow Chart .................................................................................................... 33
Figure 5.2: Layer Stacking ............................................................................................... 35
Figure 5.3: Training Cases ............................................................................................... 36
Figure 5.4: Decision Tree ............................................................................................... 37
Figure 5.5: Classification Rules ....................................................................................... 39
Figure 6.1: LISS-III Classified Image using Maximum Likelihood Classification ............ 42
Figure 6.2: Training Set Sizes and Classifier Accuracy on Training Cases ....................... 44
Figure 6.3: Training Set Size and Classifier Accuracy on Test Cases ............................... 45
Figure 6.4: LISS-III Classified Image using Extracted Knowledge ................................. 48
Figure 6.5: LISS-III Classified Image using MLC and Prior Probabilities ......................... 50
Figure 6.6: LISS-III Classified Image using Post Classification Sorting ............................ 52
Figure 6.7: Comparison of User’s Accuracy of Different Classes using Different Method .... 53
Figure 6.8: Comparison of Producer’s Accuracy of Different Classes using Different Method.... 54
List of Tables

| Table 6-1: Accuracy Assessment of Classified LISS-III image using Maximum Likelihood Classification | 43 |
| Table 6-2: Summary of the Results of Pruning Method Applied on Four Training Sets | 45 |
| Table 6-3: Summary of the Results of Boosting Applied on Four Training Sets | 46 |
| Table 6-4: Summary of the Results of Fuzzy Decision Tree Applied on Four Training Sets | 46 |
| Table 6-5: Accuracy Assessment of Classified LISS-III Image using Extracted Knowledge | 49 |
| Table 6-6: Estimated Prior Probabilities | 50 |
| Table 6-7: Accuracy Assessment of Classified LISS-III Image using MLC and Prior Probability | 51 |
| Table 6-8: Accuracy Assessment of Classified LISS-III Image using Post Classification Sorting | 52 |
| Table 6-9: Summary of Classification Results of Different Method | 53 |
1. Introduction

1.1. Overview

Supervised classification is one of important tasks in remote sensing image interpretation, in which the image pixels are classified to various predefined land use/land cover classes based on the spectral reflectance values at different bands. In reality some classes have very close spectral reflectance values and therefore they overlap in the feature space. This produces spectral confusion among the classes and results in inaccurate classified images.

It is very difficult to classify such classes correctly using traditional parametric classifier like Maximum Likelihood Classifier. The other drawback of these traditional parametric classifiers is that they require training data to be normally distributed (Jensen, 1996). Due to this fact it becomes difficult to add ancillary layers into classification procedures to improve accuracy of the classification. Ancillary layers generally have bi- or multimodal distribution. Therefore maximum likelihood classification does not give good results after addition of ancillary layers. To remove such spectral confusion we require extra spectral and spatial knowledge. Extra knowledge can come from the ancillary information or from experience and knowledge of domain experts. Classification done using such knowledge is known as knowledge base classification. But such classification needs strong knowledge base, which sometimes become drawback of this process because of the knowledge acquisition process. Generally knowledge base is created with the help of knowledge acquired by interacting with the experts. The traditional way of knowledge acquisition is that the knowledge engineer interacts with the corresponding domain expert; write up his/her experience and knowledge in an interpretable form and then feed the entire acquired knowledge in the computer in a symbolic form such as if-then rules (Kontoes, 1993). This is usually long and repeated process takes too much of time. It is also not always possible that expert is available all the time.

There is large amount of knowledge hidden in spatial databases that can be used in classification of satellite images. Some knowledge can be extracted by simple GIS query and other knowledge is so deep such as classification rules, spatial distribution rules, spatial association rules that are not stored explicitly in the database but we can be extracted them by computation and machine learning process (Li, 2000).

The question arises, from where and how we can extract this hidden knowledge automatically? This thesis explores a decision tree classifier based on machine learning to extract knowledge in the form of classification rules from the satellite and topographical data.
1.2. Decision Tree

Decision tree is one of the inductive learning algorithms that generate a classification tree to classify the data. It is based on the “divide and conquer” strategy. The classification tree is made by recursive partitioning of the feature space, based on a training set. At each branching, a specific decision rule is implemented, which may involve one or more combinations of the attribute inputs or features (Quinlan, 1993).

A decision tree is composed of a root node, a set of interior nodes, and terminal nodes, called “leaves”. The root node and interior nodes, referred to collectively as non-terminal nodes, are linked into decision stages. The terminal nodes represent final classification. The classification process is implemented by a set of rules that determine the path to be followed, starting from the root node and ending at one terminal node, which represents the label for the object being classified. At each non-terminal node, a decision has to be taken about the path to the next node. Fig1.1 illustrates a simple decision tree using pixel reflectance as input.

1.2.1. Decision Tree Algorithm

Step1: Let $T$ be the set of training instances.

Step2: Choose an attribute that best differentiates the instances in $T$.

Step3: Create a tree node whose value is the chosen attribute.

- Create child links from this node where each link represents a unique value for the chosen attribute.

- Use the child link values to further subdivide the instances into subclasses.

Step4: For each subclass created in step 3:

- If the instances in the subclass satisfy predefined criteria or if the set of remaining attribute choices for this path is null, specify the classification for new instances following this decision path.

- If the subclass does not satisfy the criteria and there is at least one attribute to further subdivide the path of the tree, let $T$ be the current set of subclass instances and return to step 2.

It is obvious that if the rules are not complete after tracing through the decision tree, some pixels will remain unclassified. Therefore the efficiency and performance of this approach is strongly affected by tree structure and choice of features selected for training.
1.3. Previous Work

Eklund et al. (1998) extracted knowledge from TM images and geographic data in soil salinity analysis using Inductive leaning. Eklund et al. (1994) used a decision tree approach to assess the effect of incremental data layers on groundwater recharge estimates.

Number of authors have conducted comparisons of decision tree and other classifiers such as MLC and neural network and found that it is best all round choice (German et al.; Pal et al. 2001). “In contrast to neural networks, decision trees can be trained quickly, are rapid in execution, takes less computational time and is not a “black box” like a neural network, working of which is concealed from view” (Gahegan et al., 1998).

Lees and Ritman (1991) examined the use of decision trees for mapping vegetation species using LandSAT and other spatial data. Byungyong and Landgrebe (1991) used decision trees to classify AVIRIS data.

Decision tree algorithms consistently outperformed maximum likelihood techniques when classifying spectral data. “It is non-parametric, simple, flexible and computationally efficient” (Friedl et al., 1997).

The advantages of decision tree classifier over traditional statistical classifier include its simplicity, ability to handle missing and noisy data, and non-parametric nature i.e., decision trees are not constrained by any lack of knowledge of the class distributions.
1.4. **Problem Definition**

The basic idea of accuracy assessment is to compare the result of the pixel classification (supervised or unsupervised) with the actual class that pixel belongs to (as indicated by ground truth). If the predicted class and ground truth do not match for some pixel, then it means that the pixel is misclassified. There are many reasons for the misclassification of the pixels, but the main source is inseparability of the classes in the feature space. Certain classes have similar spectral reflectance values due to which they overlap with the other classes in the feature space. This produces spectral confusion i.e., same class with different spectral values and different classes with the same spectral values.

For example, in India the material used to build houses consists of sand and stones, which are brought from the riverbed. Therefore urban area and riverbed have similar spectral reflectance. When supervised classification was done using Maximum Likelihood Classifier, the two classes overlapped in feature space and that resulted in misclassification of some pixels. It was found, that maximum portion of the riverbed was classified as urban in the resultant image. This leads to wrong results of image analysis. In most of the cases it was found that the maximum likelihood classifier is unable to solve such spectral confusion. This problem is also faced for other classes, such as agriculture and tea garden, shadows and water. Therefore classifying such complex features with the maximum likelihood or other parametric classifier alone is not a good choice. Extra knowledge is required to accurately classify such classes.

1.5. **Motivation**

Though extra knowledge can come from expert systems and individual experience, but the point here is to automatically extract knowledge to build a knowledge base. The main motivation is to use decision tree approach, to discover knowledge from satellite and topographical data, for the classification of the satellite images.

1.6. **Research Objectives**

The main objective of the research is to improve the accuracy of the classification process by applying extracted knowledge from the spatial data.

Knowledge in the form of classification rules will be extracted from the spatial data by using the Decision Tree approach. Classification rules will then be applied to classify the satellite image and then the accuracy assessment of the whole process will be done to check, whether there is any change in the classification accuracy.

Other sub-objective is to check how cardinality (the size) of the training set used for the decision tree classifier affects the accuracy of the classification process.
1.7. **Research Questions**

- How to extract knowledge from the Spatial Data using Decision tree classifier?
- How to use extracted knowledge in the classification process of satellite images?
- What is the effect of pruning, boosting on the performance of the decision tree classifier?
- Comparison of fuzzy and crisp decision tree based on accuracy of training and test cases.
- What is the effect on the overall accuracy of the classified image after using the knowledge extracted using decision tree?
- What is the effect of the training set size (cardinality or number of rows) on the accuracy of the classified image?

1.8. **Proposed Method**

Following are the proposed steps to achieve the goal of the research and to answer the research question.

**Step 1:** Fieldwork to collect the ground truth.

**Step 2:** Training set will be prepared based on the ground truth and then the image (IRS P6 LISS-III, Doon valley) will be classified using maximum likelihood classifier.

**Step 3:** Another training set will be prepared using the topographical data and spectral bands of LISS-III, for the decision tree classifier. This training set will be a text editor file containing rows with number of training samples and columns consists of different attributes e.g. Red band, NIR band, Blue Band, Elevation, Slope and a Class attribute. The Class attribute will contain different land use class e.g. urban, riverbed, and agriculture for corresponding training sample.

**Step 4:** Decision tree algorithm will be used to generate a decision tree from the training set in step3.

**Step 5:** Classification rules will be deduced from the decision tree.

**Step 6:** Once classification rules are deduced from step5 we would apply several approaches to use them for further classification of the satellite image.
Step 7: At last accuracy assessment would be done. An error matrix would be generated to check the accuracy of the different classified image resulted from different approaches. Accuracy of the different classified image would be compared with each other. Then conclusion will be derived from the research that whether there is any change in the accuracy of the classified image after using the extracted knowledge.

The size of the training set (cardinality) will be varied and its effect on the accuracy of the final classified image would be studied.

1.9. Software Used

Erdas 8.6, See5 data mining software, ArcGIS.

1.10. Thesis Structure

The whole thesis is divided into 7 chapters explaining the concepts, the data preparation steps, the methods used and the results of the research. The contents of the chapters are outlined below.

Chapter 2 describes the concepts of image classification and various types of classification algorithms including supervised and unsupervised classifiers, parametric and non-parametric classifiers, fuzzy and knowledge base classifiers. The theory behind maximum likelihood classification is described in section 2.4.1 Accuracy assessment is explained in terms of overall accuracy, Kappa statistics and user’s and producer’s accuracy in section 2.7.

Chapter 3 describes the decision tree and its various aspects such as pruning, boosting, and fuzzy decision tree. The theory behind decision tree classification is described, with particular reference to the implemented algorithms i.e., See5.

Chapter 4 describes the data pre-processing and data preparation steps. It also describes the study area and the ancillary layers chosen for this research.

Chapter 5 describes the method followed to achieve objectives and sub-objectives of the research.

Chapter 6 discusses the result of the research. The accuracies of the classified image that can be achieved using extracted knowledge and other proposed methods are assessed, and the results are compared with those achieved using maximum likelihood techniques. Conclusions of the results of the thesis and recommendations for future work are presented in chapter 7.
2. Image Classification

2.1 Introduction

Image classification is a particular case of Pattern Recognition. The overall objective of the classification process is to automatically classify all pixels in an image into land cover classes based on the pre-defined classification model. The term pattern in case of image classification refers to the set of radiance measurements obtained in the various wavelength bands for each pixel.

There are numerous classification algorithms. This chapter gives a brief introduction of the most popular classifiers in the field of remote sensing. Classifiers are described under broad categories such as supervised and unsupervised classifiers, parametric and non-parametric, fuzzy classifiers and knowledge base classifiers. Only maximum likelihood and decision tree would be explained in this chapter under parametric and non-parametric classifiers respectively. Fuzzy classification and knowledge base classification are explained in section 2.6 and 2.8 respectively.

2.2 Supervised Classification

In this type of classification the image analyst supervises the pixel categorisation process by specifying to the algorithm specific information of the various land cover types present in a scene. To do this, representative sample site of known cover type, called training areas, are used to compile a numerical interpretation key that describes the spectral attributes for each feature type of interest. Reflectance value of each pixel in the image is then compared numerically to each category in the interpretation key labelled with the name of the category it looks most like.

Generally there are three major steps involved in the typical supervised classification procedures as follows:

- **Training Stage**: The analyst identifies representative training areas and develops a numerical description of the spectral attributes of each land cover type of interest in the scene.

- **Classification Stage**: Each pixel in the image is categorized into the land cover class it most resembles. If the pixel is not matching to any predefined class then it is labelled as unknown.

- **Accuracy Assessment**: The classified image is compared with some reference image or ground truth to check the accuracy of the classification.


2.3 Unsupervised Classification

Unlike supervised classifiers, unsupervised ones do not utilize training data as the basis for classification. These classifiers try to aggregate reflectance value of pixels in the feature space into well separated clusters. Clusters are considered as classes. Once the spectral grouping has been done, the analyst identifies the obtained classes to some form of reference data.

There are numerous clustering algorithms that can be used to determine the natural spectral clusters present in the image. The most common algorithm is “K-means”. In this approach user has to define the number of clusters or classes to be located in the image. The algorithm automatically locates the centre means of various clusters present in the image and each pixel in the image is then assigned to the cluster whose mean is closest. After all pixels have been classified, revised mean vectors for each of the cluster is computed. The whole process is repeated again until there is no further change in the location of class means vectors.

2.4 Parametric Classifier

Parametric classification algorithms assume that the observed measurement vectors $X_c$ obtained for each class in each spectral band during the training phase of the supervised classification follow some statistical distribution such as Gaussian distribution (Jensen, 1996). The major parametric classifiers under this category are minimum distance, Mahalanobis distance, and maximum likelihood classifier. Maximum likelihood gives better accuracy than others and frequently used in the remote sensing image classification. Therefore Maximum likelihood algorithm is described here as a representative of parametric classifiers.

2.4.1 Maximum Likelihood Classification

The MLC quantitatively evaluates both the variance and covariance of the category spectral response pattern when classifying an unknown pattern. An assumption is made that the distribution of the training set is Gaussian. Under this assumption, the distribution of a training set of a class can be completely described by the mean vector and covariance matrix. Given these parameters, we may compute the statistical probability of a given pixel being a member of a particular class.

Multivariate normal statistical theory describes the probability that an observation $X$ will occur, given that it belongs to a class $k$, as the following function:

$$
\Phi_k(X) = (2\pi)^{-1/2} \left| \Sigma_k \right|^{-1/2} \times e^{-1/2 (X - \mu_k) \Sigma_k^{-1} (X - \mu_k)}
$$

\hspace{1cm} (2.1)

The quadratic product
\[ \chi^2 = (X - \mu_k)^\top \Sigma_k^{-1} (X - \mu_k) \]  

\[(2.2)\]

can be thought of as a squared distance function between the observation and the class mean as scaled and corrected for variance and covariance of the class. As applied in a maximum likelihood decision rule, Equation (2.1) allows the calculation of the probability that an observation is a member of each of \( k \) classes. The individual is then assigned to the class for which the probability value is greatest. In an operational context, observed means, variances, and co-variances substituted by the \( \log \) form of the Equation (2.1).

\[\ln[\Phi(X_i)] = -\frac{1}{2} p \ln(2\pi) - \frac{1}{2} \ln|\Sigma_k| - \frac{1}{2} (X_i - m_k)^\top D_k^{-1} (X_i - m_k)\]

\[(2.3)\]

Since the \( \log \) of the probability is a monotonic increasing function of the probability, the decision can be made by comparing values for each class as calculated from the right hand side of this equation. A simpler decision rule, \( RI \), can be derived from Equation (2.3) by eliminating the constants

\[ R1: \text{Select } k \text{ which minimizes } F_{ik}(X_i) = \ln|D_k| + (X_i - m_k)^\top D_k^{-1} (X_i - m_k) \]

\[(2.4)\]

### 2.4.2 Bayes Rule

An extension of the maximum likelihood classification approach is Bayesian classifier. The maximum likelihood decision rule can be modified easily to take into account in the population of observations as a whole. The prior probability itself is simply an estimate of the objects which will fall into a particular class. These prior probabilities are sometimes termed "weights" since the modified classification rule will tend to weigh more heavily those classes with higher prior probabilities. For example, when classifying a pixel, the probability of the rarely occurring “sand” category might be weighted lightly, and the more likely “urban” class weighted heavily. The analyst gets the a priori probabilities by evaluating historical summaries of the region. The prior probabilities have proved to be a useful way of separating classes with similar spectral reflectance values (Strahler, 1980).

Prior probabilities are incorporated into the classification through a manipulation of the law of Conditional Probability. To begin, two probabilities are defined: \( P(\omega_k) \), the probability that an observation will be drawn from class \( \omega_k \); and \( P(X_i) \), the probability of occurrence of the measurement vector \( X_i \). The law of Conditional Probability or Bayes theorem states that

\[ P(\omega_k | X_i) = \frac{\Phi_k(X_i)P(\omega_k)}{\sum_{k=1}^{\tilde{k}} \Phi_k(X_i)P(\omega_k)} = \frac{\Phi_k^+(X_i)}{\sum_{k=1}^{\tilde{k}} \Phi_k^+(X_i)} \]

\[(2.5)\]
This Equation provides the basis for the decision rule which includes prior probabilities. Since the denominator remains constant for all classes, the observation is simply assigned to the class for which $F_k(X_i) \times P(\omega_k)$, is a maximum.

2.5 Non-Parametric Classifier

A non-parametric classifier is not based on statistics, therefore, it is independent of the properties of the data. Non-Parametric classification algorithm does not take into account the distribution of the training set. They do not require that that the observed measurement vectors $X_c$ obtained for each class in each spectral band during the training phase of the supervised classification should follow Gaussian distribution. Best known classifiers in this category are parallelepiped, decision tree and neural network. A brief introduction of decision tree is given here. The complete working and its various aspects are explained in chapter 3.

2.5.1 Decision Tree

Decision tree is non parametric classifier. Decision tree is an example of machine learning algorithm. They involve a recursive partitioning of the feature space, based on a set of rules that are learned by an analysis of the training set. A tree structure is developed where at each branching a specific decision rule is implemented, which may involve one or more combinations of the attribute inputs. A new input vector then “travels” from the root node down through successive branches until it is placed in a specific class.

The thresholds used for each nodal decision are chosen using minimum entropy or minimum error measures. It is based on using the minimum number of bits to describe each decision at a node in the tree based on the frequency of each class at the node. With minimum entropy, the stopping criterion is based on the amount of information gained by a rule (the gain ratio).

There are several well-established decision tree classifier implementations e.g. C4.5 (Quinlan, 1993), CART (Breiman, 1984). Decision trees are not constrained by any lack of knowledge of the class distributions, as they do not try to model them in any way.

2.6 Fuzzy Classifier

Representing a geographical object is very difficult, as in most of the cases they do not have well defined boundaries, meaning that the boundaries between different phenomena are fuzzy, and/or there is heterogeneity within the class. If the class does not have sharp boundary then the assignment of the pixel to a class is uncertain and this uncertainty can be expressed by fuzzy class membership function (Jensen, 1996). Fuzzy set theory provides useful concepts and methods to deal with uncertain informa-
tion. It is achieved by applying a function called “membership function” on remotely sensed images. The set associated with a membership function and each element in this set has its own membership value towards that particular set. The membership values range between 0 and 1. If the membership value of an element is 0, it means that, it does not belong to that set and if it is 1, then it belongs completely. But, in crisp sets, the membership value is 1 or 0. For fuzzy classification, this function takes values between 0 and 1. Therefore, every pixel has certain membership values in every class. For example, a vegetation classification might include a pixel with grades of 0.68 for class “forest”, 0.29 for class “urban” and 0.03 for “riverbed”. We can see that pixel has higher membership value in class forest than other classes, and therefore it will be assigned to forest class.

### 2.7 Accuracy Assessment

Classification process is not complete until its accuracy is assessed. Accuracy assessment can be performed by comparing two sources of information (Jensen, 1996):

- Remote-sensing derived classification data and
- Reference test data

The relationship of these two sets is summarized in an error matrix where columns represent the reference data while rows represent the classified data. An error matrix is a square array of numbers laid out in rows and columns that expresses the number of sample units assigns to a particular category relative to the actual category as verified in the field. From the error matrix various accuracies can be derived as explained in following sections.

#### 2.7.1 Overall Accuracy

The overall accuracy is weighted by the number of samples (pixels) in each class, i.e. the sum of all samples on the diagonal divided by the total number of samples. However, as a single measure of accuracy, the overall accuracy (or percentage classified correctly) gives no insight into how well the classifier is performing for each of the different classes. In particular, a classifier might perform well for a class which accounts for a large proportion of the test data and this will bias the overall accuracy, despite low class accuracies for other classes. Therefore error matrix itself is not a sufficient way to predict the accuracy of the classified image.

#### 2.7.2 User’s and Producer’s Accuracy

Other measures derived from the error matrix are ‘error of omission’ (or producer’s accuracy) and ‘error of commission’ (or user’s accuracy). Error of omission represents an error from including a pixel to a particular class, which is actually not a part of the class. Commission error represents that a pixel, which should be part of a particular class but is not included.
2.7.3 The Kappa Statistic

The Kappa statistic was derived to include measures of class accuracy within an overall measurement of classifier accuracy (Congalton, 1991). It provides a better measure of the accuracy of a classifier than the overall accuracy, since it considers inter-class agreement. KAPPA analysis yields a $K_{hat}$ statistics that is a measure of agreement or accuracy. The $K_{hat}$ statistic is computed as:

$$K_{hat} = \frac{N \sum_{i=1}^{r} x_{ii} - \sum_{i=1}^{r} (x_{i+} \times x_{+i})}{N^2 - \sum_{i=1}^{r} (x_{i+} \times x_{+i})}$$  \hspace{1cm} (2.6)

Where $r$ is the number of rows in the matrix, $x_{ij}$ is the number of observation in row $i$ and column $i$, and $x_{i+}$ and $x_{+i}$ are the marginal totals for row $i$ and column $i$, respectively, and $N$ is the total number of observations.

2.8 Knowledge Base Classification

Besides the spectral data, expert’s knowledge can also play an important role in improving accuracy of the classification of the satellite images. Human experience and knowledge about the topology, geology etc. of the study area can be embodied in the classification procedures to prepare accurate classified maps. Such classification is known as knowledge base classification. The most difficult part of knowledge base classifier is the creation of the knowledge base (Avelino, 1993).

Generally, knowledge base is created with the help of knowledge acquired by interacting with experts. By acquiring such knowledge we can build knowledge based system that could help us in improving the classification accuracy. But Building a knowledge base is very difficult task, because of the knowledge acquisition process (Argialas, 1990). It requires lot of time to acquire knowledge from domain experts. This acquired knowledge is used further for knowledge base image classification.

2.8.1 Automatic Extraction of Knowledge

Knowledge base creation is a time consuming job and requires expert’s knowledge. Due to this, generally knowledge base classification is not used so frequently in the remote sensing field. If we somehow make the knowledge base creation process automatic then we can make this process easier. The question arises that how we can extract knowledge (IF-THEN rules) automatically from the data? This thesis proposes a method to extract knowledge in the form of IF-THEN classification rules from the satellite and ancillary data. A data mining algorithm See5 based on the decision tree classifier is used to extract knowledge in the form of classification rules.
2.8.2 Knowledge Representation

Among the several kinds of knowledge representation available in the literature, there are at least three that are often used in machine learning (Avelino, 1993):

- **Logical conjunctions**: Logical conjunctions is the kind of knowledge representation used to express IF-THEN prediction rules, where the antecedent (IF part) consists of a conjunction of conditions and the rule consequent (THEN part) predicts a certain goal attribute value for a data instance that satisfies the rule antecedent.

- **Threshold concepts**: Unlike logical conjunctions, which are associated with an “all or none” matching between a concept description and a data instance, threshold concepts are associated with a partial matching between a concept description and a data instance. The basic idea is that a data instance satisfies the conditions of a concept description if it exceeds some threshold. This type of knowledge representation is typically used in neural networks.

- **Competitive concepts**: It is same as threshold concepts. The difference is that, rather using a threshold to represent the necessary degree of matching, an algorithm computes the degree of matching and selects the best competitor. This is generally used in instance based learning or nearest neighbour algorithms.

Generally logical conjunction frequently is used to create knowledge base because it is easy to understand and to interpret. This form of knowledge representation would be used in this research.

2.8.3 Knowledge from Decision Tree Classifier

Once a decision tree has been constructed, it is a simple matter to convert it into an equivalent set of rules. To generate rules, trace each path in the decision tree, from root node to leaf node, recording the test outcomes as antecedents and the leaf-node classification as the consequent. These “if-then” rules extracted from decision tree can serve as a knowledge base for further image classification. Converting a decision tree to rules has three main advantages:

- Converting to rules allows distinguishing among the different contexts in which a decision node is used. Since each distinct path through the decision tree node produces a distinct rule.
- Converting to rules removes the distinction between attribute tests that occur near the root of the tree and those that occur near the leaves.
- Converting to rules improves readability. Rules are often easier for people to understand.
3. Decision Tree

3.1. Introduction

Learning systems based on decision trees are the easiest to use and to understand among all other machine-learning methods. The automatic construction of decision trees begins with the studies developed in the social sciences by Morgan and Sonquist (1963) and Morgan and Messenger (1973). In statistics, the CART (Classification and Regression Trees) algorithm to generate decision trees proposed by Breiman et al. (1984) is one of the most important contributions. At around the same time decision tree induction was beginning to be used in the field of machine learning, notably by Quinlan (1979-1997). The main difference among the various algorithms used, is the criterion followed to carry out the partitions of training samples. The See5 algorithm (Quinlan, 1997) is the latest version of the ID3 and C4.5 algorithms developed by Quinlan. The criterion employed in See5 algorithm to carry out the partitions is based on some concepts from Information Theory. The main idea shared with this algorithm is to choose a variable that provides more information to realize the appropriate partition in each branch in order to classify the training set.

The advantages of decision tree classifier over traditional statistical classifier include its simplicity, ability to handle missing and noisy data, and non-parametric nature i.e., decision trees are not constrained by any lack of knowledge of the class distributions (Friedl et al., 1997). It can be trained quickly, takes less computational time and is not a “black box” like a neural network, in which user does not know how the prediction has been taken or how the outcome has been achieved (Gahegan et al., 1998).

Only few major studies have been done to explore decision trees for the classification of remotely sensed data. Lees and Ritman (1991) examined the use of decision trees for mapping vegetation species using LandSAT and other spatial data. Byungyong and Landgrebe (1991) used decision trees to classify AVIRIS data. Eklund et al. (1994) used a decision tree approach to assess the effect of incremental data layers on groundwater recharge estimates. Friedl and Brodley (1997) showed that decision tree algorithms consistently outperformed maximum likelihood techniques when classifying spectral data.

This chapter is organized explaining different aspects of the decision tree classifier. Section 3.2 explains the working of the decision tree classifier, followed by section 3.3 explaining splitting criteria based on information theory. Section 3.3.1 tells method to decide appropriate attribute and threshold for each node of the tree. Fitting a decision tree until all leaves contain data for a single class causes over-fitting. The result is often a very complex tree that over-fits the data. Over-fitting can be removed by pruning the tree, which is explained in section 3.4 and 3.4.1. Fuzzy decision tree and its principles are
explained in section 3.5. How decision tree is converted into rules is explained in section 3.6. Section 3.7 gives a small introduction about the boosting technique, which is used to increase the accuracy of the weak classifier. At last, section 3.8 and 3.9 outlines some advantages and disadvantages of decision trees respectively.

3.2. Decision Tree Classifier

A decision tree classifier is a hierarchical structure where at each level a test is applied to one or more attribute values that may have one of two outcomes as shown in figure 3.1. The outcome may be a leaf, which allocates a class, or a decision node, which specifies a further test on the attribute values and forms a branch or sub-tree of the tree. Classification is performed by moving down the tree until a leaf is reached. The method for constructing a decision tree as summarized by Quinlan (1993) is as follows:

- If there are \( k \) classes denoted \( \{C_1, C_2, \ldots, C_k\} \), and a training set, \( T \), then
- If \( T \) contains one or more objects which all belong to a single class \( C_j \), then the decision tree is a leaf identifying class \( C_j \).
- If \( T \) contains no objects, the decision tree is a leaf determined from information other than \( T \). If \( T \) contains objects that belong to a mixture of classes, then a test is chosen, based on a single attribute, that has one or more mutually exclusive outcomes \( \{O_1, O_2, \ldots, O_n\} \). \( T \) is partitioned into subsets \( T_1, T_2, \ldots, T_n \), where \( T_i \) contains all the objects in \( T \) that have outcome \( O_i \) of the chosen test. The same method is applied recursively to each subset of training objects to build the decision tree.

![Figure 3.1: Decision Tree](image)

Decision tree classifiers differ in the ways they partition the training sample into subsets and thus form sub-trees. That is, they differ in their criteria for evaluating splits into subsets. The See5 or C4.5 induction algorithm uses information theory (Shannon, 1949) to evaluate splits. CART uses Gini Index to split the training samples (Breiman, 1984) and some methods use Chi-Square measure.
Many studies have been done comparing See5 decision tree algorithm with other classifiers and found that See5 based on the Information theory is more accurate and gives reliable results (Blackmore et al., 2002; Eklund et al., 1998; German et al., 1999). The other advantage of See5 algorithm is that it can convert decision tree into corresponding classification rules. Rules are more comprehensive, easy to understand and easy to implement. Therefore the working of the decision tree explained in this chapter is in context to See5 algorithm.

3.3. Criteria for Evaluating Splits – Information Theory

As mentioned above that different decision tree algorithms have different criteria for splitting the training samples, See5 uses criteria, which is based on the information theory. It defines a statistical property called information gain that measures how well a given attribute separates the training samples according to their target classification. See5 uses this information gain measure to select among the candidate attributes at each step while growing the tree.

For any subset $S$ of $X$, where $X$ is the population, let $freq(j, S)$ be the number of objects in $S$, which belongs to class $j$. Then consider the ‘message’ that a randomly selected object belongs to class $j$. The ‘message’ has probability $freq(j, S) / |S|$, where $|S|$ is the total number of objects in subset $S$. The information conveyed by the message (in bits) is given by $-\log_2(freq(j, S) / |S|)$. Summing over the classes gives the expected information (in bits) from such a message:

$$Info(s) = -\log_2 \left( \frac{freq(C, S)}{|S|} \right)$$

(3.1)

When applied to a set of training objects, $Info(T)$ gives the average amount of information needed to identify the object of a class in $T$. This amount is also known as the entropy of the set $T$. Consider a similar measurement after $T$ has been partitioned in accordance with the $n$ outcomes of a test $X$. The expected information requirement can be found as a weighted sum over the subsets $\{T_i\}$:

$$Info_T(T) = \sum_{i=1}^n \frac{|T_i|}{|T|} Info(T_i)$$

(3.2)

The quantity

$$gain(X) = info(T) - info X(T)$$

measures the information that is gained by partitioning $T$ in accordance with the test $X$. The gain criterion selects a test to maximize this information gain. The gain criterion has one significant disadvantage in that it is biased towards tests with many outcomes. The gain ratio criterion (Quinlan, 1993) was developed to avoid this bias. The information generated by dividing $T$ into $n$ subsets is given by
The proportion of information generated by the split that is useful for classification is

$$gain\ ratio\ (X) = \frac{gain\ (X)}{split\ info\ (X)}.$$ \hfill (3.4)  

If the split is near trivial, split information will be small and this ratio will be unstable. Hence, the gain ratio criterion selects a test to maximize the gain ratio subject to the constraint that the information gain is large.

### 3.3.1. Tests on Continuous Attributes

The main crux of a decision tree lies in deciding appropriate attributes and the corresponding threshold for each node of the tree. The algorithm for finding appropriate thresholds for continuous attributes (Breiman et al., 1984, and Quinlan, 1993) is explained as follows:

In the induction of decision trees from continuous-valued data, a suitable threshold $T$, which discretizes the continuous attribute $A$ into two intervals $A_1 = (\min\ (A), T)$ and $A_2 = (T, \max\ (A))$, is determined based on the classification information gain generated by the corresponding discretization. Given a threshold, the test $A = T$ is assigned to the left branch of the decision node while $A > T$ is assigned to the right branch. Assuming we are to select an attribute for a node having a set $S$ of $N$ examples, these examples are sorted according to the values of the continuous attribute $A$; and an ordered sequence of distinct values $a_1, a_2, \ldots a_N$ is formed. Every pair of adjacent data points suggest a potential threshold $T = (a_i + a_{i+1}) / 2$ to create a cut point and generate a corresponding partition of $A$. Fayyad (1992) had proved that only the class boundary points could be the cut points to obtain the maximum information in classification, which implies if $a_i$ and $a_{i+1}$ belong to the same class, a cut point between them can not lead to a partition that has maximum information gain. Therefore, we can generate a smaller set of candidate cut points from the class boundary points. Let there be $k$ classes $c_1, c_2, \ldots c_k$, and let $p(c_j, S)$ be the proportion of examples in $S$ that belong to class $c_j$. The residual uncertainty in classification is expressed as the class entropy:

$$E(S) = -\sum_{j=1}^{k} p(c_j, S) \log( p(c_j, S))$$ \hfill (3.5)

After the set of example $S$ is partitioned into two subsets $S_1$ and $S_2$ by a threshold $T$, the class information entropy is expressed as the weighted average of their resulting class entropy:

$$E(A,T; S) = \frac{N_1}{N} E(S_1) + \frac{N_2}{N} E(S_2)$$ \hfill (3.6)
Where, \( N_1 = |S_1|, N_2 = |S_2|, \) and \( N = |S| \) and are the number of examples in \( S_1, S_2, \) and \( S, \) and \( p(c_j, S_1) \) and \( p(c_j, S_2) \) are the proportion of examples of class \( c_j \) in \( S_1 \) and \( S_2 \) respectively. The cut point for which \( E(A, T_A; S) \) is minimal among all the candidate cut points of attribute \( A \) is used; and the attribute \( A_j \), for which the \( E(A_j, T_{A_j}; S) \) is minimum, or the information gain \( E(S) - E(A_j, T_{A_j}; S) \) is maximum, will be then selected to generate two child nodes. In each child node, discretization and attribute selection are performed again based on the partitioned examples. The above process is repeated recursively until a stopping criterion is matched.

### 3.4. Pruning

Decision tree classifiers aim to refine the training sample \( T \) into subsets, which have only a single class. However, training samples may not be representative of the population they are intended to represent. In most cases, fitting a decision tree until all leaves contain data for a single class causes over-fitting. Generally there are two types of pruning methods:

- Stopping or Pre-pruning
- Post pruning

Pre-pruning tries to look at the best way of splitting the subset and assess the split in terms of information gain, gain ratio or some other criteria such as chi-square (\( \chi^2 \)). If this assessment falls below some threshold, the division is rejected. In this way the tree building and pruning process works simultaneously at each node of the tree.

But such stopping rules based on threshold would not give a best-pruned tree, because finding appropriate threshold is itself a difficult job (Breiman, 1984). Too high threshold can terminate division before the benefits of subsequent splits become evident, while too low results in little simplification.

On other hand, post pruning first grow the full over fitted tree and then prune it. Though growing and then pruning is a time consuming process, it gives more reliable results than pre-pruning. Post pruning calculates error at each node and then discards sub tree, which gives maximum error. This is also known as error-based pruning, as explained in the next section.
3.4.1. Error Based Pruning

If we know the error rate of a tree and its sub tree then we can simple discard those branches of the tree, which are responsible for incorporating maximum error in the tree. But question arises, how can we predict these error rates? Error estimates calculated on the basis of the training set do not reflect actual and real error values (Breiman, 1984).

Two techniques are defined to predict error rates. First technique is based on predicting the error rate of the tree and its sub-tree using the new set of cases that is distinct from the training set, known as test cases. Since these cases were not examined at the time the tree was constructed, the estimates obtained from them are clearly unbiased and reliable. Based on this first technique there are two approaches:

- **Cost-complexity pruning:** This approach predicts the error rate of a tree as a weighted sum of its complexity and its error on the training cases, with the separate cases used primarily to determine an appropriate weighting.

- **Reduced error pruning:** It assesses the error rates of the tree and its components directly on the set of separate test cases.

A drawback of this technique is the requirement of separate test cases, which is not possible if the data is scarce. To overcome this problem the second technique uses cross-validation approach. The training cases are divided into $C$ equal-sized blocks and, for each block; a tree is constructed from cases in all other blocks and tested on cases in the holdout blocks. Therefore we do not need separate test cases to predict error rate.

3.5. Fuzzy Decision Tree

The method described above for partitioning the training samples to build a crisp decision tree results in hard discretization. For some domains, this sudden change is quite appropriate for instance, there are hard-and-fast cut-offs for bands of the income tax table. For other applications, such as remote sensing images, it is more reasonable to expect classification decisions to change more slowly with changes in attribute values. Crisp decisions works well where class boundaries are non-overlapping and clearly defined. But this never happens in remote sensing satellite imagery where spectral classes do not have well defined boundaries, meaning that the boundaries between different phenomena are fuzzy, and/or there is heterogeneity within the class. To overcome this problem, we could use fuzzy decision trees, which are based on the fuzzy set theory. Using fuzzy theory we can convert our hard and crisp discrete partitions into soft and fuzzy divisions of the training set.

The induction of fuzzy decision trees follows the same steps as that of a classical decision tree with modified induction criteria (Janikow, 1998). In fuzzy approach, the continuous attributes are partitioned
into several fuzzy sets prior to the tree induction, heuristically based on expert experiences and the data characteristics.

A crisp set $A_c$ is expressed with a sharp characterization function $A_c(a): \Omega \rightarrow \{0,1\}$, $a \in \Omega$, alternatively a fuzzy set $A$ is characterized with a membership function $A(a): \Omega \rightarrow [0,1]$, $a \in \Omega$. The membership function $A(a)$ is called the possibility of $A$ to take a value $a \in \Omega$ (Zadeh, 1978).

Generally hard discretization defined by a threshold generates two crisp sets. On the other hand, a soft discretization is defined by a fuzzy set pair, which forms a fuzzy partition. The soft discretization is defined with three parameters/functions, one is the cross point $T$, the other two are the membership functions of the fuzzy set pair $A_1$ and $A_2$: $A_1(a) + A_2(a) = 1$ (Peng, 2000; Quinlan, 1997). The cross point $T$, i.e. the localization of soft discretization, is determined based on whether it can maximize the information gain in classification, and the membership functions of the fuzzy set pair are determined according to the characteristics of attribute data, such as the uncertainty of the associated attribute. Usually, wide overlapping is used for a high uncertain attribute; for example, we can use the average distance of data points as the overlapping width. The fuzzy class entropy in $S$ is defined as,

$$E_F(S) = \sum_{j=1}^{k} p(c_j, S) \log p(c_j, S)$$  \hspace{1cm} (3.9)$$

Where, $p(c_j, S) = \sum a_i \in c_j (A_1(a_i) + A_2(a_i))$ is the fuzzy proportion of examples in $S$. After soft discretization, the class information entropy is calculated with the probability of fuzzy partition, as:

$$E_F(A,T;S) = \frac{N_s^F}{N_s^F} E_F(S_1) + \frac{N_s^F}{N_s^F} E_F(S_2)$$  \hspace{1cm} (3.10)$$

$$E_F(S_1) = -\sum_{j=1}^{k} p(c_j, S_1) \log p(c_j, S_1)$$  \hspace{1cm} (3.11)$$

$$E_F(S_2) = -\sum_{j=1}^{k} p(c_j, S_2) \log p(c_j, S_2)$$  \hspace{1cm} (3.12)$$

$$p(c_j, S_k) = \frac{N_{s,c_j}^F}{N_s^F}$$  \hspace{1cm} (3.13)$$

Where

$$N_s^F = \sum_{i=1}^{[s]} (A_1(a_i) + A_2(a_i))$$  \hspace{1cm} (3.14)$$

$$N_s^F = \sum_{i=1}^{[s]} A_1(a_i)$$  \hspace{1cm} (3.15)$$

$$N_s^F = \sum_{i=1}^{[s]} A_2(a_i)$$  \hspace{1cm} (3.16)$$
A DECISION TREE APPROACH TO EXTRACT KNOWLEDGE FOR IMPROVING SATELLITE IMAGE CLASSIFICATION

\[ N_F^{S_{c_j}} = \sum_{a_i \in c_j} A_k(a_i) \] (3.17)

And, \((k = 1, 2)\). Similar to classical decision induction, the information gain, \(E_F(S) - E_F(A, T; S)\), is used as the criterion to generate the best discretization for the corresponding attribute. After choosing the best attribute among all other attributes, which is having minimum value for \(E_F\), the truth value is calculated for each branch as follows:

\[ \eta_1 = \frac{N_F^{S_{1j}}}{N_F^S}, \quad \eta_2 = \frac{N_F^{S_{2j}}}{N_F^S} \] (3.18)

If \(\eta_1 = a\) or \(\eta_2 = a\), then delete the corresponding branch. If \(\eta_1 > a\) or \(\eta_2 > a\), then calculate the truth level of each branch belonging to the \(j^{th}\) class:

\[ \mu_{1,j} = \frac{\sum_{a_i \in c_j} A_k(a_i)}{N_F^{S_{1j}}}, \quad \mu_{2,j} = \frac{\sum_{a_i \in c_j} A_k(a_i)}{N_F^{S_{2j}}} \] (3.19)

If \(\max_{j=1}^k(\mu_{1,j}) \geq \beta\) or \(\max_{j=1}^k(\mu_{2,j}) \geq \beta\) then the corresponding branch is terminated as a leaf, and this leaf is assigned as the class \(c_j\). Otherwise, the \(S\) is partitioned into \(S_1\) and \(S_2\):

\[ S_1 = \{ s \mid A_k(a_i) \geq \lambda, a_i \in S \} \] (3.20)
\[ S_2 = \{ s \mid A_k(a_i) \geq \lambda, a_i \in S \} \] (3.21)

And the above steps are repeated for each child nodes until the above criterion (Eq. (3.18) or Eq. (3.19)) are satisfied. Usually, \(a = 0.1 - 0.2, \beta = 0.8 - 0.9\) and \(\lambda = 0.5\) can be selected. Smaller \(a\) and bigger \(\beta\) will generate a bigger tree with higher accuracy in training. However, when the data is uncertain/noisy, too small \(a\) and too big \(\beta\) can cause the induced decision tree to become over-fitted.

3.6. Decision Trees to Rules

Though pruning makes a decision tree both simpler and more accurate but still decision trees can be cumbersome and complex. In fact, it does not matter whether it is a pruned or unpruned, decision trees are always difficult to understand. To simplify a decision tree we can convert them into rules, which are easier to understand and to implement. Every path from the root to a leaf is converted to an initial rule by regarding all the test conditions appearing in the path as the conjunctive rule antecedents while regarding the class label held by the leaf as the rule consequence (Quinlan, 1993). From figure 3.1, we can derive rules from the decision tree, such as:

- If Red < 45 and MIR < 155 and NIR > 80 → Urban
- If NIR < 54 → River
After that, each initial rule is generalized by removing antecedents that do not seem helpful for distinguishing a specific class from other classes, which is performed by a pessimistic estimate of the accuracy of the rule. In detail, the accuracy of the initial rule and that of its variant where an antecedent is removed are estimated. If the latter is not worse than the former then the initial rule is replaced by the variant of it.

It is worth noting that usually there are several rule antecedents that could be removed. In such cases, See5 Rule carries out a greedy elimination, that is, the removal of the antecedent that produces the lowest pessimistic error rate of the generalized rule is kept, and such kind of removal is repeatedly performed until the rule could not be generalized further.

After all the initial rules are generalized, they are grouped into rule sets corresponding to the classes respectively. All rule sets are polished with the help of the Minimum Description Length (MDL) Principle so that rules that do not contribute to the accuracy of a rule set are removed (Quinlan, 1993). Then, the rule sets are sorted according to the ascending order of their false positive error rates. Finally, a default rule is created for dealing with instances that are not covered by any of the generated rules. The default rule has no antecedent and its consequence is the class that contains the most training instances not covered by any rule.

3.7. Boosting

Boosting is the technique to improve the accuracy of the weak classifier. The idea is to generate several classifiers rather than just one. And each classifier tries to classify accurately the cases, which were misclassified by the previous classifier. But how can we generate several classifiers from a single dataset? As the first step, a single decision tree is constructed from the training data. This classifier will usually make mistakes on some cases in the training set. When the second classifier is constructed, more attention is paid to these cases in an attempt to get them right. As a consequence, the second classifier will generally be different from the first. It also will make errors on some cases, and these errors become the focus of attention during construction of the third classifier. This process continues for a pre-determined number of iterations or trials, but stops if the most recent classifiers are either extremely accurate or inaccurate.

This improvement is achieved by weighting the individual elements of the training data set. The weights are initially set to be equal. Comparison is done to identify those elements of the training set which have been misclassified. These misclassified training data elements are then given an increased weight, and the classifier is run again. Increased weight of the misclassified elements forces classifier to focus on these cases. See5 uses AdaBoost algorithm (Quinlan, 1997) to implement boosting algorithm.
3.8. **Strengths of Decision Tree Methods**

- **Ability to Generate Understandable Rules:** The ability of decision trees to generate rules that can be translated into comprehensible English or SQL is the greatest strength of this technique.

- **Ease of Calculation at Classification Time:** Although, as we have seen, a decision tree can take many forms, in practice, the algorithms used to produce decision trees generally yield trees with a low branching factor and simple tests at each node. Typical tests include numeric comparisons, set membership, and simple conjunctions. When implemented on a computer, these tests translate into simple Boolean and integer operations that are fast and computationally inexpensive.

- **Ability to Handle Both Continuous and Categorical Variables:** Decision-tree methods are equally adept at handling continuous and categorical variables. Categorical variables, which pose problems for neural networks and statistical techniques, come ready-made with their own splitting criteria: one branch for each category. Continuous variables are equally easy to split by picking a number somewhere in their range of values.

- **Ability to Clearly Indicate Best Fields:** Decision-tree building algorithms put the field that does the best job of splitting the training records at the root node of the tree.

3.9. **Weaknesses of Decision Tree Methods**

Decision trees are less appropriate for estimation tasks where the goal is to predict the value of a continuous variable such as remote sensing data. Decision trees are also problematic for time-series data unless a lot of effort is put into presenting the data in such a way that trends and sequential patterns are made visible.

- **Computationally Expensive to Train:** The process of growing a decision tree is computationally expensive. At each node, each candidate splitting field must be sorted before its best split can be found. In some algorithms, combinations of fields are used and a search must be made for optimal combining weights. Pruning algorithms can also be computationally expensive since many candidate sub-trees must be formed and compared.

- **Trouble with Non-Rectangular Regions:** Most decision-tree algorithms only examine a single field at a time. This leads to rectangular classification boxes that may not correspond well with the actual distribution of records in the decision space.
4. Study Area and Data

4.1. Study Area

The study area chosen for the research comes in Doon Valley, state of Uttaranchal, India. The geographic extent of the study area is 77° 45′ E to 77° 50′ E Longitude and 30° 25′ N to 30° 28′ N Latitude. The total area is 64 km². It is 20 kms away from the Dehradun city, capital of newly formed Himalayan state of Uttaranchal. The city has the Himalayas to its north, the Shivalik range to its south, the sacred Ganges to its east and the Yamuna to its west. Located in one of the most scenic and tranquil hill regions of India, Doon valley is blessed with a moderate climate. Rajpur, Chakrata, Haridwar and East canal Road are some of the major roads in the area. The valley is well connected to other Indian cities as New Delhi. The valley is surrounded by some Reserve forest and National parks. The clock tower, railway station, bus terminal are some of the important landmarks of the area.

The climate of the area is temperate. Average annual rainfall of the area is 2073.3 mm. Most of the annual rainfall is received within the period of June to September. There is a great variation in the temperature; it is hot during the summer and drops to freezing point during the winter. The average annual temperature is 20°C (Max. 27.8°C and Min. 13.3°C). The Dehradun district is nestled within the mountain ranges of the Himalayas and most of the area is comprised of hilly regions. The rice and wheat is the most important food crop. Other major crops grown in the area are sugarcane, maize, pulses, vegetables and some fruits crops. Potato is the major crop among the vegetable; litchi, lemon, mango etc among the fruit crops and cowpea, pigeon pea etc among the pulses. Tea plantation is also done in some areas. But the area is famous for its Basmati rice, litchi and the tea.

The Doon valley has some peculiarity of its own. Mixing of spectral signatures is a characteristic of this area; mainly urban and riverbed, tea gardens and forest area mix up with each other. The study area consists mainly of fairly leveled land and some hilly portion with altitude varying between 400 to 1450 m above MSL. The Sal forest (Shorea robusta), orchards, and the tea garden covers major portion of the study area and generally are misclassified due to similar spectral reflectance values. Tea gardens and orchards are planted somewhere very near to the each other, therefore these areas have high probability to mix up with each other. There is no pattern for agriculture, somewhere within the forest there is agriculture land. Orchards are somewhere planted very near to the forest area. The roofs of the houses are made up of sand and stone, which are brought from the riverbed therefore these two classes, give similar spectral signatures and are misclassified often.
A DECISION TREE APPROACH TO EXTRACT KNOWLEDGE FOR IMPROVING SATELLITE IMAGE CLASSIFICATION

Figure 4.1: Location of Study Area
4.2. Satellite Data

Space technology is improving day-by-day giving us improved satellite and sensors for acquisition of remote sensing data. Recently, on 17th August, 2004 India has launched its new satellite IRS-P6 (Resource Sat). IRS-P6 carries three cameras with vastly improved spatial resolutions, a high resolution Linear Imaging Self Scanner (LISS-IV) operating with 5.8 meter spatial resolution steerable up to ±26 deg across track to obtain stereoscopic imagery and achieve five day revisit capability; a medium resolution LISS-III operating in three spectral bands visible, VNIR and SWIR band with 23.5 meter spatial resolution; and an Advanced Wide Field Sensor (AWiFS) operating in three spectral bands in VNIR and one band in SWIR with 56 meter spatial resolution. Within LISS-3 the spatial resolution of SWIR band is 23.5 meter, which was previously used to come at 70.5 meters.

Therefore working on LISS-III image acquired on May 2004 with improved spatial resolution also gives good opportunity to explore the usability of the sensor for land use classification based on the Decision tree approach. Other image, IRS 1C PAN image at 5.8 m pixel resolution acquired on May 2001 was used to generate texture map.

Figure 4.2: LISS-III FCC Image of Study Area


4.3. **Reconnaissance Survey**

Initial reconnaissance was done to identify various land use classes. Total nine land use classes were selected as representing the range of class types within the study area. Classes identified were urban, tea garden, dense Sal forest (Shorea robusta), degraded forest, canal, fallow land, agriculture land, orchards, and riverine grass. These classes were defined by a set of 75 training sites ranging in size from approximately twenty to fifty pixels in the satellite image.

4.4. **Geometric Image Correction**

Two sets of images were used in the study. LISS-III images acquired on May 2004 and IRS 1C PAN acquired on May 2001. Both the satellite images were geo-referenced in Polyconic projection system and Everest spheroid using ERDAS IMAGINE image processing software. The PAN image was geo-referenced with the help of 25 well-distributed points using topographic map. Image to image registration was then done to geo-reference LISS-III image with PAN. Both the image registration was done with the RMS error of less than a pixel. Re-sampling was done using Nearest Neighbourhood algorithm because it most closely preserves the spectral integrity of the image pixels (Lillisand, 2000).

4.5. **Feature Selection**

Once the training data have been systematically collected for each class, a judgement has been made to determine the bands that are most effective to discriminate each class from all others. This process is commonly called feature selection (Jensen, 1996). It involves statistical and graphical analysis to determine the degree of between-class separability in the training data. Two methods were used to determine this, Transformed Divergence based on statistical methods and Feature space plot based on graphical method.

4.5.1. **Transformed Divergence**

Transformed Divergence (TD) is a covariance-weighted distance between class means. Larger TD values correspond to greater statistical distance between training patterns and higher probability of correct classification of classes. Divergence is computed using the mean and covariance matrices of the class statistics collected in the training phase of the supervised classification. Suppose there are two classes $c$ and $d$ then the degree of divergence or separability between $c$ and $d$, $\text{Diver}_{cd}$, is computed as

\[
\text{Diver}_{cd} = \left[ \frac{1}{2} \text{tr}(V_c^{-1} - V_d^{-1}) + \frac{1}{2} \text{tr}(V_c^{-1} + V_d^{-1})(M_c - M_d)(M_c - M_d)^T \right]
\]

Where $\text{tr}$. is the trace of the matrix, $V_c$ and $V_d$ are the covariance matrixes for the two classes; $M_c$ and $M_d$ are the mean vectors of two classes.
If there are more than two classes then the average divergence is computed, $Diver_{avg}$. This involves computing the average over all possible pairs of classes $c$ and $d$, while holding the subsets of bands $q$ constant. Then, another subset of bands $q$ is selected for the $m$ classes and analysed. The bands having the maximum average divergence are then used. It is calculated as

$$Diver_{avg} = \frac{\sum_{c=1}^{m} \sum_{d=1}^{m} Diver_{cd}}{C}$$ (4.2)

Unfortunately, outlying easily separable classes will weight average divergence upward in a misleading fashion to the extent that sub-optimal reduced feature subsets might be indicated as best (Jensen, 1996). Therefore Transformed Divergence is computed.

$$TDiver_{cd} = 2000[1 - \exp(-\frac{Diver_{cd}}{8})]$$ (4.3)

This gives an exponentially decreasing weight to increasing distances between the classes. It also scales the divergence values to lie from 0 to 2000. A transformed divergence value of 2000 suggests excellent between-class separation.

Transformed divergence was calculated for different bands combination and finally it was found that all the four bands of LISS-3 are not sufficient for discriminating all the classes in the study area. Cell report of transformed divergence using four bands is shown in Fig 4.3. Transformed divergence for other combination of bands is given in Annexure A.

**Figure 4.3: Transformed Divergence for all Four Bands of LISS-III**

From the above cell report we can see almost all the classes are well separable using all four bands except some classes like Urban and Riverbed, Teagarden and Orchards which, are still not completely separable and therefore showing values less than 2000.
4.5.2. Feature Space Plot

Feature space plot is a graphical representation to show distribution of all the pixels of various classes in the feature space. Two bands are taken at a time and the pixel values of these two bands are plotted to show how the various classes are distributed in the feature space. It is also best visual method to see the overlapping of the classes.

Feature space plot was plotted for different two-band combination and it was analyzed that classes like Urban and Riverbed, Orchards, Dense Sal Forest and Tea garden are overlapping each other in all the bands and hence are inseparable. Feature space plot for two bands 3 and 4 is shown in the Figure 4.4. Feature space plots for other combinations of bands are given in the Annexure B.

![Overlapping Classes in Feature Space](image)

Figure 4.4: Overlapping of Classes in Feature Space
4.6. Ancillary Layers

Ancillary data are any type of spatial and non-spatial information that may be of value in the image classification, including elevation, slope, aspect, geology, soils, hydrology and vegetation maps, etc. Generally ancillary layers are used when the spectral bands are not enough to classify the image accurately. To discriminate different classes sometimes we need extra information, which is provided by adding ancillary layers. Various studies have been done to improve the accuracy of the classification using different ancillary layers and found that such methods have mixed results (Jones et al., 1988; Franklin et al., 1992).

It was clear from the feature space plot and transformed divergence that the four spectral bands of LISS-III are not sufficient to classify the image accurately. Therefore three ancillary layers DEM, texture and slope were added as ancillary information after analyzing the topography of the area, because topographic information is often important in discriminating features that have similar spectral characteristics.

4.6.1. Digital Elevation Model

Digital elevation map was generated after digitizing contours at 20 m interval from topographic map. The reason to use elevation as an ancillary layer was the topography of the study area. It was found from the initial reconnaissance that elevation could help in differentiating tea garden and dense sal forest. Tea gardens and orchards were found generally at lower elevation than the dense sal forest, therefore it was decided to add elevation as an ancillary layer to discriminate these classes.

4.6.2. Slope Map

Mostly tea gardens are planted in the area where there is a slope and hence slope is one of the factors which can differentiate tea gardens from other classes such as orchards. Slope is expressed as the change in elevation over a certain distance. In this case, the certain distance is the size of the pixel. Slope is most often expressed as a percentage, but can also be calculated in degrees. It is also used as contextual information and helps in improving classification accuracy. Slope map was generated directly from the DEM using ERDAS software. A 3 × 3 pixel window was used to calculate the slope at each pixel. For a pixel at location X, Y, the elevations around it were used to calculate the slope. Slope was calculated as:

\[
slope(s) = \frac{\sqrt{(\Delta x)^2 + (\Delta y)^2}}{2}
\]

Where, \(\Delta x\) and \(\Delta y\) are the average elevation changes per unit of distance in \(x\) and \(y\) direction.
4.6.3. Texture

In pixel-based approach, each pixel is classified individually, without considering contextual information (i.e., characteristics or label assigned to neighbour pixels). Several studies have explored the potential for using these texture statistics derived from satellite imagery as input features for land cover classification (Haralick et al., 1973, Harris, 1980, Shih et al., 1983). Given a feature, we might gain additional class discrimination power by considering contextual variability, in addition to the feature's ability to organize class labels based solely on its spectral values, which in turn could help in improving the accuracy of the classification.

It can be seen from the feature space plot that riverbed and urban areas have similar spectral reflectance values and therefore overlap with each other. Texture can be used to discriminate these two classes because urban areas have high texture than riverbed.

Texture image was generated using IRS 1C PAN image at 5.8 m resolution using ERDAS image processing software. Second order Variance algorithm using 3 × 3 window was used to generate texture image (ERDAS, 2003).

\[
\text{Variance} = \frac{\sum (x_{i,j} - M)^2}{n - 1}
\]

Where,

- \( x_{i,j} \) = DN value at \((i, j)^{th}\) position
- \( n \) = number of pixels in the window
- \( M \) = Mean of the moving window

This chapter described the study area, satellite data used in the research and the preparation steps of the three ancillary layers DEM, slope and texture. Reason to use these particular layers was explained in the respective sections. The next chapter describes the methods and the procedures used to incorporated ancillary layers with the spectral bands to prepare the training dataset for the decision tree classifier.
5. Methods

5.1. Introduction

This thesis proposes that the accuracy of the land use classification using LISS-III data could be improved by using extracted knowledge using the Decision tree approach. The main crux of the research is to automatically extract knowledge from the data using decision tree classifier and to use this knowledge in land use classification. Generally knowledge base is created using expert’s knowledge and experience. This method of knowledge acquisition is a long and repeated process. On the other hand, knowledge base in form of classification rules generated using decision tree is comprehensive, easier to interpret and to use.

This chapter explains about the methodology followed during the research on classification based on the decision tree classifier and its various aspect related to the training size, fuzzy decision tree, boosting, and pruning. Fig 5.1 explains the conceptual flow chart of the full methodology. The training dataset for the decision tree is different as it is non-parametric classifier. The preparation of the training dataset is explained in section 5.4.1. Decision tree was generated using See5 decision tree software. The main advantage of See5 is that it can convert a decision tree into classification rules.

The decision tree generated from the ancillary and spectral data was then converted to classification rules to form a knowledge base. The knowledge base created from this process was then used in further classification of the image. Basically three types of classification were performed:

- The image was classified using the knowledge base consists of the classification rules extracted using the decision tree classifier.
- The prior probabilities were evaluated from the image classified from first method and then were used in maximum likelihood classification.
- The classification rules were applied using the post classification sorting method.

Finally the accuracy assessment was done for all the classification methods. Two methods were used to assess accuracy:

- Error matrix and
  - Overall Accuracy
  - User’s and Producer’s Accuracy
- KAPPA statistics.
5.2. **Conceptual Flow Chart**

![Flow Chart Diagram]

**Figure 5.1: Flow Chart**

- **Spatial Database**
  - Satellite Images
  - Topographic And Texture Data

- **Decision Tree Classifier**
  - Learning Dataset for Decision Tree
  - Decision Tree Algorithm

- **Knowledge Base Classification**
  - Knowledge Base Classifier
  - Extracted Knowledge

- **Accuracy Assessment**
  - Comparison and Evaluation of accuracy of classified images

- **Supervised Classification**
  - Maximum Likelihood
  - Training set for MLC
5.3. Maximum Likelihood Classification (MLC)

Maximum likelihood classification (MLC) has traditionally been used as a baseline for the classification of remotely sensed data. It is the most commonly used supervised method and is supposed to provide better results compared to other supervised methods (Foody et al., 1992). This section uses maximum likelihood classification to form a baseline against which the results achieved using other classifiers can be compared. First of all, the image was classified using MLC with four spectral bands of the LISS-III image. At last the accuracy assessment was carried out.

MLC can obtain minimum classification error under the assumption that the spectral data of each class is normally distributed. Generally, there is much spectral confusion between classes. That is, same class with different spectral and different class with the same spectral values. The MLC method itself cannot solve the problem of spectral confusion. And because of the requirement of statistical distribution, the auxiliary data cannot be incorporated in MLC. Maximum likelihood procedures do not provide an effective framework for combining disparate data sets because it becomes difficult to assume a known joint probability distribution between the spectral data and other spatial data sets (Evans, 1998).

Once you add ancillary layers into the classification process, it becomes difficult to get normally distributed training samples because the ancillary layers give bi- and multimodal distribution. Due to this fact MLC gives less accurate classified image. This was checked using the histograms of the training sets. The same training sets were used which were used earlier for MLC with four spectral bands. The histograms were plotted for all the training sets and it was found that most of the training sets give the multimodal distributions. Therefore MLC could not be performed after adding ancillary layers.

5.4. Decision Tree Classifier

Decision tree is a non-parametric classifier and therefore it doesn’t take into account the statistical distribution of the training data. Therefore it is easy to add ancillary layers with the spectral bands. The training data for decision tree is entirely different than what we use in MLC. It is a set of records written in a text editor. Each training pixel in itself is a record and the various spectral and ancillary layers serve as different attributes of the record. The following subsection explains the steps for preparation of the training dataset for the decision tree classifier.

5.4.1. Data Preparation for Decision Tree

Layer stacking operation was performed using ERDAS Layer Stack operation to incorporate ancillary layers with the spectral bands. Total seven layers were taken for layer stacking operation. Four layers RED, GREEN, NIR, and MIR from LISS-3 image, which were at 23.5 m resolution. One texture layer generated from PAN image at 5.8 m resolution and two layers of DEM and Slope. Resampling was done to bring all the seven layers to one common spatial resolution. Different pixel sizes were analyzed...
to get one common resolution at which minimum texture and a spectral value is lost. Finally 12 m pixel size was chosen and all the layers were re-sampled to this pixel size. Nearest Neighbourhood algorithm was used for re-sampling because it preserves the spectral values of the image pixels.

Output of layer stacking operation was a stacked image at 12 m pixel resolution composed of seven layers. This stacked image was used further for the creating training dataset for the decision tree classifier.

5.4.2. Training and Testing Dataset

Stratified random sampling methods were used to collect separate training and test data sets in the study area using ground reference data generated from field observations. The pixels collected by random sampling were divided into two subsets, one of which was used for training and the other for testing the classifiers, so as to remove any bias resulting from the use of the same set of pixels for both training and testing.

Three files were created:

- **Name file:** All the attributes (Spectral and Ancillary layers) and land use classes were defined in this file. It serves as a meta-data file for the decision tree training dataset. The snapshot of the attribute file is attached in Appendix C.
Training File: It is a notepad file, which provides information on the training cases from which classification rules, or knowledge is extracted. One by one training pixels were selected and their corresponding values for all the seven bands were written in this file. Land use classes for corresponding pixels were written at the end of the record. For example:

73, 45, 108, 104, 442.587, 0, 7.486, Orchard
99, 97, 91, 150, 725.386, 8, 4.306, Agriculture
70, 43, 117, 110, 454.678, 0, 2.306, Teagarden
65, 37, 112, 85, 595.067, 4, 3.792, Dense Sal Forest

Each training case above gives values at different spectral and ancillary layers of the stacked image and class associated with each pixel from where the values have been taken. Each such record about a pixel becomes a training data for the decision tree classifier.

Test file: The third file consists of test cases on which the classifier can be evaluated. The test and the training files are almost similar only cases inside are different.

Figure 5.3: Training Cases
Using the training set samples created above, the classifier was built in the form of a decision tree. Figure 5.4 shows the decision tree generated using See5 algorithm.

![Decision Tree](image)

**Figure 5.4: Decision Tree**

### 5.4.2.1. Training Sample Size and Accuracy

To investigate the affect of the size of training set on the accuracy of the classifier, different training samples of different sizes were prepared (for example 225, 500, 5000 samples etc). Each classifier generated from particular training sample dataset was then tested on unseen test cases to know how the accuracy of the classifier is related to the size of the training set. A training set of 225 pixels with 25 pixels from all the nine classes were taken first and then subsequently equal number of pixels from all the nine classes were increased. For every training set the classifier’s accuracy on training and test cases was checked and graphs were plotted to study the patterns.
5.4.3. Fuzzy Decision Tree

See5 decision tree classifier also provides option to generate fuzzy decision tree. To bring out the effect of fuzzy decision tree on the accuracy of the classifier, fuzzy option was tested on both training and test cases. Due to the time constraint it was not possible to test fuzzy option with all the training sets, therefore it was only tested with four training datasets of 2700, 3650, 4100, and 4500 training samples. It was observed that there was no significant change in the accuracy using fuzzy decision tree.

5.4.4. Pruning

Basically the pruning option is used when tree is complex and too large i.e., when tree is over-fitted. See5 also provides pruning option to avoid over-fitting. This option was used to see the effect of pruning on the accuracy of the classifier on the training and test cases. Pruning option was tested with the same four datasets used for testing fuzzy decision tree. There was no significant change observed in the accuracy of the decision tree classifier.

5.4.5. Boosting

Boosting was done with ten iterations. Quinlan (1993) proposed that ten iterations are more than sufficient to get good accuracy and any additional iteration would not affect significantly on the accuracy. Boosting was again tested with the same four datasets as used for fuzzy decision tree and pruning option. It was noted that there was slight increase in the accuracy of the classification of training and test data samples.

5.4.6. Classification Rules

Decision trees can sometimes be quite difficult to understand. An important feature of See5 is its ability to generate classifiers called rule sets that consist of unordered collections of (relatively) simple “if-then” rules, as shown in the Figure 5.5. Each rule consists of:

- A rule number, which serves only to identify the rule.
- Statistics \((n, lift \, x)\) or \((n/m, lift \, x)\) that summarizes the performance of the rule. Similarly to a leaf, \(n\) is the number of training cases covered by the rule and \(m\), if it appears, shows how many of them do not belong to the class predicted by the rule. The rule’s accuracy is estimated by the Laplace ratio \((n-m+1) / (n+2)\). The lift \(x\) is the result of dividing the rule’s estimated accuracy by the relative frequency of the predicted class in the training set.
- One or more condition that must all be satisfied if the rule is to be applicable.
- A class predicted by the rule.
- A value between 0 and 1 that indicates the confidence with which this prediction is made.
For each corresponding path from the root of a decision tree to a leaf, a separate classification rule is generated. There are chances that some redundant and undesirable rules are also extracted, therefore care should be taken to remove such rules which do not contribute in improving the accuracy of the classification. The following strategy was adopted to filter out the desired rules out all extracted rules:

- If only one rule is activated, which means the attribute values match the conditions of this rule, let the final class be the same as stated by this rule.
- If several rules are activated, let the final class be the same as stated by the rule with the maximum confidence.
- If several rules are activated and the confidence values are the same, then let the final class be the same as stated by the rule with the maximum coverage of learning samples.
- If no rule is activated, then let the final class be the same as stated by the default class.

Figure 5.5: Classification Rules
5.5. Knowledge Base Classification

Once the classification rules are generated using decision tree classifier, they can serve as a knowledge base. This knowledge base can be used for classification of the satellite images. Three approaches were followed to use the extracted classification rules for the classification. In first approach, classification rules were used directly using knowledge base classifier to classify the image. In second approach prior probability of the class distribution was used to classify the image. A new method was proposed to calculate the prior probability from already classified image using first approach. In third approach, post-classification sorting method was used to reclassify the pixels, which were misclassified during maximum likelihood classification.

5.5.1. Classification using Extracted Knowledge

The LISS-III image was classified with extracted classification rules using Knowledge base classifier in ERDAS. The extracted classification rules served as the knowledge base to classify the image. The accuracy assessment was then carried out at last to compare the validity of the extracted rules. For this the resulted classified image was compared by the image classified with MLC.

5.5.2. Prior Probabilities

This approach involves the use of prior probabilities in the maximum likelihood classification (Strahler, 1980). Prior probability is the probability of the class distribution in the study area. Generally it is evaluated using the historical summaries of the region; previous accurately classified maps or topographical maps. It can help in separating classes with similar spectral response (Mather, 1985).

A new method has been tried here to evaluate the class distribution in the study area using the user’s accuracy. First of all, the image was classified by the classification rules as described in above section 5.5.1. An error matrix was generated for the accuracy assessment of the classified map to calculate user’s accuracy. User’s accuracy is the number of pixels that really belongs to a particular class. Now suppose, we get user’s accuracy of forest class as 60%, which means that out of all pixels classified as forest, 60% pixels are actual forest. For example if 1000 pixels are classified as forest then out of these classified pixels 60% are actual forest i.e., 600 pixels are pure forest. Now if we divide 600 by total number of pixels in the image, we will get the frequency of forest area occurrence in the image. Let there are 5000 total no pixels in the image then, $\frac{600}{5000} = 0.12$ will give the probability of forest class being present in the study area. This probability can be used directly in the maximum likelihood classification algorithm as a weight assigned to the forest class.

Prior probabilities were evaluated using above described method and then were used in MLC. Again accuracy assessment was done to know the influence of the prior probability on the accuracy of the classified map.
5.5.3. Post-Classification Sorting

This method involves the application of very specific rules to already classified image. The rules in our case are automatically extracted using decision tree classifier. A strategy as described by Hutchinson (1982) was followed to apply the rules to the classified image for post classification. Specific rules were analysed to learn which attributes are contributing in classifying pixels to particular class. Once the attributes and their threshold values were known, the rules were implemented to reclassify the pixels, which were misclassified by MLC.

For example, if a pixel was initially misclassified as teagarden using MLC and if the elevation > 487 (this elevation value is discovered from the classification rule extracted from decision tree classifier), then the pixel was reclassified to dense sal forest. This eliminated confusion between several classes such as teagarden and dense sal forest, orchards and dense sal forest. At last accuracy assessment was done to check how far this method is able to solve spectral confusion between spectral classes.

5.6. Accuracy Assessment

This study adapted error matrix and Kappa statistic accuracy methods as described in section 2.7. Accuracy assessment was done using independent sample points. The choice of sampling technique depends upon several factors, including the size of the study area, the type and distribution of features being mapped, and the costs of acquiring verification data. Stratified random sampling was chosen to take reference samples to test the accuracy of the classifier. Total 181 samples points were taken from all the nine classes.

Sampling units should be at least the size of one pixel but, in order to take geometric distortions also into consideration, these sampling units should be more than one pixel. To take into account the positional error introduced by GPS, sample size was taken more than 23 meters.
6. Results and Discussions

6.1. Maximum Likelihood Classification

Maximum likelihood classification was done to provide a baseline for the comparison of the results of other classification methods. MLC could not perform well where classes spectrally overlap or where training samples are not normally distributed. There was too much spectral overlapping of the classes due to which the accuracy came very less. Due to this fact the classification done by MLC did not give satisfactory result.

As it can be seen in Figure 6.1 almost whole dense sal forest (Shorea robusta) is classified as teagarden, riverine grass and orchards. This is mainly because classes have similar spectral reflectance value. Teagardens are classified somewhere as orchards and riverine. We can also notice that riverbed is misclassified somewhere as urban areas. For other classes such as fallow and agriculture land, degraded forest, MLC gave good results. The producer’s and user’s accuracies of all the classes are given in Table 6-1. User’s accuracy is very poor in case of riverine grass and teagardens Overall accuracy of 68.62 % and Kappa accuracy of 0.6410 was achieved with MLC.

Figure 6.1: LISS-III Classified Image using Maximum Likelihood Classification
Table 6-1: Accuracy Assessment of Classified LISS-III image using Maximum Likelihood Classification

<table>
<thead>
<tr>
<th>Class</th>
<th>Reference</th>
<th>Classified</th>
<th>No. Correct</th>
<th>Producer’s Accuracy</th>
<th>User’s Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense Sal Forest</td>
<td>24</td>
<td>3</td>
<td>1</td>
<td>4.17</td>
<td>33.33</td>
</tr>
<tr>
<td>Degraded Forest</td>
<td>36</td>
<td>35</td>
<td>30</td>
<td>83.33</td>
<td>85.71</td>
</tr>
<tr>
<td>Urban</td>
<td>21</td>
<td>20</td>
<td>17</td>
<td>80.95</td>
<td>85</td>
</tr>
<tr>
<td>Riverine Grass</td>
<td>2</td>
<td>16</td>
<td>2</td>
<td>100</td>
<td>12.5</td>
</tr>
<tr>
<td>Agriculture land</td>
<td>40</td>
<td>34</td>
<td>32</td>
<td>80</td>
<td>94.12</td>
</tr>
<tr>
<td>Fallow Land</td>
<td>20</td>
<td>20</td>
<td>16</td>
<td>80</td>
<td>80</td>
</tr>
<tr>
<td>Orchards</td>
<td>18</td>
<td>22</td>
<td>16</td>
<td>88.89</td>
<td>72.73</td>
</tr>
<tr>
<td>Teagardens</td>
<td>13</td>
<td>22</td>
<td>4</td>
<td>30.77</td>
<td>18.18</td>
</tr>
<tr>
<td>Riverbed</td>
<td>14</td>
<td>16</td>
<td>11</td>
<td>78.57</td>
<td>68.75</td>
</tr>
<tr>
<td>Total</td>
<td>188</td>
<td>188</td>
<td>129</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Overall Classification Accuracy = **68.62%**
Overall Kappa Accuracy = **0.6410**

6.2. Training Set Statistics

To evaluate the effects of training set size on classification accuracy using a DT classifier, twelve subsets of training data for the study area were formed by randomly sampling the set of available training data. The numbers of pixels in each of these training data subsets are 225, 450, 900, 1350, 2250, 2700, 3150, 3600, 4050, 4500 and 5000 pixels, respectively, with an equal number of pixels per class for the twelve classes, giving 25, 50, 100, 150, 250, 300, 350, 400, 450, 500 and 550 pixels per class, respectively. A separate set of 2000 pixels was used for testing the classifier. The test set did not include any pixels from the training data sets.

Figure 6.2 show the relationship of the accuracy and the training set sizes using a decision tree classifier. Figure 6.3 show the accuracy of the different classifiers generated using different training set sizes on the test cases. These two graphs show two types of accuracies; one is learning accuracy and second is prediction accuracy. Learning accuracy is the accuracy of the classifier to learn from the training or known cases. Prediction accuracy is the accuracy of the classifier to predict the class of test or unseen cases.

These results indicate that the level of learning accuracy decreases with the increase of the size of the training set. A classifier knows about the classes of the training samples from which it is built. Therefore it is easy for the classifier to accurately classify the known cases. If we increase the number of
training cases then the classifier could not classify all the training cases accurately and its learning accuracy decreases.

For each classifier generated from each set of training set its accuracy was tested using the test cases. The result indicates that the prediction accuracy of the classifier increases with the increase of the training samples. The more training samples you take better the classifier learns and predicts on unseen cases i.e. more examples you give to a student for practice, more easily he can solve unseen questions in examination. These results also indicate that the prediction accuracy of a decision tree classifier improves, as the size of the training set is increased, but only up to a point only.

It is very clear in the Figure 6.3 that at training dataset with 5000 pixels there is a slight dip in the graph, which shows that further increase in the size of the training size would decrease the accuracy. We can also see a large dip at training dataset with 2250 pixels and then again a linear increase up to dataset with 4500 pixels.

Finally we can conclude that the accuracy of the classifier depends on the training set size also. More training samples we provide better the classifier learns and predicts on unseen or unknown cases but up to a certain level. Simply increasing the training size will not always gives better results because it also depends on the study area and its class variability. If the study area consist crisp classes with no spectral overlapping and less spectral variability then we can achieve better accuracy with less number of training samples. But if the study area is complex and contains classes with spectral overlapping then we need large number of training samples to get better accuracy.

![Training set Size Vs Classifier Accuracy on Training Cases](image)

**Figure 6.2: Training Set Sizes and Classifier Accuracy on Training Cases**
6.2.1. Pruning

Basically pruning is done when the classification tree becomes too complex and large. Though the decision tree generated from the training data using the See5 was not too complex and large but still pruning option was used to see the effect on the accuracy of the classifier. Four training sets were prepared to build the decision tree and then pruning option was applied to notice the changes on the learning and prediction accuracies. Table 6.2 shows the summary of the results. No significant change was noticed on both the accuracies of the classifier. Actually pruning is beneficial where tree too large but the tree generated from the training samples was not large and complex therefore no significant change could be seen in the accuracy of training and test cases after using the pruning option.

Table 6-2: Summary of the Results of Pruning Method Applied on Four Training Sets

<table>
<thead>
<tr>
<th>Training Set Size</th>
<th>Training cases Accuracy</th>
<th>Test Cases Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Without Pruning</td>
<td>With Pruning</td>
</tr>
<tr>
<td>2700</td>
<td>98.7</td>
<td>98.6</td>
</tr>
<tr>
<td>3650</td>
<td>98.2</td>
<td>98.2</td>
</tr>
<tr>
<td>4100</td>
<td>98.5</td>
<td>98.3</td>
</tr>
<tr>
<td>4500</td>
<td>98.2</td>
<td>98</td>
</tr>
</tbody>
</table>

* Test Cases = 2000 Pixels

Figure 6.3: Training Set Size and Classifier Accuracy on Test Cases
6.2.2. Boosting

Boosting method was used on four training datasets with ten iterations. The accuracies of all the training sets were increased subsequently. The prediction accuracy of the training set with 2700 pixels showed significant change with improved in the accuracy by 5% from 76.7 to 81.2%. Therefore it was suggested that boosting is good technique for improving the performance of the decision tree classifier. The summary of the result is given in Table 6-3.

<table>
<thead>
<tr>
<th>Training Set Size</th>
<th>Training cases Accuracy</th>
<th>Test Cases Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Without Boosting</td>
<td>With Boosting</td>
</tr>
<tr>
<td>2700</td>
<td>98.7</td>
<td>99.9</td>
</tr>
<tr>
<td>3650</td>
<td>98.2</td>
<td>99.9</td>
</tr>
<tr>
<td>4100</td>
<td>98.5</td>
<td>99.8</td>
</tr>
<tr>
<td>4500</td>
<td>98.2</td>
<td>99.8</td>
</tr>
</tbody>
</table>

6.2.3. Fuzzy Decision Tree

Fuzzy option provided by See5 was used to compare the learning and prediction accuracies of the crisp and the fuzzy decision tree. Table 6-4 shows the summary of the results of fuzzy decision tree on the four training sets used earlier. No change was seen in the accuracy of the decision tree classifier. In fact the accuracy remained same.

<table>
<thead>
<tr>
<th>Training Set Size</th>
<th>Training cases Accuracy</th>
<th>Test Cases Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Without Fuzzy</td>
<td>With Fuzzy</td>
</tr>
<tr>
<td>2700</td>
<td>98.7</td>
<td>98.7</td>
</tr>
<tr>
<td>3650</td>
<td>98.2</td>
<td>98.2</td>
</tr>
<tr>
<td>4100</td>
<td>98.5</td>
<td>98.5</td>
</tr>
<tr>
<td>4500</td>
<td>98.2</td>
<td>98.2</td>
</tr>
</tbody>
</table>

6.3. Classification using Extracted Knowledge

Decision tree generated using See5 decision tree classifier was converted to classification rules. These extracted rules were directly applied using Erdas knowledge base classifier. The validation of the classification rules was done using the method as described in the section 5.4.6. The classification rules filtered out and used in the classification of the LISS-III image are given in following section.
6.3.1. Classification Rules Generated by See5 Decision Tree Classifier

Total 14 rules were chosen and used to classify the image. The classification rules are as follows:

Rule 1: If Red > 90 and Red ≤ 120 and MIR ≤ 141 and Texture > 21 and Slope ≤ 1 and DEM ≤ 480 → Urban

Rule 2: If Green ≤ 77 and NIR > 91 and Slope ≤ 1 → Orchard

Rule 3: If Red > 60 and Red ≤ 90 and NIR > 57 and NIR ≤ 80 and MIR > 93 → Riverine Grass

Rule 4: If Red ≤ 82 and Green > 49 and NIR > 81 and NIR ≤ 96 → Degraded Forest

Rule 5: If Green < 77 and NIR > 96 and DEM > 487 → Dense Sal Forest

Rule 6: If Green ≤ 54 and NIR > 91 and DEM > 475 and Slope > 1 → Dense Sal Forest

Rule 7: If Green ≤ 49 and NIR > 78 and DEM > 475 → Dense Sal Forest

Rule 8: If Red > 111 and Green > 72 and MIR ≤ 162 and Texture < 20 and DEM > 450 → Riverbed

Rule 9: If Red > 118 and NIR ≤ 79 → Riverbed

Rule 10: If Red > 82 and MIR > 162 → Fallow Land

Rule 11: If Red > 82 and Red ≤ 109 and MIR > 141 → Agriculture Land

Rule 12: If Red ≤ 69 and Red ≤ 109 and MIR > 98 and MIR ≤ 107 and DEM > 448 and DEM ≤ 470 and Slope ≤ 1 → Teagarden

Rule 13: If Green ≤ 44 and MIR > 107 and MIR ≤ 107 and DEM < 470 and Slope < 1 → Teagarden

Rule 14: If Red ≤ 78 and Green ≤ 49 and NIR > 100 and NIR ≤ 118 and MIR > 107 and Dem > 449 and DEM < 470 and Slope < 1 → Teagarden
The above mentioned classification rules were then applied to the LISS-III image. Accuracy assessment was done to compare the result of the classified image with the MLC results.

By examining the confusion matrix, it can be seen that there is less misclassification than in case of MLC. Seasonal variation between data acquisition and ground data collection has a lot of influence on the results. Agriculture is mainly dependent on season. During May, generally crop is harvested and therefore agriculture gives very bright signatures as fallow or barren land but the same area is under agriculture during October-November, when ground truth and accuracy assessment was done. These issues lead to a lot of confusion during accuracy assessment.

Figure 6.4 shows the classified image using the classification rules. Major improvement that can be noticed is in the classification of the dense sal forest, which is now better classified than MLC. There is no more overlapping of teagardens, riverine grass and orchards in the classified image.

Another improvement is in the classification of the urban and the riverbed. There is no misclassification of the riverbed. The pixels are shown unclassified which were misclassified as urban in MLC classified image.

Significant improvements could be seen in the classification of the riverine grass which was previously overlapping with the dense sal forest, teagardens and orchards in the MLC classified image.

Figure 6.4: LISS-III Classified Image using Extracted Knowledge

The user’s and producer’s accuracy of dense sal forest is significantly increased. If we compare it with MLC then both accuracies are increased by 87% and 51% respectively. User’s and producer’s accuracy of teagardens also increased by 58.74% and 46.15% respectively due to slope and elevation layers used as ancillary information.
The summary of the error matrix is shown Table 6-5. The overall accuracy of the final result is 76.06% with Kappa accuracy of 0.7247. Comparing the result with the result produced by MLC, the overall accuracy is increased by 7.44%. The results show that the extracted knowledge is valid and helps in removing the spectral confusion to some extent.

**Table 6-5: Accuracy Assessment of Classified LISS-III Image using Extracted Knowledge**

<table>
<thead>
<tr>
<th>Class</th>
<th>Reference</th>
<th>Classified</th>
<th>No. Correct</th>
<th>Producer’s Accuracy</th>
<th>User’s Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense Sal Forest</td>
<td>24</td>
<td>26</td>
<td>22</td>
<td>91.66</td>
<td>84.61</td>
</tr>
<tr>
<td>Degraded Forest</td>
<td>36</td>
<td>30</td>
<td>26</td>
<td>72.22</td>
<td>86.66</td>
</tr>
<tr>
<td>Urban</td>
<td>21</td>
<td>20</td>
<td>15</td>
<td>71.42</td>
<td>75</td>
</tr>
<tr>
<td>Riverine Grass</td>
<td>2</td>
<td>6</td>
<td>2</td>
<td>100</td>
<td>33.33</td>
</tr>
<tr>
<td>Agriculture land</td>
<td>40</td>
<td>34</td>
<td>31</td>
<td>77.5</td>
<td>91.17</td>
</tr>
<tr>
<td>Fallow Land</td>
<td>20</td>
<td>21</td>
<td>10</td>
<td>50</td>
<td>47.61</td>
</tr>
<tr>
<td>Orchards</td>
<td>18</td>
<td>20</td>
<td>17</td>
<td>94.44</td>
<td>85</td>
</tr>
<tr>
<td>Teagardens</td>
<td>13</td>
<td>13</td>
<td>10</td>
<td>76.92</td>
<td>76.92</td>
</tr>
<tr>
<td>Riverbed</td>
<td>14</td>
<td>12</td>
<td>10</td>
<td>71.42</td>
<td>83.33</td>
</tr>
<tr>
<td>Undefined</td>
<td>-</td>
<td>6</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>188</strong></td>
<td><strong>188</strong></td>
<td><strong>143</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Overall Classification Accuracy = **76.06%**
Overall Kappa Accuracy = **0.7247**

**6.4. Maximum Likelihood Classification with Prior Probabilities**

Prior probabilities were estimated as described in the section 5.5.2 using the user’s accuracy of the classes derived from the decision tree classified image. Table 6-6 shows the various steps, which were followed to calculate the prior probabilities of all the classes using the user’s accuracy. The prior probabilities estimated from this method were used in maximum likelihood classifier. The priors were used as weights and were incorporated in the MLC using Erdas software.

No significant change in the accuracy was noted using the prior probabilities calculated by this method. There was only slight change in the accuracy of the teagardens and dense sal forest. The use of the priors in MLC increased overall accuracy by 1.06% with kappa of 0.6532. Table 6-7 shows the summary of the error matrix. The results show that the priors estimated using the proposed method is not an effective method and do not contribute much in improving the accuracy of the classification. Though, it can be noticed that the overall accuracy comes slightly better than MLC with this method also.
Table 6-6: Estimated Prior Probabilities

<table>
<thead>
<tr>
<th>Class</th>
<th>User’s Accuracy</th>
<th>Number of Classified Pixels</th>
<th>Actual Number of Pixels</th>
<th>Prior Probability = (Actual Number of pixels) / (Total number of Pixels*)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense Sal Forest</td>
<td>84.61%</td>
<td>53791</td>
<td>45512</td>
<td>0.125081</td>
</tr>
<tr>
<td>Degraded Forest</td>
<td>87.5%</td>
<td>57401</td>
<td>50225</td>
<td>0.138034</td>
</tr>
<tr>
<td>Urban</td>
<td>75%</td>
<td>13813</td>
<td>10359</td>
<td>0.02847</td>
</tr>
<tr>
<td>Riverine Grass</td>
<td>25%</td>
<td>5081</td>
<td>1270</td>
<td>0.00349</td>
</tr>
<tr>
<td>Agriculture land</td>
<td>91.17%</td>
<td>89187</td>
<td>81311</td>
<td>0.223468</td>
</tr>
<tr>
<td>Fallow Land</td>
<td>47.82%</td>
<td>67515</td>
<td>32285</td>
<td>0.088729</td>
</tr>
<tr>
<td>Orchards</td>
<td>85%</td>
<td>62011</td>
<td>52709</td>
<td>0.144861</td>
</tr>
<tr>
<td>Teagardens</td>
<td>76.92%</td>
<td>9690</td>
<td>7453</td>
<td>0.020483</td>
</tr>
<tr>
<td>Riverbed</td>
<td>83.33%</td>
<td>5039</td>
<td>4198</td>
<td>0.011537</td>
</tr>
</tbody>
</table>

* Total Number of Pixels = 363860

Figure 6.5: LISS-III Classified Image using MLC and Prior Probabilities
Table 6-7: Accuracy Assessment of Classified LISS-III Image using MLC and Prior Probability

<table>
<thead>
<tr>
<th>Class</th>
<th>Reference</th>
<th>Classified</th>
<th>No. Correct</th>
<th>Producer's Accuracy</th>
<th>User's Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense Sal Forest</td>
<td>24</td>
<td>5</td>
<td>3</td>
<td>12.5</td>
<td>60</td>
</tr>
<tr>
<td>Degraded Forest</td>
<td>36</td>
<td>33</td>
<td>27</td>
<td>75</td>
<td>81.81</td>
</tr>
<tr>
<td>Urban</td>
<td>21</td>
<td>24</td>
<td>17</td>
<td>80.95</td>
<td>70.83</td>
</tr>
<tr>
<td>Riverine Grass</td>
<td>2</td>
<td>16</td>
<td>2</td>
<td>100</td>
<td>12.5</td>
</tr>
<tr>
<td>Agriculture land</td>
<td>40</td>
<td>34</td>
<td>33</td>
<td>82.5</td>
<td>97.05</td>
</tr>
<tr>
<td>Fallow Land</td>
<td>20</td>
<td>20</td>
<td>15</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>Orchards</td>
<td>18</td>
<td>19</td>
<td>17</td>
<td>94.44</td>
<td>89.47</td>
</tr>
<tr>
<td>Teagardens</td>
<td>13</td>
<td>21</td>
<td>6</td>
<td>46.15</td>
<td>28.57</td>
</tr>
<tr>
<td>Riverbed</td>
<td>14</td>
<td>16</td>
<td>11</td>
<td>78.57</td>
<td>68.75</td>
</tr>
<tr>
<td>Total</td>
<td>188</td>
<td>188</td>
<td>131</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Overall Classification Accuracy = 69.68%
Overall Kappa Accuracy = 0.6532

6.5. Post Classification Sorting

The classification rules discovered from the See5 decision tree classifier were used to reclassify the pixels, which were misclassified during maximum likelihood classification using post classification sorting method as described in section 5.5.3 of this thesis. Classified image with post sorting method is shown in Figure 6.6.

It was noticed that the MLC misclassified dense sal forest, orchards and teagardens but at the same time other classes such as fallow land, agriculture land, and degraded forest were classified accurately. The accuracy of these classes using MLC was achieved to 80%, 94.12%, and 85.7%. On the other hand, decision tree classifier accurately classified dense sal forest, tea gardens, riverbed and urban but could not achieve same accuracy for the fallow land and agriculture land. It was decided to use results of both MLC and decision tree classifier to get final classified image. This was achieved by post classification sorting of the MLC classified image using the classification rules discovered by the decision tree classifier. The pixels misclassified by the MLC, were reclassified using the extracted knowledge. The texture, DEM and slope values, which were discovered by the extracted rules, were applied to MLC classified image to re-classify those pixels which were misclassified.

The overall accuracy of the final classified image obtained is 78.19% with kappa value of 0.7479. Comparing the accuracy with MLC and decision tree classifier, it increased by 9.99% and 2.13 respectively. The summary of the error matrix is given in Table 6-8.
Figure 6.6: LISS-III Classified Image using Post Classification Sorting

Table 6-8: Accuracy Assessment of Classified LISS-III Image using Post Classification Sorting

<table>
<thead>
<tr>
<th>Class</th>
<th>Reference</th>
<th>Classified</th>
<th>No. Correct</th>
<th>Producer’s Accuracy</th>
<th>User’s Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense Sal Forest</td>
<td>24</td>
<td>27</td>
<td>23</td>
<td>95.83</td>
<td>85.19</td>
</tr>
<tr>
<td>Degraded Forest</td>
<td>36</td>
<td>43</td>
<td>33</td>
<td>91.67</td>
<td>76.74</td>
</tr>
<tr>
<td>Urban</td>
<td>21</td>
<td>12</td>
<td>11</td>
<td>55</td>
<td>91.67</td>
</tr>
<tr>
<td>Riverine Grass</td>
<td>2</td>
<td>8</td>
<td>2</td>
<td>100</td>
<td>25</td>
</tr>
<tr>
<td>Agriculture land</td>
<td>40</td>
<td>34</td>
<td>32</td>
<td>80</td>
<td>94.12</td>
</tr>
<tr>
<td>Fallow Land</td>
<td>20</td>
<td>20</td>
<td>15</td>
<td>78.95</td>
<td>75</td>
</tr>
<tr>
<td>Orchards</td>
<td>18</td>
<td>15</td>
<td>15</td>
<td>78.95</td>
<td>100</td>
</tr>
<tr>
<td>Teagardens</td>
<td>13</td>
<td>5</td>
<td>5</td>
<td>35.71</td>
<td>100</td>
</tr>
<tr>
<td>Riverbed</td>
<td>14</td>
<td>16</td>
<td>11</td>
<td>78.57</td>
<td>68.75</td>
</tr>
<tr>
<td>Undefined</td>
<td>-</td>
<td>8</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>188</strong></td>
<td><strong>188</strong></td>
<td><strong>147</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Overall Classification Accuracy = **78.19%**
Overall Kappa Accuracy = **0.7479**
6.6. Summary

Figure 6.7 and 6.8 show the comparison of User’s and Producer’s accuracies between four different methods of classification followed in this research. In most of the cases, accuracy of MLC and MLC with priors is lower than the other two methods of decision tree classifier and post classification sorting.

Table 6-9 shows the summary of the results of the different classification methods used during the research. From this case study we can suggest that the post classification method gives better results than MLC and decision tree classifier. On the other hand, if we compare MLC with decision tree classifier then we can say that the decision tree classifier is better choice if the image consists of spectrally overlapping classes. Decision tree is also a good and easy method to incorporate ancillary layers to improve classification accuracy. It is also good choice if training samples are not normally distributed.

<table>
<thead>
<tr>
<th>Method</th>
<th>Overall Accuracy</th>
<th>KAPPA Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum Likelihood</td>
<td>68.62</td>
<td>0.6410</td>
</tr>
<tr>
<td>Decision Tree Classifier</td>
<td>76.06</td>
<td>0.7247</td>
</tr>
<tr>
<td>MLC with Prior Probabilities</td>
<td>69.68</td>
<td>0.6532</td>
</tr>
<tr>
<td>Post Classification Sorting</td>
<td>78.19</td>
<td>0.7479</td>
</tr>
</tbody>
</table>

Figure 6.7: Comparison of User’s Accuracy of Different Classes using Different Method
Comparision of Producer's Accuracy of Different Classes Using Different Methods

Figure 6.8: Comparison of Producer’s Accuracy of Different Classes using Different Method
7. Conclusions and Recommendation

The main objective of this study is to improve the accuracy of the classification of satellite images using extracted knowledge in the form of classification rules using a decision tree approach. Generally, accuracy can be improved by adding ancillary information or by incorporating expert’s knowledge into classification process. The main emphasis is given here to automatically extract knowledge in the form of classification rules using decision tree classifier. Different methods have been tried to use extracted classification rules in classification of a satellite image. The conclusions and the recommendations of the thesis are given in this section.

7.1. Conclusions

The following conclusions were derived on various steps followed to achieve the objective of the research.

7.1.1. Training Set

This study shows that the classification accuracy increases with the increase in the size of the training set but up to a certain limit, as was expected. Better accuracy can be achieved by increasing the number training samples during learning process. Number of training samples depends on the complexity of the study area. In our case more than 4000 pixels were required to build the classifier that could give an acceptable accuracy of approximately 88 percent on test cases. If study area is simple and it consists of well-defined crisp classes then less number of pixels can also give better accuracy.

The training pixels taken for learning purpose should be well distributed on the image because See5 decision tree algorithm works on “divide and conquer” strategy that is based on “greedy approach”. Due to this fact, it is somewhere biased if one takes pixels for learning process only from one area. It will try to make classification rules with only those pixels, which you have selected and in turn will generate classification rules that could not be applied on the whole image. Therefore care should be taken so that well distributed pixels are taken from the image for preparing training dataset.

The building of training set is a tedious job because one has to pick the training data pixel by pixel that takes too much time. In contrast, the learning process is fast and takes very little time to build a classifier from the training samples.
7.1.2. Decision Tree

This study also concludes that the knowledge extracted using decision tree approach gives better results than traditional statistical classifier such as maximum likelihood classifier. Due to its non-parametric nature it is easy to add ancillary layers to it. It also doesn’t require any statistical assumption about the distribution of the training sets such as normal distribution as required by MLC.

Decision trees are easy to train and they learn quickly from examples. The main advantage of the decision tree is that we can convert decision tree into classification rules. Classification rules in turn are easy to interpret and can serve as a knowledge base for further classification of satellite image. Knowledge base created using decision tree is a valid approach and can help in removing spectral confusion to some extent.

7.1.3. Accuracy

Finally the accuracy obtained by different methods allows us to conclude that accuracy of the satellite images classification can be improved by using knowledge in the form of classification rules extracted using decision tree classifier. In all the three cases accuracy comes out to be better than maximum likelihood classifier.

Though the overall accuracy using the prior probability is better than the MLC, it is concluded that using the prior probabilities estimated with user’s accuracy is not an effective method, because it only has a minor effect on the accuracy of the classification.

Direct application of the extracted classification rules gives much better results than those of MLC and of prior probability. The overall accuracy comes out to be 76.06 percent, which is acceptable for land use classification. It is suggested that the decision tree classifier gives better accuracy than MLC if the classes in the image are spectrally overlapping and producing spectral confusion.

Finally the research shows that the post classification sorting is the best method for application of the extracted classification rules into the pre-classified image. This method can use the best part of both MLC and decision tree classifier. MLC can classify some classes accurately and so can do decision tree classifier. With the help of post classification sorting we can reclassify the (classified) image to get improved results, which cannot be achieved if we use the MLC or decision tree classifier alone.
7.2. Recommendations

In future, dimensionality (number of attributes or bands) of the training set can be increased by using hyper spectral images or by adding more ancillary layers such as aspect, soil maps, geology maps etc. to see how dimensionality of the training set influences the accuracy of the satellite image classification using a decision tree classifier.

In this study classification rules were extracted using the decision tree classifier. Still, there are some classes that are misclassified in the final output. To improve the accuracy we can also include the association information of the pixels with each other in the form of association rules. There are number of algorithms for discovering association rules:

- Apriori and variations (AprioriTID, AprioriHybrid)
- DHP (Direct Hashing and Pruning) – use hash function for efficient generation of frequent itemsets and effective reduction on transaction database sizes
- FP-growth – use Frequent Pattern (FP) tree to avoid candidate generation
- Partition – divide the database into small partitions such that they can be processed independently and efficiently in memory to find out their frequent itemsets.
- DIC – Dynamic Itemset Counting

It is recommended to add classification and as well as association rules for the classification of satellite images. With the addition of association rules we can expect higher accuracy of image classification.
8. Bibliography


Annexure A

Transformed divergence using one band

Transformed divergence using two band

Transformed divergence using three band
Annexure B

Feature Plot between Bands 2 and 4

Feature Plot between Bands 1 and 3
Feature Plot between Bands 1 and 2

Feature Plot between Band 2 and 3
Feature Plot between Bands 3 and 4
Annexure C

<table>
<thead>
<tr>
<th>Attribute-Netpad</th>
<th>Attribute Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Visited satellite spectral and GIS data</td>
<td></td>
</tr>
<tr>
<td>2. Sources:</td>
<td></td>
</tr>
<tr>
<td>(a) Amit Bharti, Sumeet Saran and Dr. Valentina Golspain</td>
<td></td>
</tr>
<tr>
<td>(b) IGP-III INS PM 2003 and GIS data base of Deen Valley</td>
<td></td>
</tr>
<tr>
<td>(c) Date: 11 August 2004</td>
<td></td>
</tr>
<tr>
<td>3. Post Usmanabad</td>
<td></td>
</tr>
</tbody>
</table>

4. Relevant information paragraphs:
- There are 4 classes;
- There are 6 categorical attributes, some nominal and some ordered. The value "NA" means does not apply. The values for attributes are encoded numerically, with the first value encoded as "0," the second as "1," and so forth. An unknown value is encoded as "-1."

5. Number of instances: 5000

6. Number of attributes: 6 (all have been nominalized)

7. Attribute Information:
   - **riverbed, urban, agriculture land, fallow land, riverine grass, bamboo sal forest, degraded forest, orchards and Tea Garden.**
   - **pixel value in red band continuous**
   - **pixel value in green band continuous**
   - **pixel value in NIR band continuous**
   - **pixel value in HIR band continuous**
   - **Elevation**
   - **Slope**

   **Classes (3):**
   - **riverbed, urban, agriculture land, fallow land, riverine grass, bamboo sal forest, degraded forest, orchards and Tea Garden.**

   **Attributes**
   - **pixel value in red band continuous**
   - **pixel value in green band continuous**
   - **pixel value in NIR band continuous**
   - **pixel value in HIR band continuous**
   - **Elevation**
   - **Slope**

Attribute File