A
Dissertation
On

Swarm Intelligence in Remote Sensing

Submitted in partial fulfilment of the requirement
For the award of Degree of

MASTER OF ENGINEERING
(Computer Technology and Application)

Submitted by
Shelly Bansal
(University Roll No. 12213)

Under the Guidance of
Dr. DAYA GUPTA
(Head of Department)
Department of Computer Engineering

DEPARTMENT OF COMPUTER ENGINEERING
DELHI COLLEGE OF ENGINEERING
BAWANA ROAD, DELHI
(DELHI UNIVERSITY)
2007-2009

CERTIFICATE

DELHI COLLEGE OF ENGINEERING
(Govt. of National Capital Territory of Delhi)
BAWANA ROAD, DELHI – 110042
Date:___________

This is to certify that project entitled "**Swarm Intelligence in Remote Sensing**" has been completed by **Shelly Bansal** in partial fulfilment of the requirement of **Master in Engineering** in **Computer Technology & Application**.

This is a record of her work carried out by him under my supervision and support. This is a beneficial work in field of Remote Sensing using the upcoming Swarm Intelligence based technologies and resources.

(Dr. DAYA GUPTA)
HOD & PROJECT GUIDE
(Dept. of Computer Engineering)
DELHI COLLEGE OF ENGINEERING
BAWANA ROAD, DELHI – 110042

ACKNOWLEDGEMENT

It is a great pleasure to have the opportunity to extend my heartiest felt gratitude to everybody who helped me throughout the course of this project. It is distinct pleasure to express my deep sense of gratitude and indebtedness to my learned supervisors Dr. Daya Gupta, Mr. V.K.Panchal and Mr. Shashi Kuamr for their invaluable guidance, encouragement and patient reviews.

This project has been a dedicated effort towards development of a highly autonomous artificial intelligence, which primarily would not have been possible at the first place without the apt guidance of the Head of Computer Science Department, respected Dr. Daya Gupta. Her motivation and encouragement during the project made me go further than expected in the development process and the redaction of this current report.

I would also like to present my special thanks to Mr. V. K. Panchal from Defence Terrain Research Lab-DRDO who provided me the Invaluable Satellite Data for the experimental study.

I also express my great thanks to Mr. Shashi Kumar from Defence Terrain Research Lab-DRDO for his technical guidance at every step in the work.

I would also like to take this opportunity to present my sincere regards to my teachers Mrs. Rajni Jindal, Dr. S.K.Saxena and Mr. Manoj Sethi for their support and encouragement.

This dedication would not be complete without expressing my gratitude towards all staff members and my dear classmates for their interest and curiosity toward my project, where definitely to talk about an insect can sound curious.
Last but not the least, special thanks to the crowd who are active in the field of Swarm Intelligence.

(SHELLY BANSAL)
Master in Engineering
(Computer Technology & Application)
Dept. of Computer Technology
DELHI COLLEGE OF ENGINEERING
BAWANA ROAD, DELHI – 110042

ABSTRACT

In recent years remote sensing is used for the knowledge elicitation of earth's surface and atmosphere on a very global scale. Land cover mapping is a method for acquiring the geo-spatial information from satellite data. The land cover problem is solved by image classification of the satellite image. There are soft computing techniques like fuzzy sets and rough sets for remote sensing image classification. In this thesis we focus on the optimised approach of image classification of satellite multi spectral images. This thesis deals with the land cover mapping by using swarm computing techniques. Here we have classified the Satellite images by Swarm Intelligence approach. We are using improved Ant Miner algorithm i.e, cAnt Miner and the Hybrid PSO-ACO algorithm for the rule-set generation, used for image classification.

The Swarm Intelligence inspired techniques produce comparable results with the classical approaches Minimum distance classification, Maximum Likelihood method. There are Fuzzy set and Rough set based approaches, also used for Image classification. Swarm Intelligence based Image classification approach gives competitive results with the Fuzzy and Rough set approach. The motivation of this work is to explore the improved swarm computing algorithms for the satellite image classification.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

In remote sensing , satellite based sensors are burgeoning as a major facilitator of geo-spatial information providing different manifestations of the terrain. The satellite image is one of the main source for capturing the geo-spatial information [26]. Remote sensing with multi spectral satellite imagery is based on the concept that different features/objects constituting the land cover reflect electro-magnetic radiations over a range of wavelengths in its own characteristics way according to its chemical composition and physical state. A multi-spectral remote sensing system operates in a limited number of bands and measures radiations in series of discrete spectral bands. The spectral response is represented by the discrete digital number (DN) spectral signatures of an object may be used for identification much like a fingerprint [26].

There are two main type of classifying techniques: *Supervised and Unsupervised classification* [10]. When spectral classes, based on numerical information, are grouped first and are then matched by the analyst to information classes, then it is termed as unsupervised classification. Clustering algorithms are used to determine the statistical structures in the data for example K-Means approach. In supervised classification, the homogeneous samples of the different surface cover types of interest are used. To recognize spectrally similar areas, numerical information in all spectral bands for the pixels comprising these areas is used. For each pixel in the image a comparison is made with these signatures and defined in the class it most closely "resembles" e.g., ‘Minimum Distance to Means’ classifier. [27].

The ground truth data, may itself contain redundant / conflict information. Recently the soft computing mechanism of Rough set and Fuzzy set has emerged as n effective measure to resolve imprecise knowledge, analysis of conflicts and generating rules.

Simultaneously there is a new wide range of computational algorithms that have emerged from the behaviour of social insects. Social insects are usually characterized by their self organization and with the minimum communication or the absence of it.

Every social insect individually is self-autonomous. They can obtain information about environment and interact with the remote insects or environment indirectly, by stigmergy. All these features characterize Swarm Intelligence [8]. We can found these features in nature such as ant colonies, bird flocking, animal herding, fish schooling etc. The two most widely used swarm intelligence approaches are Ant Colony Optimisation (ACO) and Particle Swarm Optimisation (PSO) [9][11]. There are some algorithms of Swarm Intelligence based on ACO and PSO such as Ant miner, cAnt miner, Hybrid PSO/ACO and Modified ACO/PSO2. We can use these strategies in the Satellite Image Classification, as on implying these strategies the information retrieved from Image Classification is found to be more correct.

1.1 Motivation:

Land is the basic building block of human civilization. By nature, this precious gift cannot be expanded or destroyed. To make best use of land and its natural resource, we need good factual knowledge of the land and its feature. Accurate knowledge on land-use is very vital for planning and efficient land use. The Satellite image is one of the main sources which can capture the temporal nature of this knowledge.

Land cover mapping is a pattern classification problem. Extensive research has been carried out to use various computational techniques for remote sensing applications. Many researchers have studied the methods, evaluated the results on varied metrics, analysed the trend and elaborated the issues with satellite image processing [26]. Computational intelligence components have been applied individually to solve the problem of satellite classification. Neural networks have been used extensively for the image classification [26] and have been reported to perform better when compared with the statistical classifiers. Fuzzy set and Rough set theory have also been beneficial for the image classification.

In addition to these algorithms which have been inspired from the nature, there is a new range of computational algorithms that have emerged from the behaviour of social insects. Social insects are usually characterized by their self organization and the absence of central control. Still, complex group behaviour emerges from the interactions of individuals who exhibit simple behaviours by themselves. In social insects, every individual is self-autonomous. They can only obtain local information, and interact with their geographical neighbours. They can also change the local environment or mark in the local environment to interact with the remote individuals indirectly, namely stigmergy. All these features characterize swarm intelligence. Examples of systems like this can be found in nature, including ant colonies, bird flocking, animal herding, fish schooling. Generally speaking, swarm intelligence denotes that more complex intelligence emerges from the interactions of individuals who behave as a swarm [8,9]. The two most widely used swarm intelligence approaches are Ant Colony Optimisation (ACO) and Particle Swarm Optimisation (PSO). By the motivation of these approaches we used some improved algorithms of ACO and PSO for Satellite Image Classification. We used the improve cAnt Miner and the modified version of Hybrid ACO/PSO for the rule set generation , which are used for image classification.

1.2 Related work:

A lot of work is being done in the field of Swarm Intelligence. A number of applications are being presented by using these optimising algorithms. But the use of Swarm intelligence in the area of remote sensing is not being evolved very much.

However in the area of Image classification T. Piatrik and E. Izquierdo gave the idea of ACO Based Image Partitional Classifier. In this paper an Ant Colony Optimization (ACO) and its learning mechanism is integrated with the K-means approach to solve image classification problems.

We came across one such paper Urban Satellite Image Classification using Biologically Inspired Techniques [S.N.Omkar et all.]. This paper focuses on optimisation algorithms inspired by swarm intelligence for satellite image classification from high resolution satellite multi- spectral images. Amongst the multiple benefits and uses of remote sensing, one of the most important has been its use in solving the problem of land cover mapping. As the frontiers of space technology advance, the knowledge derived from the satellite data has also grown in sophistication. Image classification forms the core of the solution to the land cover mapping problem. No single classifier can prove to satisfactorily classify all the basic land cover classes of an urban region. In both supervised and unsupervised classification methods, the evolutionary algorithms are not exploited to their full potential. This work tackles the land map covering by Ant Colony Optimisation (ACO) and Particle Swarm Optimisation (PSO) which are arguably the most popular algorithms in this category. Some other researches have also been done in the context of the use of Particle Swarm Optimization for Multi Layer Perceptron neural networks, which is also used for Land Cover mapping classification.

1.3 Problem Statement:

With the extensive use of images classification for the Land Cover Mapping it is required that the approach used for the image classification should be more accurate and efficient. Image recognition techniques facilitate the classification of images into semantically-meaningful categories and then map the land by the labels that have been manually assigned to the categories. Automatic image classification is the task of classifying images into semantic categories with or without supervised training.

As the traditional techniques for the automatic Image Classification have their certain shortcomings like high resolution images needed for better information retrieval, and the accuracy limitations are also there. Also the recent Soft Computing approaches for Image Classification are not able to provide good results in case of ambiguity.

So for achieving the better accuracy even with the *low resolution satellite images* and better land cover mapping we are using the **Swarm Intelligence for Remote Sensing Image Classification.**

Our objective is to use the Swarm intelligence for the image classification for Land Cover Mapping. Here we are using the different algorithms of Swarm intelligence as cAnt miner and modified Hybrid ACO/PSO for the Satellite Image Classification , so that we can retrieve the more accurate information about any Land Area even with the *low resolution satellite images*. This approach is used because it provides the greater speed and accuracy in its computation.

1.4 Scope of Work:

Our approach to use the *Swarm Intelligence for the Satellite image Classification* is able to perform some very important targets successfully completed.

- By using this approach we are able to classify the satellite image according to different areas like water, urban area, vegetation and rocky region with different colours.
- This approach provides the increase in accuracy of classification as compared to the recent and tradition image classification with minimum computation and resources.
- This approach also shows the competitive results with the latest Fuzzy Set and Rough set approach of Image Classification.

1.5 Organization of Thesis:

The remainder part of this thesis is organised in the following sections:---

Section 2: It contains the Remote Sensing Fundamentals. It presents the basic introduction of Remote Sensing and information of the Satellite Images formation. The Digital number representation of these images and the Satellite Communication are also discussed here.

Section 3: This section describes the image classification techniques. The different supervised and unsupervised traditional approaches described here. Here the recent Soft Computing approaches for image classification are also given.

Section 4: In this Section the detail description of Swarm Intelligence is given. It describes the Ant Colony Optimization and the Particle Swarm Optimization.

Section 5: This section describes proposed algorithm for the Satellite image Classification which we have developed using the Swarm intelligence approach. In this section the accuracy measurement approach which we have followed is also described.

Section 6: This Section describes the case study which is based on the experiments performed at DTR Lab-DRDO for the classification of satellite image using our proposed algorithm. Also it establishes the results obtained by our algorithm provide more accuracy then the earlier approaches of image classification.

Section 7: This section gives the details of the publications from thesis and the conference details.

Section 8: In this section the conclusion of the thesis work and the future scope of the work are presented.

Section 9: this section gives the references details of the thesis.

Appendix A: Introduction to Matlab Software

Appendix B: Introduction to ERDAS software

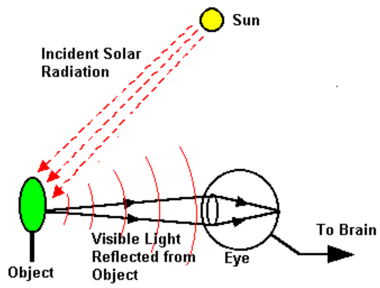
Appendix C: Introduction to Myra and open Source tool for Hybrid ACO/PSO

Chapter 2

Remote sensing fundamentals

2.1 Introduction to Remote Sensing:

Remote sensing is the science (and to some extent, art) of acquiring information about the Earth's surface without actually being in contact with it. *Kiefer et. all* gave a very useful information in their book on remote sensing, from which we are presenting some remote sensing basics. Acquiring information about earth remotely is done by sensing and recording reflected or emitted energy and processing, analyzing, and applying that information [26]. We perceive the surrounding world through our five senses. Some senses (touch and taste) require contact of our sensing organs with the objects. However, we acquire much information about our surrounding through the senses of sight and hearing, which do not require close contact between the sensing organs and the external objects. In another word, we are performing Remote Sensing all the time. Generally, remote sensing refers to the activities of recording/observing/perceiving (sensing) objects or events at far away (remote) places. In remote sensing, the sensors are not in direct contact with the objects or events being observed [27]. The information needs a physical carrier to travel from the objects/events to the sensors through an intervening medium. The electromagnetic radiation is normally used as an information carrier in remote sensing. The output of a remote sensing system is usually an image representing the scene being observed. A further step of image analysis and interpretation is required in order to extract useful information from the image. The human visual system is an example of a remote sensing system in this general sense [27].



In a more restricted sense, remote sensing usually refers to the technology of acquiring information about the earth's surface (land and ocean) and atmosphere using sensors onboard airborne (aircraft, balloons) or space borne (satellites, space shuttles) platforms.

The speed of EM energy is constant and frequency and wavelength are related:

299,893 km/sec = WAVELENGTH * FREQUENCY [27]

The EM spectrum is arbitrarily segmented into major divisions. There are no natural breaks in the EM spectrum. These separations are made for our convenience.

In much of remote sensing, the process involves an interaction between incident radiation and the targets of interest. The two basic processes involved are DATA ACQUISITIONS and DATA ANNALYSIS. The elements of data acquisition processes are energy sources, propagation of energy through the atmosphere, energy interactions with earth surface features, retransmission of energy through the atmosphere, airborne sensors, resulting in the generation of sensor data in pictorial or digital form. In short, we use sensors to record variations in the way earth surface features reflect and emit electromagnetic energy[27]. The data analysis process involves examining the data using various viewing and interpretation devices to analyze pictorial data or a computer to analyze digital data. Reference data about the resources being studied are used when and where available to assist in the data analysis. With the aid of the reference data, the analyst extracts information about the type, extent, location and condition of various resources over which the sensor data were collected. This information is then compiled in the form of hard copy maps, tables or as computer files that can be merged with other layers of information in geographical information system. Finally the information is presented to users who apply it to their decision making process. This is exemplified by the use of imaging systems where the following seven elements are involved [27].

1. **Energy Source or Illumination (A)** - the first requirement for remote sensing is to have an energy source, which illuminates or provides electromagnetic energy to the target of interest.

2. **Radiation and the Atmosphere (B)** - as the energy travels from its source to the target, it will come in contact with and interact with the atmosphere it passes through. This interaction may take place a second time as the energy travels from the target to the sensor.

3. **Interaction with the Target (C)** - once the energy makes its way to the target through the atmosphere, it interacts with the target depending on the properties of both the target and the radiation.

4. **Recording of Energy by the Sensor (D)** - after the energy has been scattered by, or emitted from the target, we require a sensor (remote - not in contact with the target) to collect and record the electromagnetic radiation.

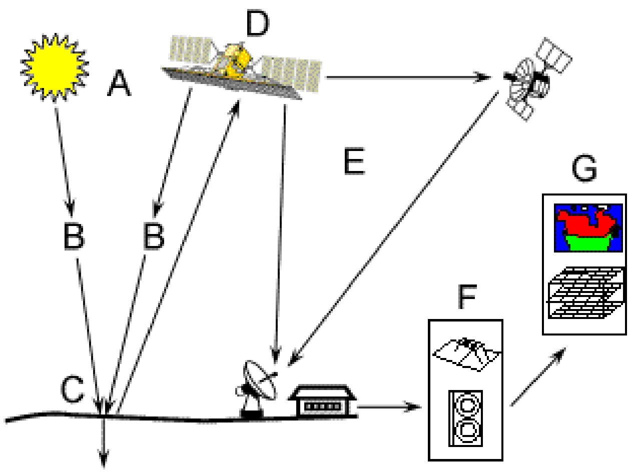


Fig.2.1 Imaging System

5. **Transmission, Reception, and Processing (E)** - the energy recorded by the sensor has to be transmitted, often in electronic form, to a receiving and processing station where the data are processed into an image (hardcopy and/or digital).

6. **Interpretation and Analysis (F)** - the processed image is interpreted, visually and/or digitally or electronically, to extract information about the target, which was illuminated.

7. **Application** - the final element of the remote sensing process is achieved when we apply the information we have been able to extract from the imagery

about the target in order to better understand it, reveal some new information, or assist in solving a particular problem.

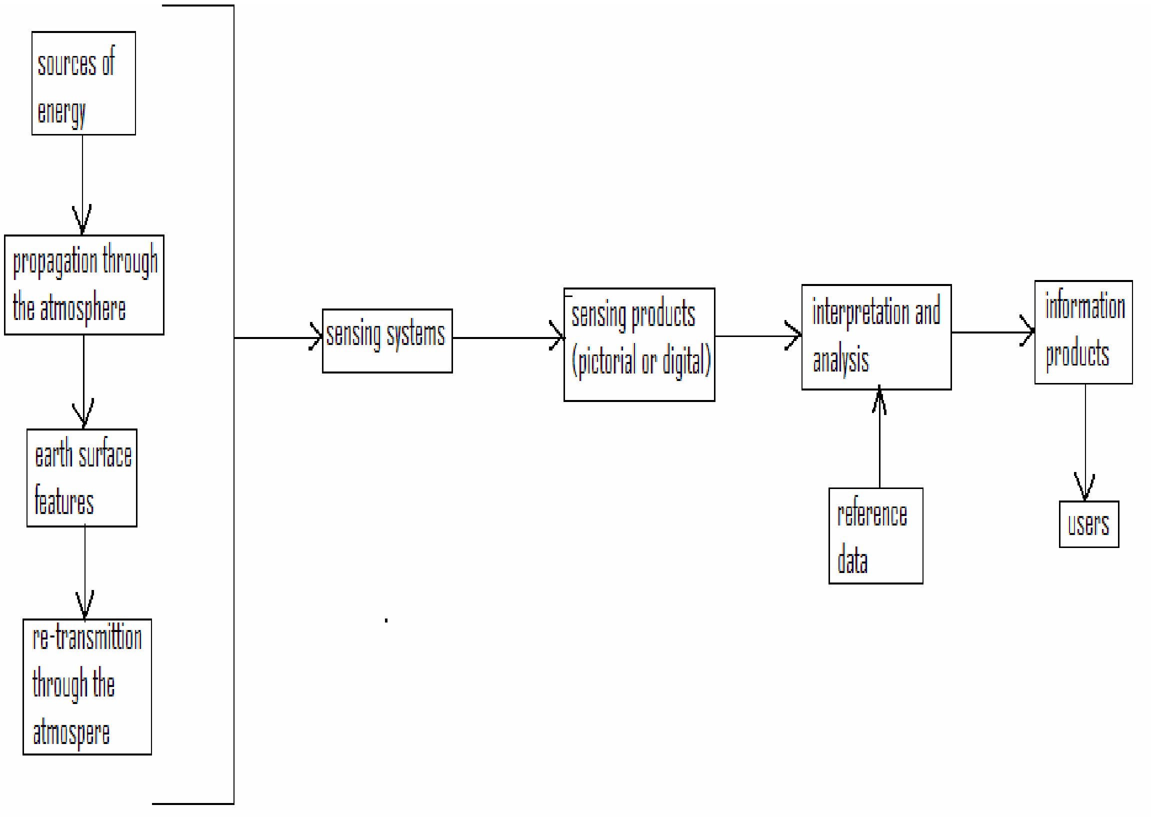


Fig.2.2 Satellite imaging System

2.2 Satellite Communication

Satellite Communications, no longer a marvel of human space activity, have evolved into an everyday, commonplace thing. Nearly all the television coverage travels by satellite today reaching directly to the home from space. Even in the age of wideband fiber optic cables and digital switching systems, satellites still serve the basic telecommunication needs of a majority of countries around the world. For example, domestic satellites have greatly improved the quality of service of public telephone system and brought nations more tightly together [27].

Kinds of Satellites

- Polar Satellites
- Geo Stationary satellites

Basic Characteristics of satellites

A communication satellite is a microwave repeater station that permits two or more users to deliver or exchange information in various forms. A system of three satellites in GEO each separated by 120 degrees of longitudes can receive and send signals over almost all the inhabited portions of the globe. The range from user to satellite is a minimum of 36000 km, which makes the design of the microwave link quite stringent in terms of providing adequate received signal power. Also, that distance introduces a propagation delay of one –quarter of a second for a single hop between a pair of users [27].

The GEO is the ideal case of the entire class of Geosynchronus orbits, which all have a 24 hour period of period of revolution but are typically inclined with respect to the equator and/or elliptical shape. Inclination is controlled by the use of an onboard propulsion system with enough fuel for corrections during the entire lifetime of the satellite [26].

Orbits that are below a mean altitude of about 36,000km have periods of revolution shorter that 24 hours and hence are termed non-GEO. The Iridium system uses multiple satellites to provide continuous coverage of a given region of the Earth. That is simply because the satellite appears to move past a point on the Earth. The Iridium mobile satellite system employs Low Earth Orbit (LEO). In which satellites are at an altitude of approximately 1,000 km and each passes a given user in only a few minutes. The advantage to using a non GEO satellite network is that the range to the user is shorter; hence, less radiated power is required and the propagation delay is reduced as well [27].

The key dimension of a Geo satellite is its ability to provide coverage of an entire hemisphere at one time. The time, difficult and expense incurred are extensive, but once established, but once established, a terrestrial infrastructure delivers very low unit service costs and can last a lifetime [72].

2.3 Digital Number

A positive integer which represents the relative reflectance or emittance of an object in a digital image is known as *Digital Number* for images. For 8 bit images, the DN or digital number lies in the range 0-255.

In a most generalized way, a digital image is an array of numbers depicting spatial distribution of a certain field parameters (such as reflectivity of EM radiation, emissivity, temperature or some geophysical or topographical elevation. Digital image consists of discrete picture elements called pixels. Associated with each pixel is a number represented as DN (Digital Number) that depicts the average radiance of relatively small area within a scene. The range of DN values being normally 0 to 255. The size of this area effects the reproduction of details within the scene. As the pixel size is reduced more scene detail is preserved in digital representation. Remote sensing images are recorded in digital forms and then processed by the computers to produce images for interpretation purposes. Images are available in two forms - photographic film form and digital form. Variations in the scene characteristics are represented as variations in brightness on photographic films. A particular part of scene reflecting more energy will appear bright while a different part of the same scene that reflecting less energy will appear black. Digital image consists of discrete picture elements called pixels. Associated with each pixel is a number represented as DN (Digital Number) [27] that depicts the average radiance of relatively small area within a scene. The size of this area effects the reproduction of details within the scene. As the pixel size is reduced more scene detail is preserved in digital representation.

2.4 Image Classification:

Image classification is formally defined as the process whereby a received pattern/signal is assigned to one of a prescribed number of classes. The overall objective of image classification procedures is to automatically categorize all pixels in an image into land cover classes or themes. Normally multi-spectral data are used to perform the classification and used to perform the classification and indeed, the spectral pattern present with in the data for each pixel is used as the numerical basis for categorize spectral pattern refers to the set of radiance measurements obtained in the various wavelength bands for each pixel.

Spectral pattern reorganization refers to the family of classification procedures that utilizes this pixel-by-pixel spectral information as the basis for automated land cover classification.

Techniques like Rough set, Rough-Fuzzy theory, Artificial neural network etc are used to classify images. Rough Set theory evolves the concept of data reduction, removing vagueness, discretization, lower and upper bound of the data set. Then rules are generated to classify the image. Fuzzy theory evolves the concept of membership function and membership grade to the objects which are vague in nature. A neural network performs image classification by first under going a training session, during which the network is repeatedly, presented a set of input patterns along with the category to which each pattern belongs. Later a new pattern is presented to the network that has not seen before, but which belongs to the same population of the patterns used to train the network.

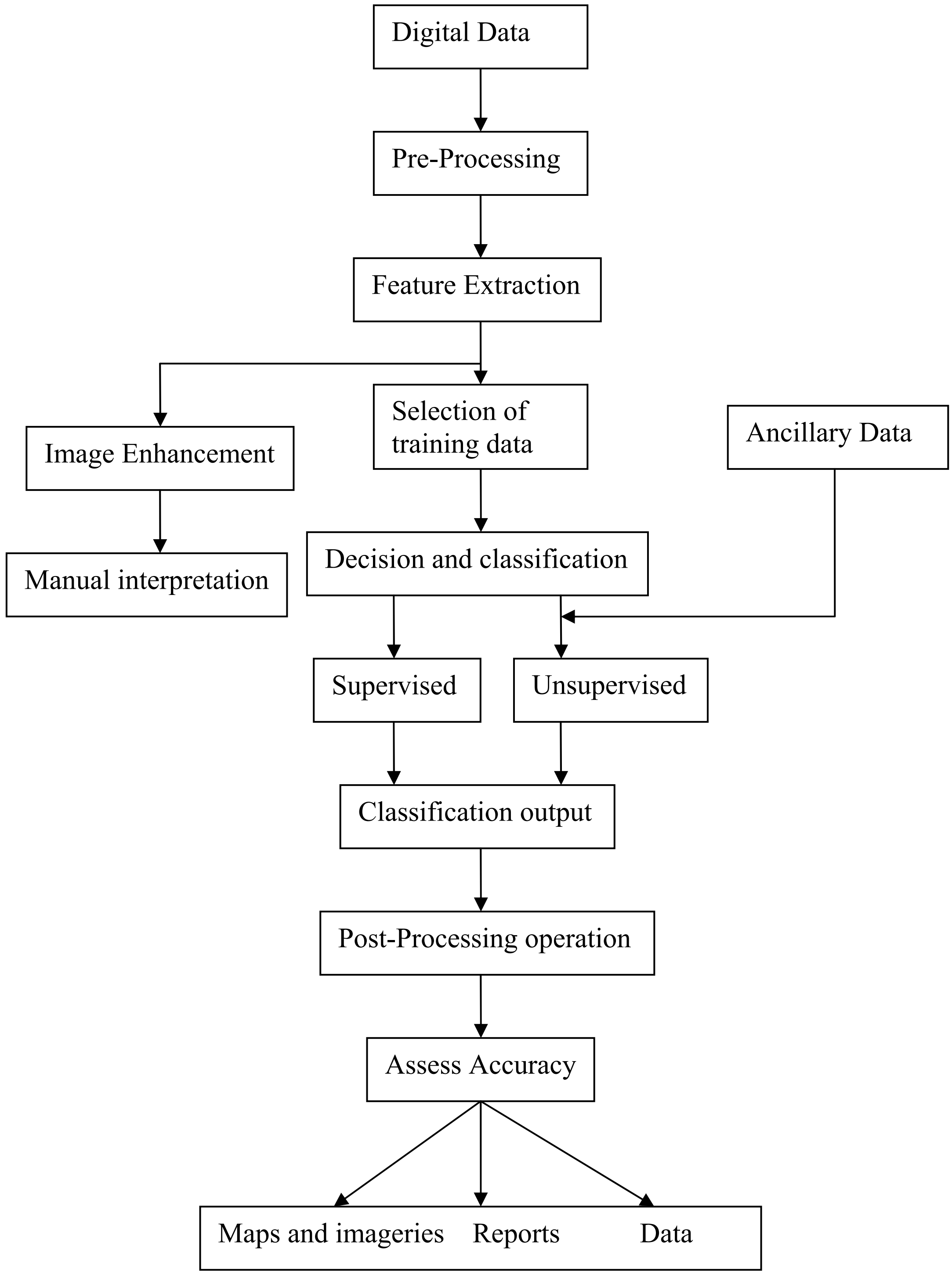


Fig 2.3 Image Classification Process

There are two type of classification.

- **Unsupervised**
- **Supervised**

2.4.1 Unsupervised classification

Unsupervised classifiers do not utilize training data as the basis for classification, rather this family of classifiers involves algorithm that examine the unknown pixel in an image and aggregate them into a number of classes based on the natural grouping or clusters present in the image values [10].

- The basic premise is that values within a given cover type should be closed together in the measurement space whereas data in different classes should be comparatively well separated.
- The classes that results from unsupervised classification are spectral classes because they are based solely on natural grouping in the image values, identity of the spectral classes will not be initially known.
- The analyst must compare the classified data within some form of referenced data to determine the identity and informational value of spectral classes
- Any individual pixel is compared to each discrete cluster to see which one it is closest to.
- A map of all pixels in the image, classified as to which cluster each pixel is most likely to belong, is produced (in black and white or more commonly in colours assigned to each cluster).
- This then must be interpreted by the user as to what the colour patterns may mean in terms of classes, etc. that are actually present in the real world scene; this requires some knowledge of the scene's feature/class/material content from general experience or personal familiarity with the area imaged.

2.4.2 Supervised classification

There are basically three steps involved in a typical supervised classification procedure [10].

- **TRAINING STAGE** – In the 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** – In classification stage, each pixel in the image data set is categorized into the land cover class it most closely resembles. If the pixel is insufficiently similar to any training data set, it is usually labeled unknown. The category labeled assigned to each pixel in this pixel is then recorded in corresponding cell of an interpreted.
- **OUTPUT STAGE** – After the entire data set has been categorized, the results are presented in the output stage. Being digital in character, the results may be used in number of ways.

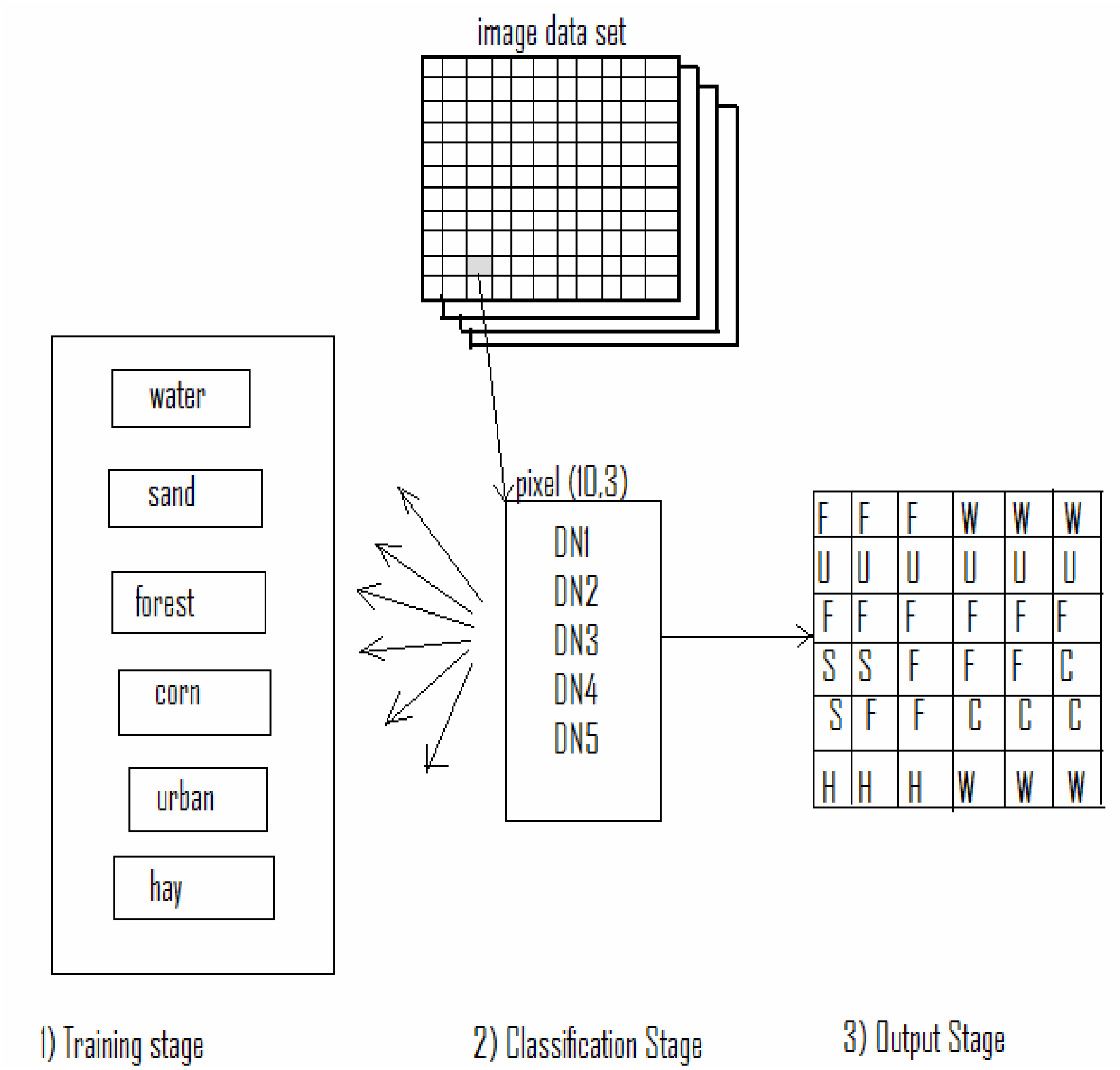


Fig 2.4 Supervised Classification

Chapter 3

IMAGE CLASSIFICATION TECHNIQUES

Land Cover Mapping is a pattern classification problem, which is solved by the classification of satellite image. For this purpose the satellite image is classified by using many different techniques.

3.1 Traditional Image Classification Techniques:

In remote sensing it is solved by using the traditional classical approaches like Parellelopiped Classification, Minimum Distance to Mean Classification, Maximum Likelihood Classification etc. Some classical supervised classification techniques are following which are in use for satellite image classification for many years [10].

3.1.1 Parallelopiped Classification

The Parallelopiped classifier is very simple supervised classifier that uses intervals of bounded regions of pixels' values to determine whether a pixel belongs to a class or not. The intervals' bounding points are obtained from the values of the pixels of samples for the class. Since this classifier is supervised there are two steps in its use: signature creation (training set) and classification[10].

3.1.2 Minimum Distance to Mean Classification

The minimum-distance-to-means strategy is mathematically simple and computationally efficient, but it has certain limitations. Most importantly, it is insensitive to different degrees of variance in the spectral response data. In Figure 7040, the pixel value plotted at point 2 would be assigned by the distance-to-means classifier to the "sand" category, in spite of the fact that the greater variability in the "urban" category suggests that "urban" would be a more appropriate class assignment. Because of such problems, this classifier is not widely used in applications where spectral classes are close to one an other in the measurement space and have high variance [10].

In this classification first of all mean or average or spectral in each band for each category is determined. These values comprise mean vector for each category. By considering the two channel pixel values as position coordinates, a pixel of unknown identity may be classified by computing the distance between the value of unknown pixel and each of the category means. In the following figure, an unknown pixel value has been plotted at point 1. The distance between this pixel value and each category mean value is illustrated by dashed lines. After computing the distance the unknown pixel is assigned to the closest class, in this case "corner"[10]. If the pixel is farther than an analyst defined distance from any category mean, it would be classified as unknown.

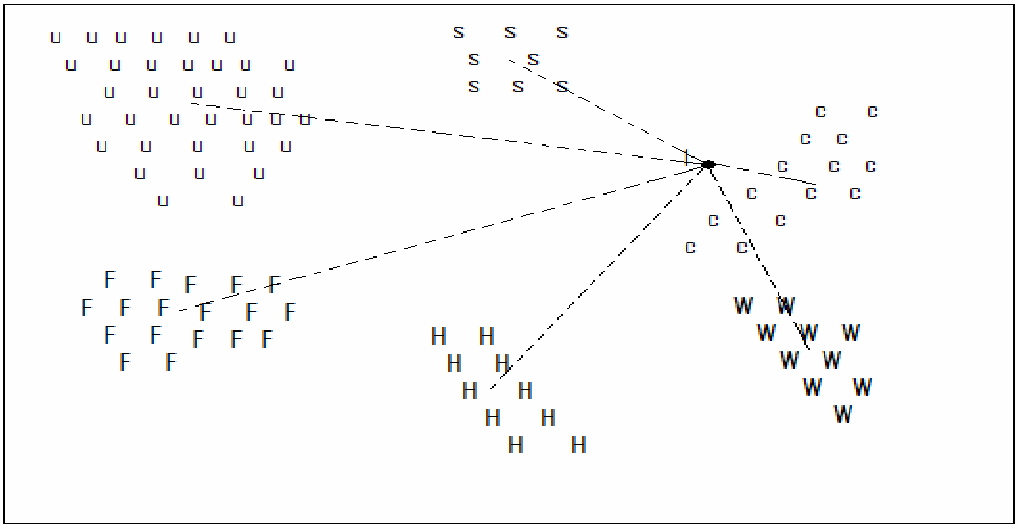


Fig 3.1 : Minimum Distance to Mean Classification

3.1.3 Gaussian Maximum Likelihood Classification

The maximum likelihood classifier quantitatively evaluates both the variance and covariance of the category spectral response patterns when classifying an unknown pixel. To do this, an assumption is made that the distribution of the cloud of points forming the category training data is Gaussian (normally distributed). This assumption of normality is generally reasonable for common spectral response distributions. In the Dimensional graph the vertical axis is associated with the probability of a pixel value being a member of one of the classes. The resulting bell-shaped surfaces are called probability density functions, and there is one such function for each spectral category. The probability density functions are used to classify an unidentified pixel by computing the probability of the pixel value belonging to each category.

3.2 Recent Soft Computing Techniques:

Soft computing differs from conventional (hard) computing in that, unlike hard computing, it is tolerant of imprecision, uncertainty and partial truth. In effect, the role model for soft computing is the human mind. The guiding principle of soft computing is: Exploit the tolerance for imprecision, uncertainty and partial truth to achieve tractability, robustness and low solution cost. The principal constituents of soft computing (SC) are fuzzy logic (FL), rough set theory (RS), neural network theory (NN) and probabilistic reasoning (PR), with the latter subsuming belief networks, genetic algorithms, chaos theory and parts of learning theory. Recently Rough Set theory and Fuzzy logic are begun to be used for the satellite image classification.

3.2.1 Rough Set Classification

Mathematician Z. Pawlak first proposed the theory of Rough Sets (RS) in 1980s. It has been used in many fields dealing with vagueness and inaccuracy, works well in artificial intelligence and analysis for decision-making. The rough set philosophy is founded on the assumptions that with every object of the universe of discourse we associate some information [16]. Any set of indiscernible (similar) objects is called elementary set, and form a basic granule of knowledge about the universe. Any union of some elementary sets is referred to as crisp (precise) set, otherwise it is rough (imprecise, vague). Each rough set has boundary line cases, i.e. objects which cannot be with certainty classified as members of the set or of its complement. Vague concepts, in contrast to precise concepts, cannot be characterized in terms of information about their elements. So we assume that any vague concept is replaced by a pair of precise concepts- called the lower and the upper approximation of the vague concept. The lower approximation consists of all objects which surely belong to the concept and the upper approximation constitutes the boundary region of the vague concept [16].

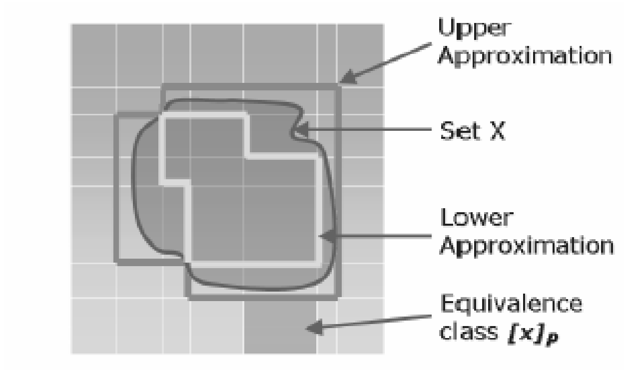


Fig 3.2. Concept of a Rough Set

The rough set philosophy is founded on the assumption that with every object of the universe of discourse we associate some information (data, knowledge). For example, if objects are patients suffering from a certain disease, symptoms of the disease form information about patients. Objects characterized by the same information are indiscernible (similar) in view of the available information about them. The indiscernibility relation generated in this way is the mathematical basis of rough set theory. Any set of all indiscernible (similar) objects is called an elementary set, and forms a basic granule (atom) of knowledge about the universe. Any union of some elementary sets is referred to as crisp (precise) set – otherwise the set is rough (imprecise, vague). Consequently each rough set has boundary-line cases, i.e., objects which cannot be with certainty classified neither as members of the set nor of its complement. Obviously crisp sets have no boundary-line elements at all. That means that boundary-line cases cannot be properly classified by employing the available knowledge. Thus, the assumption that objects can be seen only through the information available about them leads to the view that knowledge has granular structure. Due to the granularity of knowledge some objects of interest cannot be discerned and appear as the same (or similar). As a consequence vague concepts, in contrast to precise concepts, cannot be characterized in terms of information about their elements. The lower approximation consists of all objects that surely belong to the concept and the upper approximation contains all objects that possibly belong to the concept. Obviously, the difference between the upper and the lower approximation constitutes the boundary region of the vague concept. Approximations are two basic operations in rough set theory [16].

One of the main objectives of rough set data analysis is to reduce data size. Various notions such as *indiscernibility, rough set, reduct* are used to approximate inconsistent information and to exclude redundant data. They are used in analysis where there is complex variability in the spectral response pattern for individual cover types present. These conditions are quite common in such applications as vegetation mapping. Under these conditions, spectral variability within cover types normally comes about both from variations within cover types and from different site conditions.

3.2.2 Fuzzy Set Classification

Fuzzy classification attempts to handle the mixed pixel problem by employing the fuzzy set concept, in which a given entity (a pixel) may have partial membership in more than 1 category. One approach to fuzzy classification is fuzzy clustering [17]. This procedure is similar to "K-Means" unsupervised classification approach described earlier. The difference is that instead of having hard boundaries between classes in spectral measurement space, fuzzy regions are established. So instead of each unknown measurement vector being assigned solely to a single class, irrespective of how close that measurement may be a partition in measurement space, membership grade values are assigned that describes how close a pixel measurement is to the means of all classes.

Another approach to fuzzy classification is fuzzy supervised classification [17]. This approach is similar to application of maximum likelihood classification, the difference being that fuzzy mean vector and covariance matrices are developed from statically waited training data. Instead of delineating training areas that are purely homogenous, a combination of pure and mixed training sites may be used. Known mixtures of various features types define the fuzzy training class weights. A classified pixel is then assigned a membership grade with respect to its membership in each information class. For example, a vegetation classification might include a pixel with grades of 0.68 for a class "forest" , 0.29 for "street" and 0.03 for "grass".

Chapter 4

SWARM INTELLIGENCE

Swarm intelligence (SI), which is an artificial intelligence (AI) discipline, is concerned with the design of intelligent multi-agent systems by taking inspiration from the collective behavior of social insects such as ants, termites, bees, and wasps, as well as from other animal societies such as flocks of birds or schools of fish [1]. Colonies of social insects have fascinated researchers for many years, and the mechanisms that govern their behavior remained unknown for a long time. Even though the single members of these colonies are non-sophisticated individuals, they are able to achieve complex tasks in cooperation. Coordinated colony behavior emerges from relatively simple actions or interactions between the colonies' individual members.

Many aspects of the collective activities of social insects are self-organized and work without a central control. For example, leafcutter ants cut pieces from leaves, bring them back to their nest, and grow fungi used as food for their larvae[1]. Weaverant workers build chains with their bodies in order to cross gaps between two leaves. The edges of the two leaves are then pulled together, and successively connected by silk that is emitted by a mature larva held by a worker. Another example concerns the recruitment of other colony members for prey retrieval. Other examples include the capabilities of termites and wasps to build sophisticated nests, or the ability of bees and ants to orient themselves in their environment.

The term swarm intelligence was first used by Beni in the context of cellular robotic systems where simple agents organize themselves through nearest-neighbor interaction [4]. Meanwhile, the term swarm intelligence is used for a much broader research field [2]. Swarm intelligence methods have been very successful in the area of optimization, which is of great importance for industry and science. Optimization problems are of high importance both for the industrial world as well as for the scientific world. Examples of practical optimization problems include train scheduling, timetabling, shape optimization, telecommunication network design, and problems from computational biology. The research community has simplified many of these problems in order to obtain scientific test cases such as the well-known traveling salesman problem (TSP) [1]. The TSP models the situation of a traveling salesman who is required to pass through a number of cities. The goal of the traveling salesman is to traverse these cities (visiting each city exactly once) so that the total traveling distance is minimal. Another example is the problem of protein folding, which is one of the most challenging problems in computational biology, molecular biology, biochemistry, and physics. It consists of finding the functional shape or conformation of a protein in two- or three-dimensional space, for example, under simplified lattice models such as the hydrophobic-polar model [1]. The TSP and the protein folding problem under lattice models belong to an important class of optimization problems known as combinatorial optimization (CO).

In general, any optimization problem P can be described as a triple (S,Ω, f),

Where

1. S is the search space defined over a finite set of decision variables, Xi, i = 1, . . . , n. In the case where these variables have discrete domains we deal with discrete optimization (or combinatorial optimization), and in the case of continuous domains P is called a continuous optimization problem. Mixed variable problems also exist. Ω is a set of constraints among the variables;

2. f :S →IR+ is the objective function that assigns a positive cost value to each element (or solution) of S.

The goal is to find a solution s ∈S such that f(s) ≤ f(s_), ¥ s_ ∈ S (in case we want to minimize the objective function), or f(s) ≥ f(s_), ¥ s_∈S (in case the objective function must be maximized). In real-life problems the goal is often to optimize several objective functions at the same time. This form of optimization is labelled multi objective optimization.

Due to the practical importance of optimization problems, many algorithms to tackle them have been developed. In the context of combinatorial optimization (CO), these algorithms can be classified as either complete or approximate algorithms. Complete algorithms are guaranteed to find for every finite size instance of a CO problem an optimal solution in bounded time. In approximate methods such as SI-based algorithms we sacrifice the guarantee of finding optimal solutions for the sake of getting good solutions in a significantly reduced amount of time. Thus, the use of approximate methods has received more and more attention in the last 30 years. This was also the case in continuous optimization, due to other reasons: Approximate methods are usually easier to implement than classical gradient-based techniques. Moreover, generally they do not require gradient information. This is convenient for optimization problems where the objective function is only implicitly given (e.g., when objective function values are obtained by simulation), or where the objective function is not differentiable.

Two of the most notable swarm intelligence techniques for obtaining approximate solutions to optimization problems in a reasonable amount of computation time are ant colony optimization (ACO) and particle swarm optimization (PSO).

4.1 Ant Colony Optimization:

Ant as a single individual has a very limited effectiveness. But as a part of a well-organized colony, it becomes one powerful agent, working for the development of the colony. The ant lives for the colony and exists only as a part of it. Ant colonies are sometimes described as superorganism because it appears to operate as a unified entity [1]. Each ant is able to communicate, learn, cooperate, and all together they are capable of develop themselves and colonize a large area. They manage such great successes by increasing the number of individuals and being exceptionally well organized. The self organizing principles they are using allow a highly coordinated behavior of the colony, furthermore bring them to accomplish complex tasks, whose difficulty far exceed the individual capabilities of a single ant.

Pierre Paul Grassé, a French entomologist, was one of the first researchers who investigate the social behavior of insects. He discovered that these insects are capable to react to what he called “significant stimuli," signals that activate a genetically encoded reaction. He observed that the effects of these reactions can act as new significant stimuli for both the insect that produced them and for the other insects in the colony. Grassé used the term *stigmergy* to describe this particular type of indirect communication in which “the workers are stimulated by the performance they have achieved”[1].

“*Stigmergy is defined as a method of indirect communication in a self-organizing emergent system where its individual parts communicate with one another by modifying their local environment*”[1]. Ants communicate to one another by laying down pheromones along their trails, so where ants go within and around their ant colony is a stigmergic system.

The ACO framework, as we know it today, was first defined by Dorigo and colleagues in 1999. The recent book by Dorigo and St¨utzle gives a more comprehensive description [1]. The definition of the ACO framework covers most—if not all—existing ACO variants for discrete optimization problems. In the following, we give a general description of this framework. The basic way of working of an ACO algorithm is graphically shown in following figure.

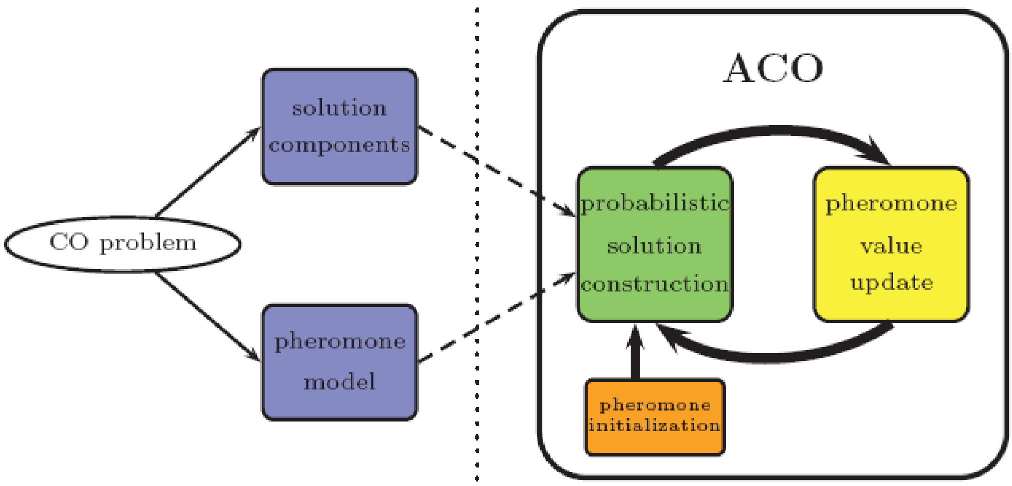


Fig.4.1 The ACO FrameWork

Given a CO (combinatorial) problem to be solved, One first has to derive a finite set C of solution components which are used to assemble solutions to the CO problem. Second, one has to define a set of pheromone values T. This set of values is commonly called the pheromone model, which is—seen from a technical point of view—a parameterized probabilistic model. The pheromone model is one of the central components of ACO. The pheromone values $\tau i \in T$ are usually associated with solution components. The pheromone model is used to probabilistically generate solutions to the problem under consideration by assembling them from the set of solution components. In general, the ACO approach attempts to solve an optimization problem by iterating the following two steps:

- candidate solutions are constructed using a pheromone model, that is, a parameterized probability distribution over the solution space;

- the candidate solutions are used to modify the pheromone values in a way that is deemed to bias future sampling towards high-quality solutions. The pheromone update aims to concentrate the search in regions of the search space containing high-quality solutions. It implicitly assumes that good solutions consist of good solution components.

4.1.1 Solution Construction

Artificial ants can be regarded as probabilistic constructive heuristics that assemble solutions as sequences of solution components. The finite set of solution components C = {c1, . . . , cn} is hereby derived from the discrete optimization problem under consideration. For example, in the case of AS applied to the TSP (travelling Salesman Problem) each edge of the TSP graph was considered a solution component. Each solution construction starts with an empty sequence s = <>. Then, the current sequence s is at each construction step extended by adding a feasible solution component from the set N(s)⊂ C \s. The specification of N(s) depends on the solution construction mechanism. The choice of a solution component from N(s) is at each construction step performed probabilistically with respect to the pheromone model. In most ACO algorithms the respective probabilities—also called the transition probabilities—are defined as follows:

$$\mathbf{p}(c_i \mid s) = \frac{[\tau_i]^{\alpha} \cdot [\eta(c_i)]^{\beta}}{\sum\limits_{c_j \in \mathcal{N}(s)} [\tau_j]^{\alpha} \cdot [\eta(c_j)]^{\beta}}, \quad \forall\, c_i \in \mathcal{N}(s),$$

Depending on the current sequence, assigns at each construction step a heuristic value η(cj) to each feasible solution component cj ϵ N(s). The values that are given by the weighting function are commonly called the heuristic information. Furthermore, the exponents α and β are positive parameters whose values determine the relation between pheromone information and heuristic information [1].

4.1.2 Pheromone Update

Different ACO variants mainly differ in the update of the pheromone values they apply. In the following, we outline a general pheromone update rule in order to provide the basic idea. This pheromone update rule consists of two parts. First, a pheromone evaporation, which uniformly decreases all the pheromone values, is performed. From a practical point of view, pheromone evaporation is needed to avoid a too-rapid convergence of the algorithm towards a suboptimal region. It implements a useful form of forgetting, favoring the exploration of new areas in the search space. Second, one or more solutions from the current and/or from earlier iterations are used to increase the values of pheromone trail parameters on solution components that are part of these solutions:

$$\tau_i \leftarrow (1-\rho)\cdot\tau_i + \rho \cdot \sum_{\{s \in \mathcal{S}_{upd} | c_i \in s\}} w_s \cdot F(s),$$

for i = 1, . . . , n. S_{upd} denotes the set of solutions that are used for the update. Furthermore, $\rho \in (0, 1]$ is a parameter called evaporation rate, and F : S _→ IR+ is a so-called quality function such that $f(s) < f(s') \Rightarrow F(s) \geq F(s')$, ¥ $s \neq s' \in S$. In other words, if the objective function value of a solution s is better than the objective function value of a solution s', the quality of solution s will be at least as high as the quality of solution s'. Above equation also allows an additional weighting of the quality function, i.e., $w_s \in$ IR+ denotes the weight of a solution s. Instantiations of this update rule are obtained by different specifications of S_{upd} and by different weight settings. In most cases, S_{upd} is composed of some of the solutions generated in the respective iteration (henceforth denoted by S_{iter}) and the best solution found since the start of the algorithm (henceforth denoted by s_{bs}). Solution s_{bs} is often called the best-so-far solution [1].

4.2 Particle Swarm Optimization:

Particle swarm optimization (PSO) is a population-based stochastic optimization technique modelled on the social behaviors observed in animals or insects, e.g., bird flocking, fish schooling, and animal herding [2]. It was originally proposed by James Kennedy and Russell Eberhart in 1995 [2]. Since its inception, PSO has gained increasing popularity among researchers and practitioners as a robust and efficient technique for solving difficult optimization problems. In PSO, individual particles of a swarm represent potential solutions, which move through the problem search space seeking an optimal, or good enough, solution. The particles broadcast their current positions to neighboring particles. The position of each particle is adjusted according to its velocity (i.e., rate of change) and the difference between its current position, respectively the best position found by its neighbors, and the best position it has found so far. As the model is iterated, the swarm focuses more and more on an area of the search space containing high-quality solutions. PSO has close ties to artificial life models. Early works by Reynolds on a flocking model Boids [2], and Heppner's studies on rules governing large numbers of birds flocking synchronously [2], indicated that the emergent group dynamics such as the bird flocking behavior are based on local interactions.

In PSO, the velocity of each particle is modified iteratively by its personal best position (i.e., the best position found by the particle so far), and the best position found by particles in its neighborhood. As a result, each particle searches around a region defined by its personal best position and the best position from its neighborhood. Henceforth we use vi to denote the velocity of the ith particle in the swarm, xi to denote its position, pi to denote the personal best position and pg the best position found by particles in its neighborhood. In the original PSO algorithm, vi and xi, for i = 1, . . . , n, are updated according to the following two equations :

$$\mathbf{vi \leftarrow vi + \phi 1 \odot (pi - xi) \odot \phi 2 \ (pg - xi), \quad Eq.(I)}$$

$$\mathbf{xi \leftarrow xi + vi, \quad Eq\ (II)}$$

where $\phi 1 = c1R1$ and $\phi 2 = c2R2$. R1 and R2 are two separate functions, each returning a vector comprising random values uniformly generated in the range [0,1]. c1 and c2 are acceleration coefficients. The symbol $\odot$ denotes pointwise vector multiplication. Equation (I) shows that the velocity term vi of a particle is determined by three parts, the "momentum", the "cognitive", and the "social" part. The "momentum" term vi represents the previous velocity term which is used to carry the particle in the direction it has travelled so far; the "cognitive" part, $\phi 1 \odot (pi - xi)$, represents the tendency of the particle to return to the best position it has visited so far; the "social" part, $\phi 2 \odot (pg - xi)$, represents the tendency of the particle to be attracted towards the positionof the best position found by the entire swarm. Position pg in the "social" part is the best position found by particles in the neighborhood of the ith particle. Different neighborhood topologies can be used to control information propagation between particles. Examples of neighborhood topologies include ring, star, and von Neumann. Constricted information propagation as a result of using small neighborhood topologies such as von Neumann has been shown to perform better on complex problems, whereas larger neighborhoods generally perform better on simpler problems . Generally speaking, a PSO implementation that chooses pg from within a restricted local neighborhood is referred to as lbest PSO, whereas choosing pg without any restriction (hence from the entire swarm) results in a gbest PSO [2]. Following algorithm summarizes a basic PSO approach and figure 4.2 shows each component of the velocity term vi in vector form, and the resulting position, xi (updated), for the ith particle. Note that the inertia coefficient w is used to scale the previous velocity term, normally to reduce the "momentum" of the particle.

4.2.1 PSO Algorithm

The PSO algorithm, assuming maximization is as follows--

- *Randomly generate an initial swarm*
- ***repeat***
- ***for** each particle i **do***
- ***if** f(**xi**) > f(**pi**) **then pi** ← **xi***
- ***pg** = max(**p**neighbours)*
- *Update velocity (see Eq. (I))*
- *Update position (see Eq. (II))*
- ***end for***
- ***until** termination criterion is met*

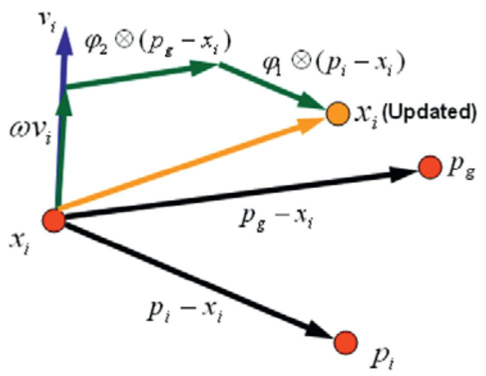


Fig.4.2 . Visualizing PSO components as vectors

Chapter 5

METHODOLOGY FOR IMAGE CLASSIFICATION USING SWARM INTELLIGENCE

5.1 Introduction

Our objective is to use the Swarm Intelligence approach for the Satellite image Classification. We have studied that image classification is the procedure of detecting different classes in an image. In the Land Cover mapping these classes are formed by different types of area like water, vegetation, urban area, barren and rocky region. We are using the new cAnt miner approach and the modified version of Hybrid PSO/ACO algorithms for rule generation in our algorithm for the satellite image classification.

Our *algorithm for the image classification using swarm intelligence* comes under the *supervised classification,* as it need the three step: the training set generation, rule set generation and the classification based on these rules.

In our algorithm for the image classification firstly the training set of the satellite image is generated by the experts. The expert knowledge is necessary for the training set generation as it is building block of the classification. On this training set which is in the Digital number presentation, we apply the cAnt miner Algorithm and the modified Hybrid PSO/ACO algorithm for the rule extraction. By applying any of these algorithms we got the rule set according to the different classes of land area. Then the region matching is done according to the different area and a label or colour code is provided to that region. In this way any satellite image is classified in different land area classes.

Our Algorithm for the Image Classification is centred around the cAnt miner Algorithm and the Hybrid PSO/ACO2 Algorithm. These algorithms are used for the rule set generation from training set. So before describing our proposed algorithm go through these algorithms and the original Ant miner and Hybrid PSO/ACO algorithms are as follows----

5.2 Ant Miner

To our knowledge, Parpinelli, Lopes and Freitas were the first to propose Ant Colony Optimization (ACO) for discovering classification rules, with the system Ant-Miner. They find out that an ant-based search is more flexible, robust and optimized than traditional approaches. Their method uses a heuristic value based on entropy measure. The goal of Ant-Miner is to extract classification rules from data [16]. The algorithm presented by Parepinelli is as follows—

- Training set = all possible training cases;
- While (No. of cases in Training set > maximum uncovered cases)
- count=0;
- Repeating count=count+1;
- Ant count incrementally constructs a classification rule;
- Prune this constructed rule;
- Update the pheromone trail followed by Ant count;
- till (count ≥ No of Ants) or (Ant count construct the same rule as the previous No of Rules Converging-1 Ants)
- Select the best rule among all constructed rules;
- Remove the cases correctly covered by the selected rule from the training set;
- End of while

Ant Miner follows a sequential covering approach to discover a list of classification rules from the given data set. It covers all or almost all the training cases. Each classification rule has the form IF <term1 AND term2 AND…> Then <CLASS>. Ant miner requires the discretization method as a pre-processing method and it is suitable only for the nominal attributes

5.3 cAnt Miner

To our knowledge, Parpinelli, Lopes and Freitas were the first to propose Ant Colony Optimization (ACO) for discovering classification rules, with the system Ant-Miner. They find out that an ant-based search is more flexible, robust and optimized than traditional approaches. Their method uses a heuristic value based on entropy measure. The goal of Ant-Miner is to extract classification rules from data (Parpinelli et al., 2002). Ant Miner follows a sequential covering approach to discover a list of classification rules from the given data set. It covers all or almost all the training cases. Each classification rule has the form IF <term1 AND term2 AND…> Then <CLASS>. Ant miner requires the discretization method as a pre-processing method and it is suitable only for the nominal attributes. Mostly real-world classification problems are described by nominal or discrete values and continuous attributes. There is a limitation with Ant-Miner that it is able to cope only with nominal attributes in its rule construction process [16]. So that discretization of continuous attributes is done in a preprocessing step. Fernando, Freitas, and Johnson proposed an extension to Ant-Miner, named cAnt Miner, which was able to cope with the continues values as well.

5.4 Hybrid PSO/ACO

The Ant Miner and cAnt Miner has already been a significant approach for data mining, but an extremely large amount of computation is required with the problem of unusually large amount of attributes and classes. This is because, a computationally expensive rule pruning procedure is required at every inner iteration of the standard ACO algorithm. Also for the algorithm to work with continuous attributes they have to be previously discretized, which can decrease classification accuracy and increase computational time at this pre-processing step.

The "standard" binary/discrete PSO algorithm [2] does not deal with categorical values in a natural fashion when compared to ACO. In particular, the standard PSO for coping with binary attributes represents a particle by a bit string, where each binary value such as true or false is encoded as 1 or 0. Sousa et al. extended the standard binary PSO to cope with multi-valued categorical attributes [11], developing a Discrete PSO (DPSO) algorithm for discovering classification rules. In particular in DPSO each value for a categorical attribute is assigned an index number.

5.5 Modified Hybrid PSO/ACO

Nicholas and Freitas proposed several modifications to the original PSO/ACO algorithm [12]. It involves the changes in the splitting of the rule discovery process into two separate phases. In the first phase a rule is discovered using nominal attributes only. In the second phase the rule is potentially extended with continuous attributes. This further increases the ability of the PSO/ACO algorithm in treating nominal and continuous attributes in different ways. Both the original PSO/ACO algorithm and the new modified version PSO/ACO2 uses a sequential covering approach to discover one classification-rule-at-a-time.

Sequential covering approach used by the hybrid PSO/ACO2 algorithm given by Nicholas and Freitas

- Initially RuleSet is empty(Φ)
- For Each class of cases Trs = {All training cases}
- While (Number of uncovered training cases of class A > Maximum uncovered cases per class)
- Run the PSO/ACO algorithm for finding best nominal rule
- Run the standard PSO algorithm to add continuous terms to Rule, and return the best discovered rule BestRule
- Prune the discovered BestRule
- RuleSet = RuleSet ∪ BestRule
- Trs = Trs – {training cases correctly covered by discovered rule}
- End of while loop
- End of for lop
- Order these rules in RuleSet by descending Quality

It is necessary to estimate the quality of every candidate rule (decoded particle). A measure must be used in the training phase in an attempt to estimate how well a rule will perform in the testing phase In PSO/ACO [12] the Quality measure used was Sensitivity * Specificity (Equation 1) [4]. Where TP, FN, FP and TN are, respectively, the number of true positives, false negatives, false positives and true negatives associated with the rule [2][16].

$$\text{Sensitivity} \times \text{Specificity} = TP / (TP + FN) \times TN / (TN + FP)$$

Equation 1: Original Quality Measure [12]

Later it is modified as (Equation 2)

$$\text{Sensitivity} \times \text{Precision} = TP / (TP + F7) \times TP / (TP + FP)$$

Equation 2: Quality Measure on Minority Class [12]

This is also modified with using Laplace correction as;

$$\text{Precision} = 1 + TP / (1+ k + TP + FP)$$

Equation 3: New Quality Measure on Minority Class [12]

Where, k is the number of classes.

So, PSO/ACO1 attempted to optimise both the continuous and nominal attributes present in a rule antecedent at the same time, whereas PSO/ACO2 takes the best nominal rule built by PSO/ACO2 and then attempts to add continuous attributes using a standard PSO algorithm.

5.6 Proposed Algorithm for Image Classification:

The algorithm for the image classification by using Swarm Intelligence Approach is given as follows-----

1. *Retrieve the original multi-spectral multi-band image in .tiff format.*
2. *Generate the training set using ERDAS software, according to the different classes by experts.*
3. *Convert this training data set into the unified Attribute Relation File Format.*
4. *Apply the cAnt Miner algorithm on the training data set obtained at step 3, by MYRA tool to get rule set-1and got to step 6.*
5. *Apply the Hybrid PSO/ACO2 on the training data set obtained at step 3, by open Source Tool to get rule set-2 and go to step6.*
6. *For each of rule set1 and rule set 2 perform the following steps—*
   - *For each pixel of original image*
   - *if the pixel validates the rule*

     Then

     the pixel is set to be in the region decided by the rule and the color code is assigned.

     Else

     Move to next rule.

     - Finally the classified image is obtained in .jpeg, jpg, .tiff or any other image format.

5.7 Description of Proposed Algorithm:

The algorithm for the image classification is built in such a way that the training set generated from original unclassified image can be in any tabular format as excel-sheet or access sheet or simple text file. This training set is the input for the rule generation by applying the cAnt miner algorithm and the Modified Hybrid PSO/ACO algorithm. Before this the training set is done unified by changing it to the Attribute Relation File Format. The cAnt miner algorithm is applied by using the MYRA tool and the Modified Hybrid PSO/ACO algorithm is applied by using the poen source tool for Hybrid PSO/ACO. These algorithms provide ruleset for the different regions. Then the regions are classified according to different rules. This algorithm can be shown diagrammatically as follows---

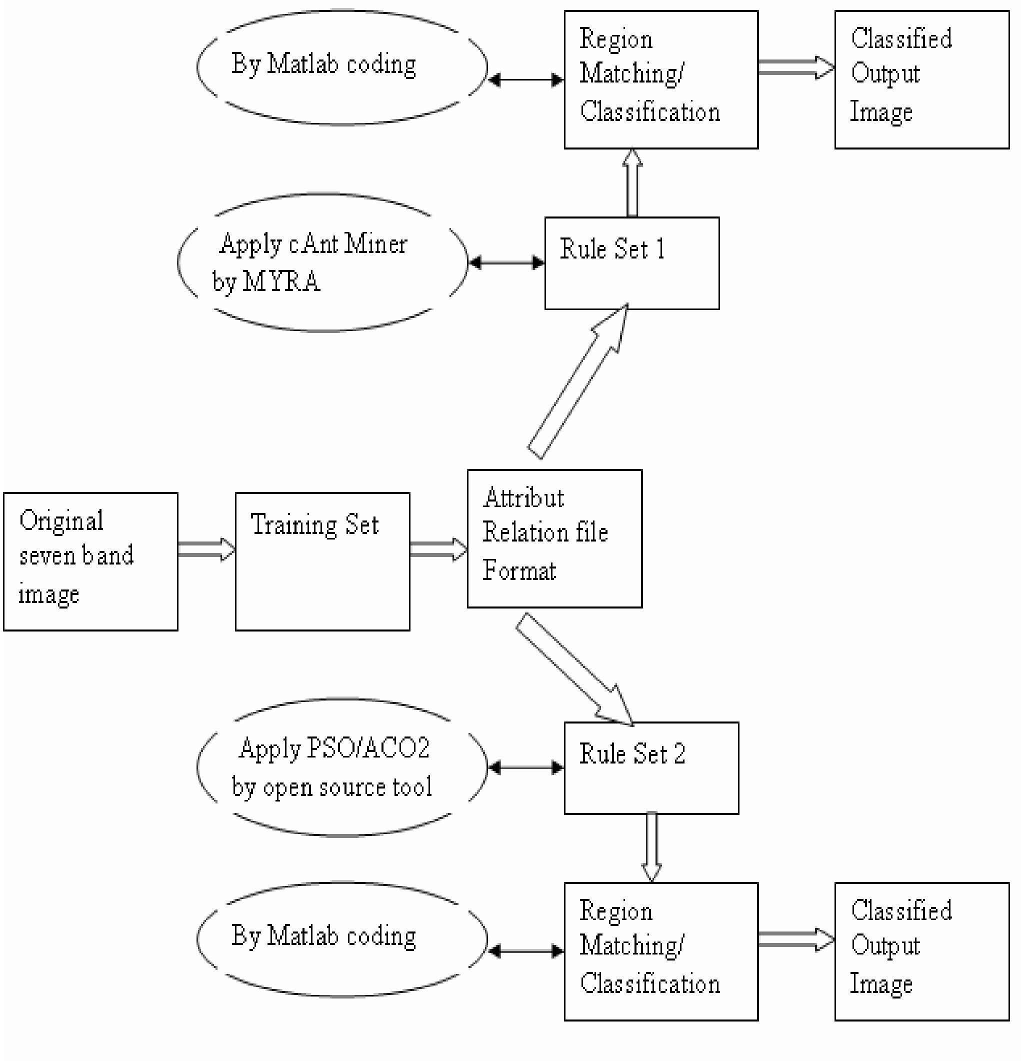


Fig:5.1 Diagrammatic Representation of the Proposed Algorithm

5.7.1 Training Set Generation

We have multi-spectral multi-band satellite image for classification. By these multi-band images the experts generate the training set. Experts of remote sensing open these images in ERDAS IMGINE. They manually select the pixels of different class regions and set a definition colour code using ERDAS Signature Editor. This data set provided by the experts in the form of digital numbers (intensity value pixel in a digital image) which were taken with help of Eradas [23]. These sets are taken by carefully selecting the areas (pixel by pixel) from all the images and noting the DN values of the pixels. This dataset can be in any Excel Sheet, Access sheet or in Text file. The Decision according to the DN values is also set in the table.

5.7.2 Rule Set Generation

From these DN values we extract rules by applying the cAntMiner and the PSO/ACO2 algorithm. To apply cAnt miner Algorithm we use MYRA tool and for applying the Modified Hybrid ACO/PSO2 we are using Open Source Tool for Hybrid ACO/PSO2. Before applying these algorithms for rule-set generation we change the training set into the unified Attribute Relation File Format, i.e., .arff format (see appendix E). The rules obtained by these algorithms are used to classify the image in different classes like: barren, rocky, crop etc.

For cAnt Miner algorithm following parameters are set—

- Fold for Cross Validation
- Minimum Covered Examples per rule
- Maximum number of Iterations
- Maximum number of uncovered cases
- Number of ants in colony

For PSO/ACO2 algorithm we have to set following parameters values according to our problem —

- Number of folds
- Number of Iterations
- Number of particles^2

5.7.3 Region Matching/ Classification

We obtain the rule sets from MYRA tool and the Open Source Tool for PSO/ACO2. But these rule set does not provide the classified image. So we code these rules in MATLAB file. In MATLAB coding the multi-band Images are read and then for each pixel of image the DN values are checked according to the rule set contained. If the pixel validates any rule then the decision of rule is set to be the class of pixel and the pixel is set to a specified colour. On executing this MATLAB file we obtain the *Final Classified Image*.

5.8 Accuracy Assessment

Another area that is continuing to receive increased attention by remote sensing specialists is that of classification accuracy assessment. Historically, the ability to produce digital land cover classifications far exceeded the ability to meaningfully quantify their accuracy. In fact, this problem sometimes precluded the application of automated land cover classification techniques even when their cost compared favourably with more traditional means of data collection. The lesson to be learned here is embodied in the expression “A classification is not complete until its accuracy is assessed."

Accuracy assessment is a general term for comparing the classification to geographical data that are assumed to be true, in order to determine the accuracy of the classification process. Usually, the assumed-true data are derived from ground truth data. It is usually not practical to ground truth or otherwise test every pixel of a classified image. Therefore, a set of reference pixels is usually used. Reference pixels are points on the classified image for which actual data are (or will be) known. The reference pixels are randomly selected.

Classification Error Matrix

One of the most common means of expressing classification accuracy preparation of a classification error matrix (sometimes called a confusion matrix or a contingency table). Error matrices compare, on a category-by category basis, the relationship between known reference data (ground truth) and the corresponding results of an automated classification. Such matrices are square, with the number of rows and columns equal to the number of categories whose classification accuracy is being assessed. This matrix stems from classifying the sampled training set pixels and listing the known cover types used for training (columns) versus the pixels actually classified into each land cover category by the classifier (rows). Several characteristics about classification performance are expressed by an error matrix. For example, one can study the various classification errors of omission (exclusion) and commission (inclusion).

Once accuracy data are collected (either in the form of pixels, cluster of pixels, or polygons) and summarized in an error matrix, they are normally subject to detailed interpretation and further statistical analysis.

A further point to be made about interpreting classification accuracies is the fact that even a completely random assignment of pixels to classes will produce percentage correct values in the error matrix. In fact, such a random assignment could result in a surprisingly good apparent classification result. The k ("KHAT") statistic is a measure of the difference between the actual agreement between reference data and an automated classifier and the chance agreement between the reference data and a random classifier. Conceptually, k can be defined as

$$k = \frac{\text{observed accuracy - chance agreement}}{\text{1 - Chance agreement}}$$

This statistic serves as an indicator of the extent to which the percentage correct values of an error matrix are due to "true" agreement versus "chance" agreement. As true agreement (observed) approaches 1 and chance agreement approaches 0, k approaches 1. This is the ideal case. In reality, k usually ranges between 0 and 1. For example, a k value of 0.67 can be thought of as an indication that an observed classification is 67 percent better than one resulting from chance. A k with the value of 0 suggests that a given classification is no better then a random assignment of pixels. In cases where chance agreement is large enough, k can take on negative values-an indication of very classification performance. (Because the possible range of negative values depends on specific matrix, the magnitude of negative values should not be interpreted as an indication of relative classification performance).

	barren	Vegetation	rocky	Urban	Waterbody	Total
Barren	30	1	2	1	0	**34**
Vegetation	1	145	1	0	2	**149**
Rocky	1	1	48	0	0	**50**
Urban	0	1	1	148	0	**150**
Waterbody	0	1	1	0	46	**48**
Total	**32**	**149**	**53**	**149**	**48**	**431**

Table.5.1 Example of Error Matrix

The KHAT statistic is computed as--

$$k = \frac{N\sum_{i=1}^{r} x_{ii} - \sum_{i=1}^{r}\left(x_{i+} \cdot x_{+i}\right)}{N^2 - \sum_{i=1}^{r}\left(x_{i+} \cdot x_{+i}\right)}$$

$$\begin{aligned}
&\textit{where} \\
&r = \textit{number of rows in the error matrix} \\
&x_{ii} = \textit{the number of observations in row } i \textit{ and column } i \\
&\qquad (\textit{on the major diagonal}) \\
&x_{i+} = \textit{total of observation in row } i \\
&x_{+i} = \textit{total of observation in column } i \\
&N = \textit{total number of observations included in matrix}
\end{aligned}$$

Where:

r = number of rows in the error matrix

x_{ii} = the number of observations in row i and column i (on the major diagonal)

x_{i+} = total of observations in row i (shown as marginal total to right of the matrix)

x_{+i} = total of observations in column i (shown as marginal total at bottom of the matrix)

N = total number of observations included in matrix

The value of 'k' lies between 0 and 1. The value of k can be 0 or 1 only in ideal case only. If value of k is near to 0 then the classification is supposed to be less efficient where as if the value is nearer to 1 then the classification is supposed to be more efficient.

Chapter 6

EXPERIMENTAL STUDY

We are using the algorithm described in the chapter 5 for the satellite image classification so that a number of information about the land cover area can be acquired with the resources and better accuracy. The complete experimental study and accuracy assessment with the result analysis is described in the following sections.

Our objective is to use the proposed swarm algorithms as an efficient Landcover classifier for satellite image. We have taken a multi-spectral, multi resolution and multi-sensor image of Alwar area in Rajasthan and Shivpuri area in Madhya Pradesh provided by DTRL-DRDO. The satellite image for 7different bands is taken (Fig 6.1). These bands are Red, Green, Near Infra Red(NIR), Middle Infra Red (MIR), Radarsat-1 (RS1), Radarsat-2(RS2), and Digital Elevation Model) DEM. We are having spectral signatures set from seven bands.

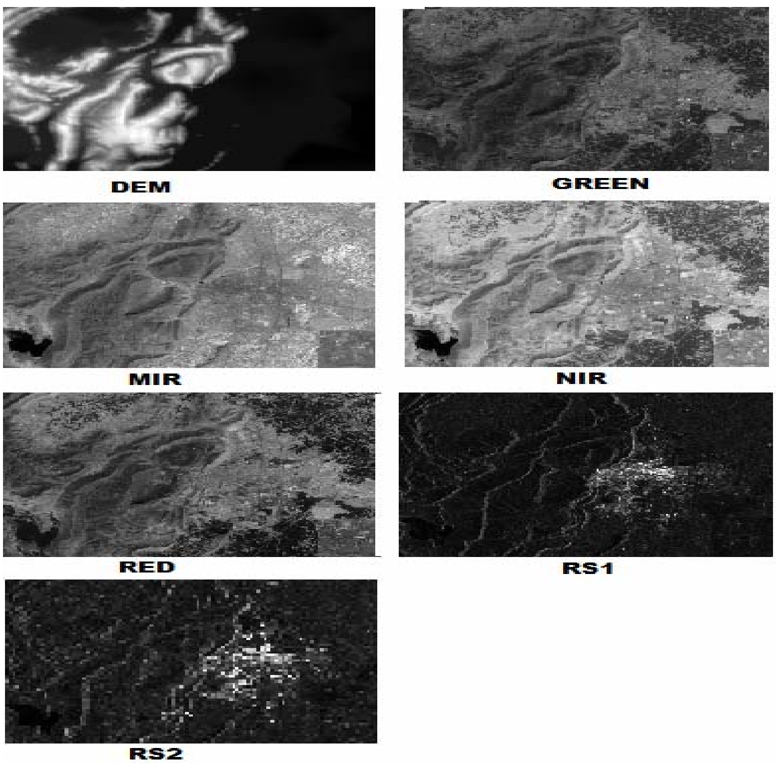


Fig.6.1 7-Band Images of Alwar Area

For the Shivpuri Region the satellite image for 3different bands is taken. These bands are Red, Green, and Blue.

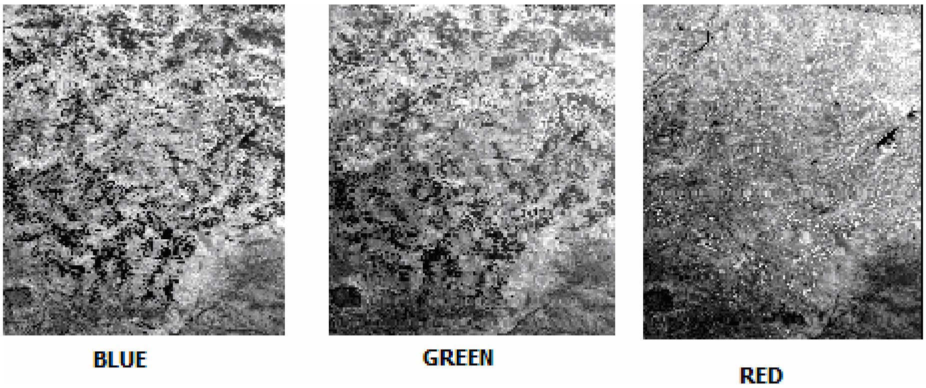


Fig.6.2. 3-Band Images of ShivPuri Area

In the Alwar area experiment we are classifying the image in 5 classes as **vegetation, barren, urban, rocky region and water**. While in the Shivpuri area experiment we are classifying the image in 5 classes as **crop, vegetation, open land, shallow water, water.**

6.1 Training Set Generation

The training set in excel sheet format for the Alwar area obtained by ERDAS software is as follows.

K182

	A	B	C	D	E	F	G	H	I
1		RED	GREEN	NIR	MIR	RS1	RS2	DEM	DECISION
155		127	96	184	131	17	32	29	Barren
156		132	104	182	142	13	28	29	Barren
157		103	83	160	135	18	20	29	Barren
158		132	104	190	146	18	39	29	Barren
159									
160	Stdev	18.0886	16.6277	15.8201	14.415	16.0566	16.8598	39.5913	
161	Avg	133.911	106.331	196.841	136.166	28.0318	34.3439	55.1083	
162									
163		62	49	135	91	44	40	94	Rocky
164		84	64	160	102	20	25	165	Rocky
165		52	45	129	85	15	29	107	Rocky
166		91	69	171	106	10	46	123	Rocky
167		87	67	168	104	8	21	157	Rocky
168		76	59	157	95	9	47	114	Rocky
169		70	51	159	95	11	46	127	Rocky
170		82	59	159	100	7	9	173	Rocky

Fig .6.3 Training Data Set for Alwar Area

The training set in excel sheet format for the Shivpuri area obtained by ERDAS software is as follows.

GREEN	BLUE	RED	decision
95	53	205	crop
102	59	198	crop
52	11	187	crop
59	17	209	crop
81	35	209	crop
59	20	230	crop
81	44	205	crop
74	32	172	crop
74	35	162	crop
59	17	201	crop
52	8	227	crop
49	5	212	crop
42	8	205	crop
91	44	201	crop
66	29	227	crop
63	23	227	crop
63	23	227	crop
59	29	223	crop
63	32	227	crop
74	38	209	crop
63	14	227	crop

Fig 6.4 . Training Data Set for Shivpuri Area

6.2 Rule Set Generation

To apply cAnt miner Algorithm we use MYRA tool and for applying the Modified Hybrid ACO/PSO2 we are using Open Source Tool for Hybrid ACO/PSO. For cAnt Miner algorithm we have set the parameters as follows—

Fold for Cross Validation=10

Minimum Covered Examples per rule=5

Maximum number of Iterations=100

Maximum number of unocovered cases=5

Number of ants in colony=50

The rule set generated by Myra is like as follows—

=== Run Information ===

Relation: Completedataset

Instances: 1431

Attributes: 7

Duplicates: yes

Test mode: 10-fold cross-validation

=== Run Information ===

Relation: Completedataset

Instances: 1287

Attributes: 7

Duplicates: yes

=== Discovered Rules ===

IF RS2 < 8.0 THEN Waterbody

IF NIR < 91.0 AND RS2 < 50.0 AND DEM < 87.0 THEN Vegetation

IF GREEN < 75.0 AND DEM >= 45.0 THEN Rocky

IF DEM < 18.0 AND RED >= 62.0 THEN Urban

IF RED >= 101.0 THEN Barren

IF <empty> THEN Vegetation

Measure (AccuracyMeasure): 0.9722222222222222

=== Run Information ===

Relation: Completedataset

Instances: 1288

Attributes: 7

Duplicates: yes

=== Discovered Rules ===

IF RS2 < 8.0 THEN Waterbody

IF DEM < 19.0 AND RED >= 82.0 THEN Urban

IF GREEN >= 70.0 THEN Barren

IF MIR >= 137.0 THEN Vegetation

IF GREEN >= 37.0 THEN Rocky

IF GREEN < 30.0 THEN Rocky

IF <empty> THEN Rocky

Measure (AccuracyMeasure): 0.965034965034965

=== Run Information ===

Relation: Completedataset

Instances: 1288

Attributes: 7

Duplicates: yes

=== Discovered Rules ===

IF GREEN >= 70.0 AND DEM < 29.0 THEN Urban

IF DEM >= 62.0 AND MIR < 115.0 THEN Rocky

IF NIR < 45.0 THEN Waterbody

IF RED < 44.0 THEN Vegetation

IF RED >= 101.0 THEN Barren

IF <empty> THEN Rocky

Measure (AccuracyMeasure): 0.965034965034965

DataSet sucessfully loaded: C:\Users\user\Desktop\New_work\training set\new_Shivp_dataset\new_Shivp_dataset\new_Shivp_dataset.arff

Number of Nominal and Binary Attributes:0

Number of Continous Attributes:3

Number of Records:1001

For PSO/ACO2 algorithm we have set the parameters as follows—

Number of folds =10

Number of Iterations=10

Number of particles^2=200

The rule set generated by Open Source Tool is like as follows—

Outputting settings and raw predictions to: C:\Users\user\Desktop\New_work\training set\new_Shivp_dataset\new_Shivp_dataset\rawoutput.txt

Starting Experiment: new_Shivp_dataset.arff

Fold: 0

Rule 0 : IF blue <= 99.08545383298684 red >= 126.79886317738477 THEN crop Quality: 0.99 (20,0)

Rule 1 : IF red >= 221.0877245423223 THEN open Quality: 0.99 (21,0)

Rule 2 : IF blue <= 108.83245476075632 THEN water Quality: 0.99 (20,0)

Rule 3 : IF green <= 143.4580025219766 blue >= 131.53067359657246 red >= 63.51540796599107 THEN vegetation Quality: 0.99 (17,0)

Rule 4 : IF green >= 167.12603104233386 blue <= 178.99695925447003 THEN shallow_water Quality: 0.99 (11,0)

Rule 5 : IF blue <= 143.1195634833576 red >= 26.322669789944076 THEN shallow_water Quality: 0.98 (6,1)

Rule 6 : IF green <= 180.21910209302524 blue >= 174.27153451342735 THEN vegetation Quality: 0.97 (2,0)

Rule 7 : IF green >= 191.95505808773763 red >= 107.55495385767463 THEN shallow_water Quality: 0.95 (2,0)

Rule 8 : IF red >= 66.32637613797527 THEN vegetation Quality: 0.22 (0,0)

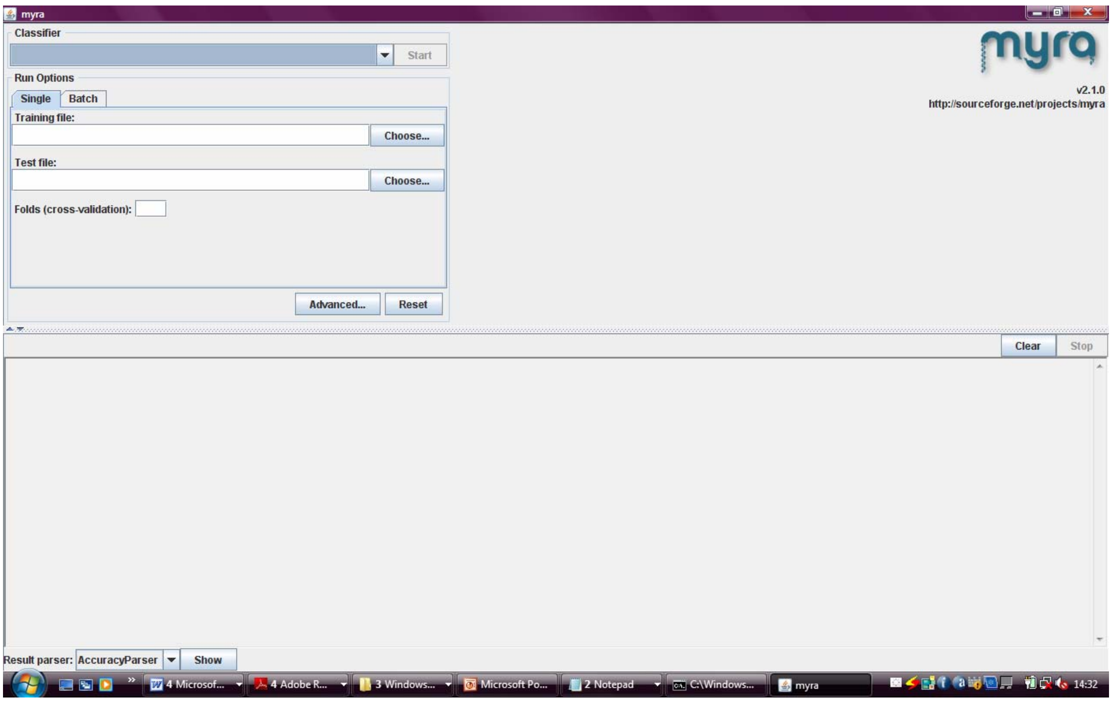


Fig:6.5 The MYRA tool for applying cAnt Miner

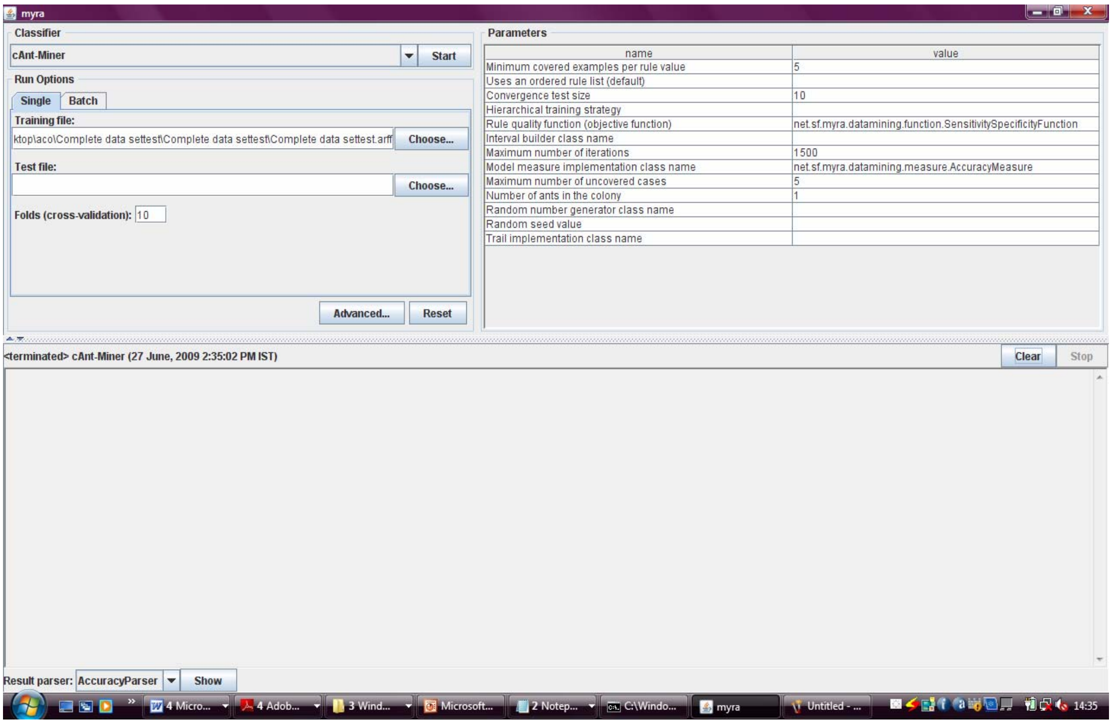


Fig:6.6 Extracting rules by MYRA tool

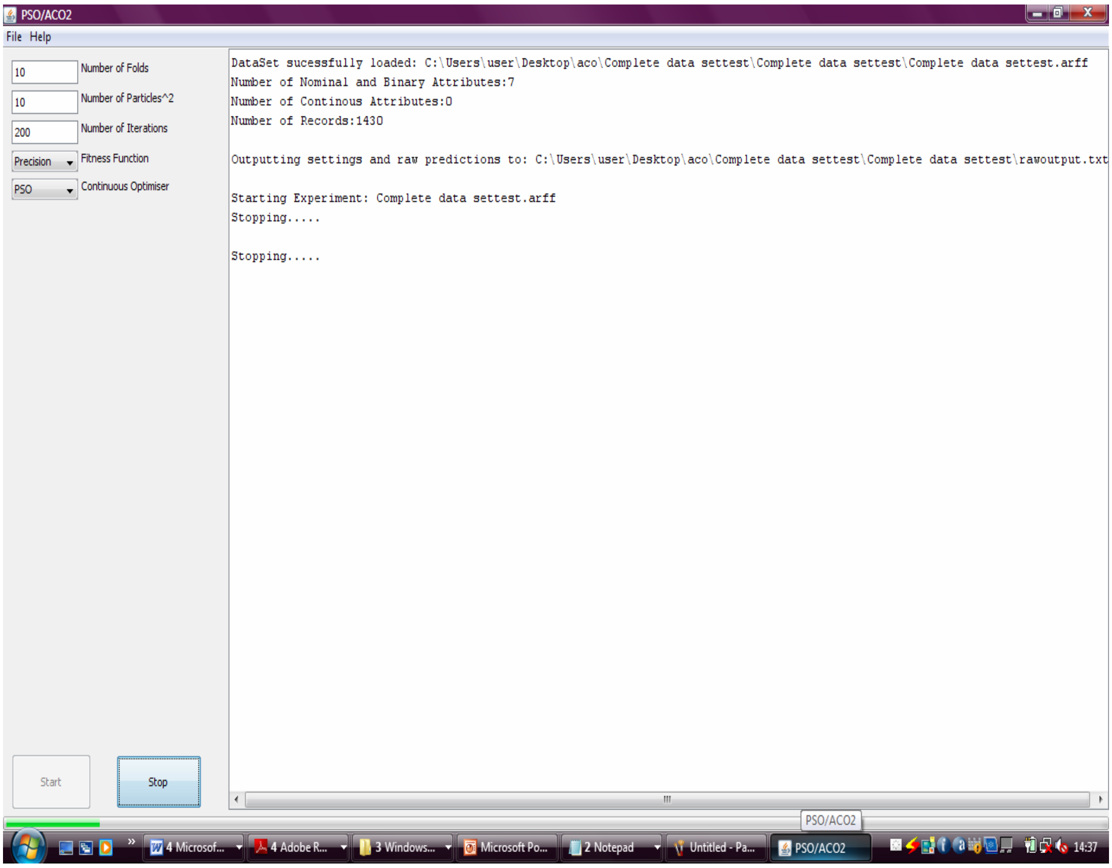


Fig: 6.7 Extracting rules by Hybrid ACO/PSO2 Tool

6.3 Region Matching

The region matching or classification coding is done in Matlab and it will give the output classified image on executing.

The Malab Coding of region matching for Alwar Area using rule set of cAntMiner approach is following-----

```
clear all;
clc;
RED=imread('C:\test_sonam\images 7_band\red.tif');
[m,n,o]=size(RED);


GREEN=imread('C:\test_sonam\images 7_band\green.tif');

NIR=imread('C:\test_sonam\images 7_band\nir.tif');
%figure;imshow(C)
MIR=imread('C:\test_sonam\images 7_band\mir.tif');
%figure; imshow(D)
RS1=imread('C:\test_sonam\images 7_band\rs1.tif');
%figure; imshow(E)
RS2=imread('C:\test_sonam\images 7_band\rs2.tif');
%figure; imshow(F)
DEM=imread('C:\test_sonam\images 7_band\dem.tif');
M(:,:,1)=NIR(:,:);
M(:,:,2)=GREEN(:,:);
M(:,:,3)=RED(:,:);

for i=1:m

    for j=1:n
if((DEM(i,j) < 18.0)&  (GREEN(i,j) >= 75.0)) %THEN Urban
            M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
elseif(GREEN(i,j) >= 70.0) % THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
elseif(MIR(i,j) >= 137.0)% THEN Vegetation
    M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif(NIR(i,j) < 74.0)% THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif((DEM(i,j) < 18.0 )&  ( RED(i,j) >= 82.0))% THEN Urban
    M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
elseif(NIR(i,j) < 45.0) % THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif((DEM(i,j) >= 45.0 )&  ( NIR(i,j) < 173.0))% THEN Rocky
    M(i,j,1) = 160;
```

```
            M(i,j,2) = 82;
            M(i,j,3) = 45;
elseif(GREEN(i,j) < 62.0) % THEN Vegetation
    M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif((RS2(i,j) >= 23.0 )&  ( RED(i,j) >= 101.0))% THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
elseif(RS1(i,j) >= 18.0) %THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
elseif(RS2(i,j) < 23.0) % THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
elseif((DEM(i,j) < 18.0 )&  ( RED(i,j) >= 82.0)) % THEN Urban
    M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
elseif(GREEN(i,j) >= 70.0) % THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
elseif(MIR(i,j) < 42.0) % THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif(MIR(i,j) >= 140.0) % THEN Vegetation
     M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif(MIR(i,j) < 137.0) % THEN Rocky
    M(i,j,1) = 160;
            M(i,j,2) = 82;
            M(i,j,3) = 45;
%IF <empty> THEN Urban
elseif((MIR(i,j) < 91.0 )&  ( RED(i,j) < 25.0)) % THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif((MIR(i,j) >= 159.0 )&  ( GREEN(i,j) < 107.0))% THEN Vegetation
     M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif((DEM(i,j) < 18.0 )&  ( GREEN(i,j) >= 62.0))% THEN Urban
    M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
elseif((MIR(i,j) < 107.0 )&  ( NIR(i,j) < 74.0))% THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif(RED(i,j) >= 97.0) % THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
elseif(MIR(i,j) < 137.0) % THEN Rocky
    M(i,j,1) = 160;
```

```
            M(i,j,2) = 82;
            M(i,j,3) = 45;
%IF <empty> THEN Vegetation
elseif((MIR(i,j) < 106.0 )&  ( RED(i,j) < 25.0))% THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif((MIR(i,j) >= 155.0 )&  ( RED(i,j) < 117.0))% THEN Vegetation
     M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif((GREEN(i,j) < 75.0 )&  ( NIR(i,j) >= 74.0)& (MIR(i,j) <
115.0)) % THEN Rocky
    M(i,j,1) = 160;
            M(i,j,2) = 82;
            M(i,j,3) = 45;
elseif((MIR(i,j) < 113.0 )&  ( RED(i,j) < 82.0)) % THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif(RED(i,j) < 62.0)% THEN Vegetation
     M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif(DEM(i,j) >= 18.0 ) %THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
elseif(GREEN(i,j) >= 96.0)  % THEN Urban
    M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
elseif(NIR(i,j) < 162.0) % THEN Urban
    M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
%IF <empty> THEN Barren
elseif(RS2(i,j) < 8.0) % THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif((NIR(i,j) < 93.0 )&  ( MIR(i,j) >= 137.0)) % THEN Vegetation
     M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif(NIR(i,j) < 74.0) % THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif((GREEN(i,j) < 70.0 )&  ( DEM(i,j) >= 45.0)) % THEN Rocky
    M(i,j,1) = 160;
            M(i,j,2) = 82;
            M(i,j,3) = 45;
elseif(GREEN(i,j) < 62.0) % THEN Vegetation
     M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif(DEM(i,j) >= 19.0) % THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
```

```
elseif(NIR(i,j) < 162.0) % THEN Urban
    M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
elseif(GREEN(i,j) < 98.0) % THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
%IF <empty> THEN Urban
elseif(RS2(i,j) < 8.0) % THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif((MIR(i,j) >= 155.0 )&  ( GREEN(i,j) < 107.0)) % THEN
Vegetation
     M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif((MIR(i,j) < 96.0 )&  ( NIR(i,j) < 74.0)) % THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif((GREEN(i,j) < 75.0 )&  ( MIR(i,j) < 115.0)) % THEN Rocky
    M(i,j,1) = 160;
            M(i,j,2) = 82;
            M(i,j,3) = 45;
elseif(GREEN(i,j) < 62.0) % THEN Vegetation
     M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif(NIR(i,j) < 173.0) % THEN Urban
    M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
elseif((DEM(i,j) >= 18.0 )&  ( RED(i,j) >= 101.0)) % THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
elseif(GREEN(i,j) >= 104.0) % THEN Urban
    M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
elseif(DEM(i,j) < 45.0) % THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
%IF <empty> THEN Rocky
elseif(RS2(i,j) < 8.0) % THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif( (RED(i,j) >= 85.0 )&  ( DEM(i,j) < 25.0)) % THEN Urban
    M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
elseif(RED(i,j) >= 101.0) % THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
elseif((MIR (i,j)>= 137.0 )&  ( RED(i,j) < 62.0)) % THEN Vegetation
     M(i,j,1)= 255;
```

```
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif(DEM(i,j) >= 45.0) % THEN Rocky
    M(i,j,1) = 160;
            M(i,j,2) = 82;
            M(i,j,3) = 45;
elseif(NIR(i,j) < 104.0) % THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
%IF <empty> THEN Urban
elseif((GREEN(i,j) >= 70.0 )&  ( DEM(i,j) < 28.0)) % THEN Urban
    M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
elseif(GREEN(i,j) >= 70.0) % THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
elseif(NIR(i,j) < 45.0) % THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif(MIR(i,j) < 137.0) % THEN Rocky
    M(i,j,1) = 160;
            M(i,j,2) = 82;
            M(i,j,3) = 45;
elseif(RED(i,j) < 62.0) % THEN Vegetation
     M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
%IF <empty> THEN Urban
elseif((MIR(i,j) < 91.0 )&  ( RED(i,j) < 27.0)) % THEN Waterbody
    M(i,j,1)=57;
             M(i,j,2)=17;
             M(i,j,3)=247;
elseif((MIR(i,j) >= 159.0 )&  (GREEN(i,j) < 107.0)) % THEN Vegetation
     M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif((GREEN(i,j) < 70.0 )&  ( MIR (i,j)< 137.0)) % THEN Rocky
    M(i,j,1) = 160;
            M(i,j,2) = 82;
            M(i,j,3) = 45;
elseif((DEM(i,j) >= 18.0 )&  ( NIR(i,j) >= 160.0)) % THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
elseif(RED(i,j) < 62.0) % THEN Vegetation
     M(i,j,1)= 255;
            M(i,j,2) = 0;
            M(i,j,3) = 0;
elseif(GREEN(i,j) >= 96.0) % THEN Urban
    M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
elseif(NIR(i,j) >= 162.0) % THEN Barren
    M(i,j,1)= 0;
            M(i,j,2) = 255;
            M(i,j,3) = 0;
elseif(DEM(i,j) < 155.0) %THEN Urban
```

```
    M(i,j,1)= 64;
            M(i,j,2) = 224;
            M(i,j,3) = 208;
%IF <empty> THEN Rocky

             end
   end
end


figure;imshow(M)
```

Similarly the matlab code is written for the PSO/ACO2 for Alwar area and for cAntMiner & PSO/ACO2 approach for ShivPuri area. These codes are executed for obtaining the required classified images.

6.4 Result Analysis:

In remote sensing k, the kappa coefficient is very important and prevalent as a measure of accuracy assessment of Land cover classification. k can be derived from the confusion matrix or error matrix. Hence,

$$k \propto \text{Knowledge}$$

if k=0 then the classifier is inconsistent and does not represent the requisite knowledge

if k=1 then the classifier is consistent and represent the requisite knowledge

Here the k coefficients for the other two probabilistic classifers i.e, Minimum Distance classifier & Maximum Likelihood Classifier and Swarm Classifier for Alwar area area experiment are compared in table1. The k coefficients for Fuzzy set and Rough set classifier & Swarm Classifier for Shivpuri area experiment are compared in table2 . Classification results for Alwar Area are shown in figure 1 and the classification results for ShivPuri Area are shown in figure 2.

Minimum Disatnce Classifier	Maximum Likelihood Classifier	cAnt Miner Classifier	PSO/ACO2 Classifier
k=0.7364	k=0.7525	k=0.9751	k=0.9784

Table 6.1. Classification Comparison for Alwar area

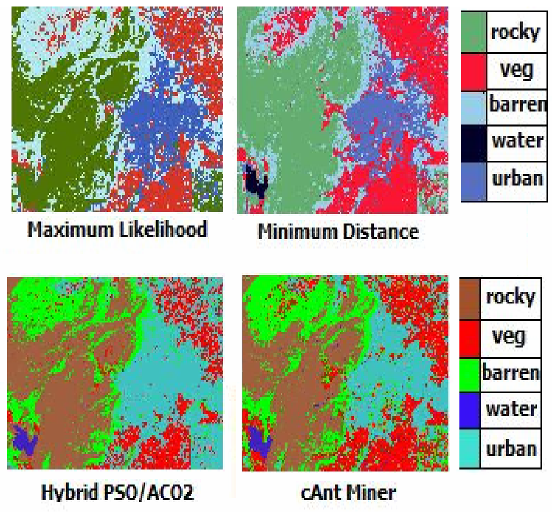


Fig.6.8 Classifiication Results for Alwar area

Fuzzy Set Classifier	Rough Set Classifier	cAnt Miner Classifier	PSO/ACO2 Classifier
k=0.785	k=0.847	k=0.964	k=0.975

Table 6.2. Classification Comparison for Shivpuri area

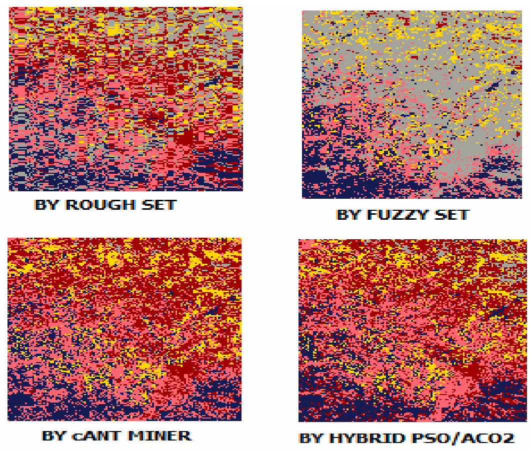


Fig.6.9 Classification Results for Shivpuri area

6.5 Application:

Accuracy of image classification is the one of important factor for analysis of natural resource management/ land cover analysis. Because, if you have very low spatial resolution land cover image with good numbers of classified accuracy will able to gives land cover analysis result similar to high resolution imageries. Thus, with lower cost & high degree of classification accuracy we will able to replace high resolution high cost satellite imageries.

Here, we have used cAnt Miner ant PSO/ACO2 algorithm for LISS III image (Alwar region of Rajasthan), which gives good degree of classification accuracy.

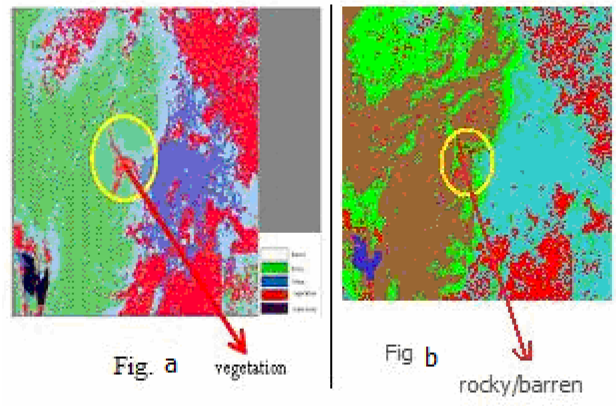


Fig: 6-10 Application 1

In the figure above, we can see the comparison between the accuracy of ACO-classification Fig.a and Minimum distance to mean classification (using ERDAS) Fig.b. Here, in Fig.b the encircled region shows the presence of vegetation which actually is not possible in a Metamorphic Rocky region, whereas PSO/ACO2 classified image Fig.b correctly shows it as barren region.

In our second experiment, we have used cAnt Miner ant PSO/ACO2 algorithm for LISS III image (Shivpuri region of Madhya Pradesh), which gives good degree of classification accuracy as compare to the Fuzzy and Rough set methods. In Fuzzy classified image the encircled region shows that there is open land nearby the water, which is not correctly interpreted. It is correctly given by PSO classified image, where it is shown as vegetation.

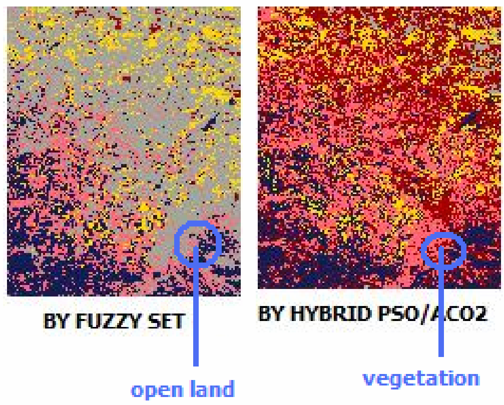


Fig.6.11 Application 2

.

Chapter 7

PUBLICATION FROM THESIS

During the period of working over this project we interacted with International community working on Artificial Intelligence. We discussed our approach for representing knowledge with them and collected reviews and worked over the suggestion send to us. Two research papers have been accepted in International conferences for presentation and will be published in their proceedings.

First paper focuses on the optimised approach of image classification of satellite multi spectral images. This paper deals with the land cover mapping by using swarm computing techniques and presents the results of experiments also. The second paper gives a comparison study of the swarm intelligence classifier to the rough and fuzzy set classifier.

7.1 Details of Conferences:

1. **Conference Name:** International Conference of Artificial Intelligence and Pattern Recognition (AIPR-09), Orlando, USA.

   URL: http://www.promoteresearch.org

   Paper Title: "Remote Sensing Image Classification by Improved Swarm Inspired Techniques"

   Authors: Shelly Bansal, Dr. Daya Gupta, V.K.Panchal, Shashi Kumar

   Location: Orlando, USA.

   Publisher/Proceedings: The paper will be included in the conference proceedings, which has an ISBN number. The proceedings will also be submitted for several database indexes.

2. **Conference name:** International Conference on Contemporary Computing (IC3-09), Noida, India.

URL: http://www.jiit.ac.in

Paper Title: "Swarm Intelligence Inspired Classifiers in comparison with Fuzzy and Rough Classifiers- A Remote sensing perspective"

Authors: Shelly Bansal, Dr. Daya Gupta, V.K.Panchal, Shashi Kumar

Location: Noida, India.

Publisher/Proceedings: Springer-Verlag, Berlin, Heidelberg 2009

Chapter 8

CONCLUSION AND FUTURE WORK

The complex problem of delineation of the Landuse/Landcover classes having similar spectral signatures depends extensively on expert's perception of the scene. Training sets, although a manifestation of the expert's knowledge is an indirect method. Most of the supervised classification techniques do not elicit the underlying knowledge contained in the training set. Swarm Intelligence approach, it is observed, facilitates the explicit characterization of the underlying knowledge in the training set in the form of < if … then …> rules. This form of knowledge representation is quite akin to the human thinking and provides a better insight into the classification mechanism.
Also, the Swarm Intelligence Approach enables us to optimize the requirement of minimum number of spectral bands, i.e. the dimensionality reduction,
Further, the κ -coefficient can also be used as a measure of elicited knowledge. This also provides us a mechanism to compare the knowledge content with other paradigms of supervised classification.

Following inferences are arrived at:

i. The use of Swarm Intelligence, in eliciting expert's knowledge, a very complex task of resolving class conflict of spectrally similar classes which otherwise is usually avoided in the conventional methodologies, is innovative.
ii. Accuracy level of the classified image by this approach is comparable and quite acceptable, especially in time-scared scenario.
iii. It is very difficult even for an expert to explicitly explain his reasoning. The approach based on Swarm Intelligence offers a very transparent knowledge representation scheme.
iv. Finally, the elicited knowledge can be measured in terms of κ-coefficient. This provides a very transparent and efficient mechanism for knowledge comparison with other black-box type of classification systems.

FUTURE DIRECTIONS

Expert's knowledge is a synthesis of various factors. In this study K- coefficient provides a measurable coefficient of the expert knowledge. Studies are undertaken how context information / knowledge can be segregated to get insight into the expert's knowledge canvas, using the Swarm Intelligence. Future research will focus on using these algorithms with granular computing such that the strengths of both the fields can be exploited. The classifiers that perform better for a particular land cover class will be considered more reliable during conflict resolution. So these swarm techniques can also be combined with fuzzy-rough approach also in future.

Chapter 9

REFERENCES

1. A.Colorni, M. Dorigo, and V. Maniezzo, Distributed Optimization by Ant Colonies, In Proceedings of the First European Conference on Artificial Life (ECAL 91), pp. 134-142, 1991
2. D. Bratton and J. Kennedy. Defining a Standard for Particle Swarm Optimization. Proceedings of the 2007 IEEE Swarm Intelligence Symposium,
3. DI Kaichang, LI Deren, LI Deyi, Remote Sensing Image Classification With Gis Data Based On Spatial Data Mining Techniques in proceedings of IEEE, 2007.
4. D.J. Hand. Construction and Assessment of Classification Rules. Wiley, 1997.
5. Fernando E.B. Otero, Alex A. Freitas, and Colin G. Johnson, cAnt-Miner: An Ant Colony Classification Algorithm to Cope with Continuous Attributes in Springer-Verlag Berlin, Heidelberg 2008.
6. I.H. Witten, E. Frank, Data Mining: Practical machine learning tools and techniques, 2nd Edition, Morgan Kaufmann, San Francisco, CA, 2005.
7. Ian H. Witten, E. Frank, Data Mining—Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kauffmann, 1999
8. J. Kennedy and R. C. Eberhart, with Y. Shi. SwarmIntelligence, San Francisco: Morgan Kaufmann/Academic Press, 2001
9. J. Kennedy and R. Mendes, Population structure and particle swarm performance. Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2002), Honolulu, Hawaii USA. 2002.
10. Long, W., III; Srihann, S, Geoscience and Remote Sensing Symposium, Unsupervised and supervised classifications: Land cover classification of SSC image: unsupervised and supervised classification using ERDAS Imagine, IGARSS '04. Proceedings, Volume 4, 20-24, Sept 2004.
11. M. Dorigo and T. Stuetzle. Ant Colony Optimization. MIT Press, 2004.
12. N. Holden and A.A. Freitas. A hybrid particle swarm/ant colony algorithm for the classification of hierarchical biological data. In: Proc. 2005 IEEE Swarm Intelligence Symposium (SIS-05), pp. 100-107, IEEE, 2005.

13. N. Holden and A.A. Freitas. Hierarchical Classification of GProtein-Coupled Receptors with a PSO/ACO Algorithm. In: Proc. IEEE Swarm Intelligence Symposium (SIS-06), pp. 77-84. IEEE, 2006.
14. Omkar S.N., Kumar Manoj, Muley Dipti, Urban Satellite Image Classification using Biologically Inspired Techniques in proceedings of IEEE, 2007.
15. Panchal VK, Singhal Naresh, Kumar Shashi, Bhakna Sonam, Rough-Fuzzy Sets Tie-Up for Geospatial Information Proceedings of International Conference on Emerging Scenarios in Space Technology and Applications (ESSTA2008), SSTA2008), Chennai, India, vol-I, 2008.
16. Pawlak, Z., “Rough Set Theory and its Applications to Data Analysis”,Cybernetics and Systems 29(7): 661-688 (1998)
17. Pawlak,Z, Skowron,A.,”Rudiments of rough sets.Inf. Sci.177(1): 3-27(2007)
18. R.S. Parpinelli, H.S. Lopes and A.A. Freitas. Data Mining with an Ant Colony Optimization Algorithm, IEEE Trans. On Evolutionary Computation, special issue on Ant Colony algorithms, 6(4), pp. 321-332, Aug 2002.
19. Thomas M. Lillesand , Ralph W. Kiefer, “Remote Sensing and Image Interpretation”, 6th Edition, Wiley, 2008.
20. T. Sousa, A. Silva, A. Neves, Particle Swarm based Data Mining Algorithms for classification tasks, Parallel Computing 30, pp. 767–783, 2004.
21. U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, From data mining to knowledge discovery: an overview, in: Advances in Knowledge Discovery and Data Mining, AAAI/MIT, Cambridge, 1996, pp. 1–34.
22. Zadeh,L.A.,"Fuzzy sets," Information and Control, vol. 8, no. 3, pp.338- 353, June 1965
23. Zadeh, L.A. , “Fuzzy logic and its application to approximate reasoning”, In: Information processing 74, Proc.IFIP Congr, 1977(3) pp.591-594.
24. ERDAS,/http://www.ERDAS.com/
25. MATLAB, http://www.mathworks.com/
26. T.M. Lillesand and R.W. kiefer “Remote Sensing & Image Interpretation ” , 3rd edition, 1994.
27. Canada Center for Remote Sensing, “Fundamentals of Remote sensing”, pg no. 1-164.

Appendix A

An Introduction to MATLAB

URL: http://mathworks.com/

MATLAB is a high performance language for technical computing. It integrates computation, visualization and programming in an easy-to-use environment where problems and solutions are expresses in familiar mathematical notation. Typical uses include:

1. Math and computation
2. Algorithm, simulation and prototyping
3. Modeling, simulation and prototyping
4. Data analysis, exploration and visualization
5. Scientific and engineering and visualization
6. Application development, including graphical user interface building

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows you to solve many technical computing problems, especially those with matrix and vector formulations in a fraction of the time it would take to write a program in a scalar non interactive language such as C or FORTRAN.

The name MATLAB stands for Matrix Laboratory. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects. Today MATLAB uses software developed by the LAPACK and ARPACK projects, which together represent the state-of-the-art in software for matrix computation.

MATLAB has evolved over a period of years with input from many users. In university environments, it is the standard instructional tool for introductory and advanced courses in mathematics, engineering and science. In industry, MATLAB is the tool of choice for high-productivity research, development and analysis.

MATLAB features a family of application-specific solutions called toolboxes. Very important to most users of MATLAB toolboxes allow you to learn and apply specialized technology. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control system neural networks, fuzzy logic, wavelets, simulation and many others.

FILE TYPES

MATLAB can read and write several types of files.There are mainly five different types of files used in MATLAB which is used for storing data or programs.

- **M-FILES**- They are the standard ASCII files, with a .m extension to the file name. There are basically two types of files and they are SCRIPT and FUNCTION file. In general, mostly MATLAB files are saved as M-FILES.
- **MAT-FILES**- They are the binary data-files, with a .mat extension to the filename. These files are created when you save the MATLAB data with the save command. The data which you save in MATLAB can only be read by mat lab as it save in a special format.
- **FIG-FILES**- They are the binary figure-file, with a .fig extension to the filename.Such files are created by saving a figure in this format by using the save and saveas option in it. These files basically create all kind of information which is used for again recreating a figure and can be opened by filename.fig.
- **P-FILES**- These are the compiled M-File , with a .p extension to the filename.These file can be executed directly without using any compiler and parsed in it.These files are created with the P-CODE command.
- **MEX-FILES-** These are MATLAB-callable Fortran and C programme , with the .mex extension to the filename.Use of these file require some experience in MATLAB and lot of patience in it .

Appendix B

An Introduction to ERDAS

URL: http://www.ERDAS.com/

Overview

ERDAS is pleased to provide ERDAS IMAGINE® version 8.4. Many private and commercial users who need to extract and interpret information from imagery recognize ERDAS IMAGINE as a must have. With ERDAS IMAGINE 8.4, ERDAS' latest, most advanced release of ERDAS IMAGINE , production workflows are enhanced and simplified like never before. As an example, the Batch Wizard streamlines repetitive procedures such as importing; reprojecting, and exporting large numbers of files at once, using a wizard approach to record and "re-play" commonly used procedures. Also featured is the IMAGINE Expert Classifier™ – a tool for graphically building and executing geographically aware, rules-based expert systems. This tool can be used to build decision support systems, classifiers for high-resolution imagery, GIS analysis techniques, etc. These can then be distributed to other users for use with their own data.

Key Features Summary

- IMAGINE Expert Classifier
- Direct read and edit of ESRI's Shapefiles
- Direct read of ESRI's SDE data
- Enhanced and expanded native raster file handling
- Re-projection of raster data on-the-fly
- Batch processing wizard
- Enhanced Viewer functionality
- Improved print versatility on Windows NT
- International 2-byte font support in Annotation layers
- Support for ERDAS IMAGINE .img files larger than 2 GB

Appendix C

An Introduction to Open Source Tool for PSO/ACO2

URL: http://www.PSOACO2.sourceforgenet.com/

PSO/ACO2 uses a hybrid Particle Swarm Optimization/Ant Colony Optimisation algorithm to generate classification rules. It takes data sets of the form of ARFF. It supports binary, nominal and continuous attributes.

This algorithm still needs to consider rule interaction properly. The PSO/ACO2 optimiser itself is sound, but it may produce sub-optimal results.

Version 0.95 feature creeps into Differential Evolution support, that very popular optimiser. Code adapted from Java implementation by Mikal Keenan and Rainer Storn. Note that it probably wont work well with a mix of continuous and nominal attributes, due to problems integrating the PSO/ACO2 nominal optimiser and DE code.

In version 0.9 you can now set a test set, as well as performing the standard cross-validation procedure. Open a training set using File > Open..., then set an optional test set using the File > Open Test Set.

This program was developed at the University of Kent at Canterbury England, by Nicholas Holden, under the supervision of Alex A Freitas. It was pulled directly out of experimental code and given a front end, so expect spaghetti code and major bugs.

While using this tool you can select the number of iterations the algorithm should run for, for each rule as 100 also. For lots of continuous attribute you will see improvements with 200 iterations. You can select the number of particles, the default is 10^2, which actually equates to 100 in the algorithm.

You can select whether to use Precision or Sensitivity*Specificity (as with Ant-Miner) as the fitness function for particles. These will work better or worse depending on the data set. With tic-tac-toe precision works really well because there are “smaller” patterns, with breast-cancer sensitivity works better. Overall precision seems to perform better.

Appendix D

An Introduction to MYRA Open Source Tool

URL: http://www.MYRA.sourceforgenet.com/

Myra is a cross-platform Ant Colony Optimization framework written in Java. It provides a specialised data mining layer to support the application of ACO to classification problems, including the implementation of Ant-Miner and cAnt-Miner algorithms.

Running: Version 2.1.0 introduces a GUI interface which can be used to run Ant-Miner and cAnt-Miner algorithms, as well as setting the run parameters. Just run the 'run.sh' script to use the GUI interface. The bin directory provides shell scripts to execute Ant-Miner (antminer) and cAnt-Miner (cantminer) from the command line. There are 3 execution modes:

1 - training: the training file is specified (e.g. cantminer.sh wine.arff)

2 - test-traning: the test and training file are specified (e.g. cantminer -test wine-test.arff wine-training.arff)

3 - batch: the file pattern (wine.arff) and the directory which contains the training (TRXX_wine.arff) and test (TSXX_wine.arff) files, where XX is a numeric id, are specified (e.g. cantminer -batch wine.arff /opt/wine)

Configuration: The configuration files can be found in the conf directory.

Running the scripts without specifying any parameter will print a list of options.

Note that if you are using the GUI interface you don't need to use the configuration file.

Cross-validation: Using the cross-validation option (net.sf.myra.datamining.folds) in the configuration file will partition the training file into n folds, and at each iteration n-1 folds will be used for training and the left-out fold will be used for test. It is only necessary to specify the training file in the command-line.

Appendix E

ARFF- Attribute Relation File Format

Overview

ARFF files have two distinct sections. The first section is the Header information, which is followed the Data information.

The Header of the ARFF file contains the name of the relation, a list of the attributes (the columns in the data), and their types. An example header on the standard IRIS dataset looks like this:

```
% 1. Title: Iris Plants Database
%
% 2. Sources:
%      (a) Creator: R.A. Fisher
%      (b) Donor: Michael Marshall (MARSHALL%PLU@io.arc.nasa.gov)
%      (c) Date: July, 1988
%
@RELATION iris

@ATTRIBUTE sepallength  NUMERIC
@ATTRIBUTE sepalwidth   NUMERIC
@ATTRIBUTE petallength  NUMERIC
@ATTRIBUTE petalwidth   NUMERIC
@ATTRIBUTE class       {Iris-setosa,Iris-versicolor,Iris-virginica}
```

The Data of the ARFF file looks like the following:

```
@DATA
 5.1,3.5,1.4,0.2,Iris-setosa
 4.9,3.0,1.4,0.2,Iris-setosa
 4.7,3.2,1.3,0.2,Iris-setosa
 4.6,3.1,1.5,0.2,Iris-setosa
 5.0,3.6,1.4,0.2,Iris-setosa
 5.4,3.9,1.7,0.4,Iris-setosa
 4.6,3.4,1.4,0.3,Iris-setosa
 5.0,3.4,1.5,0.2,Iris-setosa
 4.4,2.9,1.4,0.2,Iris-setosa
 4.9,3.1,1.5,0.1,Iris-setosa
```

Lines that begin with a % are comments. The @RELATION, @ATTRIBUTE and @DATA declarations are case insensitive.

The ARFF Header Section

The ARFF Header section of the file contains the relation declaration and attributes declarations.

1. **The @relation Declaration**

The relation name is defined as the first line in the ARFF file. The format is:

```
@relation <relation-name>
```

where <*relation-name*> is a string. The string must be quoted if the name includes spaces.

2. **The @attribute Declarations**

Attribute declarations take the form of an orderd sequence of **@attribute** statements. Each attribute in the data set has its own **@attribute** statement which uniquely defines the name of that attribute and its data type. The order the attributes are declared indicates the column position in the data section of the file. For example, if an attribute is the third one declared then Weka expects that all that attributes values will be found in the third comma delimited column.

The format for the **@attribute** statement is:

```
@attribute <attribute-name> <datatype>
```

where the <*attribute-name*> must start with an alphabetic character. If spaces are to be included in the name then the entire name must be quoted. The <datatype> can be any of the four types supported by Weka:

- *numeric*
- *integer* is treated as *numeric*
- *real* is treated as *numeric*
- <nominal-specification>
- *string*
- *date* [<date-format>]

where <nominal-specification> and <date-format> are defined below. The keywords numeric, real, integer, string and date are case insensitive.

Numeric attributes

Numeric attributes can be real or integer numbers.

Nominal attributes

Nominal values are defined by providing an <nominal-specification> listing the possible values: {<nominal-name1>, <nominal-name2>, <nominal-name3>, ...}

For example, the class value of the Iris dataset can be defined as follows:

```
@ATTRIBUTE class        {Iris-setosa,Iris-versicolor,Iris-virginica}
```

Values that contain spaces must be quoted.

String attributes

String attributes allow us to create attributes containing arbitrary textual values. This is very useful in text-mining applications, as we can create datasets with string attributes, then write Weka Filters to manipulate strings (like StringToWordVectorFilter). String attributes are declared as follows:

```
@ATTRIBUTE LCC    string
```

The ARFF Data Section

The ARFF Data section of the file contains the data declaration line and the actual instance lines.

1. **The @data Declaration**

The @data declaration is a single line denoting the start of the data segment in the file. The format is:

```
@data
```

2. **The instance data**

Each instance is represented on a single line, with carriage returns denoting the end of the instance.
Attribute values for each instance are delimited by commas. They must appear in the order that they were declared in the header section (i.e. the data corresponding to the nth @attribute declaration is always the nth field of the attribute).
Missing values are represented by a single question mark, as in:

```
@data
4.4,?,1.5,?,Iris-setosa
```

Values of string and nominal attributes are case sensitive, and any that contain space must be quoted.

Sparse ARFF files

Sparse ARFF files are very similar to ARFF files, but data with value 0 are not be explicitly represented.
Sparse ARFF files have the same header (i.e @relation and @attribute tags) but the data section is different. Instead of representing each value in order, like this:

```
@data
0, X, 0, Y, "class A"
0, 0, W, 0, "class B"
```

the non-zero attributes are explicitly identified by attribute number and their value stated, like this:

```
@data
{1 X, 3 Y, 4 "class A"}
{2 W, 4 "class B"}
```

Each instance is surrounded by curly braces, and the format for each entry is: <index> <space> <value> where index is the attribute index (starting from 0).
Note that the omitted values in a sparse instance are 0, they are not "missing" values! If a value is unknown, you must explicitly represent it with a question mark (?).