CLASSIFICATION OF SATELLITE IMAGE USING MAXIMUM LIKELIHOOD AND ISODATA TECHNIQUES

CHUA XIN YING

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CLASSIFICATION OF SATELLITE IMAGE USING MAXIMUM LIKELIHOOD AND ISODATA TECHNIQUES

CHUA XIN YING

This report is submitted in partial fulfillment of the requirements for the Bachelor of [Computer Science (Networking)] with Honours.

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2021

DECLARATION

I hereby declare that this project report entitled

CLASSIFICATION OF SATELLITE IMAGE USING MAXIMUM LIKELIHOOD AND ISODATA TECHNIQUES

is written by me and is my own effort and that no part has been plagiarized without citations.

STUDENT : ________________________ Date : 09/09/2021

(CHUA XIN YING)

I hereby declare that I have read this project report and found this project report is sufficient in term of the scope and quality for the award of Bachelor of [Computer Science (Networking)] with Honours.

SUPERVISOR : ________________________ Date : 09/09/2021

(GS. DR. OTHMAN BIN MOHD)

DEDICATION

This study is dedicated to my parents, Chua Yong Peng and Ong Swee Kiau who inspired me and support me on financial, moral and emotional in completing this study. Without their support, I may not be able to complete this. Besides, I decided this study to my supervisor, Gs. Dr. Othman Bin Mohd. He gave me a lot of advice, help me in understanding more about image processing and guide me in completing this study. Lastly, I dedicate to my friend, Chong Zi Qing who share her knowledges in image processing with me and encourage me to complete my project.

ACKNOWLEDGEMENTS

First and fore most, I would like to express my appreciation to my supervisor, Gs. Dr. Othman Bin Mohd for giving me a lot of advice and guidance, helping me in understanding more about image processing throughout this project. The motivation and inspiration that he gave to me, motivated me to contribute to this project. Besides, I would like to thanks the authority of Universiti Teknikal Malaysia Melaka(UTeM) for providing the guideline and giving me the opportunity to complete this project.

Furthermore, I was highly thankful to my friends, who share their knowledges to me and encourage me when I was facing problems in completing my project. Lastly, I would like to express my gratitudes my family members for their supports and encouragement which help me in finishing in project.

ABSTRACT

Satellite Image Processing is important in Research and Development field. It is taken by the artificial satellite, and the photo taken is processed by computer to extract the data in the photo. Processing the image by using manual methods consumes a lot of time. Besides, the expert must know well about the area covered by the satellite image and the knowledge and familiarity of the expert will directly affect the efficiency and accuracy of the classification. Therefore, image classification by computer is introduced. There are a lot of techniques introduced by the previous researcher to classify the satellite image and the techniques are mainly divided into two types, which are Supervised and Unsupervised Classification. The example of supervised classification method is Maximum Likelihood and unsupervised classification method is ISODATA. The two techniques were used to identify the objects in the satellite image. The process of comparing results based on the two techniques were carried out to identify the best classification techniques. The comparison of the output was conducted based on the percentage of accuracy.

ABSTRAK

Pemprosesan imej satelit adalah antara bidang yang penting dalam bidang Penyelidikan dan Pembangunan. Foto akan diambil oleh satelit buatan, dan diproses oleh komputer untuk mengekstrak data dalam foto tersebut. Pemprosesan gambar dengan menggunakan kaedah manual memerlukan banyak masa. Selain itu, penganalisis mesti mengetahui bidang yang diliputi oleh imej satelit dan pengetahuan dan keakraban penganalisis akan mempengaruhi kecekapan dan ketepatan klasifikasi secara langsung. Oleh itu, pengkelasan gambar oleh komputer diperkenalkan. Terdapat banyak teknik yang diperkenalkan oleh pengkaji sebelumnya untuk mengklasifikasikan imej satelit dan teknik tersebut dibahagikan kepada dua jenis, iaitu Klasifikasi Terbimbing dan Tidak Terbimbing. Antara contoh kaedah Klasifikasi Terbimbing ialah Kemungkinan Maksimum dan Klasifikasi Tidak Terbimbing ialah ISODATA. Kedua-dua teknik digunakan untuk mengenalpasti objek dalam gambar satelit. Proses perbandingan hasil dapatan antara dua teknik dilakukan untuk mengenalpasti teknik klasifikasi yang terbaik. Perbandingan hasil dapatan akan dilakukan berdasarkan peratusan ketepatan.

TABLE OF CONTENTS

		PAGE
DECLARATION		ii
DEDICATION		iii
ACKNOWLEDGEMENTS		iv
ABSTRACT		v
ABSTRAK		vi
TABLE OF CONTENTS		vii
LIST OF TABLES		x
LIST OF FIGURES		xii
LIST OF ABBREVIATIONS		xiv
LIST OF APPENDIXS		xv
CHAPTER 1		
1.1	Introduction	1
1.2	Problem Statement	2
1.3	Project Question	3
1.4	Project Objective	3
1.5	Project Scope	4
1.6	Project Contribution	4
1.7	Report Organisation	5
1.8	Conclusion	5
CHAPTER 2		
2.1	Introduction	6
2.2	Related Work/Previous Work	7
	2.2.1 Image Processing	8
	2.2.2 Image Processing for Satellite Image	9
	2.2.3 Image Classification	11
	2.2.4 Image Classification based on Maximum Likelihood	13

		Image Classification based on ISODATA	14
2.3	Critical review of current problem and justification		16
2.4	Proposed Solution/further project		19
2.5	Conclusion		19
CHAPTER 3			
3.1	Introduction		21
3.2	Methodology		21
	3.2.1	Preprocessing	22
	3.2.2	Image Classification	22
	3.2.3	Accuracy Assessment	23
	3.2.4	Evaluation of Classification Method	23
3.3	Classification Techniques		23
	3.3.1	Maximum Likelihood	23
	3.3.2	ISODATA	25
3.4	Project Milestones		26
3.5	Conclusion		31
CHAPTER 4			
4.1	Introduction		32
4.2	Environment setup		32
	4.2.1	Software Requirements	32
	4.2.2	Hardware Requirements	33
4.3	Satellite Image Classification Process		33
	4.3.1	Maximum Likelihood Classifier	34
	4.3.2	ISODATA Classifier	38
4.4	Conclusion		42
CHAPTER 5			
5.1	Introduction		43
5.2	Test Results and Analysis		43
	5.2.1	Pre-processing	43
	5.2.2	Testing process	44
		5.2.2.1 Maximum Likelihood's Testing Process	45

5.2.2.1 ISODATA's Testing Process 48

5.3 Result Analysis 51

5.3.1 Testing Result 51

5.3.2 Testing Result Analysis 53

5.4 Conclusion 54

CHAPTER 6

6.1 Introduction 56

6.2 Project Summarization 56

6.3 Project Contribution 57

6.4 Project Limitation 57

6.5 Future Works 57

6.6 Conclusion 57

REFERENCES 59

LIST OF TABLES

		PAGE
Table 1.1	Problem Statements	2
Table 1.2	Project Questions	3
Table 1.3	Project Objectives	4
Table 2.1	Image Processing Techniques	10
Table 2.2	Classification Techniques	12
Table 2.3	Comparative Study for Image Classification Techniques	18
Table 3.1	Project Milestone	26
Table 3.2	Gantt Chart	27
Table 4.1	Software Requirement	32
Table 4.2	Hardware Requirement	33
Table 5.1	Confusion Matrix and Accuracy for Maximum Likelihood Classifier	51
Table 5.2	Sensitivity, Specificity and Overall Accuracy for Maximum Likelihood Classifier	52
Table 5.3	Confusion Matrix and Accuracy for each of the classes for ISODATA Classifier	52
Table 5.4	Sensitivity, Specificity and Overall Accuracy for ISODATA Classifier	53
Table 5.5	Comparison for the accuracy of Maximum Likelihood and ISODATA	53
Table 5.6	Advantage and disadvantage of MLC and ISODATA	54

LIST OF FIGURES

		PAGE
Figure 2.1	Image Processing Summary	8
Figure 2.2	General Stage of ISODATA	15
Figure 3.1	Detail Process of Image Processing	22
Figure 3.2	Stage of Maximum Likelihood classification	24
Figure 3.3	Stage of ISODATA classification	25
Figure 4.1	Original Satellite Image	34
Figure 4.2	Flow of Maximum Likelihood Classification	35
Figure 4.3	Coding of importing satellite image	35
Figure 4.4	Coding of importing training data	35
Figure 4.5	Coding of calculating mean and standard deviation	36
Figure 4.6	Coding of applying MLE's formula	36
Figure 4.7	Coding of applying MLE’s formula	36
Figure 4.8	Coding of defining the colors of classes	36
Figure 4.9	Coding of color changing in the final image	37
Figure 4.10	Coding of getting the final output	37
Figure 4.11	Final result of Maximum Likelihood	37
Figure 4.12	Process of ISODATA classification	38
Figure 4.13	Coding of initialize the value of variable	39
Figure 4.14	Coding of importing satellite image	39
Figure 4.15	Coding of finding the nearest distance between	39
Figure 4.16	Coding of splitting process	40
Figure 4.17	Coding of merging process	40
Figure 4.18	Coding of removing clusters with small number of elements	41
Figure 4.19	Coding of applying color changing and getting the final image.	41
Figure 4.20	Final result of ISODATA	42

Figure 5.1	Ground Truth Image	44
Figure 5.2	Flow of testing process	44
Figure 5.3	Coding of getting matrices of classified image	45
Figure 5.4	Matrices of the classified image	46
Figure 5.5	Coding of getting the matrices of ground truth image	46
Figure 5.6	Matrices of the ground truth image	47
Figure 5.7	Coding of obtaining the confusion matrix	47
Figure 5.8	Coding of calculate the sensitivity, specificity accuracy and overall accuracy	47
Figure 5.9	Calculation result of Maximum LikelihoodFigure 0.9 Calculation result	48
Figure 5.10	Coding of inputting the classified image	48
Figure 5.11	Coding of getting the matrices of ground truth image	48
Figure 5.12	Matrices of the classified image	49
Figure 5.13	Coding of getting the matrices of the ground truth image	49
Figure 5.14	Matrices of the ground truth image	49
Figure 5.15	Coding of obtaining the confusion matrix	50
Figure 5.16	Coding of calculating the sensitivity, specificity accuracy and overall accuracy	50
Figure 5.17	Calculation result of ISODATA	50
Figure 5.18	Accuracy of each class and each algorithm	54

LIST OF ABBREVIATIONS

FYP	-	Final Year Project
UTEM	-	Universiti Teknikal Malaysia Melaka
ISODATA	-	Iterative Self-Organizing Data Analysis
MLE	-	Maximum Likelihood Estimation
MLC	-	Maximum Likelihood Classification
GUI	-	Graphical User Interface
TP	-	True Positive
TN	-	True Negative
FP	-	False Positive
FN	-	False Negative

LIST OF APPENDIXS

		PAGE
Appendix A	Coding for Maximum Likelihood Classification	69
Appendix B	Coding for ISODATA Classification	76
Appendix C	Coding for getting matrices for ground truth image	82
Appendix D	Coding for Confusion Matrix and Accuracy Calculation	83

CHAPTER 1

INTRODUCTION

1.1 Introduction

Satellite images are one of the most important and powerful tools used to collect the photo of Earth (Oakfield,WI Tornado, n.d.). It contains useful information, such as the shape of the river, shape of the road, coverage of the urban area and forest area. The evolution of satellite images enhanced the process of data collection, by reducing the time taken for data collection and facilitate the process of collecting data.

In the beginning, satellite image is used in the environmental and military field. It is used more and more in the field of agriculture, map production, planning of national land, forestry, and establishment of city plan lately. Therefore, recognition of the object in satellite images is important and it is necessary to detect the objects in the satellite images accurately and distribute the information to the institutes.

The satellite images are being processed with image processing techniques. Image processing techniques can be divided into several classes, such as image classification, image segmentation, image enhancement and image compression. Image classification refers to the categorizing of images into one of a number of classes which are predefined. The feature of the satellite image was label into different classes, enable the system to differentiate between different objects. Recognition and classification of the objects in the satellite image were done by comparing the image patterns with the target pattern.

The challenge that faced by the researchers is over-classification or under-classification of the area of classes in the satellite images. It causes failure of classification process and lead to a poor classification result. The similarity between features causes difficulties in differentiating the difference between objects in the satellite image and affected the accuracy in the classification process.

According to the previous research, there are several types of image classification methods to recognize the object in the satellite images, such as Maximum Likelihood in supervised method, ISODATA in unsupervised method. These methods were able to successfully recognize the objects in satellite images and it was a contribution for the future research and study in this field. In this project, the study was be focused on Maximum Likelihood and ISODATA techniques to identify the best technique for satellite image classification.

1.2 Problem Statement

There are a lot of image classification methods to classify the satellite images such as Supervised Classification, Unsupervised Classification and Object-Based Classification. According to the previous research, there are some common techniques that being used to classify the satellite image. One of the supervised classification methods is Maximum Likelihood and ISODATA is one of the unsupervised classification methods.

The ability to classify the pixels of each classes accurately is very important, problems such as overclassify or under-classify may lead to a poor result. One of the challenges in satellite image classification is it may have difficulties in differentiating the difference between objects in the satellite image and not able to classify the objects accurately. Besides, there are a lot of classification methods, different method may give output with different accuracy. The aim of this project is to study and identify the best techniques use to obtain the most accurate classification output. The problem statements are summarized in Table 1.1.

Table 1.1 Problem Statements

PS	Problem Statement
PS1	Difficulties in differentiating the difference between objects in the satellite image
PS2	Consists of various techniques in satellite image classification

1.3 Project Question

Based on the problems stated in Table 1.1, the project questions are being identified to overcome the problem statements. The project questions are:

1. It is important to identify the techniques that are suitable to classify satellite image with a good output result. Therefore, what classification techniques can be used to classify satellite image?
2. Based on the research conducted by researcher, there are a lot of classification techniques can be used to classify satellite image. It is important to figure out which techniques can be implemented to obtain the best result in differentiating objects in satellite image.
3. Based on the techniques selected, the result will be compared to identify the best output. Hence, it is important to determine which of the proposed technique can give the best output.

The summary of the Problem Statement and Project Questions as describe in Table 1.2.

Table 1.2 Project Questions

PS	PQ	Project Question
PS1	PQ1	What classification techniques can be used to classify satellite image?
PS2	PQ2	Which techniques can be implemented to obtain the best result in differentiating objects in satellite image?
PS2	PQ3	Which of the proposed technique can give the best output?

1.4 Project Objective

Based on the problem statements and project questions stated in Table 1.1 and Table 1.2, there are three objectives implemented in this project. The project objectives are:

1. To identify the classification techniques in satellite image classification.
2. To apply Maximum Likelihood and ISODATA techniques for satellite image classification.
3. To evaluate the best technique in satellite image classification.

The summary of the Problem Statement, Project Questions and Project Objectives as describe in Table 1.3.

Table 1.3 Project Objectives

PS	PQ	PO	Project Objective
PS1	PQ1	PO1	To identify the classification techniques in satellite image classification.
PS2	PQ2	PO2	To apply Maximum Likelihood and ISODATA techniques for satellite image classification.
PS2	PQ3	PO3	To evaluate the best technique in satellite image classification.

1.5 Project Scope

The satellite image of Malacca city in Malaysia was used to conduct the image classification process in this project and the satellite images was downloaded from Google Earth. Google Earth was selected because this platform is free and easy for us to download the image. The image was downloaded on 24/5/2021 and the highest resolution available in Google Earth is 4800 x 2679 pixels.

This project was done by using MATLAB software running on Window platform. MATLAB is chosen because it is one of the most effective software to code and debug program. The satellite image was processed by using one supervised classification method and one unsupervised classification method in MATLAB.

1.6 Project Contribution

Image processing is widely used in technology and science's field such as computer vision and remote sensing. The study contributed to the experts in the field of technology and science such as land cover mapping, regional planning and environmental monitoring.

1.7 Report Organisation

In this project, there are six chapters to be discussed. The content that discussed in each of the chapter are:

Chapter 1, this chapter explained about the introduction of the project. It also discussed about the problems facing in current situation, the project questions, and the project objectives which needed to be accomplished. Besides, this chapter also explained about the project scope and project contribution of the project.

Chapter 2, this chapter discussed about the literature review, previous research done by the researchers, such as the method used in image classification, and the challenges faced in current situation.

Chapter 3, this chapter explained about the methodology used when conducting this project. The procedures of each of the method were described here.

Chapter 4, this chapter discussed about the implementation of the project. The software and hardware used, and the procedures in conducting the project were clearly stated.

Chapter 5, this chapter explained about the result and analysis of the project. Comparison between the two techniques was carried out to obtain the best result to identify which techniques is more effective for this project.

Chapter 6, this chapter discussed about the conclusion of the project. Project summarization, project contribution, project limitation, and future works were discussed in this chapter.

1.8 Conclusion

Satellite images are one of the most important and powerful tools used to collect the photo of Earth. It contains useful information, such as the shape of the river, shape of the road, coverage of the urban area and forest area. There are a lot of techniques can be implemented to classify the satellite image. The techniques that focused in this project is Maximum Likelihood and ISODATA.

There are three objectives in this project, that is to identify the classification techniques in satellite image classification, to apply supervised classification method, Maximum Likelihood and unsupervised classification method, ISODATA for satellite image classification, and to evaluate the best technique in image classification. The area to be covered in this project is Melaka and the satellite image was downloaded from Google Earth. MATLAB software was used to process the images. At the end of the project, the satellite image was successfully classified using two different techniques, that is Maximum Likelihood and ISODATA. With these image classification techniques, the elements in the satellite image were recognized efficiently and accurately.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In the field of satellite image classification, there are many technologies being introduced to classify the element in the satellite image. The purpose of classification process is to arrange and sort all pixels in a digital image into one of the classes, and allow the system to recognize the element in the digital image. According to the previous research carried out by the researchers, there are a lot of techniques can be used to classify image, such as Minimum Distance, Maximum Likelihood, Parallelepiped, K-means and ISODATA. However, each of the techniques have both strengths and weaknesses, it is hard to find a technique that is suitable to classify all types of images.

Image classification is the process in computer vision that can classify an image according to its visual content and it is widely used in satellite image classification. There are a few methods can be used to classify the satellite image, and the purpose of this project is to identify the method that can produce a best result in satellite image classification.

2.2 Related Work/Previous Work

In this project, both supervised and unsupervised classification was covered. The two classification techniques to be implemented in this project are Maximum Likelihood and ISODATA. The summary of image processing is as shown in Figure 2.1.

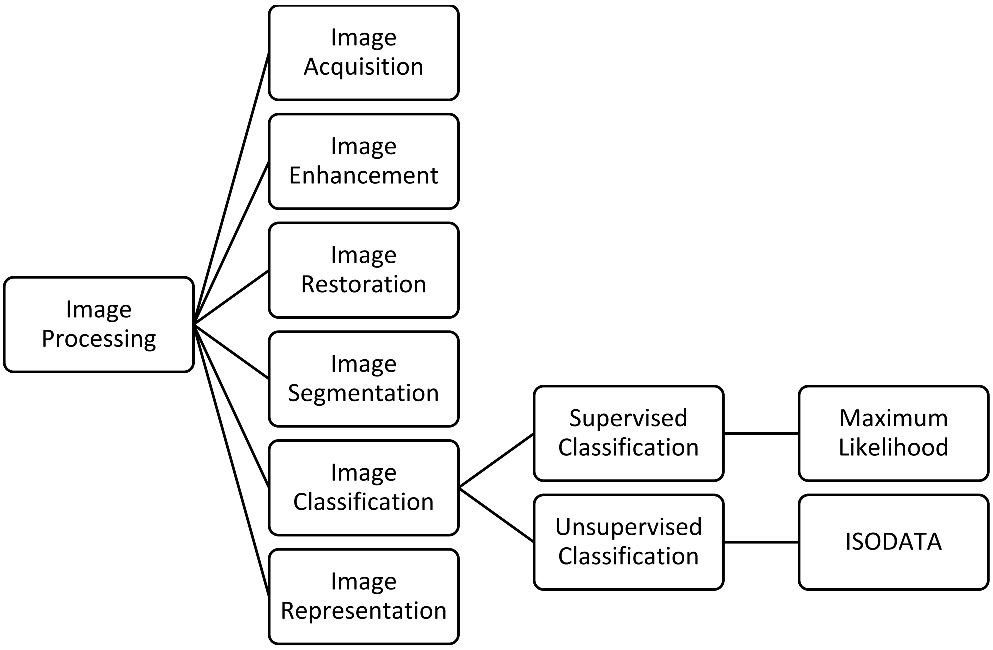

Figure 2.1 Image Processing Summary

Image classification can be divided into two categories, that is supervised classification and unsupervised classification. The two methods under supervised and unsupervised classification were discussed and implemented in this project.

2.2.1 Image Processing

Image processing is a process to extract useful data from certain image, by performing some operations on that image. It is one of the types of signal processing, and the process can be conducted by using programming languages such as MATLAB, Java or C++. The process of image processing is about input a source of image and get the output in the form of enhanced image, desired characteristic or features of the image by applying some mathematical operation.

There are three phases in image processing, which are use image classification tools to import the image, analyze and manipulate the image, and output is the final step where result can be altered image or report which is based on image analysis. The quality of the processed image remains the same after undergoing image processing steps.

Image processing can be divided into five categories, each category has their own purposes and function. The five categories of image processing are visualization, image restoration, image retrieval, pattern measurement and image recognition.

With the fast growth of technology in 21^{st} century, image processing is widely used in the field of technology and science. The range of image processing's application covered from medicine to agriculture, meteorology and entertainment. There is a few important image processing's applications in the field of technology and science include remote sensing, forecasting, feature extraction and pattern recognition.

2.2.2 Image Processing for Satellite Image

Satellite image is also known as Earth observation imagery. It consists of Earth image collected by Earth observation satellites. These Earth observation satellites are run by government or business institution around all the world. There are some business companies collect images by operating imagery satellites and licenses the image to governments or other businesses. The example of the business companies are Apple Maps, Google Maps and Baidu Maps.

Satellite image processing can be divided into several phrases: image enhancement, image restoration, image compression, image segmentation, representation description and image classification. The satellite image will be converted into digital form through these processes. The main intent of satellite image processing is to improve the quality of captured images, make the raw image easier to be analyze and obtain useful data from the image. This project focus only on image classification by using both supervised and unsupervised classification method. The selected classification techniques were explained in other subchapter.

Table 2.1 explain about the image processing phases. Each of the phases has their function and purposes.

Table 2.1 Image Process Techniques

Image Processing Techniques	Description
Image Enhancement	In image enhancement, the images are adjusted to be more suitable for display or further image analysis. The image can be adjusted by sharpen, brighten, eliminate noise, allowing the key features to be easier to identify.
Image Restoration	Image restoration is the process estimating a original image from a corrupted image. It improves the quality of blurred images, recover the resolution that was lost to the human eye during some degradation process.
Image Compression	Image compression is an implementation of the data compression which encodes actual image with some bits. The purpose of the image compression is to decrease the redundancy and irrelevance of image data to be capable to record or send data in an effective form.
Image Segmentation	Image Segmentation is the process by which a digital image is partitioned into various subgroups of pixels, which can decrease the complexity of the image, and simplify the image analysing process. The main intent of image segmentation is to divide an image into many sections for the further analysis and get only useful or required segment of an information.
Image Representation	Image Representation refers to the way that the delivered information, such as colour, is coded digitally and how the image is stored. Several open or patented standards were proposed to create, manipulate store and exchange digital images. They describe the format of image files, the algorithms of image encoding such as

	compression as well as the format of additional information often called metadata.
Image Classification	Image classification is the process of categorizing and labelling pixels in an image according to specific rules. One or more spectral or textural can be used to devise the categorization law. There are two methods of image classification: supervised classification and unsupervised classification.

2.2.3 Image Classification

Image Classification is the process of classifying pixels into a number of classes based on their data values. The pixel that conformed the rules to fit in the class will be assigned to a particular class.

Image classification techniques can be divided in to two categories, which are supervised classification and unsupervised classification. Supervised classification method required training set to classify other pixels in the image. The sample pixels for each land cover class in a satellite image is selected as training sets. Unsupervised classification analyzes the satellite image without the references of training data. The computer uses algorithms to calculate and decide the class of the pixels and categorize them into the related classes.

In Supervised classification, sample pixels that represent specific classes are selected by user and the training sites are used by the image processing software as references to classify other pixels in the image. Training data is needed in supervised classification technique and the training data is used to teach the classifier to define the decision boundary. The number of output classes can be decided by user.

Unsupervised classification is where the pixels with common characteristics are grouped without sample data and are based on the software analysis of an image using algorithms. The computer will decide which pixels are related and sorts the pixels into groups by using techniques. Users need to specify the number of output classed and decide the software algorism and there is no training set needed in unsupervised classification.

There are several supervised and unsupervised techniques can be applied to classify a satellite image. Supervised techniques included Maximum Likelihood, Minimum Distance, Parallelepiped, Support Vector Machines (SVM) and Artificial Neural Networks (ANN) whereas Unsupervised techniques included K-means and ISODATA. The techniques that focus in this project are Maximum Likelihood and ISODATA. In Table 2.2, some of the classification techniques were described.

Table 2.2 Classification Techniques

Classification Techniques	Description
Maximum Likelihood	Maximum Likelihood is a supervised classification technique. It is derived from Bayes theorem and makes use of discriminant function. The pixel with highest likelihood is grouped to the class which is related. Standard deviation and mean are evaluated and to assign the pixels into particular classes.
Minimum Distance	Euclidean distance from each unknown pixel to the mean vector for each classes are computed by using the mean vectors for each class in Minimum Distance. The pixels will be classified to the nearest class.
K-means	K-means is a clustering algorithm which divides observations into k clusters. (Mudassir Khan, 2017). The K cluster mean vectors were initialize randomly and each pixel will be assigned to any of the K clusters based on the minimum feature distance. Each cluster mean is recomputed after all pixels are grouped to the K clusters.
ISODATA	ISODATA is a improvement of k-means clustering algorithm to overcomes the weakness of k-means. If the separation distance in multispectral feature space is less than a specified value, the cluster will be merged. It also contains a rule of splitting clusters.

2.2.4 Image Classification based on Maximum Likelihood

Maximum Likelihood is one of the supervised classification methods. It can be expressed as statistical approach to pattern recognition and it is derived from Bayes theorem. The pixel is grouped into the class with highest likelihood after the likelihood of the pixel belonging to each of a predefined set of classes is evaluated and compared. With the Maximum Likelihood algorithm, probability density functions are built for each class based on the training data's spectral values.

The variances and covariances of the class are considered to assign the pixels to the classes represented. Assumed that the sample of the class is a normal distribution, and characterized the class with mean vector and the covariance matrix. The likelihood of each pixel is calculated and compared to decide and the pixels were assigned to the class with highest likelihood.

Maximum Likelihood uses training data to estimate the mean and the variance of each class based on Bayes probability formula which states that a posteriori probability that the pixel with the feature vector x is:

$$\mathbf{P}(\omega \mathbf{j} \mid \mathbf{x}) = \frac{\mathbf{P}(\mathbf{x}|\omega\mathbf{j})\mathbf{P}(\omega\mathbf{j})}{\mathbf{P}(\mathbf{x})}$$

Where P (ωj) is the priori probability, P(ωj |x) is the posterior probability, that is the probability of class being ωj with the feature value x. Beisdes, P(x| ωj) is the likelihood of ωj with the feature value x. (S. S Wanarse, et. al.,2014).

Besides that, the Likelihood of d categories is using this formula, whereby j is equal to one and d represents the number of classes.

$$\boldsymbol{P}(\omega\mathbf{j}) = \sum_{1}^{d} \boldsymbol{P}(\omega\mathbf{j}|\mathrm{x}) \ \boldsymbol{P}(\boldsymbol{x})$$

The formula of discriminant function is:

$$\boldsymbol{g_i}(\boldsymbol{x}) = \mathbf{ln}\,\boldsymbol{P}(\omega\mathrm{i}|\mathrm{x}) = -\frac{1}{2}(x - \mu_i) - \frac{\boldsymbol{d}}{2}\mathbf{ln}\,2\boldsymbol{\pi} - \frac{1}{2}\mathbf{ln}\,|\sum\nolimits_{\boldsymbol{i}}|$$

Furthermore, the Maximum Likelihood Estimation also can be calculated with this formula, whereby σ is standard deviation and μ is mean.

$$\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

2.2.5 Image Classification based on ISODATA

ISODATA is a method of unsupervised classification method. It is a improvement of K-means where the clusters can be varied by splitting a single cluster into two clusters or merging clusters into one single cluster to achieve a prespecified number of clusters. ISODATA algorithm is an iterative procedure, instead of just two passes, it allows a huge number pass through the remote sensing dataset until specified results are acquired. (Anon, n.d.). Figure 2.2 shows the general stage of the ISODATA algorithm.

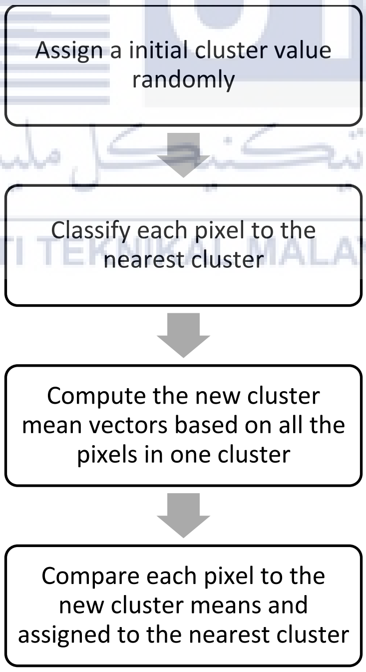

Figure 2.2 General Stage of ISODATA

Firstly, arbitrarily establishing N cluster means based on the means and standard deviations of the bands in the input file in which "N" is a number specified by the user (Anon. 2016). Next, each pixel was assigned to the nearby cluster according to the minimum distance criterion. (A. W. Abbas, et. al., 2016). Next, the new cluster mean vectors were calculated based on all the pixels in one cluster in step 3. The second and third step will be iterated until obtaining a little change between the iteration is obtained. After that, each of the individual pixels was re-compared to the new cluster means and the pixel was be sorted to the nearest cluster.

To improve the result of ISODATA algorithm, the clusters were split or merged. If the centers of two clusters are closer than a certain threshold or either the number of members (pixel) in a cluster is less than a certain threshold or, clusters are merged. Besides that, if the cluster standard deviation exceeds a predefined value and the number of members (pixels) is twice the threshold for the minimum number of members, clusters were split.

2.3 Critical review of current problem and justification

Research conducted by **Satish S Wanarse, et. al. (2014)** in their research paper entitled "*Class Quantification of Aerial Images using Maximum Likelihood Estimation.*", mentioned the unique properties of Maximum Likelihood estimator to prove its ascendancies. The ascendancies of Maximum Likelihood are Consistency, Parameterization Invariance, Efficiency and Sufficiency. In their research, firstly the number of categories is decided, then the training pixels is selected with some consideration such as size, shape, number of training areas, placement and uniformity. The mean and variance of each category were calculated from the training pixels and Maximum Likelihood Classifier is used to assign the pixels into different categories. In conclusion, Maximum Likelihood Classifier developed in this research works very well and was able to classify each pixel to its proper class. Proper selection of the area is one of the main factors that can produce a high accuracy result.

Based on the research conducted by **Adil Nawaz, et. al. (2015)** in their research paper "*Simplified Maximum Likelihood Classification for Hyperspectral Data in Cluster Space*", discussed that there are many methods can be used to classify the image such as Parallelepiped, Maximum Likelihood, and Mahalanobis Distance. An accurate decision rule which is less consuming and require lesser resources need to be found between these classification methods. The imagery was divided into four classes based on Maximum Likelihood and there is a good separation between different classes in the satellite image. The vegetation and non-vegetation classes are nicely separated and the overall accuracy of Maximum Likelihood classifier is 99.1%. The Kappa coefficient of Maximum Likelihood is 0.96 and it is nearly perfect. Besides, they also found that spatial resolution has no major effect on the classification accuracy, but it affects the efficiency of classification.

Based on the **Pushpendra Singh Sisodia, et. al. (2014)** in their research paper entitled "*Analysis of Supervised Maximum Likelihood Classification for Remote Sensing Image*", discussed about classify remoted sensed images using Maximum Likelihood Classification. They mentioned that remotely sensed images are widely used in mapping and keep track of the urban change, enable the planners to manage the city and understand deeply about the growth in urban. Therefore, Maximum Likelihood Classification is used to classify the remotely sensed images in the way of recognizing different patterns in the images, including urban area, vegetation, water, hill and land. Maximum Likelihood is used to achieve an accurate classification result. Lastly, they found that there are very less chances of Maximum Likelihood misclassification if the probability of the pixel from class is normal distributed. However, the training set should be chosen properly since normal distribution is not always achieved.

Researchers **Eka Miranda, et. al. (2016)** conducted a research about "*Classification of Land Cover from Sentinel-2 Imagery Using Supervised Classification Technique (preliminary study)*". In this research, they focused on the satellite image classification with cloud cover consideration. They found that Maximum Likelihood provides index of certainty associated with each pixel chosen and the assignment class (Eka Miranda, et. al., 2016). However, the weakness of Maximum Likelihood Classification is the statistical value cannot always deal with complicated images, some of the pixels cannot be classified accurately.

Based on the research carried out by **Asmala Ahmad & Suliadi Firdaus Sufahani (2012)** in their research paper "*Analysis of Landsat 5 TM Data of Malaysian Land Covers Using ISODATA Clustering Technique*", ISODATA was used as a classification method to classify the land covers recorded by Landsat 5 TM Satellite. They found that K-means assumes the number of clusters is known as a priori whereas ISODATA allows different numbers of clusters to be specified. The main advantage of ISODATA compared to K-means algorithm is that ISODATA allows different numbers of clusters to be specified, in which the cluster can be ranged from a minimum to a maximum number of clusters (A. Ahmad & S. F. Sufahani, 2012). Hence, ISODATA is more flexible if compared to K-means. Besides, they also discussed about the main steps in ISODATA clustering.

Based on research "*An Entropy-Based Multispectral Image Classification Algorithm*" conducted by **D. Long & V. P. Singh (2013)**, found that supervised and unsupervised classification have both strength and weakness. Unsupervised classification will be more suitable if the field samples are not available. For example, if want to map a large area which is not well known before this, unsupervised classification such as ISODATA and K-means is a better strategy. In addition, classification method can be divided into parametric and non-parametric classification. Maximum Likelihood and ISODATA is grouped as parametric classification. Parametric classification assumed that the digital number (DN) vectors of different classes in remotely sensed images follow a multivariate normal distribution (D. Long & V. P. Singh, 2013).

According to researcher **M. Förster & B. Kleinschmit (2014)** in their research "*Significance Analysis of Different Types of Ancillary Geodata Utilized in a Multisource Classification Process for Forest Identification in Germany*", mentioned that ISODATA is one of the most efficient and approved methods to extract land-cover information and it is an efficient method of partitioning multispectral image feature space. They found that ISODATA clustering was able to find coherent pattern accurately. Also, ISODATA can give negative indicators to classification where the spectral value is only partly utilized.

Research conducted by **C. K S, et. al. (2020)** in their research paper "*Classification of Homogeneous Sites using IRSP5 Satellite Imagery*", discussed about the performance of ISODATA and Mahalanobis Distance algorithm in satellite image classification. They compared the performance between these two techniques and concluded that ISODATA classification provides 89% accuracy whereas Mahalanobis Distance provides 93% accuracy.

Table 2.3 shows the comparative study for the image classification techniques.

Table 2.3 Comparative Study for Image Classification Techniques

Author and Year	Image Classification Techniques				
	Maximum Likelihood	Parallelepiped	Mahalanobis Distance	ISODATA	K-means
S. S Wanarse, et. al. (2014)	x				
A. Nawaz, et. al. (2015)	x	x	x		
P. S. Sisodia, et. al. (2014)	x				
E. Miranda, et. al. (2016)	x				
A. Ahmad & S. F.				x	

Sufahani (2012)					
D. Long & V. P. Singh (2013)	x			x	x
M. Förster & B. Kleinschmit (2014)			x	x	
C. K S, et. al. (2020)				x	

2.4 Proposed Solution/further project

There are a lot of techniques can be used to classify the satellite image. For example, unsupervised classification method is more suitable if the field samples are not available. Nevertheless, supervised classification method is more suitable to be implemented. In this project, both of the classification techniques were implemented and the suitable techniques to be used were identified. The satellite images of Malacca city was download from Google Earth and the satellite image was processed with the techniques mention above, compare the classification result to evaluate the most suitable technique in satellite image classification.

2.5 Conclusion

In this chapter, the related or previous work done by the researcher before were discussed. There are a lot of techniques can be used to classify the satellite image. The techniques are divided into two categories, which are supervised and unsupervised classification. The example of popular techniques in supervised classification are Maximum Likelihood, Minimum Distance, Parallelepiped, Support Vector Machines (SVM) and Artificial Neural Networks (ANN). For unsupervised classification, there are two techniques which are ISODATA and K-means. Every technique has their advantages and disadvantages, it is hard to find a technique which is perfect in image classification.

Besides, the critical review of current problem and justification also discussed. The current problem faced by the researched with the proposed classification methods is listed out and well elaborated in this chapter, therefore the solution can be suggested to solve the current problems.

There are a lot of techniques can be implemented to classify the satellite image. Based on the previous research, two techniques which are Maximum Likelihood and ISODATA were decided to implement in this project. After that, comparison between the classification results were carried out to identify the best classification technique.

In next chapter, the methodology of this project was discussed, describing the activities that done in every stage. The activities was recorded in a Gantt chart.

CHAPTER 3

PROJECT METHODOLOGY

3.1 Introduction

This chapter consists of four sub-chapter, which are introduction, methodology, project milestones and conclusion. In methodology, the stages of the project were clearly listed out and the activities in every stage were explained. In this project, there are four stages involved, which are image pre-processing, image classification using different classifiers, accuracy assessment and evaluation of classification method. Besides, the detail procedures of Maximum Likelihood and ISODATA classification techniques were described in this chapter. Furthermore, a project milestone wad scheduled to ensure a smooth project flow and ensure that the project can be completed according to the estimated time.

3.2 Methodology

Methodology is developed to make the research well planning and allow the project to be done within the estimated time. There are a few stages in image classification and in every stage, and several techniques can be used to conduct the activity. Figure 3.1 shows the detail process of image classification that implemented in this project.

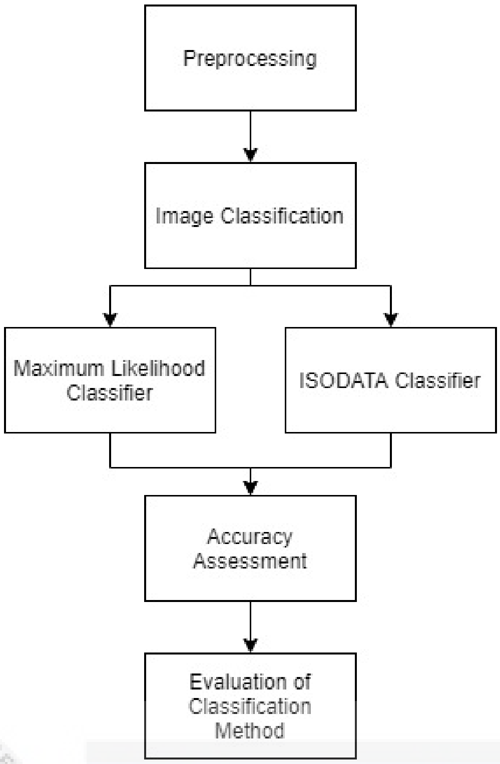

Figure 3.1 Detail Process of Image Processing

3.2.1 Preprocessing

First, the satellite image underwent image preprocessing which is collect the data and improve the image by enhancing the important features in the image. The purpose of preprocessing is to ensure the quality of the image and benefits the further classification process. Raw image may contain flaws and deficiencies such as geometric error, atmospheric constituents and radiometric error. Image preprocessing correct the image, make the image clearer in order to obtain a better image.

3.2.2 Image Classification

Image classification is the process assign land cover classes into pixels. For unsupervised classification, it is pixel-based computer automated. The pixels are grouped together according to their similarity in spectral. The computer uses feature space to analyze and sort the pixels into classes. In supervised classification, representative samples for each class are chosen as training sites. Then, the software will use these training sites and applies them to the entire satellite image. The classification techniques are used to train and extract the features in the satellite image.

3.2.3 Accuracy Assessment

In this stage, the accuracy of classification results were assessed using confusion matrix. The percentage of accuracy of the two techniques were calculated and the result of accuracy were recorded in a table.

3.2.4 Evaluation of Classification Method

In this stage, the classification result by using the two techniques, which are Maximum Likelihood and ISODATA were obtained. The more suitable classification technique was determined based on the result of satellite image classification.

3.3 Classification Techniques

In this project, there are two techniques to be used, which are Maximum Likelihood and ISODATA. The classification result of these techniques were compared to identify which techniques give the most accurate result.

3.3.1 Maximum Likelihood

Maximum Likelihood is a type of supervised classification method and training data is needed. The stages of Maximum Likelihood classification method is shown in Figure 3.2.

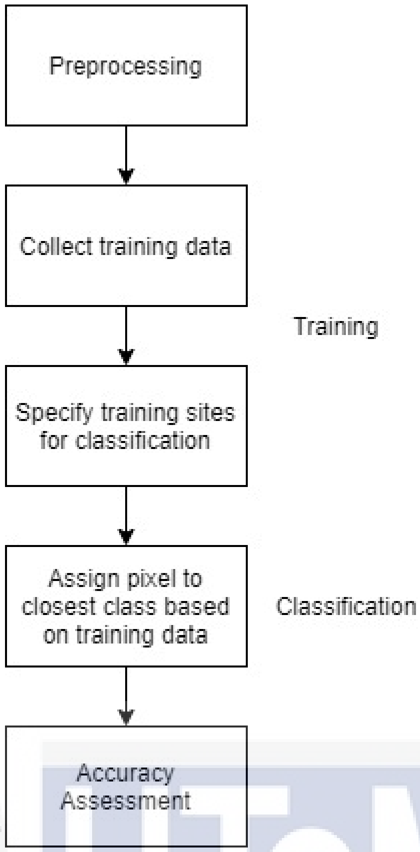

Figure 3.2 Stage of Maximum Likelihood classification

Initially, pre-processing was carried out to enhance the quality of the satellite image. Pre-processing corrected the satellite image by removing flaws and deficiencies such as geometric error, atmospheric constituents and radiometric error.

Next, the area that used as the training sites were defined based on the color image. Establish the area and the area will be delineated according to its spectrally homogenous sub-areas. The representative samples for each class are chosen as training sites.

After the training sites are set up, Maximum Likelihood classifier was used to classify the satellite image. The probability of each given pixel was computed and the pixel was categories into classes based on the highest probability. The calculation of Maximum Likelihood Classifier is based on the Bayesian probability formula.

The last step of classification is accuracy assessment. The measurements of accuracy assessment were carried out to calculate the probability of error of Maximum Likelihood classifier.

3.3.2 ISODATA

ISODATA is a type of unsupervised classification method and no training data is required. The stages of ISODATA classification method is shown in Figure 3.3.

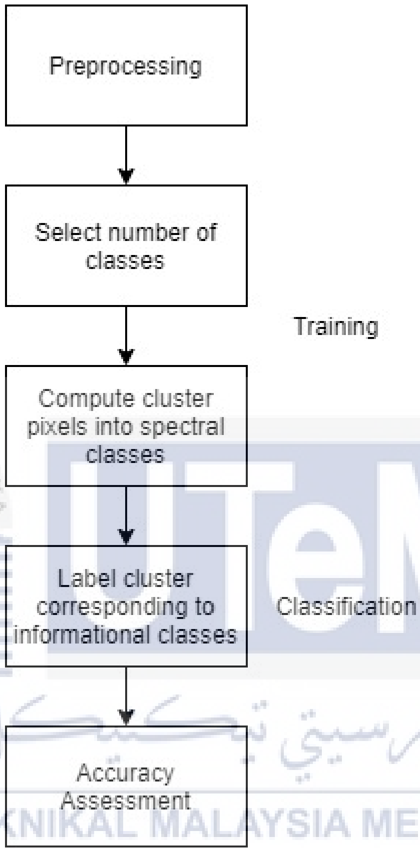

Figure 3.3 Stage of ISODATA classification

For ISODATA classification method, firstly improve the quality of the satellite image using pre-processing. The raw image may contain flaws and deficiencies and the purpose of pre-processing is to remove these errors and make the image clearer.

Next, Specify the number of classes that want to generate in the classification process. Then, the clusters pixels were grouped into cluster according to their properties.

For unsupervised classification, it is pixel-based computer automated. In the classification stage, the pixels were grouped together according to their similarity in spectral. The computer uses feature space to analyze and sort the pixels into classes.

Lastly, assess the accuracy of classification. The probability of error for ISODATA classifier were calculated.

3.4 Project Milestones

Project Milestones and Gantt Chart were created to ensure the project can be completed in estimated time. Project Milestones is listed in Table 3.1 and Gantt Chart is listed in Table 3.2.

Table 3.1 Project Milestone

PSM 1		
Week	Activity	Note/Action
1 (15/3 – 21/3)	Submit proposal to supervisor for approval	Deliverable – Proposal Action – Student
	Proposal assessment and verification	Action – Supervisor
2 (22/3 – 28/3)	Proposal correction and improvement Submit final proposal to PSM Committee via email	Action – Student
	Proposal Approval List of Supervisor/Title	Action – Committee PSM/PD
3 (29/3 – 4/4)	Submit proposal via PSM Ulearn	Action – Student
	Develop Chapter 1 (Introduction of project)	Action – Student
4 (5/4 – 11/4)	Submit Chapter 1	Deliverable – Chapter 1 Action – Student
5 (12/4 – 18/4)	Develop Chapter 2 (Literature Review)	Action – Student
6 (19/4 – 25/4)	Submit Chapter 2	Deliverable – Chapter 2
	Student Status	Action – Supervisor, Committee PSM/PD Warning Letter 1

7 (26/4 – 2/5)	Develop Chapter 3 (Project Methodology)	Action – Student
8 (3/5 – 9/5)	Submit Chapter 3	Deliverable – Chapter 3 Action – Student
9 (10/5 – 16/5)	MID SEMESTER BREAK	
10 (17/5 – 23/5)	Develop Chapter 4 (Implementation)	Action – Student
	Student Status	Action – Supervisor, Committee PSM/PD Warning Letter 2
11 (24/5 – 30/5)	Project Demo	Action – Student, Supervisor
	Determination of student status (Continue/Withdraw)	Action – Supervisor, Committee PSM/PD
12 (31/5 – 6/6)	Project Demo Submit PSM1 Report	Action – Student, Supervisor
13 (7/6 – 13/6)	Project Demo PSM1 Report Schedule the presentation	Action – Student, Supervisor Action – Committee PSM/PD Presentation Schedule
14 (14/6 – 20/6)	Project Demo	Deliverable – Complete PSM 1 Draft Report Action – Student, Supervisor
15 (21/6 – 27/6)	Final Presentation Submit PSM 1 Report onto PSM Ulearn	Action – Student, Supervisor, Evaluator, Committee
16 (28/6 – 4/7)	Revision Week Correction of the draft report. Submit PSM1 Logbooks and EoS Survey Form.	Action – Student, Supervisor
PSM2		

1 (19/7 – 25/7)	Improvement for Chapter 4	Deliverable – Chapter 4 Action – Student, Supervisor
2 (26/7 – 1/8)	Develop Chapter 5 (Testing and Analysis)	Action – Student
3 (2/8 – 8/8)	Chapter 5	Deliverable – Chapter 5 Action – Student, Supervisor
	Student Status	Action – Supervisor, Committee PSM/PD Warning Letter 1
4 (9/8 – 15/8)	Chapter 5	Deliverable – Chapter 5 Action – Student, Supervisor
	Project Progress	Deliverable – Chapter 5 Action – Student
5 (16/8 – 22/8)	Develop Chapter 6 (Project Conclusion)	Action – Student
	Submit Chapter 5 & 6	Action – Student, Supervisor
	Student Status	Action – Supervisor, Committee PSM/PD Warning Letter 2
	Presentation Schedule	Action – Committee PSM/PD Presentation Schedule
6 (23/8 – 29/8)	Project Demo PSM2 Draft Report	Deliverable – PSM2 Draft Report Action – Student, Supervisor, Evaluator
7 (30/8 – 5/9)	Final Presentation & Project Demonstration	Final Presentation Project Demonstration

		Action – Student, Supervisor, Evaluator
8 (6/9 – 12/9)	Final Examination Weeks	Deliverable – Complete PSM2 Logbooks Action – Student, Supervisor
9 (13/9 – 19/9)	Inter-Semester Break Submit final complete report Upload PSM2 report to Ulearn	Deliverable – Complete Final PSM Report, Complete PSM2 Logbooks, Plagiarism Report Action – Student, Supervisor

Table 3.2 Gantt Chart

No	Task	Week															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	Prepare proposal																
2	Proposal Improvement																
3	Chapter 1 Introduction																
4	Chapter 2 Literature Review																
5	Chapter 3 Methodology																
6	Project Demonstration																
7	Chapter 4 Analysis and Design																
8	Chapter 5 Implementation																
9	PSM Report																
10	PSM1 Final Presentation																
11	Draft report correction based on supervisor's comments																
12	Chapter 6 Testing																
13	Chapter 7 Project Conclusion and Final Presentation																

3.5 Conclusion

In this chapter, the methodology of the project were discussed. The activities in every stage were clearly described to make the flow of project become smoother and ensure the project can be completed within the estimated time. There are four stages in our project, which are image preprocessing, image classification, accuracy assessment and evaluation of method.

Besides that, the detail procedure of the two classification methods which are Maximum Likelihood and ISODATA were clearly listed out. The procedures were plotted in a flow chart, enable us to understand and know the following step. In addition, the project milestone is arranged to ensure the task was completed within the estimated time.

In next chapter, project implementation was discussed, in which the hardware and software that required in the project were listed out and clearly described. Also, the steps of experiments were elaborated in next chapter.

CHAPTER 4

IMPLEMENTATION

4.1 Introduction

In this chapter, the environment setup that used in the project were discussed. The software and hardware requirements that used in this project were listed out to ensure a smooth process when conducting our project. Furthermore, the steps that underwent to classify the satellite image were explained. The techniques that implemented were Maximum Likelihood and ISODATA. The coding that used to process the image were listed in this chapter. Besides, the expected output after applying the two methods were presented in this chapter as well.

4.2 Environment setup

In this project, several software and hardware requirements to carry out the classification process were identified and installed. The software and hardware are listed in Table 4.1 and Table 4.2 and the functions are clearly described.

4.2.1 Software Requirements

In this project, there are two software are used to conduct our project. The software is MATLAB R2021a and Microsoft Word 2016. The function of the software is mentioned in the Table 4.1.

Table 4.1 Software Requirement

Software	Function
MATLAB R2021a	This software was used to process the satellite image and classify the elements

	in the satellite image using different classification methods.
Microsoft Word	This software was used for documentation, which are writing proposal and report.

4.2.2 Hardware Requirements

In this project, there are one hardware are used to conduct the project. The hardware is Asus VivoBook S14 M433IA. The function of the hardware is mentioned in the Table 4.2.

Table 4.2 Hardware Requirement

Hardware	Function
Asus VivoBook S14 M433IA 	Asus laptop was used to write report and run the MATLAB software to carry out the classification process.

4.3 Satellite Image Classification Process

In this project, two techniques were used to classify the satellite image which are supervised classification technique, Maximum Likelihood and unsupervised classification technique, ISODATA. In this section, the two techniques were implemented to classify the satellite image and the more suitable technique were identified. The original satellite image is as shown in Figure 4.1.

Figure 4.1 Original Satellite Image

4.3.1 Maximum Likelihood Classifier

Firstly, the training data was prepared by cropping some of the parts in the satellite image. There are five classes to be classify, which are water, forest, road, building and barren land. Therefore, five sets of training data were prepared.

Next, both satellite image and training data were imported into MATLAB. The mean and standard deviation of each layer (R, G, B) of each class were calculated. After that, Maximum Likelihood Estimation's formula was applied to calculate and assign the pixels to classes relatively. Lastly, the color of classes was changed in the final image to distinguish the classes. Blue color is for water, white color for building, green color for forest, red color for barren land and black color for road.

The flow of the Maximum Likelihood classification is shown in Figure 4.2.

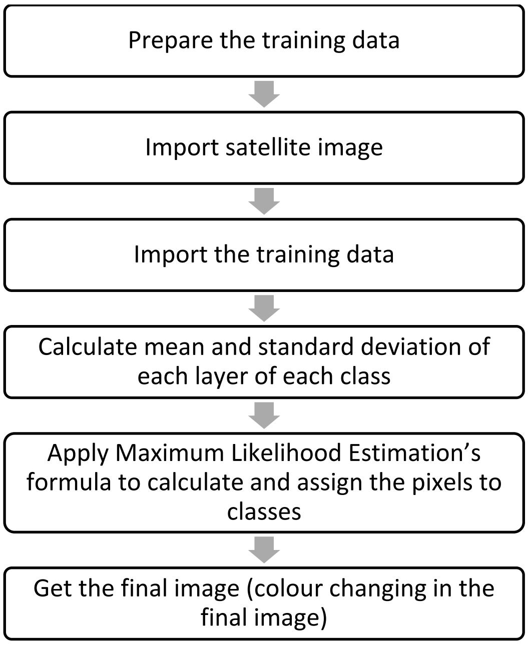

Figure 4.2 Flow of Maximum Likelihood Classification

Figure 4.3 shows the coding of importing image into MATLAB.

```
img = double(imread('map.png'));  % read the image
```

Figure 4.3 Coding of importing satellite image

Figure 4.4 shows the coding of importing training data into MATLAB.

```
Water_I = double(imread('water1.jpg'));         % class 1
Building_I = double(imread('Buildings.jpg'));   % class 2
Forest_I = double(imread('Forest.jpg'));        % class 3
Barren_I = double(imread('Barren.jpg'));        % class 4
Road_I = double(imread('Road.jpg'));            % class 5
```

Figure 4.4 Coding of importing training data

Figure 4.5 shows the coding of calculating mean and standard deviation of each layer (R, G, B) of each class.

```
% mC1L1 (the mean for class 1 layer 1)
mC1L1 = 0;
for counter = 1:length(L1c1)
    mC1L1 = mC1L1 + L1c1(counter);
end
mC1L1 = mC1L1/length(L1c1);

% dC1L1
dC1L1 = 0;
for counter = 1:length(L1c1)
    dC1L1 = dC1L1 + (L1c1(counter)-mC1L1)^2;
end
dC1L1 = dC1L1/length(L1c1);
dC1L1 = sqrt(dC1L1);
```

Figure 4.5 Coding of calculating mean and standard deviation

Figure 4.6 and Figure 4.7 shows the coding of applying Maximum Likelihood Estimation's formula to calculate and assign the pixels to classes.

```
Pc(1) = (1/(dC1L1*sqrt(2*pi)))*exp(-0.5*((img(R,C,1)-mC1L1)/dC1L1)^2)*(1/(dC1L2*sqrt(2*pi)))
Pc(2) = (1/(dC2L1*sqrt(2*pi)))*exp(-0.5*((img(R,C,1)-mC2L1)/dC2L1)^2)*(1/(dC2L2*sqrt(2*pi)))
Pc(3) = (1/(dC3L1*sqrt(2*pi)))*exp(-0.5*((img(R,C,1)-mC3L1)/dC3L1)^2)*(1/(dC3L2*sqrt(2*pi)))
Pc(4) = (1/(dC4L1*sqrt(2*pi)))*exp(-0.5*((img(R,C,1)-mC4L1)/dC4L1)^2)*(1/(dC4L2*sqrt(2*pi)))
Pc(5) = (1/(dC5L1*sqrt(2*pi)))*exp(-0.5*((img(R,C,1)-mC5L1)/dC5L1)^2)*(1/(dC5L2*sqrt(2*pi)))
```

Figure 4.6 Coding of applying MLE's formula

```
*exp(-0.5*((img(R,C,2)-mC1L2)/dC1L2)^2)*(1/(dC1L3*sqrt(2*pi)))*exp(-0.5*((img(R,C,3)-mC1L3)/dC1L3)^2);
*exp(-0.5*((img(R,C,2)-mC2L2)/dC2L2)^2)*(1/(dC2L3*sqrt(2*pi)))*exp(-0.5*((img(R,C,3)-mC2L3)/dC2L3)^2);
*exp(-0.5*((img(R,C,2)-mC3L2)/dC3L2)^2)*(1/(dC3L3*sqrt(2*pi)))*exp(-0.5*((img(R,C,3)-mC3L3)/dC3L3)^2);
*exp(-0.5*((img(R,C,2)-mC4L2)/dC4L2)^2)*(1/(dC4L3*sqrt(2*pi)))*exp(-0.5*((img(R,C,3)-mC4L3)/dC4L3)^2);
*exp(-0.5*((img(R,C,2)-mC5L2)/dC5L2)^2)*(1/(dC5L3*sqrt(2*pi)))*exp(-0.5*((img(R,C,3)-mC5L3)/dC5L3)^2);
```

Figure 4.7 Coding of applying MLE's formula

Figure 4.8 and Figure 4.9 shows the coding of changing the color of classes in the final image.

```
% define the colors of classes%
blue = [0 0 255];     %water
white = [255 255 255];    %building
green = [0 128 0];     %forest
black = [0 0 0];     %road
red = [255 0 0];    %barren land
```

Figure 4.8 Coding of defining the colors of classes

```
% color changing in the final image.
  if class == 1
      Final_I(R,C,1:3) = blue;
  elseif class == 2
      Final_I(R,C,1:3) = white;
  elseif class == 3
      Final_I(R,C,1:3) = green;
  elseif class == 4
      Final_I(R,C,1:3) = red;
  elseif class == 5
      Final_I(R,C,1:3) = black;
  end
```

Figure 4.9 Coding of color changing in the final image

Figure 4.10 shows the coding of getting the final output.

```
imshow(Final_I)
```

Figure 4.10 Coding of getting the final output

Figure 4.11 shows the final result obtained after color changing.

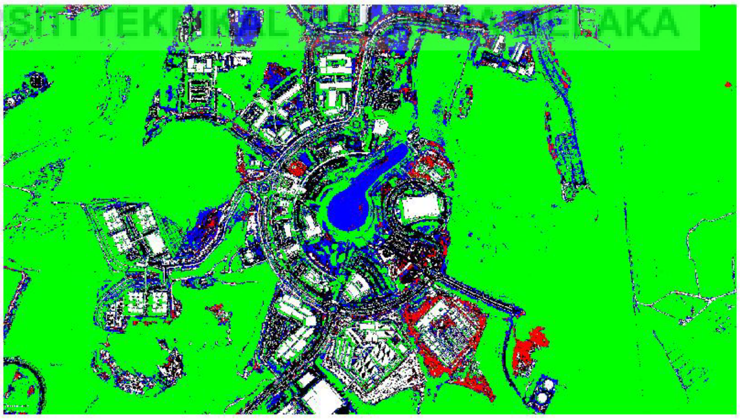

Figure 4.11 Final result of Maximum Likelihood

4.3.2 ISODATA Classifier

Firstly, initialize the expected number of cluster centers, minimum number of samples in each cluster, standard deviation, minimum distance between two clusters, maximum number of cluster center that can be combined and number of iterations. The information was needed in the calculation later.

Next, the satellite image was imported into MATLAB. After that, the iteration was started to find the nearest distance between each pixel and center. If the standard deviation calculated is larger than defined standard deviation, the class splits. If the distance between cluster is smaller than defined threshold, then the classes merge. The clusters with small number of elements were removed. The iteration was continued until the threshold was reached.

After that, the color of classes in the final image was changed to distinguish the classes. The color used is blue, white, green, yellow and black.

Figure 4.12 shows the process of ISODATA classification.

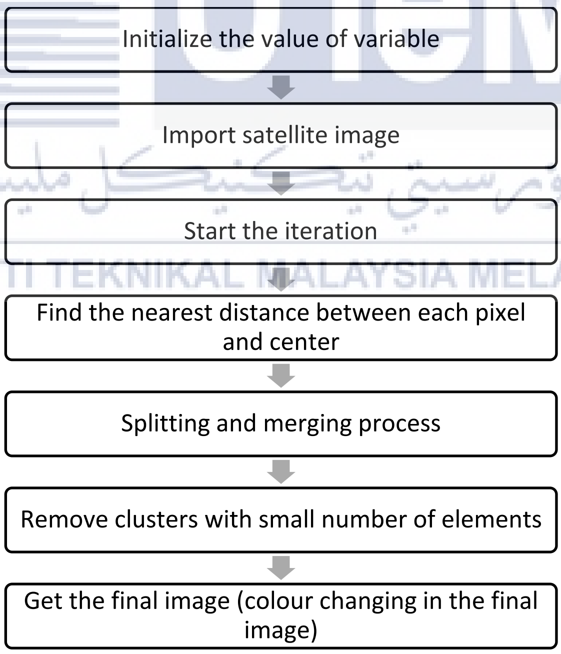

Figure 4.12 Process of ISODATA classification

Figure 4.13 shows the coding of initialize value of variable.

```
k = 10;  %expected no of cluster centers
leastNum = 1;    %minimum number of sample in each
classStandard = 1;  %Standard Deviation of sample
minDis = 1;    %Minimum distance between 2 cluster
maxCombine = 20;    %Maximum number of cluster cen
iterationTime = 100;  %Number of iteration
```

Figure 4.13 Coding of initialize the value of variable

Figure 4.14 shows the coding of importing satellite image into MATLAB.

```
%input image
A = imread('map.jpg');
```

Figure 4.14 Coding of importing satellite image

Figure 4.15 shows the coding of finding the nearest distance between each pixel and center.

```
while I<=iterationTime %iteration start
    class=zeros(currentNum,r*c);
    classCounter = zeros(currentNum,1);
    for i=1:num
        [value,index]=min(distance(i,:)); %Find the nearest distance between
        dtemp=[i];
        classCounter(index) = classCounter(index) + 1;
        class(index,classCounter(index))=dtemp;
    end
```

Figure 4.15 Coding of finding the nearest distance between

Figure 4.16 shows the coding of splitting process.

```
if flag==1   %split
    Cstandard=ComputCstandard(class,img,centre,currentNum,classCounter);
    %calculate the standard deviation of sample and center
    maxCstandard = zeros(currentNum,1);
    for i=1:currentNum
        [value,index]=max(Cstandard(i,:));
        maxCstandard(i)=value;
        if maxCstandard(i)>classStandard
            if (classAverage(i)>totalAverage && size(classCounter(i),2)>2*(leastNum+1))||currentNum<=k/2
                %if class average( between 2 clusters) more than total average, split it
                lcentre=centre(i,:)-factor*maxCstandard(i);
                hcentre=centre(i,:)+factor*maxCstandard(i);
                temp=centre((i+1):currentNum,:);
                currentNum=currentNum+1;
                centre(i,:)=lcentre;
                centre(i+1,:)=hcentre;
                centre((i+2):currentNum,:)=temp;
            end
        end
    end
```

Figure 4.16 Coding of splitting process

Figure 4.17 shows the coding of merging process.

```
elseif flag==0 % merge
    if I==iterationTime % Last Iteration
        minDis=0;
    end
    centreDis=ComputCentreDistance(centre,currentNum);
    %calculate all center Distance
    [indrow indcol] = find(centreDis<minDis);
    %  find the center which is small than the centre Distance
    [indrow indcol]=selectsort(centreDis,indrow,indcol);
    for i=1:size(indrow,1)
        if i<=maxCombine
            lengthI=size(class(indrow(i),:),2);
            lengthJ=size(class(indcol(i),:),2);
            tempCentre(i,:)=(1/(lengthI+lengthJ))*(lengthI*centre(indrow(i),:)+lengthJ*centre(indcol(i),:));
            ss = size(tempCentre) % to be removed
            if indrow(i)<indcol(i)
                centre(indrow(i),:)=tempCentre(i,:);
                temp=centre((indcol(i)+1):currentNum,:);
                currentNum=currentNum-1;
                centre(indcol(i):currentNum,:)=temp;
            else
                centre(indcol(i),:)=tempCentre(i,:);
                temp=centre((indrow(i)+1):currentNum,:);
                currentNum=currentNum-1;
                centre(indrow(i):currentNum,:)=temp;
            end
        else
            break;
        end
    end
end
```

Figure 4.17 Coding of merging process

Figure 4.18 shows the coding of removing clusters with small number of elements.

```
% remove clusters with small number of elements

for i=1:currentNum    % number of clusters
    if size(classCounter(i),2)<leastNum
        class=move(class,i);
        currentNum=currentNum-1;
    end
end
```

Figure 4.18 Coding of removing clusters with small number of elements

Figure 4.19 shows the coding of applying color changing and getting the final image.

```
COLOR(1,1:3) = [0 0 255];      % blue
COLOR(2,1:3) = [255 255 255]; % white
COLOR(3,1:3) = [0 128 0];      % green
COLOR(4,1:3) = [255 215 0];    % gold
COLOR(5,1:3) = [0 0 0];        % black
COLOR(6,1:3) = [255 0 0];      % red


% Getting the final Image
for i = 1:size(class,1)
    for j = 1:size(class(i,:),2)
        if class(i,j) ~= 0
            Final_Image(INDEX(class(i,j),1),INDEX(class(i,j),2),1:3) = COLOR(i,1:3);
        end
    end
end
```


Figure 4.19 Coding of applying color changing and getting the final image.

Figure 4.20 shows the final result of ISODATA.

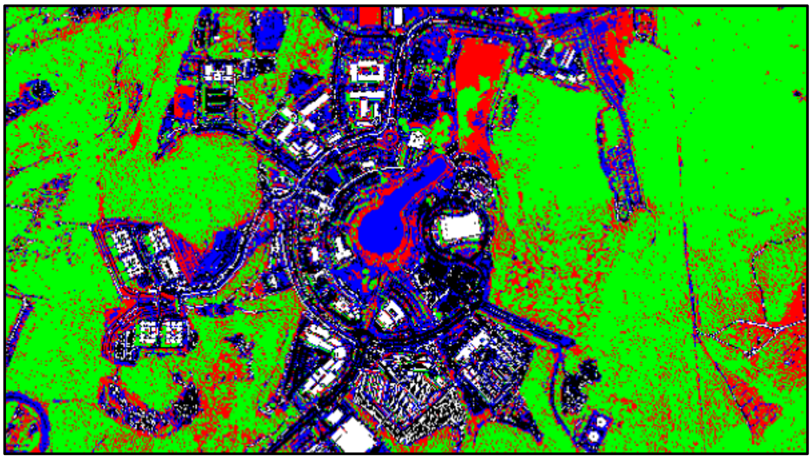

Figure 4.20 Final result of ISODATA

4.4 Conclusion

In this chapter, the hardware and software required in this project were stated. The software required are MATLAB 2021a and Microsoft Office 2016. MATLAB was used to process the satellite image to identify the elements in the satellite image whereas Microsoft Office was used in writing report. Besides, ASUS Laptop was used to write report and run the MATLAB software to carry out the classification process.

The flow of classifying the satellite image using Maximum Likelihood and ISODATA classifier was listed in this chapter. All the coding that applied in MATLAB and the final output was shown in this chapter. Based on the result shown, Maximum Likelihood has a better result comparing to ISODATA.

In next chapter, the result and analysis were discussed. The classification process was repeated to verify the result in Chapter 4. The most suitable classification technique was identified in next chapter.

CHAPTER 5

TESTING

5.1 Introduction

In this chapter, the results that obtained from the previous chapter were tested and the overall accuracy of the satellite image classification were evaluated. Based on the overall accuracy, the classification method with higher accuracy was identified. Therefore, the objective of this project was met.

5.2 Test Results and Analysis

In this subchapter, the classification results obtained from the previous chapter were analyzed. The accuracy of the classification results was evaluated to identify the best technique in satellite image classification. There are three stages in testing, which are pre-processing, testing process and result analysis.

5.2.1 Pre-processing

Before the testing process take part, pre-processing was carried out to benefits the further testing process. As mentioned in the previous chapter, there are five classes to be classified in this project, which are water, road, building, forest and barren land. A ground truth image was prepared using manual classification: each of the classes was classified manually and painted with different color. Then, the ground truth image and classified images were resized into same size. Figure 5.1 shows the ground truth image which is manually classified and painted.

Figure 5.1 Ground Truth Image

5.2.2 Testing process

Testing process was started with getting the matrices for both ground truth image and classified images. The matrices were used to obtain confusion matrix and compute sensitivity, specificity and overall accuracy. The flow of the testing process is shown in Figure 5.2.

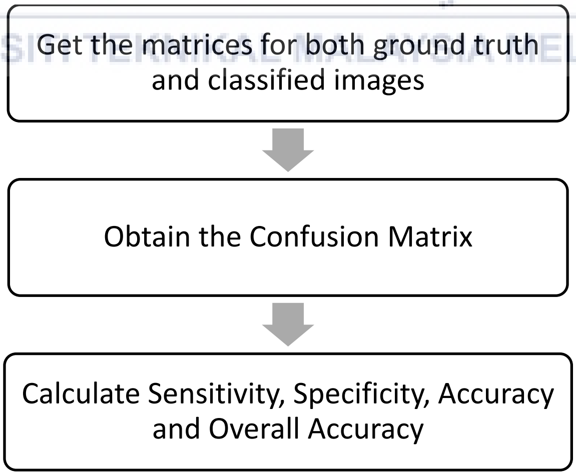

Figure 5.2 Flow of testing process

Firstly, the matrices for both ground truth image and classified images were obtained. To smoothen the testing process, the five classes were assigned with number

1 to 5: Water is class 1, Building is class 2, Forest is class 3, Barren Land is class 4 and Road is class 5. There are two important terms in Confusion Matrix's concept, which is Actual Value and Predicted Value. In this project, Ground Truth Image is the Actual Value whereas Classified Image is the Predicted Value. The Actual Value was compared to the Predicted Value to get the True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) value and these values were used in the calculation.

There are four metrics to be calculated in this project, which are Accuracy for each class, Overall Accuracy, Sensitivity and Specificity. Accuracy was calculated by dividing correctly classified pixels by total number of pixels of the class. Overall Accuracy is the total number of pixel which is correctly classified divided by the total number of pixels of the image. Sensitivity is TP divided by the sum of TP and FN whereas Specificity is TN divided by the sum of TN and FP.

5.2.2.1 Maximum Likelihood's Testing Process

Figure 5.3 shows the coding of getting the matrices of the classified image.


```
if class == 1
    Final_I(R,C,1:3) = blue;
    Class_M(R,C) = 1;
elseif class == 2
    Final_I(R,C,1:3) = white;
     Class_M(R,C) = 2;
elseif class == 3
    Final_I(R,C,1:3) = green;
     Class_M(R,C) = 3;
elseif class == 4
    Final_I(R,C,1:3) = red;
     Class_M(R,C) = 4;
elseif class == 5
    Final_I(R,C,1:3) = black;
     Class_M(R,C) = 5;
end
```

Figure 5.3 Coding of getting matrices of classified image

Figure 5.4 shows the matrices of the classified image.

1	2	3	4	5	6	7	8	9	10	11	12	13
3	3	3	1	3	3	3	3	3	3	3	3	3
3	3	1	5	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	5	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	1	3	3	3	3	3	3	3	3

Figure 5.4 Matrices of the classified image

Figure 5.5 shows the coding of getting the matrices of the ground truth image.

```
if (6 >IM_input(R,C,1)&& IM_input(R,C,1)> -1) && (6 >IM_input(R,C,2) && IM_input(R,C,2)> -1) && (256 >IM_input
    Final_I(R,C,1:3) = [0 0 255];
    GT_Matrix(R,C) = 1;
elseif (256 >IM_input(R,C,1) && IM_input(R,C,1)> 200) && (256 >IM_input(R,C,2) && IM_input(R,C,2)> 200) && (25
    Final_I(R,C,1:3) = [255 255 255];
    GT_Matrix(R,C) = 2;
elseif (20 >IM_input(R,C,1) && IM_input(R,C,1)> -1) && (256 >IM_input(R,C,2) && IM_input(R,C,2)> 100) && (20 >
    Final_I(R,C,1:3) = [0 128 0];
    GT_Matrix(R,C) = 3;
elseif (256 >IM_input(R,C,1) && IM_input(R,C,1)> 200) && (50 >IM_input(R,C,2) && IM_input(R,C,2)> -1) && (50 >
    Final_I(R,C,1:3) = [255 0 0];
    GT_Matrix(R,C) = 4;
elseif (5 >IM_input(R,C,1) && IM_input(R,C,1)> -1) && (5 >IM_input(R,C,2) && IM_input(R,C,2)> -1) && (5 >IM_in
    Final_I(R,C,1:3) = [0 0 0];
    GT_Matrix(R,C) = 5;
end
```


Figure 5.5 Coding of getting the matrices of ground truth image

Figure 5.6 shows the matrices of the ground truth image.

1	2	3	4	5	6	7	8	9	10	11	12	13
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3

Figure 5.6 Matrices of the ground truth image

Figure 5.7 shows the coding of obtaining the confusion matrix.

```
%% OBTAIN THE CONFUSION MATRIX
  for i=1:m
      for j=1:n
          if(ImgTruth(i,j)==0)
              continue;
          end
          t=ImgResult(i,j);
          k=ImgTruth(i,j);
          if ((t ~= 0) && (k ~= 0));
              CM(k,t)=CM(k,t)+1;
          end
      end
  end
```

Figure 5.7 Coding of obtaining the confusion matrix

Figure 5.8 shows the coding of calculate the sensitivity, specificity and overall accuracy.

```
%% CALCULATE THE METRICS
 Kappa=(N*Diagsum-Pc)/(N*N-Pc);
 OveAcc=Diagsum/N;
for i=1:m
    ProAcc(i)=CM(i,i)/rowsum(i);
    UserAcc(i)=CM(i,i)/columnsum(i);
    W = N - (columnsum(i) - rowsum(i) - CM(i,i));
    specificity(i) = W / (W + rowsum(i) - CM(i,i));
end
```

Figure 5.8 Coding of calculate the sensitivity, specificity accuracy and overall accuracy

Figure 5.9 shows the calculation result for Maximum Likelihood Classfier, where OverAcc is the Overall Accuracy, ProAcc is the accuracy for each class.

Name	Value
N	142159
Numofclass	5
OveAcc	0.8369
Pc	1.2652e+10
ProAcc	[0.3760;0.6639;0.9694;...
r	5
R	335
red	[255,0,0]
Road_I	*163x190x3 double*
rowsum	[2907,9657,110230,12...
sensitivity	[0.1548;0.6732;0.9471;...
specificity	[0.9871,0.9786,0.9865,...

Figure 5.9 Calculation result of Maximum Likelihood

5.2.2.2 ISODATA's Testing Process

Figure 5.10 and Figure 5.11 shows the coding of getting the matrices of the classified image.

```
IM_input = (imread('iso2.png'));
[r,c,l] = size(IM_input);
```

Figure 5.10 Coding of inputting the classified image

```
if (6 >IM_input(R,C,1)&& IM_input(R,C,1)> -1) && (6 >IM_input(R,C,2) && IM_input(R,C,2)> -1) && (256 >IM_input
    Final_I(R,C,1:3) = [0 0 255];
    GT_Matrix(R,C) = 1;
elseif (256 >IM_input(R,C,1) && IM_input(R,C,1)> 200) && (256 >IM_input(R,C,2) && IM_input(R,C,2)> 200) && (25
    Final_I(R,C,1:3) = [255 255 255];
    GT_Matrix(R,C) = 2;
elseif (20 >IM_input(R,C,1) && IM_input(R,C,1)> -1) && (256 >IM_input(R,C,2) && IM_input(R,C,2)> 100) && (20 >
    Final_I(R,C,1:3) = [0 128 0];
    GT_Matrix(R,C) = 3;
elseif (256 >IM_input(R,C,1) && IM_input(R,C,1)> 200) && (50 >IM_input(R,C,2) && IM_input(R,C,2)> -1) && (50 >
    Final_I(R,C,1:3) = [255 0 0];
    GT_Matrix(R,C) = 4;
elseif (5 >IM_input(R,C,1) && IM_input(R,C,1)> -1) && (5 >IM_input(R,C,2) && IM_input(R,C,2)> -1) && (5 >IM_in
    Final_I(R,C,1:3) = [0 0 0];
    GT_Matrix(R,C) = 5;
end
```

Figure 5.11 Coding of getting the matrices of ground truth image

Figure 5.12 shows the matrices of the classified image.

1	2	3	4	5	6	7	8	9	10	11	12	13
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	0	3	3	3	3
3	3	3	3	3	3	3	0	4	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
0	3	3	3	3	3	3	3	3	3	3	3	3
0	3	3	3	3	3	3	3	3	3	0	0	3
3	3	3	3	3	3	3	3	3	3	0	0	0
3	3	3	3	3	3	3	3	3	3	0	4	4

Figure 5.12 Matrices of the classified image

Figure 5.13 shows the coding of getting the matrices of the ground truth image.

```
if (6 >IM_input(R,C,1)&& IM_input(R,C,1)> -1) && (6 >IM_input(R,C,2) && IM_input(R,C,2)> -1) && (256 >IM_input
    Final_I(R,C,1:3) = [0 0 255];
    GT_Matrix(R,C) = 1;
elseif (256 >IM_input(R,C,1) && IM_input(R,C,1)> 200) && (256 >IM_input(R,C,2) && IM_input(R,C,2)> 200) && (25
    Final_I(R,C,1:3) = [255 255 255];
    GT_Matrix(R,C) = 2;
elseif (20 >IM_input(R,C,1) && IM_input(R,C,1)> -1) && (256 >IM_input(R,C,2) && IM_input(R,C,2)> 100) && (20 >
    Final_I(R,C,1:3) = [0 128 0];
    GT_Matrix(R,C) = 3;
elseif (256 >IM_input(R,C,1) && IM_input(R,C,1)> 200) && (50 >IM_input(R,C,2) && IM_input(R,C,2)> -1) && (50 >
    Final_I(R,C,1:3) = [255 0 0];
    GT_Matrix(R,C) = 4;
elseif (5 >IM_input(R,C,1) && IM_input(R,C,1)> -1) && (5 >IM_input(R,C,2) && IM_input(R,C,2)> -1) && (5 >IM_in
    Final_I(R,C,1:3) = [0 0 0];
    GT_Matrix(R,C) = 5;
end
```

Figure 5.13 Coding of getting the matrices of the ground truth image

Figure 5.14 shows the matrices of the ground truth image.

1	2	3	4	5	6	7	8	9	10	11	12	13
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3

Figure 5.14 Matrices of the ground truth image

Figure 5.15 shows the coding of obtaining the confusion matrix.

```
%% OBTAIN THE CONFUSION MATRIX
for i=1:m
    for j=1:n
        if(ImgTruth(i,j)==0)
            continue;
        end
        t=ImgResult(i,j);
        k=ImgTruth(i,j);
        if ((t ~= 0) && (k ~= 0));
            CM(k,t)=CM(k,t)+1;
        end
    end
end
```

Figure 5.15 Coding of obtaining the confusion matrix

Figure 5.16 shows the coding of calculating the sensitivity, specificity accuracy and overall accuracy.

```
%% CALCULATE THE METRICS
 Kappa=(N*Diagsum-Pc)/(N*N-Pc);
 OveAcc=Diagsum/N;
for i=1:m
    ProAcc(i)=CM(i,i)/rowsum(i);
    UserAcc(i)=CM(i,i)/columnsum(i);
    W = N - (columnsum(i) - rowsum(i) - CM(i,i));
    specificity(i) = W / (W + rowsum(i) - CM(i,i));
end
```

Figure 5.16 Coding of calculating the sensitivity, specificity accuracy and overall accuracy

Figure 5.17 shows the calculation result for ISODATA Classifier, where OverAcc is the Overall Accuracy, ProAcc is the accuracy for each class.

Name	Value
N	108738
Numofclass	5
OveAcc	0.7682
Pc	6.9464e+09
ProAcc	[0.3941;0.6118;0.8577;...
r	5
R	334
rowsum	[2761,6417,88175,737...
sensitivity	[0.1543;0.8072;0.9849;...
specificity	[0.9844,0.9787,0.9398,...

Figure 5.17 Calculation result of ISODATA

5.3 Result Analysis

In this subchapter, the result of confusion matrix was discussed. Confusion Matrix was obtained by comparing ground truth image with classified image. After obtaining the confusion matrix, the sensitivity, specificity, accuracy for each class and overall accuracy were evaluated. The overall accuracy for Maximum Likelihood classifier and ISODATA classifier were compared to identify which is the best algorithm in satellite image classification.

5.3.1 Testing Result

Table 5.1 shows the Confusion Matrix and Accuracy for each of the classes for Maximum Likelihood Classifier. The total number of pixels in the satellite image is 142159. The accuracy of each class was recorded in Table 5,1, where the accuracy for Water is 37.60%, accuracy for Building is 66.39%, accuracy for Forest is 96.94%, accuracy for Barren Land is 12.41%, and the accuracy for Road is 44.22%.

Table 5.1 Confusion Matrix and Accuracy for Maximum Likelihood Classifier

Class	Water	Building	Forest	Barren Land	Road	Accuracy
Water	1093	0	1794	10	10	37.60%
Building	337	6411	1022	30	1857	66.39%
Forest	1860	231	106853	143	1143	96.94%
Barren Land	2910	2053	1785	1543	4139	12.41%
Road	859	828	1371	810	3067	44.22%

Table 5.2 shows the number of pixels which was correctly classified and incorrectly classified, Sensitivity and Specificity for each class and the Overall Accuracy for Maximum Likelihood Classifier. The Overall Accuracy for satellite image classification using Maximum Likelihood Classifier is 83.69%.

Table 5.2 Sensitivity, Specificity and Overall Accuracy for Maximum Likelihood Classifier

Class	Correctly Classified	Incorrectly Classified	Sensitivity	Specificity	Overall Accuracy
Water	1093	1814	15.48%	98.71%	83.69%
Building	6411	3246	67.32%	97.86%	
Forest	106853	3377	94.71%	98.65%	
Barren Land	1543	10887	60.48%	93.38%	
Road	3067	3868	30.02%	97.35%	

Table 5.3 shows the Confusion Matrix and Accuracy for each of the classes for ISODATA Classifier. The total number of pixels in the satellite image is 142159. The accuracy of each class was recorded in Table 5.3, where the accuracy for Water is 39.41%, accuracy for Building is 61.18%, accuracy for Forest is 85.77%, accuracy for Barren Land is 11.45%, and the accuracy for Road is 50.92%.

Table 5.3 Confusion Matrix and Accuracy for each of the classes for ISODATA Classifier

Class	Water	Building	Forest	Barren Land	Road	Accuracy
Water	1088	0	389	1282	2	39.41%
Building	562	3926	274	237	1418	61.18%
Forest	1642	55	75628	10550	300	85.77%
Barren Land	2768	647	146	845	2971	11.45%
Road	989	236	349	393	2041	50.92%

Table 5.4 shows the number of pixels which was correctly classified and incorrectly classified, Sensitivity and Specificity for each and Overall Accuracy for ISODATA Classifier. The Overall Accuracy for ISODATA Classifier is 76.82%.

Table 5.4 Sensitivity, Specificity and Overall Accuracy for ISODATA Classifier

Class	Correctly Classified	Incorrectly Classified	Sensitivity	Specificity	Overall Accuracy
Water	1088	1673	15.43%	98.44%	
Building	3926	2491	80.72%	97.87%	
Forest	75628	12547	98.49%	93.98%	76.82%
Barren Land	845	6532	6.35%	94.07%	
Road	2041	1967	30.32%	98.21%	

5.3.2 Testing Result Analysis

Table 5.5 and Figure 5.18 shows the comparison for the accuracy of Maximum Likelihood Classifier and ISODATA Classifier. The total number of pixels of the image is 142159 and the number of pixels is the same for both classifiers. The accuracy of Maximum Likelihood for Building class, Forest class and Barren Land class is higher than ISODATA. The accuracy of ISODATA for Water class and Road class is higher than Maximum Likelihood.

Table 5.5 Comparison for the accuracy of Maximum Likelihood and ISODATA

	Maximum Likelihood	ISODATA
Water	37.60%	39.41%
Building	66.39%	61.18%
Forest	96.94%	85.77%
Barren Land	12.41%	11.45%
Road	44.22%	50.92%
Overall Accuracy	83.69%	76.82%

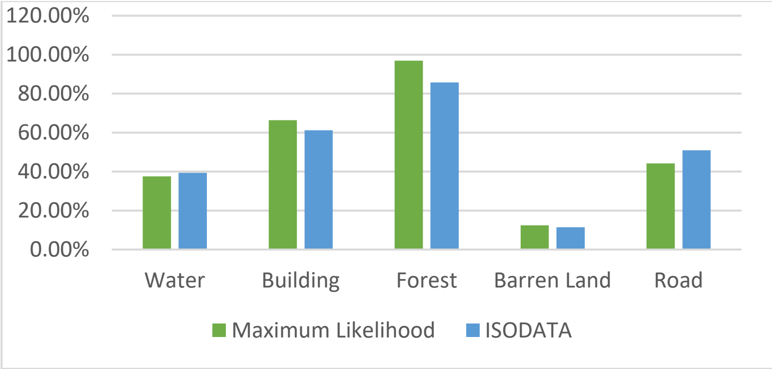

Figure 5.18 Accuracy of each class and each algorithm

The overall accuracy of Maximum Likelihood Classifier is higher than ISODATA Classifier. Therefore, it can be concluded that Maximum Likelihood is the best technique in satellite image classification. Maximum Likelihood is more efficient and reliable comparing to ISODATA.

Table 5.6 shows the advantage and disadvantage of Maximum Likelihood Classifier and ISODATA Classifier which were obtained in this project. Maximum Likelihood is relatively accurate but it requires prior knowledge of the region which needed to be classified. ISODATA does not require prior knowledge of the region, but it is less accurate comparing to Maximum Likelihood.

Table 5.6 Advantage and disadvantage of MLC and ISODATA

Maximum Likelihood		ISODATA	
Advantage	Disadvantage	Advantage	Disadvantage
Relatively Accurate	Required prior knowledge of the region	Does not required prior knowledge of the region	Less accurate

5.4 Conclusion

In this chapter, the testing experiments were conducted and the testing experiment was successful. The testing process was carried out to verify the result obtained in the previous chapter. Confusion matrix was used to evaluate the accuracy of the classification result. Accuracy for each class, sensitivity, specificity and overall accuracy were calculated to identify the best technique in satellite image classification.

Based on the calculation result obtained, the accuracy of Maximum Likelihood is 83.69% whereas the accuracy of ISODATA is 76.82%. It can be concluded that Maximum Likelihood has a better performance comparing to ISODATA.

In conclusion, the objective of this project which is to identify the classification techniques in satellite image classification, to apply Maximum Likelihood and ISODATA techniques for satellite image classification and to evaluate the best technique in satellite image classification were achieved.

CHAPTER 6

PROJECT CONCLUSION

6.1 Introduction

In this chapter, there are six sub-chapter, which are introduction, project summarization, project contribution, project limitation, future works and conclusion. The summary of the project, contribution of the project, limitation of the project and the future works that can be conducted by the researcher in the future were discussed.

6.2 Project Summarization

The project started with defining the problem statement and project question. Based on the problem statement and project question, three project objective were defined, which are to identify the classification techniques in satellite image classification, to apply Maximum Likelihood and ISODATA techniques for satellite image classification and to evaluate the best technique in satellite image classification. The first objective was achieved in Chapter 2, where the classification techniques, Maximum Likelihood and ISODATA were identified after referring to the previous work done by the researcher before.

Next, the second objective was achieved in Chapter 4. Maximum Likelihood and ISODATA algorithms were implemented in Chapter 4 to classify the satellite image. There are five classes needed to be classified, which is water, building, forest, barren land and road. The classification process is successful. The output that obtained in Chapter 4 is captured and recorded for further testing and result analysis in Chapter 5.

The third objective was achieved in Chapter 5, where the testing experiments were conducted to evaluate the best technique in satellite image classification. To make sure the testing result obtained is reliable and consistence, Confusion Matrix was used to calculate the accuracy of the classification results. In conclusion, Maximum Likelihood is the best technique in satellite image classification.

6.3 Project Contribution

The contribution of this project was finding the best classification technique for satellite image. Intellectual with expertise in this field can use the best technique found in this project to conduct their image classification project especially in satellite image classification. Besides, this project benefits the experts who work in the related field such as military, environmental field, map production, agriculture, forestry, planning of national land and establishment of city plan since the unexplored area can be identified using these methods.

6.4 Project Limitation

There are a few limitations in this project. The program was developed with MATLAB programming language, therefore MATLAB software is required to execute the program. Besides, the program that developed in this project does not has GUI, user without prior knowledge in MATLAB may face difficulties in using the program.

6.5 Future Works

The program can be improved by designing an application which can run on different operating system including mobile devices such as tablet. Furthermore, a user-friendly GUI can be designed to make the program easier to be used.

6.6 Conclusion

Finally, after almost half year of research period, the project was done. The objective that defined in this project were all achieved. The techniques for satellite image classification were identified, the two techniques to classify the satellite image were applied, and the best techniques to classify the satellite image was evaluated. This project contributes finding the best technique for satellite image classification. Although there are some limitations in this project, the future researcher may make some improvements to this study to develop a better program with a higher accuracy of classification result.

REFERENCES

Allison Ragan. (2018). Taking the Confusion Out of Confusion Matrices. Retrieved 1 August 2021 from

https://towardsdatascience.com/taking-the-confusion-out-of-confusion-matrices-c1ce054b3d3e

Aniruddha Bhandari. (2020). Everything you Should Know about Confusion Matrix for Machine Learning. Retrieved 1 August 2021 from

https://www.analyticsvidhya.com/blog/2020/04/confusion-matrix-machine-learning/

Anonymous. (2016). Confusion Matrix. Retrieved 1 August 2021 from

http://scaryscientist.blogspot.com/2016/03/confusion-matrix.html

Anonymous. (2014). Image Classification: Supervised and Unsurpervised. Retrieved 10 June 2021 from https://remotesensinginactionlearningblog.wordpress.com/2014/11/29/image-classification-supervised-and-unsurpervised/

Anonymous. (n.d.). How Maximum Likelihood Classification works. Retrieved 13 May 2021 from https://desktop.arcgis.com/en/arcmap/10.3/tools/spatial-analyst-toolbox/how-maximum-likelihood-classification-works.htm

Anonymous. (2016). Image Classification Unsupervised. Retrieved 16 April 2021 from http://knightlab.org/rscc/labs/Lab04_Image_Classification_Unsupervised-2016.pdf

Anonymous. (n.d.). Matlab implementation of Kmeans and ISODATA algorithm. Retrieved 9 June 2021 from https://www.programmersought.com/article/2184107262/

Anonymous. (2014). Simple guide to confusion matrix terminology. Retrieved 1 August 2021 from https://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/

Anonymous. (n.d.). Supervised Classification. Retrieved 13 April 2020 from http://gsp.humboldt.edu/OLM/Courses/GSP_216_Online/lesson6-1/supervised.html

Anonymous. (n.d.). Unsupervised Classification. Retrieved 13 April 2021 from http://gsp.humboldt.edu/OLM/Courses/GSP_216_Online/lesson6-1/unsupervised.html

Anonymous. (n.d.). Unsupervised Classification algorithms. Retrieved 9 June 2021 from http://www.wu.ece.ufl.edu/books/EE/communications/UnsupervisedClassification.html

Compute confusion matrix for classification problem - MATLAB confusionmat - MathWorks Switzerland. (n.d.). MathWorks. Retrieved 1 August 2021, from https://ch.mathworks.com/help/stats/confusionmat.html

Confusion matrix. (2021, July 4). In Wikipedia. https://en.wikipedia.org/wiki/Confusion_matrix

Jason Brownlee. (2016). What is a Confusion Matrix in Machine Learning. Retrieved 1 August 2021 from https://machinelearningmastery.com/confusion-matrix-machine-learning/

Jonny Brooks-Bartlett. (2018). Probability concepts explained: Maximum likelihood estimation. Retrieved 13 May 2021 from https://towardsdatascience.com/probability-concepts-explained-maximum-likelihood-estimation-c7b4342fdbb1

Joydwip Mohajon. (2020). Confusion Matrix for Your Multi-Class Machine Learning Model. Retrieved 1 August 2021 from https://towardsdatascience.com/confusion-matrix-for-your-multi-class-machine-learning-model-ff9aa3bf7826

Kavish Sanghvi. (2020). Image Classification Techniques. Retrieved 12 April 2021, from https://medium.com/analytics-vidhya/image-classification-techniques-83fd87011cac

Mapasyst. (2019). What's the difference between a supervised and unsupervised image classification? . Retrieved 13 April 2021, from https://mapasyst.extension.org/whats-the-difference-between-a-supervised-and-unsupervised-image-classification/

MLK. (2019). Demystifying Confusion Matrix and Performance Metrics in Machine Learning. Retrieved 1 August 2021 from https://machinelearningknowledge.ai/confusion-matrix-and-performance-metrics-machine-learning/

Muhammad Sarfraz. (2020). Introductory Chapter: On Digital Image Processing. Retrieved 13 April 2021, from https://www.intechopen.com/books/digital-imaging/introductory-chapter-on-digital-image-processing

NorthGod. (2013). k-means & isodata clustering. Retrieved 9 June 2021 from https://blog.csdn.net/houston11235/article/details/8511379

Rose Mary. (2011). Introduction to image processing. Retrieved 13 April 2021, from https://www.engineersgarage.com/article_page/introduction-to-image-processing/

Rosaria Silipo. (2019). Confusion Matrix and Class Statistics. Retrieved 1 August 2021 from https://towardsdatascience.com/confusion-matrix-and-class-statistics-68b79f4f510b

Sarang Narkhede . (2018). Understanding Confusion Matrix. Retrieved 1 August 2021 from https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

Serafeim Loukas. (2020). Multi-class Classification: Extracting Performance Metrics From The Confusion Matrix. Retrieved 1 August 2021 from https://towardsdatascience.com/multi-class-classification-extracting-performance-metrics-from-the-confusion-matrix-b379b427a872

Sunit Prasad. (n.d.). What is Image Segmentation? Retrieved 13 April 2021, from https://www.analytixlabs.co.in/blog/what-is-image-segmentation/

UKEssays. (2018). Supervised Image Classification Techniques. Retrieved 13 April 2021 from https://www.ukessays.com/essays/engineering/supervised-image-classification-9746.php?vref=1

BIBLIOGRAPHY

A. Ahmad and S. F. Sufahani, "Analysis of Landsat 5 TM data of Malaysian land covers using ISODATA clustering technique," 2012 IEEE Asia-Pacific Conference on Applied Electromagnetics (APACE), Melaka, Malaysia, 2012, pp. 92-97, doi: 10.1109/APACE.2012.6457639.

Abbas, Arbab & Minallh, N. & Ahmad, Nasir & Abdur Rehman Abid, Sahibzada & Khan, Muhammad. (2016). K-Means and ISODATA Clustering Algorithms for Landcover Classification Using Remote Sensing. 48. 315-318.

A. Nawaz, Z. Iqbal and S. Ullah, "Performance analysis of supervised image classification techniques for the classification of multispectral satellite imagery," 2015 Fourth International Conference on Aerospace Science and Engineering (ICASE), Islamabad, Pakistan, 2015, pp. 1-5, doi: 10.1109/ICASE.2015.7489513.

C. K.S., S. G.S. and A. L. Choodarathnakara, "Classification of Homogeneous Sites using IRS-P5 Satellite Imagery," 2020 International Conference on Computation, Automation and Knowledge Management (ICCAKM), Dubai, United Arab Emirates, 2020, pp. 184-189, doi: 10.1109/ICCAKM46823.2020.9051510.

D. Long and V. P. Singh, "An Entropy-Based Multispectral Image Classification Algorithm," in IEEE Transactions on Geoscience and Remote Sensing, vol. 51, no. 12, pp. 5225-5238, Dec. 2013, doi: 10.1109/TGRS.2013.2272560.

Domadia, S. G., & Zaveri, T. (2011, May). Comparative analysis of unsupervised and supervised image classification techniques. In Proceeding of National Conference on Recent Trends in Engineering & Technology (pp. 1-5).

E. Miranda, A. B. Mutiara, Ernastuti and W. C. Wibowo, "Classification of Land Cover from Sentinel-2 Imagery Using Supervised Classification Technique (Preliminary Study)," 2018 International Conference on Information Management and Technology (ICIMTech), Jakarta, Indonesia, 2018, pp. 69-74, doi: 10.1109/ICIMTech.2018.8528122.

K. Vani, "Satellite image processing," 2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN), Chennai, India, 2017, pp. 1-3, doi: 10.1109/ICSCN.2017.8085410.

M Shinozuka, B Mansouri,
4 - Synthetic aperture radar and remote sensing technologies for structural health monitoring of civil infrastructure systems,
Editor(s): Vistasp M. Karbhari, Farhad Ansari,
In Woodhead Publishing Series in Civil and Structural Engineering,
Structural Health Monitoring of Civil Infrastructure Systems,
Woodhead Publishing,
2009,
Pages 113-151,
ISBN 9781845693923,
https://doi.org/10.1533/9781845696825.1.114.

Olle Hagner, Heather Reese,
A method for calibrated maximum likelihood classification of forest types,
Remote Sensing of Environment,
Volume 110, Issue 4,
2007,
Pages 438-444,
ISSN 0034-4257,
https://doi.org/10.1016/j.rse.2006.08.017.
(https://www.sciencedirect.com/science/article/pii/S0034425707001757)

P. S. Sisodia, V. Tiwari and A. Kumar, "Analysis of Supervised Maximum Likelihood Classification for remote sensing image," International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014), Jaipur, India, 2014, pp. 1-4, doi: 10.1109/ICRAIE.2014.6909319.

Singh, Manjari & Kumar, Sushil & Chouhan, Siddharth & Shrivastava, Manish. (2016). Various Image Compression Techniques: Lossy and Lossless. International Journal of Computer Applications. 142. 23-26. 10.5120/ijca2016909829.

Gouet-Brunet V. (2009) Image Representation. In: LIU L., ÖZSU M.T. (eds) Encyclopedia of Database Systems. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-39940-9_1438

S. S. Wanarse, T. G. Patil, S. S. Patankar and J. V. Kulkarni, "Class quantification of aerial images using Maximum Likelihood Estimation," 2014 First International Conference on Networks & Soft Computing (ICNSC2014), Guntur, India, 2014, pp. 345-347, doi: 10.1109/CNSC.2014.6906691.

Thakur, N., & Maheshwari, D. (2017). A Review of Image Classification Approaches and Techniques. International Journal of Recent Trends in Engineering and Research, 3(3), 1-5 https://doi.org/10.23883/ijrter.2017.3033.xts7z

Ahmad, Asmala & Quegan, S.. (2012). Analysis of maximum likelihood classification on multispectral data. Applied Mathematical Sciences. 6. 6425-6436.

Mudassir Khan. (2017). KMeans Clustering for Classification https://towardsdatascience.com/kmeans-clustering-for-classification-74b992405d0a#:~:text=KMeans%20is%20a%20clustering%20algorithm,than%20the%20number%20of%20classes.

APPENDICES

Appendix A - Coding for Maximum Likelihood Classification

```
img = double(imread('mapS.png'));  % read the image
[r,c,l] = size(img);        % get the num of rows and columns of the image.

% ........ Making the Training Vectors ..........%
Water_I = double(imread('water1.jpg'));        % class 1
Building_I = double(imread('Buildings.jpg'));  % class 2
Forest_I = double(imread('Forest.jpg'));       % class 3
Barren_I = double(imread('Barren.jpg'));       % class 4
Road_I = double(imread('Road.jpg'));           % class 5

% define the colors of classes%
blue = [0 0 255];     %water
white = [255 255 255];   %building
green = [0 128 0];     %forest
black = [0 0 0];     %road
red = [255 0 0];   %barren land

             % first class vectors
L1c1 = reshape(Water_I(:,:,1),[numel(Water_I(:,:,1)),1]);   %R
L2c1 = reshape(Water_I(:,:,2),[numel(Water_I(:,:,2)),1]);   %G
L3c1 = reshape(Water_I(:,:,3),[numel(Water_I(:,:,3)),1]);   %B

            % second class vectors
L1c2 = reshape(Building_I(:,:,1),[numel(Building_I(:,:,1)),1]);
L2c2 = reshape(Building_I(:,:,2),[numel(Building_I(:,:,2)),1]);
L3c2 = reshape(Building_I(:,:,3),[numel(Building_I(:,:,3)),1]);

            % third class vectors
L1c3 = reshape(Forest_I(:,:,1),[numel(Forest_I(:,:,1)),1]);
L2c3 = reshape(Forest_I(:,:,2),[numel(Forest_I(:,:,2)),1]);
L3c3 = reshape(Forest_I(:,:,3),[numel(Forest_I(:,:,3)),1]);

             % forth class vectors
L1c4 = reshape(Barren_I(:,:,1),[numel(Barren_I(:,:,1)),1]);
L2c4 = reshape(Barren_I(:,:,2),[numel(Barren_I(:,:,2)),1]);
L3c4 = reshape(Barren_I(:,:,3),[numel(Barren_I(:,:,3)),1]);
```

```
                % fifth class vectors
L1c5 = reshape(Road_I(:,:,1),[numel(Road_I(:,:,1)),1]);
L2c5 = reshape(Road_I(:,:,2),[numel(Road_I(:,:,2)),1]);
L3c5 = reshape(Road_I(:,:,3),[numel(Road_I(:,:,3)),1]);

% calculate the mean and standard deviation for each class:
%..........class 1..........%
% class 1 layer 1

% mC1L1 (the mean for class 1 layer 1)
mC1L1 = 0;
for counter = 1:length(L1c1)
   mC1L1 = mC1L1 + L1c1(counter);
end
mC1L1 = mC1L1/length(L1c1);

% dC1L1
dC1L1 = 0;
for counter = 1:length(L1c1)
   dC1L1 = dC1L1 + (L1c1(counter)-mC1L1)^2;
end
dC1L1 = dC1L1/length(L1c1);
dC1L1 = sqrt(dC1L1);

%class 1 layer 2

% mC1L2 (the mean for class 1 layer 2)
mC1L2 = 0;
for counter = 1:length(L2c1)
   mC1L2 = mC1L2 + L2c1(counter);
end
mC1L2 = mC1L2/length(L2c1);

% dC1L2
dC1L2 = 0;
for counter = 1:length(L2c1)
   dC1L2 = dC1L2 + (L2c1(counter)-mC1L2)^2;
end
dC1L2 = dC1L2/length(L2c1);
dC1L2 = sqrt(dC1L2);

% mC1L3 (the mean for class 1 layer 3)
mC1L3 = 0;
for counter = 1:length(L3c1)
   mC1L3 = mC1L3 + L3c1(counter);
end
mC1L3 = mC1L3/length(L3c1);

% dC1L3
dC1L3 = 0;
```

```
        for counter = 1:length(L3c1)
    dC1L3 = dC1L3 + (L3c1(counter)-mC1L3)^2;
end
dC1L3 = dC1L3/length(L3c1);
dC1L3 = sqrt(dC1L3);

%..........class 2..........%
% class 2 layer 1

% mC2L1 (the mean for class 2 layer 1)
mC2L1 = 0;
for counter = 1:length(L1c2)
    mC2L1 = mC2L1 + L1c2(counter);
end
mC2L1 = mC2L1/length(L1c2);


dC2L1 = 0;
for counter = 1:length(L1c2)
    dC2L1 = dC2L1 + (L1c2(counter)-mC2L1)^2;
end
dC2L1 = dC2L1/length(L1c2);
dC2L1 = sqrt(dC2L1);

% class 2 layer 2

% mC2L1 (the mean for class 2 layer 2)
mC2L2 = 0;
for counter = 1:length(L2c2)
    mC2L2 = mC2L2 + L2c2(counter);
end
mC2L2 = mC2L2/length(L2c2);

% dC2L2
dC2L2 = 0;
for counter = 1:length(L2c2)
    dC2L2 = dC2L2 + (L2c2(counter)-mC2L2)^2;
end
dC2L2 = dC2L2/length(L2c2);
dC2L2 = sqrt(dC2L2);

% class 2 layer 3

mC2L3 = 0;
for counter = 1:length(L3c2)
    mC2L3 = mC2L3 + L3c2(counter);
end
mC2L3 = mC2L3/length(L3c2);
```

```
% dC2L3
dC2L3 = 0;
for counter = 1:length(L3c2)
   dC2L3 = dC2L3 + (L3c2(counter)-mC2L3)^2;
end
dC2L3 = dC2L3/length(L3c2);
dC2L3 = sqrt(dC2L3);

%..........class 3..........%

% class 3 layer 1

mC3L1 = 0;
for counter = 1:length(L3c2)
   mC3L1 = mC3L1 + L1c3(counter);
end
mC3L1 = mC3L1/length(L1c3);

% dC3L1
dC3L1 = 0;
for counter = 1:length(L1c3)
   dC3L1 = dC3L1 + (L1c3(counter)-mC3L1)^2;
end
dC3L1 = dC3L1/length(L1c3);
dC3L1 = sqrt(dC3L1);

% class 3 layer 2

mC3L2 = 0;
for counter = 1:length(L3c2)
   mC3L2 = mC3L2 + L2c3(counter);
end
mC3L2 = mC3L2/length(L2c3);

% dC3L2
dC3L2 = 0;
for counter = 1:length(L2c3)
   dC3L2 = dC3L2 + (L2c3(counter)-mC3L2)^2;
end
dC3L2 = dC3L2/length(L2c3);
dC3L2 = sqrt(dC3L2);

% class 3 layer 3

mC3L3 = 0;
for counter = 1:length(L3c2)
   mC3L3 = mC3L3 + L3c3(counter);
end
mC3L3 = mC3L3/length(L3c3);
```

```
% dC3L3
dC3L3 = 0;
for counter = 1:length(L3c3)
    dC3L3 = dC3L3 + (L3c3(counter)-mC3L3)^2;
end
dC3L3 = dC3L3/length(L3c3);
dC3L3 = sqrt(dC3L3);

%..........class 4..........%

% class 4 layer 1

mC4L1 = 0;
for counter = 1:length(L1c4)
    mC4L1 = mC4L1 + L1c4(counter);
end
mC4L1 = mC4L1/length(L1c4);

% dC3L1
dC4L1 = 0;
for counter = 1:length(L1c4)
    dC4L1 = dC4L1 + (L1c4(counter)-mC4L1)^2;
end
dC4L1 = dC4L1/length(L1c4);
dC4L1 = sqrt(dC4L1);

% class 4 layer 2

mC4L2 = 0;
for counter = 1:length(L2c4)
    mC4L2 = mC4L2 + L2c4(counter);
end
mC4L2 = mC4L2/length(L2c4);

% dC3L1
dC4L2 = 0;
for counter = 1:length(L2c4)
    dC4L2 = dC4L2 + (L2c4(counter)-mC4L2)^2;
end
dC4L2 = dC4L2/length(L2c4);
dC4L2 = sqrt(dC4L2);

% class 4 layer 3

mC4L3 = 0;
for counter = 1:length(L3c4)
    mC4L3 = mC4L3 + L3c4(counter);
end
mC4L3 = mC4L3/length(L3c4);
```

```
% dC4L3
dC4L3 = 0;
for counter = 1:length(L3c4)
   dC4L3 = dC4L3 + (L3c4(counter)-mC4L3)^2;
end
dC4L3 = dC4L3/length(L3c4);
dC4L3 = sqrt(dC4L3);

%..........class 5..........%

% class 5 layer 1

mC5L1 = 0;
for counter = 1:length(L1c5)
   mC5L1 = mC5L1 + L1c5(counter);
end
mC5L1 = mC5L1/length(L1c5);

% dC5L1
dC5L1 = 0;
for counter = 1:length(L1c5)
   dC5L1 = dC5L1 + (L1c5(counter)-mC5L1)^2;
end
dC5L1 = dC5L1/length(L1c5);
dC5L1 = sqrt(dC5L1);

% class 5 layer 2

mC5L2 = 0;
for counter = 1:length(L2c5)
   mC5L2 = mC5L2 + L2c5(counter);
end
mC5L2 = mC5L2/length(L2c5);

% dC5L2
dC5L2 = 0;
for counter = 1:length(L2c5)
   dC5L2 = dC5L2 + (L2c5(counter)-mC5L2)^2;
end
dC5L2 = dC5L2/length(L2c5);
dC5L2 = sqrt(dC5L2);

% class 5 layer 3

mC5L3 = 0;
for counter = 1:length(L3c5)
   mC5L3 = mC5L3 + L3c5(counter);
end
mC5L3 = mC5L3/length(L3c5);
```

```
% dC5L3
dC5L3 = 0;
for counter = 1:length(L3c5)
   dC5L3= dC5L3 + (L3c5(counter)-mC5L3)^2;
end
dC5L3 = dC5L3/length(L1c3);
dC5L3 = sqrt(dC5L3);

% Classification process

for R = 1:r
   for C = 1:c
Pc(1) = (1/(dC1L1*sqrt(2*pi)))*exp(-0.5*((img(R,C,1)-
mC1L1)/dC1L1)^2)*(1/(dC1L2*sqrt(2*pi)))*exp(-0.5*((img(R,C,2)-
mC1L2)/dC1L2)^2)*(1/(dC1L3*sqrt(2*pi)))*exp(-0.5*((img(R,C,3)-
mC1L3)/dC1L3)^2);
Pc(2) = (1/(dC2L1*sqrt(2*pi)))*exp(-0.5*((img(R,C,1)-
mC2L1)/dC2L1)^2)*(1/(dC2L2*sqrt(2*pi)))*exp(-0.5*((img(R,C,2)-
mC2L2)/dC2L2)^2)*(1/(dC2L3*sqrt(2*pi)))*exp(-0.5*((img(R,C,3)-
mC2L3)/dC2L3)^2);
Pc(3) = (1/(dC3L1*sqrt(2*pi)))*exp(-0.5*((img(R,C,1)-
mC3L1)/dC3L1)^2)*(1/(dC3L2*sqrt(2*pi)))*exp(-0.5*((img(R,C,2)-
mC3L2)/dC3L2)^2)*(1/(dC3L3*sqrt(2*pi)))*exp(-0.5*((img(R,C,3)-
mC3L3)/dC3L3)^2);
Pc(4) = (1/(dC4L1*sqrt(2*pi)))*exp(-0.5*((img(R,C,1)-
mC4L1)/dC4L1)^2)*(1/(dC4L2*sqrt(2*pi)))*exp(-0.5*((img(R,C,2)-
mC4L2)/dC4L2)^2)*(1/(dC4L3*sqrt(2*pi)))*exp(-0.5*((img(R,C,3)-
mC4L3)/dC4L3)^2);
Pc(5) = (1/(dC5L1*sqrt(2*pi)))*exp(-0.5*((img(R,C,1)-
mC5L1)/dC5L1)^2)*(1/(dC5L2*sqrt(2*pi)))*exp(-0.5*((img(R,C,2)-
mC5L2)/dC5L2)^2)*(1/(dC5L3*sqrt(2*pi)))*exp(-0.5*((img(R,C,3)-
mC5L3)/dC5L3)^2);

[value,class] = max(Pc);

% color changing in the final image.
  if class == 1
     Final_I(R,C,1:3) = blue;
     Class_M(R,C) = 1;
  elseif class == 2
     Final_I(R,C,1:3) = white;
      Class_M(R,C) = 2;
  elseif class == 3
     Final_I(R,C,1:3) = green;
      Class_M(R,C) = 3;
  elseif class == 4
     Final_I(R,C,1:3) = red;
      Class_M(R,C) = 4;
  elseif class == 5
     Final_I(R,C,1:3) = black;
```

```
Class_M(R,C) = 5;
 end

   end
end
imshow(Final_I)
```

Appendix B - Coding for ISODATA Classification

```
k = 10;  %expected no of cluster centers
leastNum = 1;   %minimum number of sample in each cluster, less than this
number of sample in each cluster is not consider as independent cluster
classStandard = 1;  %Standard Deviation of sample distance distribution
minDis = 1;   %Minimum distance between 2 cluster, if less than this number,
clusters merged
maxCombine = 20;   %Maximum number of cluster centers that can be combined
with one in an iterative operation
iterationTime = 100;  %Number of iteration
[class, INDEX] =
ISODATA(k,leastNum,classStandard,minDis,maxCombine,iterationTime)

COLOR(1,1:3) = [0 128 0];
COLOR(2,1:3) = [0 128 0];
COLOR(3,1:3) = [255 0 0];
COLOR(4,1:3) = [0 0 255];
COLOR(5,1:3) = [0 0 0];
COLOR(6,1:3) = [255 255 255];

% Getting the final Image
for i = 1:size(class,1)
    for j = 1:size(class(i,:),2)
        if class(i,j) ~= 0
            Final_Image(INDEX(class(i,j),1),INDEX(class(i,j),2),1:3) =
COLOR(i,1:3);
        end
    end
end
imshow(Final_Image);

function
[class,INDEX]=ISODATA(k,leastNum,classStandard,minDis,maxCombine,iterat
ionTime)

%input image
A = imread('mapS.png');
```

```
% get the num of rows and columns of the image.
[r c] = size(A(:,:,1));

i = 1;
img1 = zeros(r*c,3);
INDEX = zeros(r*c,2);
for R = 1:r
   for C = 1:c
   img1(i,1:3) = [A(R,C,1) A(R,C,2) A(R,C,3)];
   INDEX(i,1:2) = [R C];     % Record the original index of a pixel
   i = i + 1;
   end
end
img = img1;

num=size(img,1);  % get the size of img
currentNum=1;  %current number of cluster
centre=img(1,:);  %default cluster center
centre(2,:)=img(20000,:);
centre(3,:)=img(50000,:);
centre(4,:)=img(70000,:);
factor = 0.5;   %factor when calculating spliting
distance=ComputDistance(img,centre,num,currentNum);  %calculate the distance
between img with center
I=1;

while I<=iterationTime %iteration start
   class=zeros(currentNum,r*c);
   classCounter = zeros(currentNum,1);
   for i=1:num
      [value,index]=min(distance(i,:)); %Find the nearest distance between sample
and center
      dtemp=[i];
      classCounter(index) = classCounter(index) + 1;
      class(index,classCounter(index))=dtemp;
   end

% remove clusters with small number of elements

   for i=1:currentNum    % number of clusters
      if size(classCounter(i),2)<leastNum
         class=move(class,i);
         currentNum=currentNum-1;
      end
   end

centre=ComputMid(img,class,currentNum,classCounter);   %recalculate the
cluster center
```

```
distance=ComputDistance(img,centre,num,currentNum);    %recalculate the
distance between sample and center

classAverage=ComputAverageClass(class,distance,currentNum,classCounter);
%calculate the single average of the distance between sample and center

totalAverage=ComputAverageTotal(class,classAverage,currentNum,num,classCo
unter);   %calculate the total average of distance between sample and center

   flag=step_judge(currentNum,I,iterationTime,k);

   if flag==1   %split
      Cstandard=ComputCstandard(class,img,centre,currentNum,classCounter);
      %calculate the standard deviation of sample and center
      maxCstandard = zeros(currentNum,1);
      for i=1:currentNum
         [value,index]=max(Cstandard(i,:));
         maxCstandard(i)=value;
         if maxCstandard(i)>classStandard
            if (classAverage(i)>totalAverage &&
size(classCounter(i),2)>2*(leastNum+1))||currentNum<=k/2
               %if class average( between 2 clusters) more than total average, split
               lcentre=centre(i,:)-factor*maxCstandard(i);
               hcentre=centre(i,:)+factor*maxCstandard(i);
               temp=centre((i+1):currentNum,:);
               currentNum=currentNum+1;
               centre(i,:)=lcentre;
               centre(i+1,:)=hcentre;
               centre((i+2):currentNum,:)=temp;
            end
         end
      end

   elseif flag==0 % merge
      if I==iterationTime % Last Iteration
         minDis=0;
      end
centreDis=ComputCentreDistance(centre,currentNum);
      %calculate all center Distance
      [indrow indcol] = find(centreDis<minDis);
      %  find the center which is small than the centre Distance
      [indrow indcol]=selectsort(centreDis,indrow,indcol);

for i=1:size(indrow,1)
         if i<=maxCombine
            lengthI=size(class(indrow(i),:),2);

lengthJ=size(class(indcol(i),:),2);
tempCentre(i,:)=(1/(lengthI+lengthJ))*(lengthI*centre(indrow(i),:)+lengthJ*centr
e(indcol(i),:));
```

```
ss = size(tempCentre)
            if indrow(i)<indcol(i)
               centre(indrow(i),:)=tempCentre(i,:);
               temp=centre((indcol(i)+1):currentNum,:);
               currentNum=currentNum-1;
               centre(indcol(i):currentNum,:)=temp;
            else
               centre(indcol(i),:)=tempCentre(i,:);
               temp=centre((indrow(i)+1):currentNum,:);
               currentNum=currentNum-1;
               centre(indrow(i):currentNum,:)=temp;
            end
         else
            break;
         end
      end
   end
   distance=ComputDistance(img,centre,num,currentNum);   %recalculate
   I=I+1;
end   %end while

end

function d=ComputDistance(img,centre,num,currentNum)
%Compute the distance between every sample and every center
d = zeros(num,currentNum);
for i=1:num
   for j=1:currentNum
      d(i,j)=sqrt((img(i,1)-centre(j,1)).^2+(img(i,2)-centre(j,2)).^2+(img(i,3)-
centre(j,3)).^2);
   end
end
end

function m=ComputMid(img,class,currentNum,classCounter)
% calculate center
for i=1:currentNum
   length=classCounter(i);
   csum1=0;
   csum2=0;
   csum3=0;
   for j=1:length
      csum1=csum1+img(class(i,j),1);
      csum2=csum2+img(class(i,j),2);
      csum3=csum3+img(class(i,j),3);
   end
      m(i,1)=csum1/length;
      m(i,2)=csum2/length;
      m(i,3)=csum3/length;
```

```
 end
end

function ac=ComputAverageClass(class,distance,currentNum,classCounter)
% Distance between / N

% calculate avg distance between every cluster to every center
for i=1:currentNum
   length=classCounter(i);
   csum=0;
   for j=1:length
      csum=csum+distance(class(i,j),i);
   end
   ac(i)=csum/length;
end
end

function
at=ComputAverageTotal(class,classAverage,currentNum,num,classCounter)
% calculte total avg
csum=0;
for i=1:currentNum
   length=classCounter(i);
   csum=csum+length*classAverage(i);
end
at=csum/num;
end
function flag=step_judge(currentNum,I,iterationTime,k)
% determine merge or split
if currentNum<=k/2   %if less than 1, split
   flag=1;
elseif I==iterationTime || mod(I,2)==0 || currentNum>=2*k  %  for last iteration,
or more than two times, merge
   flag=0;
else
   flag=1;
end
end

function
Cstandard=ComputCstandard(class,img,centre,currentNum,classCounter)
%  calculate standard deviation
for i=1:currentNum
   length=classCounter(i);       % the class includes index only
   csum1=0;
   csum2=0;
   csum3=0;
   for j=1:length
      csum1=csum1+(img(class(i,j),1)-centre(i,1))^2;
      csum2=csum2+(img(class(i,j),2)-centre(i,2))^2;
```

```
csum3=csum3+(img(class(i,j),3)-centre(i,3))^2;
   end
   Cstandard(i,1)=sqrt(csum1/length);
   Cstandard(i,2)=sqrt(csum2/length);
   Cstandard(i,3)=sqrt(csum3/length);
end
end

function centreDis=ComputCentreDistance(centre,currentNum)
% calculate all of the distance between cluster center
for i=1:currentNum
   j=1;
   while j<=currentNum
      if j<=i
         centreDis(i,j)=32768; %if element index smaller than the class number
then D = 32768
      else
         centreDis(i,j)=sqrt((centre(i,1)-centre(j,1))^2+(centre(i,2)-
centre(j,2))^2+(centre(i,3)-centre(j,3))^2);
      end
      j=j+1;
   end
end
end

function [indrow indcol]=selectsort(centreDis,indrow,indcol)
% arrange according to the distance between class
for i=1:size(indrow,2)-1
   min=i;
   for j=1:size(indrow,2)
      if centreDis(indrow(min),indcol(min))>centreDis(indrow(j),indcol(j))
         min=j;
      end
   end
   if min~=i
      temprow=indrow(i);
      tempcol=indcol(i);
      indrow(i)=indrow(min);
      indcol(i)=indcol(min);
      indrow(min)=temprow;
      indcol(min)=tempcol;
   end
end
end

function class=move(class,i)
% move the class to front
for j=i:size(class,1)-1
      class(j)=class(j+1);
end
```

```
end

function flag=isModify(currentNum,k,class,centreDis,maxCstandard,num)
%  determine whether to modify the parameter
flag=0;
for i=1:size(centreDis,2)  % calculate is centerDistance larger than standard
deviation of the sample
    for j=1:size(centreDis,2)
        if j>i
            if centreDis(i,j)<maxCstandard(i) || centreDis(i,j)<maxCstandard(j)
                flag=1;
                break;
            end
        end
    end
 end

for i=1:currentNum
    if size(class(i,:),2)<num/(k+1)
        flag=1;
        break;
    end
end

if currentNum~=k
    flag=1;
end
end
```


Appendix C - Coding for getting matrices for ground truth image

```
IM_input = (imread('colorMap.jpg'));  % the input image to be processed.
[r,c,l] = size(IM_input);        % num of rows and columns of the iput image.

for R = 1:r
    for C = 1:c

if (6 >IM_input(R,C,1)&& IM_input(R,C,1)> -1) && (6 >IM_input(R,C,2) &&
IM_input(R,C,2)> -1) && (256 >IM_input(R,C,3) && IM_input(R,C,3)> 200)
      Final_I(R,C,1:3) = [0 0 255];
      GT_Matrix(R,C) = 1;
  elseif (256 >IM_input(R,C,1) && IM_input(R,C,1)> 200) && (256
>IM_input(R,C,2) && IM_input(R,C,2)> 200) && (256 >IM_input(R,C,3) &&
IM_input(R,C,3)> 200)
      Final_I(R,C,1:3) = [255 255 255];
      GT_Matrix(R,C) = 2;
  elseif (20 >IM_input(R,C,1) && IM_input(R,C,1)> -1) && (256
```

```
>IM_input(R,C,2) && IM_input(R,C,2)> 100) && (20 >IM_input(R,C,3) &&
IM_input(R,C,3)> -1)
      Final_I(R,C,1:3) = [0 128 0];
      GT_Matrix(R,C) = 3;
  elseif (256 >IM_input(R,C,1) && IM_input(R,C,1)> 200) && (50
>IM_input(R,C,2) && IM_input(R,C,2)> -1) && (50 >IM_input(R,C,3) &&
IM_input(R,C,3)> -1)
      Final_I(R,C,1:3) = [255 0 0];
      GT_Matrix(R,C) = 4;
  elseif (5 >IM_input(R,C,1) && IM_input(R,C,1)> -1) && (5 >IM_input(R,C,2)
&& IM_input(R,C,2)> -1) && (5 >IM_input(R,C,3) && IM_input(R,C,3)> -1)
      Final_I(R,C,1:3) = [0 0 0];
      GT_Matrix(R,C) = 5;
  end

    end
end
imshow(Final_I);
GT_Matrix
```


Appendix D - Coding for Confusion Matrix and Accuracy Calculation

```
ImgResult = Class_M;
ImgTruth = GT_Matrix;
Numofclass = 5;

%% OBTAIN INFORMATION
 [m,n] = size(ImgTruth);
 CM = zeros(5,5);                %initialize the confusion matrix
%% OBTAIN THE CONFUSION MATRIX
 for i=1:m
    for j=1:n
       if(ImgTruth(i,j)==0)
          continue;
       end
       t=ImgResult(i,j);    %obtain the  label from the classification result
       k=ImgTruth(i,j);     %obtain the true label
       if ((t ~= 0) && (k ~= 0));
          CM(k,t)=CM(k,t)+1;    %confusion matrix assignment
       end
    end
 end

%% CALCULATE EVALUATION METRICS
 [m,n] = size(CM);
 r=m;
```

```
rowsum=zeros(1,m);         %store the sum of the row value
 columnsum=zeros(1,n);     %store the sum of the column value
 N=0;                      %store the total number of the pixels
 Diagsum=0;                %sum of the diag
 ProAcc=zeros(m,1);        %store the producer accuracy for every class
 sensitivity=zeros(m,1);   %store the user accuracy for every class
 AveAcc=zeros(2,1);        %store the average producer and user accuracy
 for i=1:m
    for j=1:n
        rowsum(i)=rowsum(i)+CM(i,j);   % sum of the rows and columns
        columnsum(j)=columnsum(j)+CM(i,j);
        N=N+CM(i,j);      % compute the total number of the pixels
        if(i==j)
            Diagsum=Diagsum+CM(i,i);     % compute the sum of the pixels which
are rightly classified
        end
    end
 end

 Pc=0;
 for i=1:r
    Pc=Pc+rowsum(i)*columnsum(i);
 end
%% CALCULATE THE METRICS
 Kappa=(N*Diagsum-Pc)/(N*N-Pc);
 OveAcc=Diagsum/N;
for i=1:m
    ProAcc(i)=CM(i,i)/rowsum(i);
    sensitivity(i)=CM(i,i)/columnsum(i);
    W = N - (columnsum(i) - rowsum(i) - CM(i,i));
    specificity(i) = W / (W + rowsum(i) - CM(i,i));
end
AveAcc(1)=sum(ProAcc)/Numofclass;
AveAcc(2)=sum(sensitivity)/Numofclass;
```