

# **SUPER RESOLUTION MAPPING OF LAND COVER CLASSES**

## **A PROJECT REPORT**

Submitted by

**P.Naveen Raj - 2009115061**

**A.Yashwanth Kumar - 2009115117**

**S.B.Yuvaraja – 2009115118**

*A project report submitted to the*

## **FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING**

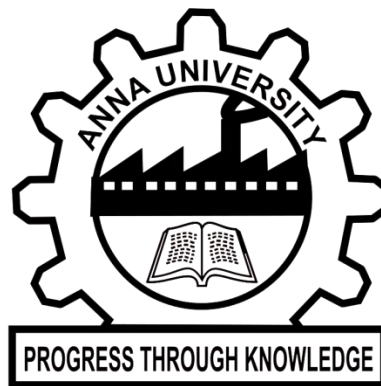
in partial fulfillment of the requirements

for the award of the degree of

## **BACHELOR OF TECHNOLOGY**

*in*

## **INFORMATION TECHNOLOGY**



DEPARTMENT OF INFORMATION SCIENCE AND TECHNOLOGY

ANNA UNIVERSITY, CHENNAI - 600025

MAY 2013

## **CERTIFICATE**

Certified that this project report titled “**SUPER RESOLUTION MAPPING OF LAND COVER CLASSES**” is the bonafide work of **P.Naveen Raj (2009115061), A.Yashwanth Kumar (2009115117), S.B.Yuvaraja (2009115118)** who carried out the project work under my supervision, for the fulfillment of the requirements for the award of the degree of Bachelor of Engineering in Information Technology. Certified further that to the best of my knowledge, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion on these are any other candidates.

**Place:** Chennai

**Date:** 14/05/2013

**Dr. K. Vani**

Associate Professor,

Department of Information Science and Technology,

Anna University,

Chennai – 600025.

## **COUNTERSIGNED**

**Dr. A.Kannan**

**Professor and Head of the Department,**

Department of Information Science and Technology,

Anna University Chennai,

Chennai – 600025.

## ACKNOWLEDGEMENT

We express our deep gratitude to our guide, **Dr. K. Vani** for guiding us through every phase of the project. We appreciate her thoroughness, tolerance and ability to share her knowledge with us. We thank her for being easily approachable and quite thoughtful. Apart from adding her own input, she has encouraged us to think on our own and give form to our thoughts. We owe her for harnessing our potential and bringing out the best in us. Without her immense support through every step of the way, we could never have it to this extent.

We are extremely grateful to **Dr.A.Kannan**, Professor and Head of the Department of Information Science and Technology, Anna University, Chennai 600025, for extending the facilities of the Department towards our project and for his unstinting support.

We express our thanks to the panel of reviewers **Dr.S.Swamynathan**, **Dr.S.SendhilKumar**, **Dr.K.Vidhya**, and **Mrs.E.Uma** for their valuable suggestions and critical reviews throughout the course of our project.

We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

**P.Naveen Raj**

**A.Yashwanth Kumar**

**S.B.Yuvaraja**

## **ABSTRACT**

Satellite images usually cover large area that will have various classes (grass, land, buildings ...etc.). So, these large areas which contain various classes will be seen as a single pixel in the satellite image. The project focuses on obtaining the exact location of the different classes present in the single pixel. Soft classification of the input image is performed to identify the spectral components (the percentage of classes present in a single pixel) of the image. PSO (Particle Swarm Optimization) is applied for each soft classified class image after calculating the attraction between neighboring pixels and correlation between the sub-pixels. Results of all the classes after PSO (Particle Swarm Optimization) are merged to obtain the final classified map. ACO (Ant Colony Optimization) is applied for all the soft classified images and results of all the class images are merged to obtain the final classified map. Classified maps of the ACO (Ant Colony Optimization) and PSO (Particle Swarm Optimization) are compared.

## திட்டப்பணிச்சுருக்கம்

செயற்கைக்கோள் மூலம் கைப்பற்றப்பட்டபடம் பொதுவாக பல்வேறு வகுப்புகள் (புல், நிலம், கட்டிடங்கள் ... முதலியன) உள்ளது. பல்வேறு வகுப்புகள் கொண்ட மிக பெரிய பகுதியில் செயற்கைக்கோள் படத்தில் ஒற்றை பிக்சல் பார்க்கப்படுகிறது. திட்ட ஒற்றை பிக்சல் வெவ்வேறு வகுப்புகள் சரியான இடம் பெற கவனம் செலுத்துகிறது. உள்ளீடு படத்தை மென்மையான வகைப்பாடு படத்தை ஸ்பெக்ட்ரம் கூறுகளை அடையாளம் காண செய்யப்படுகிறது. PSO (துகள் திரள் தேர்வுமுறை) துணை பிக்சல்கள் இடையே அண்டை பிக்சல்கள் மற்றும் தொடர்பு இடையே ஈர்ப்பு கணக்கிட்டு பின்னர் ஒவ்வொரு மென்மையான விளம்பரங்கள் வர்க்கம் படத்தை செலுத்தப்படுகிறது. PSO பின்னர் அனைத்து வகுப்புகள் (துகள் திரள் தேர்வுமுறை) முடிவு இறுதி வரி வரைபடம் பெற இணைக்கப்பட்டது. ACO (எறும்பு கூட்ட தேர்வுமுறை) அனைத்து மென்மையான விளம்பரங்கள் படங்கள் மற்றும் அனைத்து வர்க்கம் படங்களை முடிவுகளை இறுதி வரி வரைபடம் பெற சேர்க்கப்படும் செலுத்தப்படுகிறது. ACO (எறும்பு கூட்ட தேர்வுமுறை) மற்றும் PSO (துகள் திரள் தேர்வுமுறை) என்ற அறிவிப்பு வரைபடங்கள் ஒப்பிடுகையில்.

## TABLE OF CONTENTS

<b>ABSTRACT</b>	<b>IV</b>
<b>ABSTRACT (TAMIL)</b>	<b>V</b>
<b>LIST OF FIGURES</b>	<b>IX</b>
<b>LIST OF TABLES</b>	<b>XI</b>
<b>LIST OF ABBREVIATIONS</b>	<b>XII</b>
<b>1. INTRODUCTION</b>	<b>1</b>
1.1 PROBLEM DOMAIN	1
1.1.1 Types Of Classification	1
1.1.2 Super Resolution Mapping	2
1.2 SCOPE OF THE PROJECT	4
1.3 PROBLEM DEFINITION	4
1.4 ORGANIZATION OF THIS REPORT	4
<b>2. LITERATURE STUDY</b>	<b>6</b>
2.1 TWO STEP APPROACH	6
2.1.1 Linear Unmixing Model	6
2.1.2 Maximum Spatial Dependence Model	6
2.2 HOP-FIELD NEURAL NETWORK	7
2.3 PARTICLE SWARM OPTIMIZATION	8
2.3.1 Sub-pixel/Pixel Spatial Attraction Model (SPSAM)	8
2.3.2 Post-Processing Spsam Results	9
2.4 LIMITATIONS OF THE EXISTING WORK	10
2.4.1 Limitations	10
2.4.2 Proposed System	10

<b>3. SYSTEM DESIGN</b>	<b>10</b>
3.1 DESIGN REQUIREMENTS	11
3.2 SYSTEM ARCHITECTURE	12
3.2.1 Soft Classification	12
3.2.2 Sub-Pixel Mapping	12
3.3 MODULE DESCRIPTIONS	13
<b>4. SYSTEM DEVELOPMENT</b>	<b>15</b>
4.1 ALGORITHM FOR SUB-PIXEL MAPPING	15
4.1.1 Spatial Dependence Theory	15
4.1.2 Sub-pixel / Pixel Spatial Attraction Model (SPSAM)	16
4.2 ALGORITHM FOR POST-PROCESSING ON SPSAM RESULTS	17
4.2.1 Pixel Swapping Algorithm	18
4.2.2 An Objective Function	19
4.3 THE PARTICLE SWARM OPTIMIZATION ALGORITHM	21
4.3.1 General PSO Algorithm	21
4.3.2 PSO After SPSAM Process	22
4.4 ANT COLONY OPTIMIZATION	26
4.4.1 Overview	28
4.4.2 Pheromone Update	28
<b>5. RESULTS AND DISCUSSIONS</b>	<b>28</b>
5.1 TWO CLASS SYNTHETIC IMAGE	28
5.1.1 PSO ALGORITHM	29
5.1.2 ANT COLONY OPTIMIZATION ALGORITHM	30

5.1.3 COMPARISON OF PSO AND ANT COLONY OPTIMIZATION ALGORITHM	30
5.2 THREE CLASS SYNTHETIC IMAGE	31
5.2.1 PSO ALGORITHM	32
5.2.2 ANT COLONY OPTIMIZATION ALGORITHM	33
5.2.3 COMPARISON OF PSO AND ANT COLONY OPTIMIZATION ALGORITHM	33
5.3 THREE CLASS SATELLITE IMAGE	34
5.3.1 PSO ALGORITHM	37
5.3.2 ANT COLONY OPTIMIZATION ALGORITHM	38
5.3.3 COMPARISON OF PSO AND ANT COLONY OPTIMIZATION ALGORITHM	39
5.4 FOUR CLASS SATELLITE IMAGE	40
5.4.1 PSO ALGORITHM	43
5.4.2 ANT COLONY OPTIMIZATION ALGORITHM	44
5.4.3 COMPARISON OF PSO AND ANT COLONY OPTIMIZATION ALGORITHM	45
<b>6. CONCLUSION</b>	46
6.1 OVERALL CONCLUSION	35
6.2 FUTURE WORK AND OPTIMIZATION	46
<b>7. REFERENCES</b>	47



## LIST OF FIGURES

<b>FIGURE NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
1. 1	Difference between Hard and Soft Classification	2
1. 2	A Mixed Pixel	3
2. 1	Super Resolution Mapping using HNN	7
3. 1	Block Diagram of Soft Classification	11
3. 2	The Block Diagram of Sub-Pixel Mapping with SPSAM and PSO	12
4. 1	Demonstration of Spatial Dependence Theory	15
4. 2	Flow Chart of PSO Algorithm	25
5. 1	Two Class Synthetic Image as Input (70x70)	28
5. 2	Presence of Class 1 (Yellow)	28
5. 3	Presence of Class 1 (Blue)	29
5. 4	Result of two class synthetic image after applying PSO Algorithm	29
5. 5	Result of two class synthetic image after applying ACO Algorithm	30
5. 6	Three Class Synthetic Image as Input (10x10)	31
5. 7	Presence of Class 1 (Black)	32
5. 8	Presence of Class 1 (Blue)	32
5. 9	Presence of Class 1 (White)	32
5. 10	Result of three class synthetic image after applying PSO Algorithm	32
5. 11	Result of three class synthetic image after applying ACO Algorithm	33
5. 12	Three Class Satellite Image as Input (227x216)	34
5. 13	Presence of Class 1 (Building)	35
5. 14	Presence of Class 2 (Grass)	35
5. 15	Presence of Class 3 (Land)	36
5. 16	Result of three class satellite image after applying PSO Algorithm	37
5. 17	Result of three class satellite image after applying ACO Algorithm	38
5. 18	Four Class Satellite Image as Input (187x195)	40
5. 19	Presence of Class 1 (Dark Blue)	40
5. 20	Presence of Class 2 (Blue)	41
5. 21	Presence of Class 3 (Orange)	41

5. 22	Presence of Class 4 (Red)	42
5. 23	Result of Four class satellite image after applying PSO Algorithm	43
5. 24	Result of four class satellite image after applying ACO Algorithm	44

## LIST OF TABLES

TABLE NO.	TITLE	PAGE NO.
5. 1	Comparison of PSO And ACO Algorithm for SRM 2 class synthetic image	
	<b>Error! Bookmark not defined.</b>	
Table 5. 1	Comparison of PSO And ACO Algorithm for SRM for 2 class synthetic image	31
Table 5. 1	Comparison of PSO And ACO Algorithm for SRM for 2 class synthetic image	31

## LIST OF ABBREVIATIONS

<b>PSO</b>	Particle Swarm Optimization
<b>SPSAM</b>	Sub Pixel Spatial Attraction Mapping
<b>ACO</b>	Ant Colony Optimization
<b>MSI</b>	Multi Spectral Image
<b>SPM</b>	Sub Pixel Mapping
<b>SRM</b>	Super Resolution Mapping

# **1. INTRODUCTION**

## **1.1 PROBLEM DOMAIN**

Maps are classified based on images taken from satellite. This is a very complex task and involves many chances of errors. If the sensor is capable of only taking low resolution images (for example: 72 meters per pixel), then the mapping process becomes less accurate.

### **1.1.1 TYPES OF CLASSIFICATION**

There are two types of classification,

- Hard Classification and
- Soft Classification.

## **HARD CLASSIFICATION**

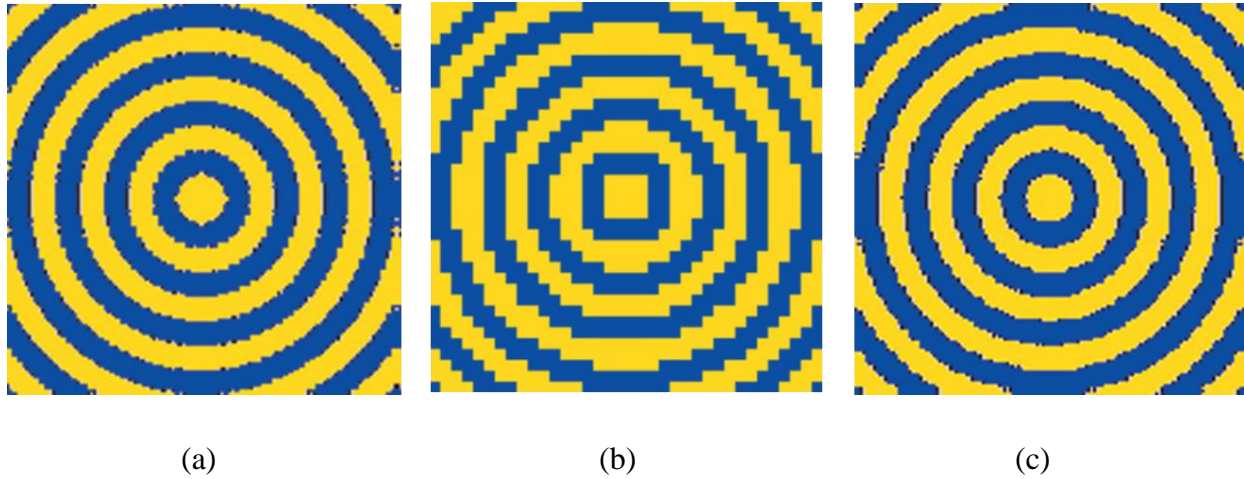
This is an approximate classification. This kind of classification maps one pixel to only a single color (or) to only a single land cover class. But this kind of classification suffers from some serious disadvantages.

If the satellite image is of poor resolution, then we cannot map a single pixel to exactly a single class. Consider for example, a poor resolution map of 73 meters per pixel. This pixel which covers an area of 72x72 square meters may not have only one land cover class. It is possible that it may contain mixed land cover classes.

## **SOFT CLASSIFICATION**

This method finds out the area proportion of each land cover class in a particular pixel. Using that proportion we can be able to generate an accurate soft

classified image for various classes. This classification addresses the disadvantages of Hard Classification but this also have problem with accurate mapping. Soft Classification can produce accurate results only for the higher resolution images.



**Figure 1. 1 Difference between Hard and Soft Classification**

Figure 1.1 – (a) is the original synthetic image.

Figure 1.1 – (b) is the Hard Classified image

Figure 1.1 – (c) Soft Classified image

### **1.1.2 SUPER RESOLUTION MAPPING**

Super-resolution mapping is a technique which allows mapping at the sub-pixel scale. Each pixel in the original satellite imagery is scaled by a scale factor and then the map is constructed. The soft classified image is fed as input to an Super Resolution Mapping (SRM) Algorithm. Using the percentage of land cover class present in each pixel, the exact spatial location of the land cover within the pixel is found. Various methods and algorithms have been adopted to produce SRMed image with more accuracy.

Several Super Resolution mapping techniques have been proposed such as spatial dependence maximization (Atkinson, 1997), sub-pixel per-classification (Aplin and Atkinson, 2001), linear optimization technique (Verhoeye and De Wulf, 2002), Hopfield neural network (Tatem et al., 2001a; Tatem et al., 2001b; Tatem et al., 2002), two-point histogram optimization (Atkinson, 2008), genetic algorithms (Mertens et al., 2003), wavelet coefficients prediction using feed-forward neural networks (Mertens et al., 2004) and pixel swapping (Thornton et al., 2006).

## SPECTRAL UNMIXING



**Figure 1. 2 A Mixed Pixel**

In a mixed pixel, there may be many land cover classes present. The Fraction of these classes is identified by a technique called Spectral Unmixing.

In Figure 1.2, the entire box represents one pixel. This pixel has three land cover classes, black, blue and grey. Black occupies 25% of the space while Grey occupies 35% of the entire space and the remaining 40% is represented by Blue class. This fractional image is identified by this Spectral Unmixing process. This process just identifies the fraction of land cover classes and not the spatial location of them.

## **SUB-PIXEL MAPPING**

Sub-pixel mapping (SPM) is a technique to predict spatial locations of land cover classes within mixed pixels in remotely sensed imagery. After the spectral unmixing process, the fraction image is fed into an algorithm for sub-pixel mapping. Then the algorithm makes an optimal arrangement of these fractional classes in that particular pixel.

### **1.2 SCOPE OF THE PROJECT**

Satellite images usually cover large area that will have various classes (grass, land, buildings ...etc.). So large area which contains various classes will be seen as the single pixel in the satellite image. The project focuses on obtaining the exact location of the different classes in the single pixel. Input satellite can be of lower resolution output will be classified map with higher accuracy .

### **1.3 PROBLEM DEFINITION**

The Satellite image obtained from a poor sensor is low in resolution (eg. 72\*72 meters per pixel). In such images many information are lost. A single pixel can be represented using a single color only. But this is not the case in real time. A 72 square meter area may not contain a single class (A particular land cover), instead it may contain many classes like building, farms, lakes, rivers, grass lands etc.

Also Super Resolution Mapping of extremely large images can take more time. Say for example a 120x120 image can take at most 5 hours to complete.



### **1.3 ORGANIZATION OF THIS REPORT**

The report is a documentation of the phases of the project in order namely the literature study, system design and implementation details .

The outline of the report is as follows.

**Chapter 2** discusses about the literature survey

**Chapter 3** discusses the System Design

**Chapter 4** explains the Algorithmic Implementations

**Chapter 5** explains the Results and Discussions

**Chapter 6** summarizes conclusion and directions for future work.

## **2. LITERATURE STUDY**

A Number of Approaches has been adopted for the problem of Super Resolution Mapping. Among those the following have been studied and referred.

### **2.1 TWO STEP APPROACH**

The two-step approach first estimates fraction images by spectral unmixing and then inputs fraction images into an SPM algorithm to generate the final sub-pixel land cover map. A shortcoming of this approach is that the information about the credibility of fraction images is not considered. In this letter, we proposed a general framework of SPM which is directly applied to original coarse resolution remotely sensed imagery by integrating spectral and spatial information. Based on the proposed framework, the linear unmixing model and the maximal spatial dependence model were combined to construct a novel SPM model aiming to minimize the least squares error of spectral signature and make the sub-pixel land cover map spatially smooth, simultaneously[2].

#### **2.1.1 LINEAR UNMIXING MODEL**

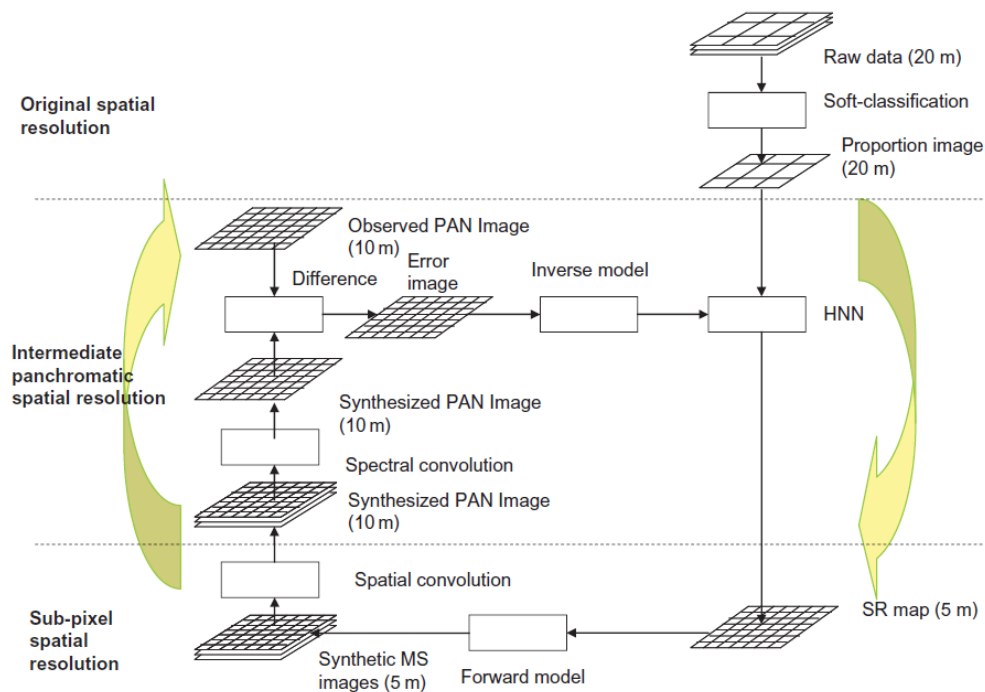
In general, LUM views spectral signatures of each mixed pixel as being made up of a weighted linear sum of spectral signatures of end members within that pixel. The weights are determined by the relative area proportions of each end member[2].

#### **2.1.2 MAXIMUM SPATIAL DEPENDENCE MODEL**

This methodology aims to minimize a weighted sum of Spectral and Spatial terms. Maximizing the land cover spatial dependence is identical to making the resulting land cover map spatially smooth. Using both the spectral and spatial terms the proper arrangement of classes inside a pixel is identified[2].

## 2.2 HOP-FIELD NEURAL NETWORK

This process is an iterative approach to spectrally map a particular region. Figure 2.1 is a graphical depiction of the method proposed for incorporating a PAN image into super-resolution mapping using a HNN. A proportion image is obtained by Soft Classification and the proportion image is fed to a Hop-Field Neural Network (HNN)[4]. The proportion images are then used to produce the sub-pixel land-cover class at the first iteration using the HNN (each sub-pixel is represented by a neuron in the HNN). From the super-resolution map at the first iteration, an estimated MS image (at the PAN image spatial resolution) is then produced using a forward model and spatial convolution. The estimated MS image is then convolved spectrally to create a synthetic PAN image. By comparing the observed and synthetic PAN images, a value is produced for all neurons covered by the same pixel in the PAN image to make the synthetic PAN converge to the observed PAN image[1].



**Figure 2. 1 Super Resolution Mapping using HNN**

## **2.3 PARTICLE SWARM OPTIMIZATION**

### **2.3.1 SUB-PIXEL/PIXEL SPATIAL ATTRACTION MODEL (SPSAM)**

This Algorithm directly estimates the spatial location of a mixed pixel. It uses the attraction between sub-pixels and its neighboring pixels. As such, the algorithm requires no iteration to achieve the spatial allocation of sub-pixel classes. The advantage of this algorithm is that it is suitable for real-time processing. Particularly, when dealing with situations involving a large scale factor, it is fast in obtaining the SPM results. However, the algorithm fails to consider adequately the correlation between sub-pixels. When the scale factor is large, there can be many isolated pixels and much noise. If we apply the algorithm to real remote-sensing imagery, where the spatial distribution of each class is diverse and changeable, there may be many saw tooth shaped edges in the SPM results[3].

## **CONCOCTION**

**Particle Swarm Optimization (PSO)** is a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. PSO optimizes a problem by having a population of candidate solutions, here dubbed particles, and moving these particles around in the search-space according to simple mathematical formulae over the particle's position and velocity. Each particle's movement is influenced by its local best known position and is also guided toward the best known positions in the search-space, which are updated as better positions are found by other particles. This is expected to move the swarm toward the best solutions[3].

### **2.3.2 POST-PROCESSING SPSAM RESULTS**

We can see that the SPSAM algorithm directly estimates the class of sub-pixels according to the class proportion of its neighboring pixels. However, the algorithm fails to adequately consider the correlation between sub-pixels, and thus it may lead to poor performance for SPM. The following methods can be used to post-process[3].

- Pixel Swapping Algorithm.
- An Objective Function.
- Particle Swapping Optimization Algorithm

In this project PSO Algorithm is used to post process SPSAM results.

## **2.4 LIMITATIONS OF THE EXISTING WORK**

### **2.4.1 LIMITATIONS**

- In the existing work Attraction with nearest pixel is not considered
- Correlation between the classes with in the pixel is not considered .
- It can provide only local optimization

### **2.4.2 PROPOSED SYSTEM**

The proposed system overcomes the limitations by,

- Attraction with nearest pixel is considered
- Correlation between the classes with in the pixel is considered .
- Particle Swarm technique and Ant Colony Techniques are applied

### 3. SYSTEM ARCHITECTURE

This chapter discusses the design methodology adopted to implement the project. First, the overall methodology adopted is explained followed by the detailed design of the components.

#### 3.1 DESIGN REQUIREMENTS

- **Hardware:** A system with 2GB RAM or above.
- **Tools:**
  - MatLab
  - ENVI
- **Operating System:** Windows

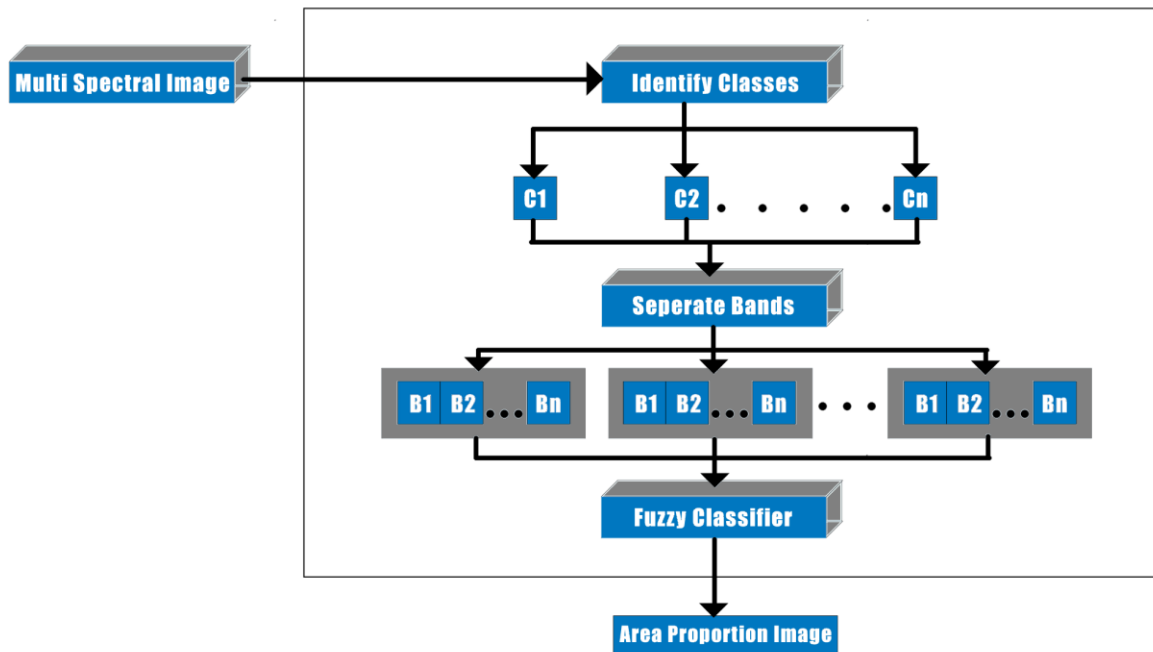
#### 3.2 SYSTEM ARCHITECTURE

The following is the Architecture Diagram of the entire project. It has been split into two parts. One for the Soft Classification step and the other for the Sub-Pixel Mapping step.

##### 3.2.1 SOFT CLASSIFICATION

Soft Classification is the technique to identify the spectral components of the various classes and to obtain the classified map for each class with the same resolution of the image. Figure 3.1 depicts the architecture of the first step in the two step process, Soft Classification after the soft classification only images can be processed for SRM (Super Resolution Mapping).

The Multi Spectral (MS) Image is given as input for the Soft Classification Algorithm and an Area Proportion Image is obtained as output. In the first step of Soft Classification the classes are identified and then separated.

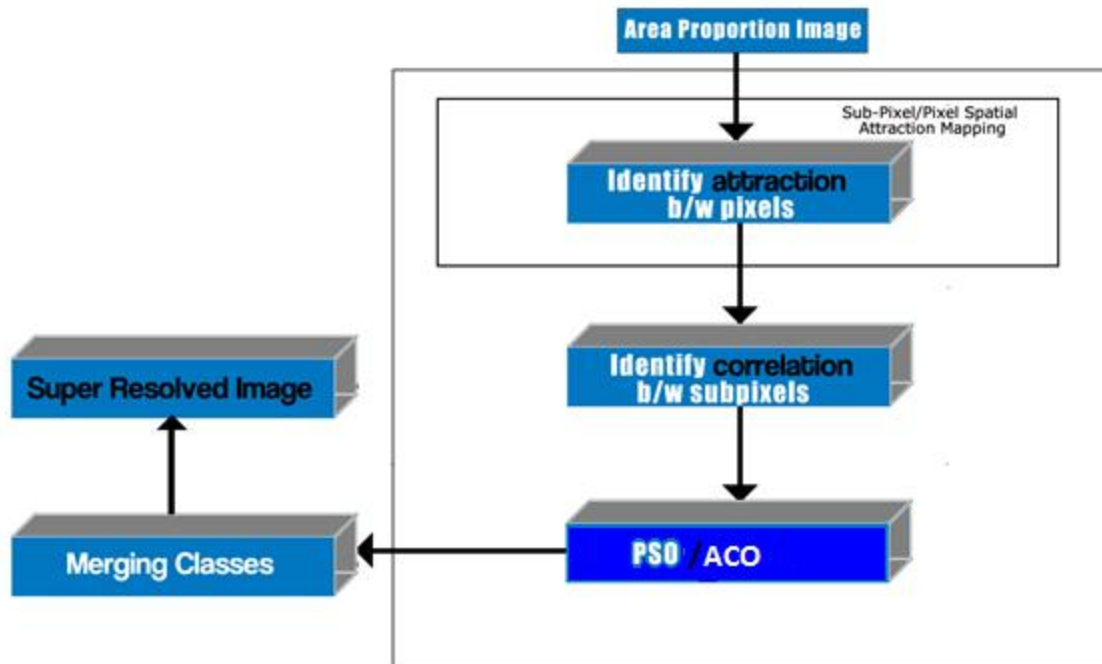


**Figure 3. 1 Block Diagram of Soft Classification**

In the next step, for each class bands are separated and given as input to a Fuzzy Classifier. This identifies the proportion of a particular class in a particular pixel. Thus the Area Proportion Image is obtained for each class. The Area Proportion image is a monochromatic image with white representing the complete presence of the class and black being the complete absence of the class.

### 3.2.2 SUB-PIXEL MAPPING

After the soft classification of the image the area proportion image for the each class is processed to obtain the classified map for each class and classified map for the various class are combined. Figure 3.2 depicts the architecture diagram of the second step of the two step process, Sub-Pixel Mapping using PSO (Particle Swarm Optimization) technique or ACO (Ant Colony Optimization).



**Figure 3. 2 The Block Diagram of Sub-Pixel Mapping with SPSAM and PSO/ACO**

The Area Proportion image obtained from the soft classification process is the input for Sub-Pixel Mapping. The output is the final Super Resolved Image. In this Particle Swarm Optimization Algorithm is used along with SPSAM process to spatially identify the classes.

### 3.3 MODULE DESCRIPTIONS

#### MODULE 1: SOFT CLASSIFICATION

- Input : Low Resolution Satellite Image
- Output : Classified Map for each selected class
- Methodology :

Low Resolution Satellite Image is chosen in the ENVI tool and different classes are selected using ROI tool and the Spectral Unmixing is done using the LUM (linear Un-mixing Model) present in the



ENIV tool and the soft classified image for the various class with the same resolution as input image is obtained

## **MODULE 2: SPSAM (Sub Pixel Spatial Attraction Mapping )**

- Input : Area Proportion Image
- Output : Matrix ( Attraction Matrix )
- Methodology :

Area Proportion image of the particular class is selected and the attraction with the nearest pixel is calculated and the attraction matrix is obtained as the output of this module

## **MODULE 3: Calculating Correlation between sub pixels**

- Input : output of module 2
- Output : Correlation Matrix
- Methodology :

Matrix obtained after altering with calculated attraction is applied as input and the correlation with the sub pixel is calculated and the Correlation matrix is obtained as the output of this module

Area Proportion image of the particular class is selected and the attraction with the nearest pixel is calculated and the attraction matrix is obtained as the output of this module

## **MODULE 4: Applying PSO Algorithm**

- Input : Matrix obtained after altering with correlation
- Output : Classified Map with better resolution
- Methodology :

Matrix obtained after altering with calculated Correlation is applied as input and the correlation with the sub pixel is calculated and the PSO Algorithm is applied and Classified map for the particular class with higher accuracy is obtained as the output

### **MODULE 5: Applying ACO Algorithm**

- Input : Area Proportion Image
- Output : Classified Map with better resolution
- Methodology :

Area Proportion image is applied as input and the PSO Algorithm is applied and Classified map for the particular class with higher accuracy is obtained as the output

### **MODULE 6: Merging of various classes classified image**

- Input : result of PSO / ACO
- Output : SRM ( Super Resolution Mapping ) image
- Methodology :

Result of PSO/ ACO is applied as input and various images are merged and we obtain the super resolution mapped image

### **MODULE 7: Comparing PSO and ACO**

- Input : SRM after PSO and ACO
- Output : quantitative result
- Methodology :

SRM after PSO and ACO is taken as the input and the images are compared with the original image . In the synthetic image the advantages and disadvantage of the this these techniques can be compared easily

## 4. SYSTEM DEVELOPMENT

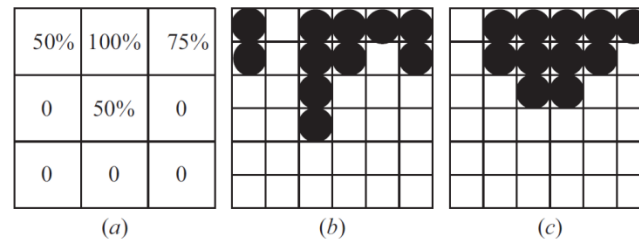
There are three algorithms uses in this project, one for Sup-Pixel Mapping and the two others for post-processing on the results obtained from Sub-Pixel Mapping.

### 4.1 ALGORITHM FOR SUB-PIXEL MAPPING

Atkinson (1997) initially proposed the concept of SPM and the spatial dependence theory with the assumption that land cover is spatially dependent both within and between pixels; that is, compared with distant pixels, neighboring pixels are more likely to be of the same land-cover class. Mertens *et al.* (2006) applied a Sub-Pixel/Pixel Spatial Attraction model (SPSAM) that realized the spatial dependence theory directly. This SPSAM is used in this project.

#### 4.1.1 SPATIAL DEPENDENCE THEORY

A simple representation of SPM is given in figure 4.1. It shows a raster grid of  $3 \times 3$  coarse spatial resolution pixels with associated proportions of one land-cover class in figure 4.1(a), which can be obtained by spectral unmixing. A single coarse resolution pixel is divided into  $2 \times 2$  sub-pixels, each corresponding to a 25% area of coarse low-resolution pixels. From the proportions, the number of sub-pixels belonging to this class can be calculated.



**Figure 4. 1 Demonstration of Spatial Dependence Theory**

For example, the fraction of 75% corresponds to three sub-pixels. One possible arrangement is shown in figure 4.1(b), where the black circles represent sub-pixels of the class. Obviously, its spatial structure conflicts with expectations of spatial dependence. This is because, in nature, the land cover coming from the same classes is more likely to stay together. Another solution is presented in figure 4.1(c). It can be seen that, compared with figure 4.1(b), the spatial dependence both within and between the coarse pixels in figure 4.1(c) is much stronger. Therefore, figure 4.1(c) is a more reasonable SPM result.

#### 4.1.2 SUB-PIXEL / PIXEL SPATIAL ATTRACTION MODEL (SPSAM)

Mertens *et al.* (2006) applied SPSAM and realized the spatial dependence theory in a simple and effective way. In SPSAM, attractions between each sub-pixel within a coarse resolution pixel and its neighbour pixels are calculated in order to determine the spatial distribution of sub-pixels per class. Assume  $p_{ij}$  is a sub-pixel in pixel  $P_{ab}$  and  $P_k$  is one of  $P_{ab}$ 's neighbors. Then the attraction from class  $c$  for sub-pixel  $p_{ij}$  is calculated as

$$\text{Attraction}_{c,p_{ij}} = \sum_{k=1}^n \lambda_k F_c(P_k), \quad (4.1)$$

where  $n$  is the total number of neighbors (in this article,  $n$  is set to 8) and  $F_c(P_k)$  is the fraction value of the  $k$ th neighboring pixel  $P_k$  for class  $c$ .  $\lambda_k$  is the measurement of spatial dependence and is calculated as

$$\lambda_k = [d(p_{ij}, P_k)]^{-1}, \quad (4.2)$$

where  $d(p_{ij}, P_k)$  is the Euclidean distance between geometric centers of sub-pixel  $p_{ij}$  and its neighboring pixel  $P_k$  and is calculated as

$$d(p_{ij}, P_k) = \sqrt{(x_{ij} - X_k)^2 + (y_{ij} - Y_k)^2}. \quad (4.3)$$

Finally,  $\text{Attraction}_{c,p_{ij}}$  for each class can be used for the assignment of sub-pixels to the different classes: sub-pixels with highest attractions are assigned first. The SPSAM algorithm is much easier for the situation of two land-cover classes. Suppose there are two classes A and B. We can conduct the SPSAM as follows: within a coarse resolution pixel  $P_{ab}$ , each  $p_{ij}$ 's  $\text{Attraction}_{c,p_{ij}}$  is first calculated by equations (1)–(3) and then the values are ranked in order. Finally, the  $FA(P_{ab}) S$  ( $FA(P_{ab})$  is the fraction value of class A within pixel  $P_{ab}$  and  $S$  is a scale factor) sub-pixels with high values of  $\text{Attraction}_{c,p_{ij}}$  are assigned to class A while the residual ones are assigned to class B (Mertens *et al.* 2004). This approach can also be extended to multiple classes.

## 4.2 ALGORITHM FOR POST-PROCESSING ON SPSAM RESULTS

We can see that the SPSAM algorithm directly estimates the class of sub-pixels according to the class proportion of its neighboring pixels. However, the algorithm fails to adequately consider the correlation between sub-pixels, and thus

it may lead to poor performance for SPM. Shen *et al* (2009) adopted a method that created a modified pixel-swapping algorithm (MPS) with initialization from SPSAM. Although its original purpose was to improve the PSA, it can be regarded as a method that enhances the performance of SPSAM as well by fully considering the correlation between sub-pixels after SPSAM. In this section, the principle of PSA is described first, and then an objective function is proposed based on this principle. The objective function will be used for the searching process by PSO in the next section and will also be used as a post-processing method after the SPSAM procedure.

#### 4.2.1 PIXEL SWAPPING ALGORITHM

Atkinson (2001, 2005) proposed the PSA. The objective was to vary the spatial arrangement of the sub-pixels in such a way that the spatial correlation between neighboring sub-pixels (both within and, perhaps more importantly, between pixels) would be maximized. Two classes are taken into account: ‘1’ and ‘0’. For each sub-pixel  $p_{ij}$ , the attraction caused by all its neighboring sub-pixels is calculated as

$$O_{p_{ij}} = \sum_{k=1}^n \lambda_k z(p_k), \quad (4.4)$$

where  $n$  is the total number of neighbors,  $z(p_k)$  is the binary value of the class for  $p_k$  and  $\lambda_s$  is the measurement of spatial correlation between sub-pixels and is calculated as

$$\lambda_k = \exp \left( \frac{-d_k}{a} \right), \quad (4.5)$$

where  $a$  is a non-linear parameter of the exponential model and  $d_k$  is the Euclidean distance between geometric centers of sub-pixel  $p_{ij}$  and its neighboring sub-pixel  $p_k$ , as in equation (3). We can see that  $O_{p_{ij}}$  indicates the attraction for  $p_{ij}$  caused by class ‘1’ from its neighboring sub-pixels. After each  $p_{ij}$ ’s  $O_{p_{ij}}$  has been calculated, the following two-stage process is conducted for each pixel  $P_{ab}$ :

**Stage 1:** Rank all  $O_{p_{ij}}$  in decreasing order. As a result, a corresponding sequence (sequence <sub>$a$</sub> ) is generated, which is composed of the binary values of the class for  $p_{ij}$ .

**Stage 2:** Identify the first ‘0’ from left and the first ‘1’ from right in the sequence. If the ‘0’ locates before the ‘1’, then the two values are swapped to increase the total attraction inside  $P_{ab}$ . Otherwise, no change is made.

The above two-stage process is repeated iteratively. The process can be stopped either at a fixed number of iterations or when little change is made. PSA was first used to work for two classes. It can be extended for multiple classes.

#### 4.2.2 AN OBJECTIVE FUNCTION

Having studied the feature of sequence <sub>$a$</sub> , we can construct an objective function that makes the correlation between sub-pixels reach maximum after the SPSAM process. Again the two classes are regarded: ‘1’ and ‘0’, i.e.  $p_{ij}$  takes 1 or

0 for an unmixed pixel  $P_{ab}$ . The objective function with the constraint condition can be written as

$$\begin{aligned} \max J &= \sum_{i=1}^S \sum_{j=1}^S \left[ p_{ij} \alpha \mathbf{N}_{p_{ij}} + (1 - p_{ij}) \alpha \left( \mathbf{I}^T - \mathbf{N}_{p_{ij}} \right) \right] \\ \text{s.t.} \quad & \sum_{i=1}^S \sum_{j=1}^S p_{ij} = F(P_{ab}) S^2, \end{aligned} \tag{4.6}$$

where  $F(P_{ab})$  is the proportion of class ‘1’ in  $P_{ab}$  and  $\mathbf{N}_{p_{ij}}$  is a vector composed of the class values of  $p_{ij}$ ’s neighboring sub-pixels:  $\mathbf{N}_{p_{ij}} = [p_{i-1,j-1}, p_{i-1,j}, p_{i-1,j+1}, p_{i,j-1}, p_{i,j+1}, p_{i+1,j-1}, p_{i+1,j}, p_{i+1,j+1}]^T$ . It should be noticed that when  $i = 1$  or  $j = 1$ , the subscript of some of  $\mathbf{N}_{p_{ij}}$ ’s elements is equal to 0.  $d_k$  ( $k = 1, 2, \dots, 8$ ) is the distance between  $p_{ij}$  and the  $k$ th neighboring sub-pixel.  $\mathbf{1}^T$  is a vector composed of elements of 1’s. If  $p_{ij}$  belongs to class ‘1’ ( $p_{ij} = 1$ ), the attraction for the sub-pixel caused by ‘1’ from its neighbors can be calculated by the first term of equation (6). In contrast, if  $p_{ij}$  belongs to class ‘0’ ( $p_{ij} = 0$ ), the attraction for the sub-pixel caused by ‘0’ from its neighbors can be calculated by the second term of equation (6). As a result, inside  $P_{ab}$ , the total attraction for all of the sub-pixels caused by the same class can be calculated by equation (6). When the correlation between sub-pixels is maximized,  $J$  reaches maximum. Hence, by solving equation (6), we can get the most suitable distribution of all sub-pixels within the mixed pixel by evaluating all possible configurations and selecting the one that makes  $J$  reach maximum. However, it mainly works well for small images with a small scale factor (Mertens *et al.* 2003b). With a large scale factor, the number of combinations of possible spatial distribution increases dramatically and the computational load may become unrealistic. For this reason, there is a need to



introduce an effective optimization algorithm to handle the problem. So a PSO technique is used in this project and it is discussed in detail in the next section.

## 4.3 THE PARTICLE SWARM OPTIMIZATION ALGORITHM

### 4.3.1 GENERAL PSO ALGORITHM

The initial PSO is operated in the continuous-valued space, where coordinates for every particle's position and velocity are coded as real numbers in each dimension. The PSO algorithm that operates on discrete binary variables is Binary PSO. In the binary PSO, coordinates for position will take on a 1 or 0 value, but it is not for velocity. The mathematical description of the binary PSO is as follows. Each particle  $i$  consists of two vectors, position and velocity, which can be represented by  $\mathbf{X}_i = [X_{i1}, X_{i2}, \dots, X_{im}]$  and  $\mathbf{V}_i = [V_{i1}, V_{i2}, \dots, V_{im}]$ , respectively, where  $m$  is the dimension of the search space. One position vector  $\mathbf{X}_i$  corresponds to one solution to the optimization problem. The position and velocity are updated as

$$V_{ip}(t) = wV_{ip}(t-1) + c_1\text{rand}_1[P_{ip}(t-1) - X_{ip}(t-1)] + c_2\text{rand}_2[G_p(t-1) - X_{ip}(t-1)], \quad (4.7)$$

$$\text{sig}[V_{ip}(t)] = \frac{1}{1 + \exp[-V_{ip}(t)]}, \quad (4.8)$$

$$X_{ip}(t) = \begin{cases} 1, & \text{rand} < \text{sig}[V_{ip}(t)] \\ 0, & \text{rand} \geq \text{sig}[V_{ip}(t)] \end{cases}, \quad (4.9)$$

where  $i = 1, 2, \dots, M$ , with  $i$  representing the  $i^{\text{th}}$  particle and  $M$  the size of the swarm;  $t$  is the number of generations; sig means sigmoid function;  $V_{ip}(t)$  and

$X_{ip(t)}$  mean the coordinate values of the  $p^{\text{th}}$  dimension of velocity  $\mathbf{V}_i$  and the  $p^{\text{th}}$  dimension of spatial position  $\mathbf{X}_i$  at the  $t^{\text{th}}$  generation;  $P_{ip}(t-1)$  is the coordinate value of the  $p^{\text{th}}$  dimension of  $\mathbf{P}_i(t-1)$  that indicates the best solution for particle  $i$  from the first to the  $(t-1)^{\text{th}}$  generation;  $G_p(t-1)$  is the coordinate value of the  $p^{\text{th}}$  dimension of  $\mathbf{G}(t-1)$  that indicates the best position or solution in the whole swarm during the  $(t-1)^{\text{th}}$  generation;  $w$  is the inertia weight coefficient;  $c1$  and  $c2$  are learning rates, that is they are non-negative constants and usually both are set to 2; and  $\text{rand}_1$  and  $\text{rand}_2$  are independent random numbers between 0 and 1.

### 4.3.2 PSO AFTER SPSAM PROCESS

At first, the correspondence between the position of a random particle and its solution is illustrated in figure 3, where the scale factor  $S = 3$ . Then, suppose the SPM result SuperA is acquired by the SPSAM procedure and PSO is implemented after it, with the whole process executed as follows.

**Stage 1:** A mixed pixel  $P_{ab}$  in a coarse low-resolution image (i.e. fraction image) is selected in order and the following six steps of processing are carried out.

**Step 1:** A swarm with  $M$  particles is formed with the dimension number  $S^2$  for each particle  $i, i = 1, 2, \dots, M$ . Necessarily, the total number (defined as  $N_i$ ) of sub-pixels that belong to class 1 must be  $F(P_{ab})S^2$ . For a mixed pixel  $P_{ab}$ , a certain particle can be extracted from SuperA, the position of which corresponds to the spatial distribution of the classes in  $P_{ab}$  itself. After that, the particular particle is cloned several times. Assume  $\rho_{\text{clone}} \in (0,1)$  is the defined ratio. As a result, there will be  $\rho_{\text{clone}}M$  clones generated in the swarm.  $\rho_{\text{clone}}$  cannot be too high, otherwise it will result in premature convergence and will fall into local optima.

**Step 2:** The initialization of the velocity for every particle:  $V_i = [V_{i1}, V_{i2}, \dots, V_i]$ ,  $i = 1, 2, \dots, M$ , where the  $p^{\text{th}}$  dimension of the  $i^{\text{th}}$  particle is initialized as

$$V_{ip} = V_{\min} + \text{rand}(V_{\max} - V_{\min}), \quad p = 1, 2, \dots, S^2. \quad (4.10)$$

We can set  $V_{\min} = -V_{\max}$  and then  $V_{ip}$  is restricted to the interval  $[-V_{\max}, V_{\max}]$ .

**Step 3:** According to equation (6), the fitness  $J_{Xi}$  of each particle is calculated. Afterwards, the particle  $G(t)$  that has the highest fitness is selected and it is just the best position in the whole swarm at the  $t^{\text{th}}$  generation. Besides, the best position  $P_i(t)$  for particle  $i$  from the first to the  $t^{\text{th}}$  generation is also selected in the same way. By using equations (7)–(9), the position and velocity are updated, after which the velocity is restricted as

$$V_{ip}(t) = \begin{cases} V_{\max}, & V_{ip}(t) > V_{\max} \\ V_{ip}(t), & -V_{\max} \leq V_{ip}(t) \leq V_{\max} \\ -V_{\max}, & V_{ip}(t) < -V_{\max} \end{cases}, \quad (4.11)$$

This can prevent the particles from falling into local optima and flying over the best position.

**Step 4:** The constraint in equation (6) is realized. After one update,  $N_i$  may not be  $F(P_{ab})S^2$  and measures should be taken to maintain the constraint, which can be realized as follows: if  $N_i$  is bigger than  $F(P_{ab})S^2$ ,  $N_i - F(P_{ab})S^2$  particles that belong to class ‘1’ are randomly selected and changed into ‘0’ while, on the contrary, the selected ones are change into ‘1’.

**Step 5:** Swarm goes through  $R$  times evolution according to steps 3 and 4.

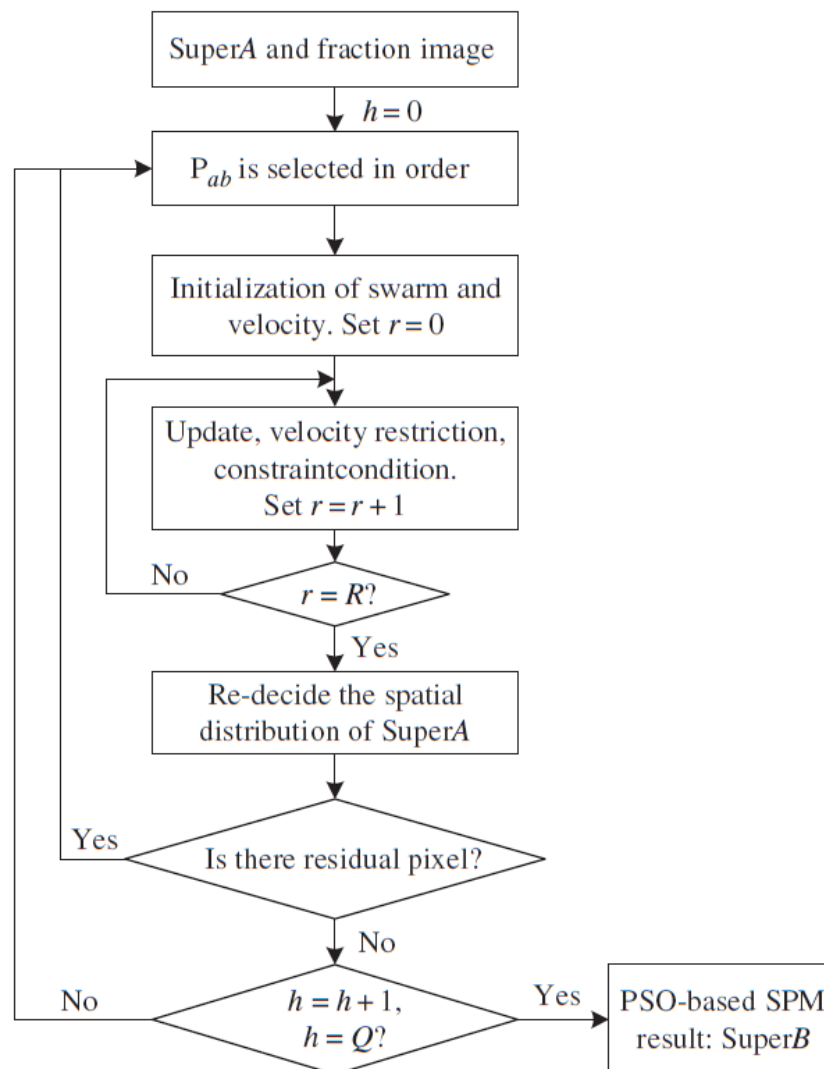
**Step 6:** The best position  $X_{\text{best}}$  during all generations is found out and used to re-decide the spatial distribution of the classes within  $P_{ab}$  in SuperA.

**Stage 2:** For all mixed pixels in the fraction image, stage 1 is processed.

**Stage 3:** The behavior of the swarm is affected by sub-pixels that are part of neighboring coarse resolution pixels (i.e.  $p_{kn}$ ) and changes in one coarse resolution pixel should have an influence on neighboring coarse resolution pixels. Therefore, stages 1 and 2 are repeated  $Q$  times, and the SPM result SuperB based on PSO is approached iteratively. Figure 4.2 displays the whole PSO-based SPM process, where  $r$  and  $h$  denote the counters of generations and iterations, respectively. When the total number of classes is  $C$ , then we can construct  $C$  one-against-rest models. In each model, one class is selected as class ‘1’ and other classes are treated as class ‘0’.

Then the spatial distribution of class ‘1’ is determined. At last,  $C$  sub-pixel maps will be generated. Theoretically, the boundaries of each class within the coarse resolution pixel may compete with each other. An effective way to integrate the  $C$  sub-pixel maps and get the SPM results for multiple classes is to choose the boundary of each class as a common boundary in turn (Ge *et al.* 2009). For each coarse resolution pixel, once the first class is allocated in agreement with its boundary, only the remaining sub-pixels are used to allocate the second class in agreement with the boundary of this class. This ordered allocation procedure continues until the final class is allocated. In this way, the order of these classes must be specified at first. Makido *et al.* (2007) have advocated a method using Moran’s  $I$  to determine the order. However, this method requires prior class information, which is not obtainable in most real situations. To avoid complex

procedures, here a simple and effective method is applied. Instead of starting to allocate sub-pixels of the dominant class surrounded by many neighbors, it may seem reasonable to first determine the distribution of the rare sub-pixels among them (Mertens *et al.* 2006). For this reason, for each selected coarse resolution pixel, the classes within it can be ranked according to their fractions from this pixel's neighbors and the class with smaller fraction is allocated before the classes with larger fractions.



**Figure 4. 2 Flow Chart of PSO Algorithm**

## 4.4 ANT COLONY OPTIMIZATION

The **ant colony optimization** algorithm (ACO) is a probabilistic technique for solving computational problems which can be reduced to finding good paths through graphs.

This algorithm is a member of the **ant colony algorithms** family, in swarm intelligence methods, and it constitutes some metaheuristic optimizations. Initially proposed by Marco Dorigo in 1992 in his PhD thesis, the first algorithm was aiming to search for an optimal path in a graph, based on the behavior of ants seeking a path between their colony and a source of food. The original idea has since diversified to solve a wider class of numerical problems, and as a result, several problems have emerged, drawing on various aspects of the behavior of ants.

### 4.4.1 Overview:

In the natural world, ants initially wander randomly, and upon finding food return to their colony while laying down pheromone trails. If other ants find such a path, they are likely not to keep travelling at random, but to instead follow the trail, returning and reinforcing it if they eventually find food see Ant communication.

Over time, however, the pheromone trail starts to evaporate, thus reducing its attractive strength. The more time it takes for an ant to travel down the path and back again, the more time the pheromones have to evaporate. A short path, by comparison, gets marched over more frequently, and thus the pheromone density becomes higher on shorter paths than longer ones. Pheromone evaporation also has the advantage of avoiding the convergence to a locally optimal solution. If there were no evaporation at all, the paths chosen by the first ants would tend to be

excessively attractive to the following ones. In that case, the exploration of the solution space would be constrained.

Thus, when one ant finds a good (i.e., short) path from the colony to a food source, other ants are more likely to follow that path, and positive feedback eventually leads to all the ants following a single path. The idea of the ant colony algorithm is to mimic this behavior with simulated ants walking around the graph representing the problem to solve.

The ACO met heuristic is:

```
Set parameters, initialize pheromone trails
SCHEDULE_ACTIVITIES
  ConstructAntSolutions
  DaemonActions      {optional}
  UpdatePheromones
END_SCHEDULE_ACTIVITIES
```

#### 4.4.2 Pheromone update:

When all the ants have completed a solution, the trails are updated by

$$\tau_{xy} \leftarrow (1 - \rho)\tau_{xy} + \sum_k \Delta\tau_{xy}^k \quad (4.12)$$

$\tau_{xy}$  is the amount of pheromone deposited for a state transition  $xy$ ,  $\rho$  is the pheromone evaporation coefficient and  $\Delta\tau_{xy}^k$  is the amount of pheromone deposited by  $k$ th ant,

$$\Delta\tau_{xy}^k = \begin{cases} Q/L_k & \text{if ant } k \text{ uses curve } xy \text{ in its tour} \\ 0 & \text{otherwise} \end{cases} \quad (4.13)$$

where  $L_k$  is the cost of the  $k$ th ant's tour (typically length) and  $Q$  is a constant.

## 5. IMPLEMENTATION AND RESULTS

In this section we will compare the results of the PSO (Particle Swarm Optimization) and the Ant Colony Optimization technique, Results of the PSO for the various input images (with different classes, Satellite image and synthetic image) are compared with the ANT Colony Optimization for the same image.

### 5.1 TWO CLASS SYNTHETIC IMAGE



**Figure 5. 1 Two Class Synthetic Image as Input (70x70)**

This is the input image that we are going to process using PSO Algorithm and ANT colony Optimization Algorithm , this image is created manually with 70\*70 pixels with two colors yellow and blue , in this yellow is taken as the one class and blue is taken as another class.

#### **Image after Soft Classification:**



**Figure 5. 2 Presence of Class 1 (Yellow)**

This is the image that we obtain after the soft classification of our input image in this white represents the area where the yellow class is present and black represents the area where the yellow class is not present.

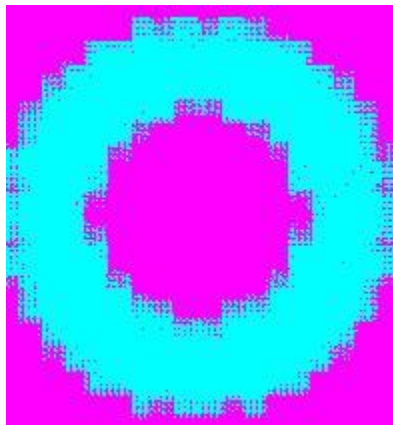




**Figure 5. 3 Presence of Class 1 (Blue)**

This is the image that we obtain after the soft classification of our input image in this white represents the area where the blue class is present and black represents the area where the blue class is not present.

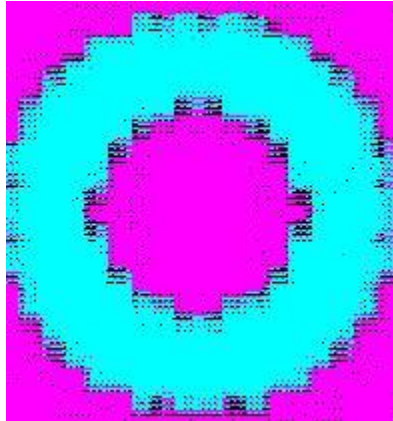
### **5.1.1 PSO ALGORITHM**



**Figure 5. 4 Result of two class synthetic image after applying PSO Algorithm**

The final image that is obtained is 210\*210 because in this we used scale factor as 3, and this image is used to test the accuracy of the PSO because only in the synthetic image accuracy can be checked , in this the blue represents the area of the Dark Blue class and rose represents the yellow class areas.

### 5.1.2 ANT COLONY OPTIMIZATION ALGORITHM



**Figure 5. 5 Result of two class synthetic image after applying ACO Algorithm**

The final image that is obtained is 210\*210 because in this we used scale factor as 3, and this image is used to test the accuracy of the ANT Colony Optimization because only in the synthetic image accuracy can be checked, in this the blue represents the area of the Dark Blue class and rose represents the yellow class areas and black dots represents the unclassified classes.

### 5.1.3 COMPARISON OF PSO AND ANT COLONY OPTIMIZATION ALGORITHM

#### SPEED

As both the algorithms are ran using the parallel processing results are obtained quicker for the both the techniques but PSO took nearly of 20 sec to process this image and for ANT colony speed cannot be told accurately because speed depends upon the number of the Ants that are generated randomly.

#### ACCURACY

PSO is more accurate than the ANT colony Optimization technique because in the ANT colony Optimization technique we are getting some classified classes

in the synthetic image itself but in the ANT Colony Optimization technique edge are clearer than the PSO algorithm.

	INPUT RESOLUTION	OUTPUT RESOLUTION	SPEED	UNCLASSIFIED PERCENTAGE	EDGES ACCURACY	Image Accuracy
<b>PSO(Particle swarm Optimization)</b>	30*30	210*210	20SEC	2%	95%	98%
<b>ACO(Ant Colony Optimization)</b>	30*30	210*210	30SEC, 40SEC	9%	98%	81%

Table 5. 1 Comparison of PSO And ACO Algorithm for SRM for 2 class synthetic image

The Results obtained from ACO Algorithm is less accurate. So, the Algorithm has to be refined further. If better accuracy is needed, then PSO Algorithm has to be applied. If better edge accuracy is needed, then ACO has to be Applied.

## 5.2 THREE CLASS SYNTHETIC IMAGE

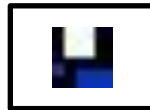


Figure 5. 6 Three Class Synthetic Image as Input (10x10)

This is the input image that we are going to process using PSO technique and ANT colony Optimization technique, this image is created manually with 10\*10 pixels with three colors black, white and blue , in this black is taken as the one class , blue is taken as other class and white is taken as other class.

### Image after Soft Classification:



**Figure 5. 7 Presence of Class 1 (Black)**

This is the image that we obtain after the soft classification of our input image in this white represents the area where the black class is present and black represents the area where the black class is not present



**Figure 5. 8 Presence of Class 1 (Blue)**

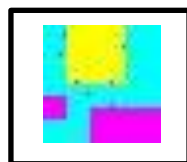
This is the image that we obtain after the soft classification of our input image in this white represents the area where the blue class is present and black represents the area where the blue class is not present.



**Figure 5. 9 Presence of Class 1 (White)**

This is the image that we obtain after the soft classification of our input image in this white represents the area where the white class is present and black represents the area where the white class is not present.

### 5.2.1 PSO ALGORITHM



**Figure 5. 10 Result of three class synthetic image after applying PSO Algorithm**

The final image that is obtained is 30\*30 because in this we used scale factor as 3, and this image is used to test the accuracy of the PSO because only in the

synthetic image accuracy can be checked, in this the yellow represents the area of the white class, rose represents the areas of dark blue class and blue represents the area of black class.

### 5.2.2 ANT COLONY OPTIMIZATION ALGORITHM



**Figure 5. 11 Result of three class synthetic image after applying ACO Algorithm**

The final image that is obtained is 30\*30 because in this we used scale factor as 3, and this image is used to test the accuracy of the ANT Colony Optimization because only in the synthetic image accuracy can be checked, in this the green represents the area of the white class, rose represents the areas of dark blue class and yellow represents the area of black class.

### 5.2.3 COMPARISON OF PSO AND ANT COLONY OPTIMIZATION ALGORITHM

#### SPEED

As both the algorithms are ran using the parallel processing results are obtained quicker for the both the techniques but PSO took nearly of 5 sec to process this image and for ANT colony speed cannot be told accurately because speed depends upon the number of the Ants that are generated randomly

#### ACCURACY

PSO is more accurate than the ANT colony Optimization technique because in the ANT colony Optimization technique we are getting some classified classes in the synthetic image itself but in the ANT Colony Optimization technique edge are clearer than the PSO algorithm.

	INPUT RESOLUTION	OUTPUT RESOLUTION	SPEED	UNCLASSIFIED PERCENTAGE	EDGES ACCURACY	Image Accuracy
<b>PSO(Particle swarm Optimization)</b>	10*10	30*30	5SEC	1%	94%	99%
<b>ACO(Ant Colony Optimization)</b>	10*10	30*30	10SEC, 8SEC	9%	97%	91%

Table 5. 2 Comparison of PSO And ACO Algorithm for SRM for 3 class synthetic image

The Results obtained from ACO Algorithm is less accurate. So, the Algorithm has to be refined further. If better accuracy is needed, then PSO Algorithm has to be applied. If better edge accuracy is needed, then ACO has to be Applied.

### 5.3 THREE CLASS SATELLITE IMAGE



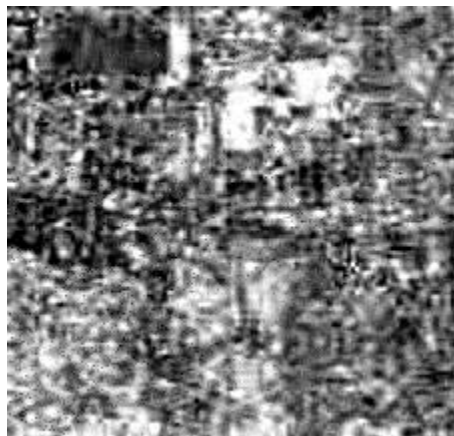
**Figure 5. 12 Three Class Satellite Image as Input (227x216)**

This is the input image that we are going to process using PSO technique and ANT colony Optimization technique, this image is satellite image with 227\*216 pixels, in this 3 class is selected which represents the buildings , grass and land area.

**Image after Soft Classification:**

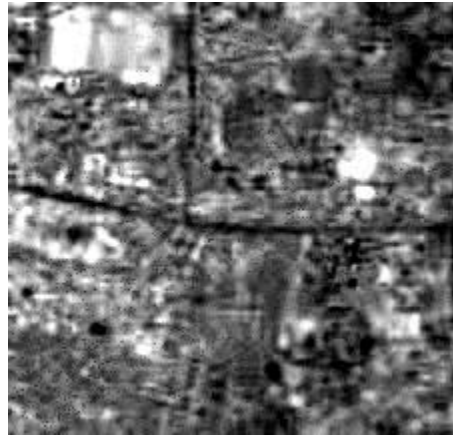
**Figure 5. 13 Presence of Class 1 (Building)**

This is the image that we obtain after the soft classification of our input image in this white represents the area where the building class is present and black represents the area where the building blue class is not present.



**Figure 5. 14 Presence of Class 2 (Grass)**

This is the image that we obtain after the soft classification of our input image in this white represents the area where the grass class is present and black represents the area where the building grass class is not present.

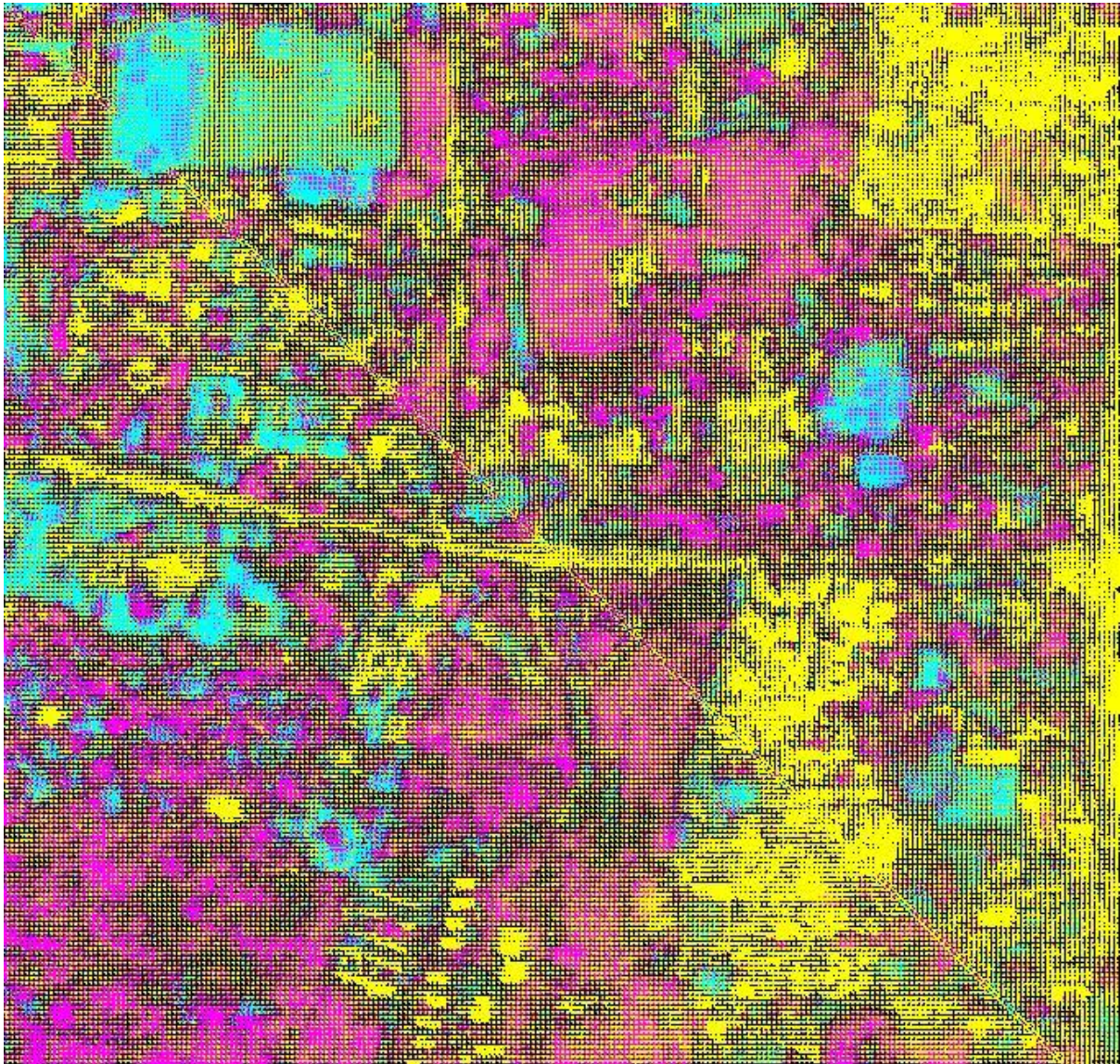


**Figure 5. 15 Presence of Class 3 (Land)**

This is the image that we obtain after the soft classification of our input image in this white represents the area where the land class is present and black represents the area where the building land class is not present.



### 5.3.1 PSO ALGORITHM

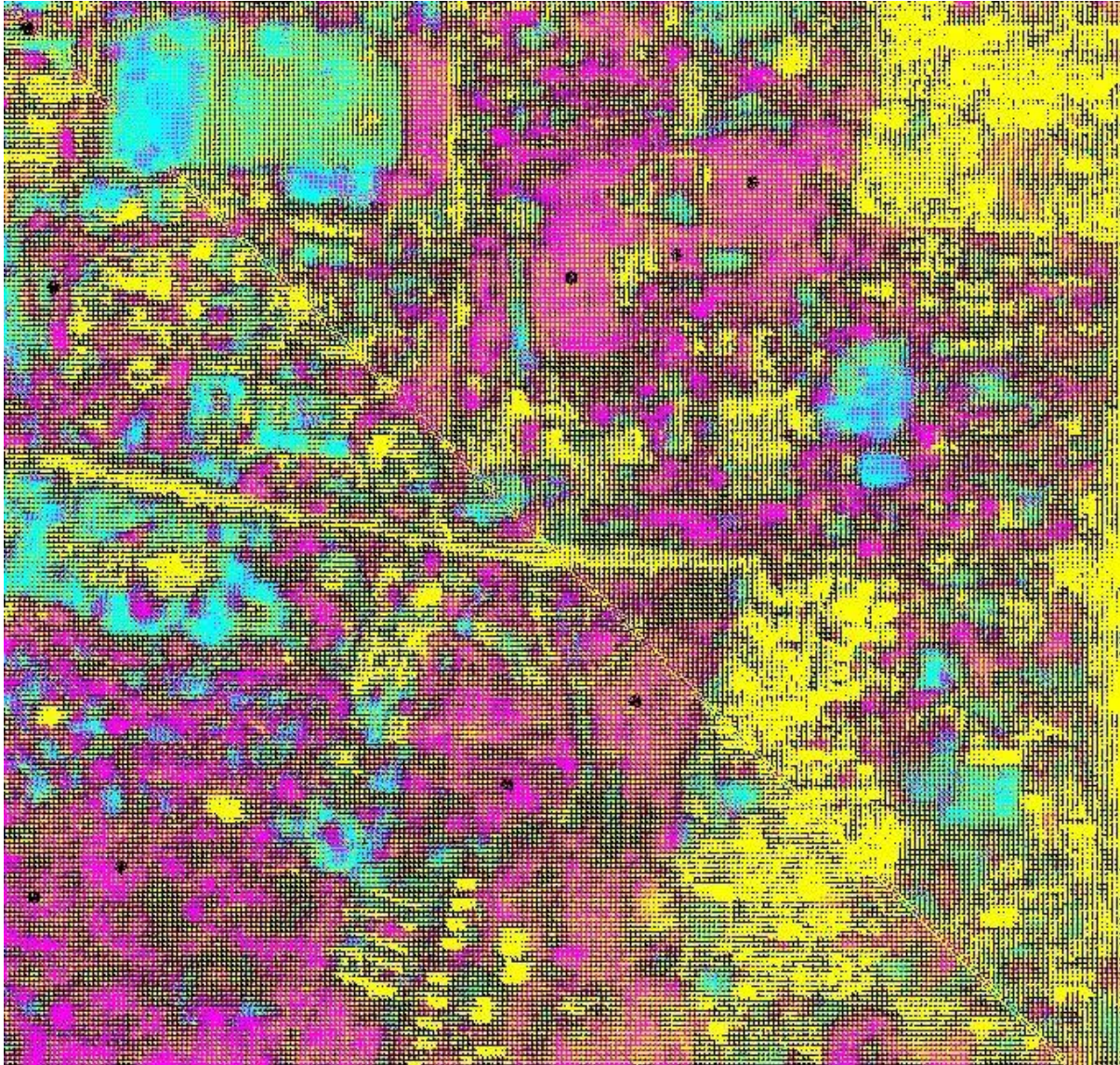


**Figure 5. 16 Result of three class satellite image after applying PSO Algorithm**

The final image that is obtained is  $681 \times 685$  because in this we used scale factor as 3 , and this image is used to test the accuracy of the PSO because only in the synthetic image accuracy can be checked.



### 5.3.2 ANT COLONY OPTIMIZATION ALGORITHM



**Figure 5. 17 Result of three class satellite image after applying ACO Algorithm**

The final image that is obtained is 661\*685 because in this we used scale factor as 3 , and this image is used to test the accuracy of the ANT Colony Optimization because only in the synthetic image accuracy can be checked, in this the yellow represents the building class, rose represents the area of grass class and blue represents the land area.

### 5.3.3 COMPARISON OF PSO AND ANT COLONY OPTIMIZATION ALGORITHM

#### SPEED

As both the algorithms are ran using the parallel processing results are obtained quicker for the both the techniques but PSO took nearly of 10 minutes to process this image and for ANT colony speed cannot be told accurately because speed depends upon the number of the Ants that are generated randomly.

#### ACCURACY

PSO is more accurate than the ANT colony Optimization technique because in the ANT colony Optimization technique we are getting some classified classes but in the ANT Colony Optimization technique edge are clearer than the PSO algorithm.

	INPUT RESOLUTION	OUTPUT RESOLUTION	SPEED
PSO(Particle swarm Optimization)	227*216	661*685	5MINS
ACO(Ant Colony Optimization)	227*216	661*685	7MINS 9MINS

Table 5. 3 Comparison of PSO and ACO Algorithm for SRM for 3 class satellite image

The Results obtained from ACO Algorithm is less accurate. So, the Algorithm has to be refined further. If better accuracy is needed, then PSO Algorithm has to be applied. If better edge accuracy is needed, then ACO has to be Applied.



## 5.4 FOUR CLASS SATELLITE IMAGE



**Figure 5. 18 Four Class Satellite Image as Input (187x195)**

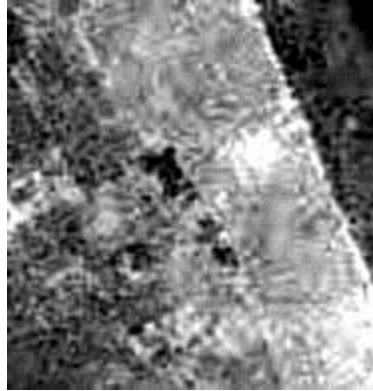
This is the input image that we are going to process using PSO technique and ANT colony Optimization technique, this image is satellite image with 187\*195 pixels, in this 4 class is selected with the different colors.

**Image after Soft Classification:**



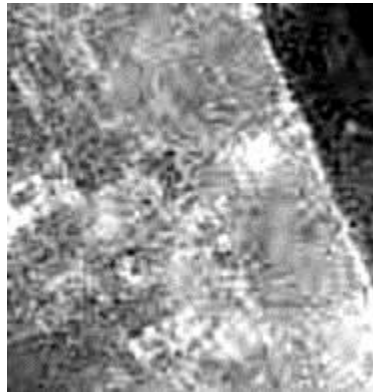
**Figure 5. 19 Presence of Class 1 (Dark Blue)**

This is the image that we obtain after the soft classification of our input image in this white represents the area where the dark blue class is present and black represents the area where the dark blue class is not present.



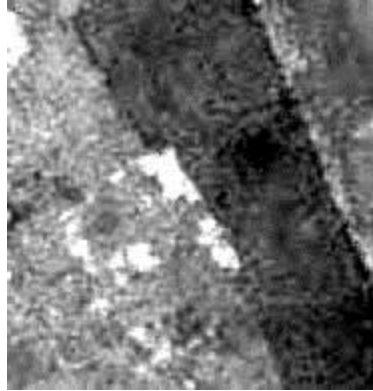
**Figure 5. 20 Presence of Class 2 (Blue)**

This is the image that we obtain after the soft classification of our input image in this white represents the area where the blue class is present and black represents the area where the blue class is not present.



**Figure 5. 21 Presence of Class 3 (Orange)**

This is the image that we obtain after the soft classification of our input image in this white represents the area where the orange class is present and black represents the area where the orange class is not present.

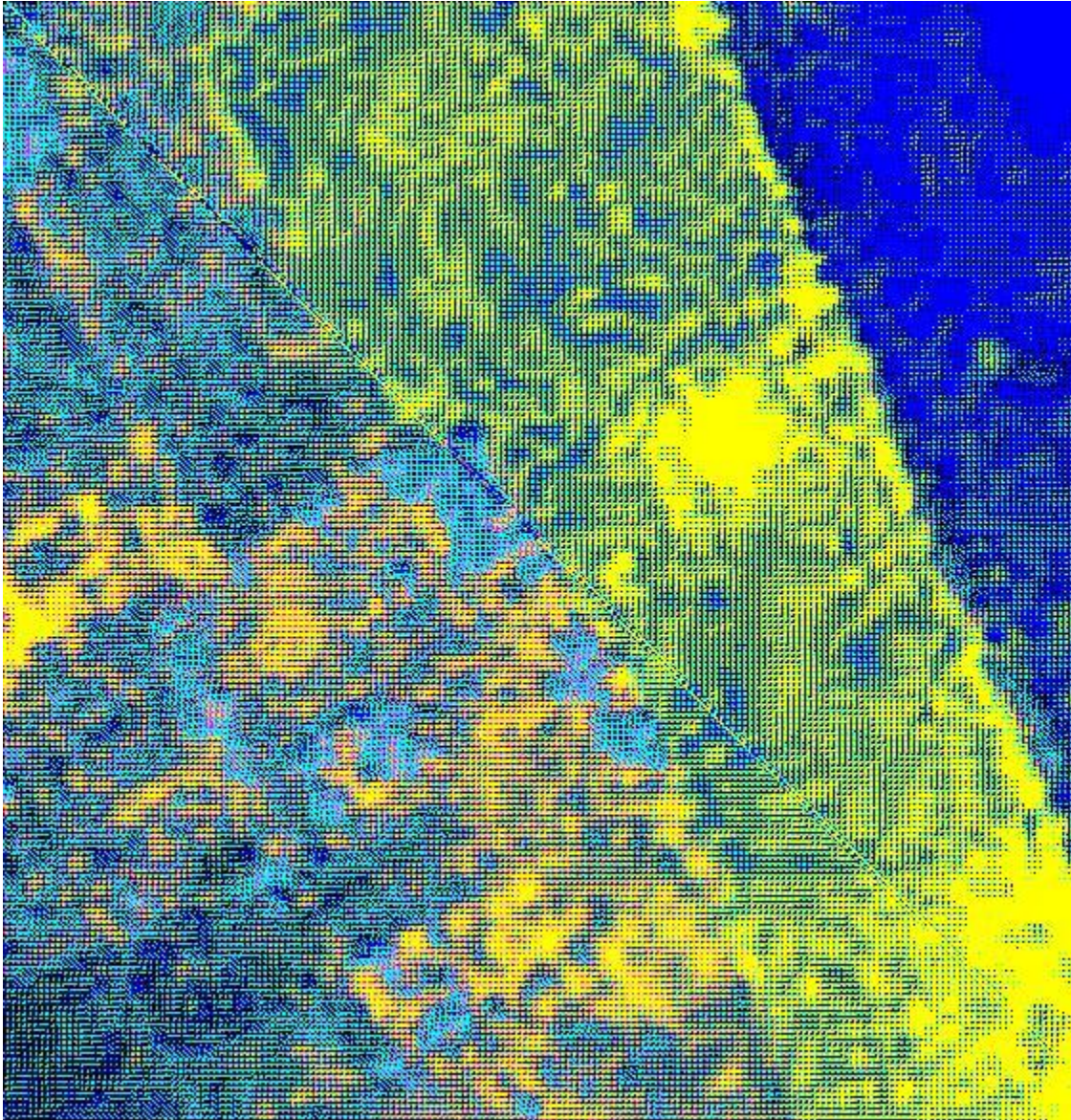


**Figure 5. 22 Presence of Class 4 (Red)**

This is the image that we obtain after the soft classification of our input image in this white represents the area where the red class is present and black represents the area where the red class is not present.



### 5.4.1 PSO ALGORITHM

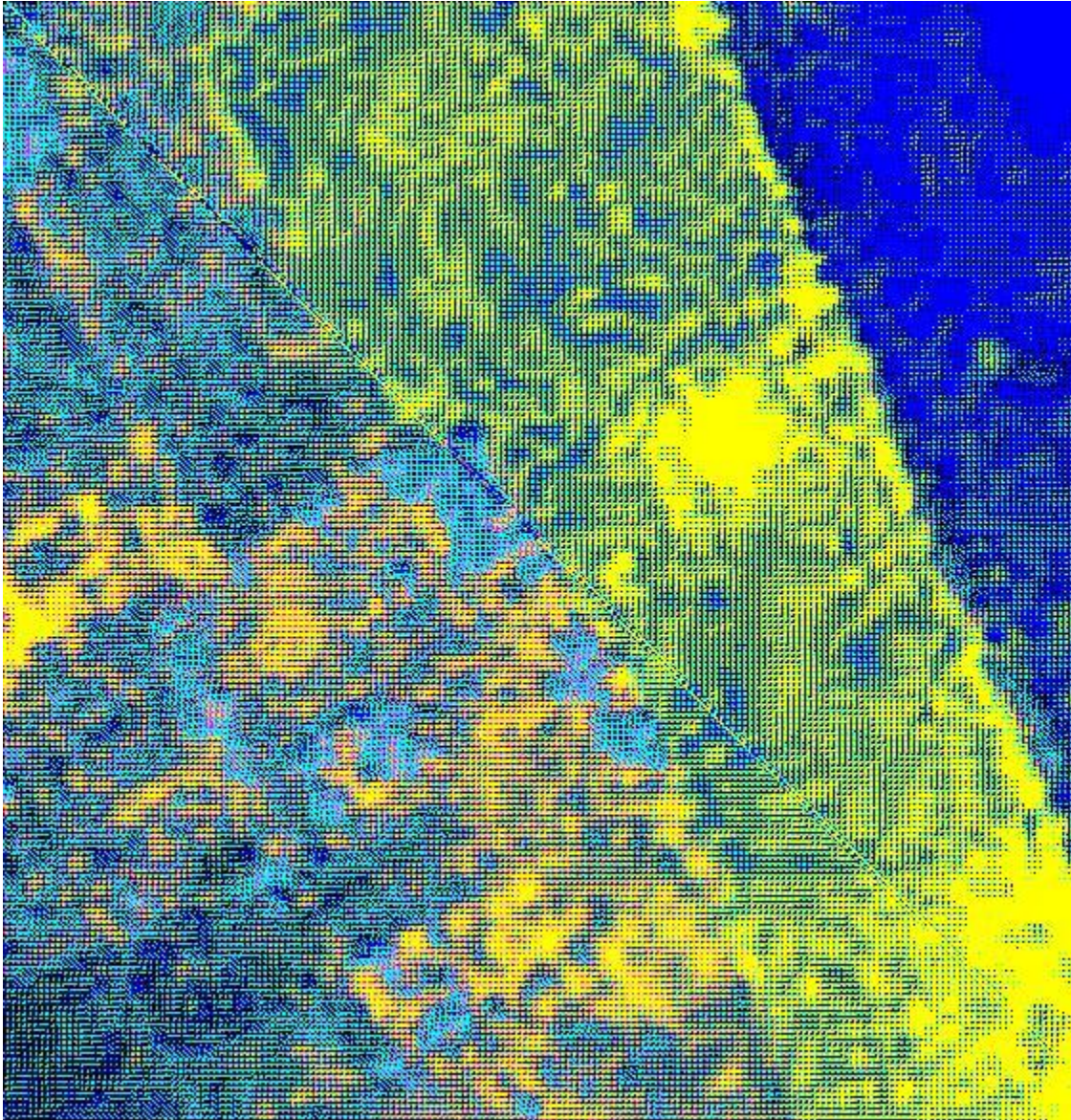


**Figure 5. 23 Result of Four class satellite image after applying PSO Algorithm**

The final image that is obtained is 561\*585 because in this we used scale factor as 3, and this image is used to test the accuracy of the PSO because only in the synthetic image accuracy can be checked, in this the dark blue represents the area of the dark blue class, yellow represents the areas of blue class, dark yellow represents the area of red class and light blue represents the area of orange class.



### 5.4.2 ANT COLONY OPTIMIZATION ALGORITHM



**Figure 5. 24 Result of four class satellite image after applying ACO Algorithm**

The final image that is obtained is 561\*585 because in this we used scale factor as 3 , and this image is used to test the accuracy of the ANT Colony Optimization because only in the synthetic image accuracy can be checked, in this the dark blue represents the area of the dark blue class , yellow represents the areas



of blue class, dark yellow represents the area of red class and light blue represents the area of orange class.

### **5.4.3 COMPARISON OF PSO AND ANT COLONY OPTIMIZATION ALGORITHM**

#### **SPEED**

As both the algorithms are ran using the parallel processing results are obtained quicker for the both the techniques but PSO took nearly of 8mins to process this image and for ANT colony speed cannot be told accurately because speed depends upon the number of the Ants that are generated randomly.

#### **ACCURACY**

PSO is more accurate than the ANT colony Optimization technique because in the ANT colony Optimization technique we are getting some classified classes but in the ANT Colony Optimization technique edge are clearer than the PSO algorithm. Optimization technique edge is clearer than the PSO algorithm.

	<b>INPUT RESOLUTION</b>	<b>OUTPUT RESOLUTION</b>	<b>SPEED</b>
<b>PSO(Particle swarm Optimization)</b>	187*195	561*585	8MINS
<b>ACO(Ant Colony Optimization)</b>	187*195	561*585	7MINS 10MINS

Table 5. 4 Comparison of PSO and ACO Algorithm for SRM for 4 class satellite image

The Results obtained from ACO Algorithm is less accurate. So, the Algorithm has to be refined further. If better accuracy is needed, then PSO Algorithm has to be applied. If better edge accuracy is needed, then ACO has to be Applied.

## **6. CONCLUSION**

### **6.1 OVERALL CONCLUSION**

In the 20<sup>th</sup> century we still experience problem with satellite image processing because satellite images can be of lesser resolution and usually cover large area so the single pixel in the satellite image represents large area with different classes .Thus in the project explored on different techniques of sub-pixel mapping to help map an area. Proposed a novel approach for post processing to enhance SPSAM process. Compared two Optimization algorithms, the Particle Swarm Optimization and Ant Colony Optimization to efficiently map a particular area.

### **6.1 FUTURE WORK AND OPTIMIZATION**

The system currently doesn't provide 100% accuracy in the processed image this could be extended to get 100% accuracy by comparing with the different algorithms for the different type of images

## 7. REFERENCES

- [1] A. J. Tatem, H. G. Lewis, P. M. Atkinson, and M. S. Nixon, "Super-resolution land cover pattern prediction using a Hopfield neural network," *Remote Sens. Environ.*, vol. 79, no. 1, pp. 1–14, Jan. 2002.
- [2] Feng Ling, Yun Du, Fei Xiao, and Xiaodong Li "Subpixel Land Cover Mapping by Integrating Spectral and Spatial Information of Remotely Sensed Imagery" *IEEE GEOSCIENCE AND REMOTE SENSING LETTERS.*, 2011 IEEE.
- [3] P. M. Atkinson, "Issues of uncertainty in super-resolution mapping and their implications for the design of an inter-comparison study," *Int. J. Remote Sens.*, vol. 30, no. 20, pp. 5293–5308, Oct. 2009.
- [4] A. J. Tatem, H. G. Lewis, P. M. Atkinson, and M. S. Nixon, "Super-resolution target identification from remotely sensed images using a Hopfield neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 4, pp. 781–796, Apr. 2001.
- [5] D. Picard, A. Revel, M. Cord, "An Application of Swarm Intelligence to Distributed Image Retrieval", *Information Sciences*, 2010
- [6] D. Picard, M. Cord, A. Revel, "Image Retrieval over Networks : Active Learning using Ant Algorithm", *IEEE Transactions on Multimedia*, 2012
- [7] [http://en.wikipedia.org/wiki/Particle\\_swarm\\_optimization](http://en.wikipedia.org/wiki/Particle_swarm_optimization)
- [8] [http://en.wikipedia.org/wiki/Ant\\_colony\\_optimization\\_algorithms](http://en.wikipedia.org/wiki/Ant_colony_optimization_algorithms)