



**HAL**  
open science

# Multilevel fusion for classification of very high resolution remote sensing images

Ahsan Ahmad Ursani

► **To cite this version:**

Ahsan Ahmad Ursani. Multilevel fusion for classification of very high resolution remote sensing images. Environmental Engineering. INSA de Rennes, 2010. English. NNT : . tel-00922645

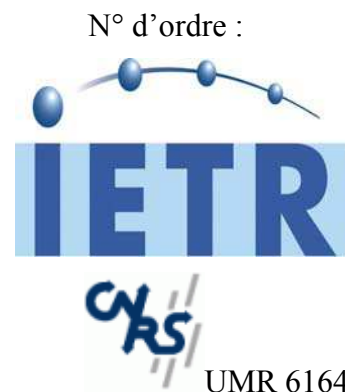
**HAL Id: tel-00922645**

**<https://theses.hal.science/tel-00922645>**

Submitted on 29 Dec 2013

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Ecole Doctorale MATISSE

# THESE

Présentée à

**L'INSTITUT NATIONAL DES SCIENCES APPLIQUEES**

Pour l'obtention du

**DOCTORAT**

Spécialité

**Traitement du Signal et de l'Image**

**Ahsan Ahmad URSANI**

---

**Multilevel fusion for classification of very high  
resolution remote sensing images**

Fusion multiniveau pour la classification d'images de  
télédétection à très haute résolution spatiale

---

*Soutenance prévue le 4 Novembre 2008*

Directeur de Thèse : **Joseph RONSIN**  
Co-encadrant : **Kidiyo KPALMA**

Membres du jury :

**Lorenzo BRUZZONE**  
Professor, University of Trento, Italy

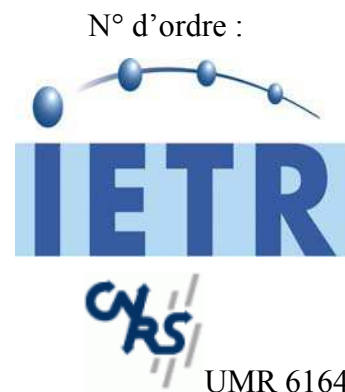
**Kidiyo KPALMA**  
Maître de conférences, INSA de Rennes

**Camille LELONG**  
chercheur au CIRAD/UMR TETIS Montpellier

**Laurence HUBERT-MOY**  
Professeur, Université de Rennes II

**Laurent FERRO-FAMIL**  
MdC (HDR), Université de Rennes I

**Joseph RONSIN**  
Professeur, INSA de Rennes



Ecole Doctorale MATISSE

# THESE

Présentée à

**L'INSTITUT NATIONAL DES SCIENCES APPLIQUEES**

Pour l'obtention du

**DOCTORAT**

Spécialité

**Traitement du Signal et de l'Image**

**Ahsan Ahmad URSANI**

---

**Multilevel fusion for classification of very high  
resolution remote sensing images**

Fusion multiniveau pour la classification d'images de  
télédétection à très haute résolution spatiale

---

*Soutenance prévue le 4 Novembre 2008*

Directeur de Thèse : **Joseph RONSIN**  
Co-encadrant : **Kidiyo KPALMA**

Membres du jury :

**Lorenzo BRUZZONE**  
Professor, University of Trento, Italy

**Kidiyo KPALMA**  
Maître de conférences, INSA de Rennes

**Camille LELONG**  
chercheur au CIRAD/UMR TETIS Montpellier

**Laurence HUBERT-MOY**  
Professeur, Université de Rennes II

**Laurent FERRO-FAMIL**  
MdC (HDR), Université de Rennes I

**Joseph RONSIN**  
Professeur, INSA de Rennes



**To my life that is  
Guided by my loving parents  
Adorned by my beautiful wife  
Made colourful by my three cute sons,  
Made easy by my helping brothers and sisters  
And  
Owned by my motherland Sindh**



## **Acknowledgement**

I would like to express my gratitude to Professor Joseph Ronsin for accepting me as his PhD student. The radiant guidance of Professor Ronsin and Professor Kidiyo Kpalma has led my research to fruit-bearing success that is vital to my professional growth. The successful completion of this thesis would have not been possible without their immense help. Critic view of the two on the write up of this thesis has really polished the presentation of this research work.

Thanks are due to Mrs. Camille Lelong of CIRAD, who generously and graciously provided not only the RS images including the panchromatic and the multispectral images for this research, but also provided the corresponding ground truth collected through field survey.

I am thankful to all the smiling faces at INSA de Rennes, especially former and the present secretaries of the department of EII, Mrs. Catherine Dicostanzo and Mrs. Jocelyne Tremier, who have been friendly and helping at many difficult times.

I am also grateful to my neighbours who have been a great help and moral support for me and my family during last more than two years. I would like to mention the names of Martine Cahour, Bernard Kittous, Corrine Kittous, Samuel Lepuissant and Aurore Lepuissant.

Last but not the least; I am indebted to Mehran University of Engineering & Technology Jamshoro Sindh for financially supporting my PhD studies at INSA de Rennes, France.





# List of contents

<b>Introduction</b>	<b>XI</b>
<b>State-of-the-art vista of remote sensing and image fusion</b>	<b>1</b>
Part A: Remote Sensing.....	2
1.1 Introduction to remote sensing .....	2
1.2 Applications of remote sensing .....	3
1.2.1 Remote sensing in assorted areas.....	3
1.2.2 Remote sensing of environment.....	3
1.3 The remote sensing imagery.....	4
1.3.1 Platforms of remote sensing.....	4
1.3.2 Types of RS imagery .....	5
1.3.2.1 Optical images.....	5
1.3.2.2 SAR images.....	10
1.3.2.3 Light detection and ranging (LIDAR) images.....	12
1.3.3 Very high resolution satellite imagery .....	15
1.4 Typical land-covers .....	15
1.5 Spectral signatures of land-covers.....	16
1.6 Vegetation covers and their mapping .....	16
1.7 Vegetation indices .....	17
1.7.1 Simple ratio (SR) .....	19
1.7.2 Normalized difference vegetation index (NDVI).....	19
1.7.3 Soil adjusted vegetation index (SAVI) .....	21
1.7.4 Modified SAVI (MSAVI).....	21
1.7.5 Transformed difference vegetation index (TDVI) .....	22
1.7.6 Normalized green-red difference index (NGRDI) .....	22
1.7.7 Atmospherically corrected vegetation indices .....	22
Part B: Image Fusion .....	24
1.8 Pixel level fusion .....	25
1.9 Feature level fusion .....	26
1.10 Decision level fusion .....	26
1.11 Multilevel Fusion.....	27
1.12 Region based fusion .....	27
1.13 Conclusion.....	28
<b>Texture features for classification of land-covers</b>	<b>29</b>
Part A: Theoretical concepts.....	30
2.1 Image texture .....	30
2.2 Texture description.....	30
2.2.1 Ideal texture description.....	31
2.2.2 Practical texture description.....	31
2.2.3 Texture features based on Gabor filters .....	32
2.2.4 DFT-based texture signatures .....	34
2.2.4.1 Extraction of LFH based features.....	34
2.3 Modifications to LFH-based features .....	36
2.3.1 Circular neighbourhood .....	36
2.3.2 The phase features .....	37
2.3.2.1 Local image gradient.....	38
2.3.2.2 Rotation invariance of the phase features.....	40
2.3.3 Smaller neighbourhoods .....	41
2.3.4 Quantisation.....	42

2.4	Computational cost analysis.....	42
Part B:	Experimental setup.....	44
2.5	Texture datasets.....	44
2.5.1	The training set.....	44
2.5.2	The test image sets.....	46
2.5.2.1	Test images without noise.....	46
2.5.2.2	The test images with Gaussian white noise.....	47
2.6	Performance evaluation.....	48
2.6.1	Recognition.....	49
2.6.2	Rotation invariant texture features.....	49
2.6.3	Retrieval.....	49
2.7	Results.....	50
2.7.1	The test images without noise.....	50
2.7.1.1	Recognition.....	50
2.7.1.2	Retrieval.....	53
2.7.2	The test images with Gaussian white noise.....	56
2.7.2.1	Recognition.....	57
2.7.2.2	Retrieval.....	57
2.8	Conclusion.....	58

**Image segmentation and classification 61**

3.1	Segmentation and classification.....	62
3.2	Unsupervised classification.....	62
3.2.1	Single-run k-means algorithm.....	64
3.2.1.1	Methods of initialisation.....	64
3.2.1.2	K-means with multiple restarts.....	65
3.2.1.3	Gene Cluster 3.0.....	65
3.2.1.4	Problems with k-means.....	65
3.2.2	Notion behind FOOS.....	66
3.2.3	Fusion of over-segmentations (FOOS).....	67
3.2.3.1	Over-segmentations.....	67
3.2.3.2	Fusion.....	68
3.2.4	Computational complexity.....	69
3.2.5	Comparison.....	70
3.2.5.1	Synthetic texture images.....	71
3.2.5.2	Non-image datasets.....	82
3.3	Supervised classification.....	83
3.3.1	The nearest neighbour classification.....	83
3.3.2	K nearest neighbours classification.....	84
3.3.3	Similarity criterion.....	84
3.3.4	Editing the lookup table.....	84
3.4	Conclusion.....	85

**Fusion of region and class information for classifying remote sensing images 87**

4.1	The source RS images.....	88
4.1.1	Panchromatic image.....	88
4.1.2	Multi-spectral (MS) image.....	88
4.1.3	The land-covers.....	90
4.1.4	The training regions.....	91
4.1.5	The test regions.....	92
4.2	Segmentation.....	95
4.2.1	Template matching.....	95
4.2.2	Texture signatures.....	97
4.2.3	Spectral signatures.....	97
4.3	Classification.....	97
4.3.1	The training phase.....	98

4.3.2	The testing phase .....	99
4.3.2.1	Region classification .....	99
4.3.2.2	Grid classification .....	99
4.4	Fusion of the region and the class information.....	100
4.4.1	Scheme 1.....	101
4.4.2	Scheme 2.....	101
4.4.3	Discussion.....	102
4.5	Performance evaluation.....	103
4.5.1	Kappa statistics .....	103
4.5.1.1	Error matrix (EM) .....	103
4.5.1.2	User's accuracy (UA).....	104
4.5.1.3	Producer's accuracy (PA).....	104
4.5.1.4	Overall classification accuracy (OCA).....	104
4.5.1.5	Product matrix (PM).....	105
4.5.1.6	Expected classification accuracy (ECA) .....	105
4.5.1.7	Kappa .....	106
4.5.2	The resulting land-cover maps and their accuracy.....	106
4.6	Further improvements .....	112
4.6.1	Polygon filling .....	112
4.6.2	Two block-sizes .....	112
4.6.3	Feature-level fusion .....	113
4.7	Conclusion.....	113
 <b>Conclusion</b>		<b>115</b>
 <b>Appendices</b>		<b>117</b>
 <b>List of figures</b>		<b>119</b>
 <b>List of pictures</b>		<b>123</b>
 <b>List of tables</b>		<b>125</b>
 <b>List of publications</b>		<b>127</b>
 <b>Bibliography</b>		<b>129</b>



## Introduction

*"So it is clear, since there will be no end to time and the world is eternal, that neither the Tanais nor the Nile has always been flowing, but that the region whence they flow was once dry: for their effect may be fulfilled, but time cannot. And this will be equally true of all other rivers. But if rivers come into existence and perish and the same parts of the earth were not always moist, the sea must need change correspondingly. And if the sea is always advancing in one place and receding in another it is clear that the same parts of the whole earth are not always either sea or land, but that all this changes in course of time."*

*Meteorologica by Aristotle*

The environment of Earth keeps changing since ever. Understanding the phenomena underneath this change has always been a challenge for Human, who is in quest of exploring the physical principles governing the universe. To understand these phenomena, one needs studying the Earth as a system and precisely tracking the changes in the physical parameters governing it. Remote sensing (RS) provides the primary means necessary to accomplish this task.

Automatic methods of artificial cognition combined with the availability of advanced space and electronic sensor technologies have made remote sensing and its tremendous applications a reality. The launch of the first satellite by USSR in 1957 marked the beginning of the era of RS. Soon USSR and USA had placed their meteorological and Earth observation satellites in the Earth's orbit. With the RS technology becoming commercially viable and feasible, it is not only broadening its scope of applications but also making inroads into the developing countries. This is evident from the fact that India launched its first RS satellite in 1997, followed by joint RS satellite of China and Brazil in 1999, and the fact that there are several other developing countries soon planning to have their own RS satellites. Those who do not possess their own RS satellites are also using commercial RS facilities to their benefit.

The RS satellites have been using various technologies, e.g. synthetic aperture radar (SAR), optical sensors in the visible and infrared bands, and LIDAR, etc. for imaging. The SAR images provide information regarding land-covers that is complimentary to the optical images. The optical images provide spectral information of the land-covers, whereas the SAR images provide structural information. Therefore, conventionally, the optical and SAR images are fused together to obtain the optimal information for land-cover classification. Sometimes, spectral features and structural features extracted from optical and SAR images, respectively, are fused before performing the classification.

Although SAR and optical RS have been providing high-resolution images since long, the optical images from RS satellites never had a sub-metre resolution before the launch of IKONOS RS satellite at the end of the 20<sup>th</sup> century and QuickBird as well as Orbview-3 RS satellites in the beginning of this century. The spatial resolution of present and near-future non-military space-

borne SAR sensors is weaker than that of the optical sensors. In addition, the inherent speckle noise in SAR images reduces the effective resolution. The availability of high-resolution optical RS images has made it possible to apply texture analysis for discrimination of land-covers that remain indistinguishable otherwise due to their similar spectral characteristics.

This thesis aims at presenting the work carried out for applying texture analysis on very high-resolution RS images and combining the results of supervised and unsupervised classifications of the RS images for improved classification of vegetation covers. It takes a site from Nîmes, France as a test case, which includes the apple trees, orchards, vineyards, fields, truck crops, fallows, and forest. The author's contribution includes an initialisation method for the unsupervised classification method called k-means clustering, a method of introducing rotation invariance into the texture features based on discrete Fourier transform, and a fusion method that merges the results of unsupervised classification and supervised classification.

Chapter 1 introduces remote sensing technology and its applications with the emphasis on environmental management. After presenting the state-of-the-art on environmental management, it goes on discussing more precisely the problem of land-cover classification by introducing the land-covers themselves and their spectral characteristics. The vegetation indices extracted from multispectral RS imagery are also explained. These indices are conventionally used by the RS community in diverse applications in remote sensing of the environment. The chapter also presents image fusion and its applications in remote sensing.

Since this thesis concentrates on texture-based identification of the land-covers, chapter 2 introduces methods of extracting texture features. It also suggests improvements in one of the texture features-sets and compares their performance for image recognition as well as retrieval. The chapter also proposes a parameter called rotation variance for performance evaluation of the texture features.

Chapter 3 presents the problems of image segmentation, and classification. It also introduces methods of supervised and unsupervised classification. Further, it discusses the pros and cons of k-means clustering in depth and then proposes improvements by suggesting an initialisation method for it.

Chapter 4 reinforces the theoretical notions presented in the preceding chapters by segmenting and classifying the parts of a high-resolution RS image. The texture features proposed in chapter 2 are applied to the RS image from the test site. Improved classification of the RS image results from the fusion of unsupervised classification driven by spectral features and supervised classification driven by texture features. The conclusion marks the end of this thesis and proposes possible extensions to this research.

# 1 State-of-the-art vista of remote sensing and image fusion

<b>Part A: Remote Sensing</b> .....	2
1.1 Introduction to remote sensing .....	2
1.2 Applications of remote sensing .....	3
1.2.1 Remote sensing in assorted areas .....	3
1.2.2 Remote sensing of environment.....	3
1.3 The remote sensing imagery.....	4
1.3.1 Platforms of remote sensing.....	4
1.3.2 Types of RS imagery.....	5
1.3.2.1 Optical images.....	5
1.3.2.1.1 Panchromatic images .....	5
1.3.2.1.2 Multi-spectral (MS) image .....	6
1.3.2.2 SAR images.....	10
1.3.2.3 Light detection and ranging (LIDAR) images.....	12
1.3.3 Very high resolution satellite imagery .....	15
1.4 Typical land-covers .....	15
1.5 Spectral signatures of land-covers .....	16
1.6 Vegetation covers and their mapping .....	16
1.7 Vegetation indices .....	17
1.7.1 Simple ratio (SR) .....	19
1.7.2 Normalized difference vegetation index (NDVI).....	19
1.7.3 Soil adjusted vegetation index (SAVI).....	21
1.7.4 Modified SAVI (MSAVI) .....	21
1.7.5 Transformed difference vegetation index (TDVI).....	22
1.7.6 Normalized green-red difference index (NGRDI).....	22
1.7.7 Atmospherically corrected vegetation indices.....	22
<b>Part B: Image Fusion</b> .....	24
1.8 Pixel level fusion .....	25
1.9 Feature level fusion .....	26
1.10 Decision level fusion.....	26
1.11 Multilevel Fusion .....	27
1.12 Region based fusion .....	27
1.13 Conclusion .....	28

This chapter introduces the reader to the remote sensing as a field of research, and image fusion as applied to remote sensing. The chapter has two parts.

The subject of the first part is the science and practice of remote sensing. It introduces the fundamental concepts in remote sensing (RS), platforms of remote sensing, types, and different applications of the RS imagery. It also highlights the use of RS imagery in various problems of environmental management.

The second part of this chapter concerns with the image fusion. It introduces image fusion and abstraction levels of image fusion, discusses different applications of image fusion in assorted areas in general and in the area of remote sensing in particular.

## **Part A: Remote Sensing**

### **1.1 Introduction to remote sensing**

Quest of understanding the Earth as a system and the physical phenomena observed therein is as old as Man's history. Apart from hypothesis, pursuing this quest also requires global but precise observations and measurements of the underlying physical parameters. Observing and measuring from a distance without any direct physical contact, namely remote sensing, fulfils this task. The remote sensing (RS) helps detect and make exact estimates of the physical parameters relating to the lands, the oceans, and the atmosphere of Earth. Platforms of kites, balloons, model airplanes, airplanes and artificial satellites collect large scale but precise data regarding the physical conditions at the surface as well as the atmosphere of Earth. The remote sensing experts later interpret these data to extract information useful for specific applications discussed in the succeeding sections.

According to one point of view, invention of photography in the 18<sup>th</sup> century founded the remote sensing. Aerial photography was introduced in France as early as in 1858 and was used for military purposes in the First World War, and therefore was under extensive use in the beginning of the 20<sup>th</sup> century. The unmanned aircrafts, including remote-controlled model crafts, have also been used to carry the remote sensing equipment for landscape survey, advertisement, and espionage. Satellite-borne remote sensing became an earnest reality after the USSR placed its first artificial satellite in the space on 4<sup>th</sup> October 1957. Satellites for meteorological measurements were soon in the orbit around Earth. The RS satellites further divide into two broad categories of meteorological (weather) satellites and the earth observation satellites. Although the weather satellites are platforms of remote sensing, usually the term remote sensing satellite refers to the Earth observation satellites.

Remote sensing has two modes of operation, the active and the passive. The passive remote sensing involves using optical sensors capable of capturing energy emitted by sun and reflected by objects on Earth's surface, in visible, near infrared and thermal infrared bands. The energy that reaches back to the satellite's sensor is subject to physical and chemical properties of the objects under observation. This provides passive remote sensing the opportunity to identify and map the objects under observation using automatic means of intelligent computing. On the other hand, the active remote sensing satellites carry synthetic aperture radars (SAR) and light detection and ranging (lidar) apparatus that emit electromagnetic energy to recapture it as it bounces back from objects on Earth's surface. This thesis is concerned with the analysis of RS images from satellite-borne optical sensors only.



## **1.2 Applications of remote sensing**

Recent developments in electronics in general and optical sensor technology in particular, satellite imagery has now attained the ground resolutions finer than ever before; resulting in the true colour optical images with sub-metre resolution and spreading its use in almost every arena of life. [Gnauck 1999] presents a case study on commercial use of remote sensing.

While fieldwork remains a source of information for some RS applications [Mumby 2004], the cost and time to carry out these methods sometimes may not be feasible for the study. Therefore, the use of RS imagery is finding increasingly more acceptability and use. This section puts some light on the applications of remote sensing in general, and then the applications of remote sensing in particular to the environmental management.

### **1.2.1 Remote sensing in assorted areas**

Defence and military necessities provided the thrust to the earliest development in the remote sensing technology. Later, with the technology being cheaper with time, the remote sensing extended its applications in civilian use. Today, the remotely sensed data find their use in as diverse applications as disaster assessment [Chen 2005b], weather forecasting, natural resources exploration, agricultural research [Doraiswamy 2007], rural land use, urban land use [Herold 2002], urban planning [Harman 2002], and ecological studies [Wang 2004]. More examples include healthcare management [Tran 2002], cartography [Akono 2000], archaeology [Pantani 2000], vegetation cover mapping [Hirose 2004], extraction of road network [Shackelford 2003], and estimation of socioeconomic attributes [Jensen 1999]. All the types of RS images including optical images, SAR images, and the lidar images [Baek 2005] are used to develop digital elevation models (DEM) and digital terrain models (DTM). The DEM and DTM provide the height information, i.e. 3D information, of the imaged area, used in positioning systems.

### **1.2.2 Remote sensing of environment**

The environmental concerns are at height for last few decades as the problems of global warming and environmental pollution have surfaced. Since then, scientific research has substantially been diverted to natural conservation, environmental protection, and sustainable development. The pertinent fields of environmental management are ocean and coastal management, agricultural management, forest management, oceanography, wetland management, assessment of damage by natural disasters, and wildlife management. The notable but non-exhaustive literature on these fields include [Ward 2000] that presents methods of identifying populations potentially exposed to agricultural pesticides using remote sensing, [Curran 1987] that shows that remote sensing was used for agricultural management as far back as in 1987, [Tsiligirides 1998] that explores remote sensing as a tool for agricultural statistics, [Wright 2007] that introduces methods of wetland mapping, [Gordon 1983] that presents a review of remote sensing of oceans as far back as in 1983,

[Chen 2005c] that demonstrates how remote sensing can be effective in monitoring the dynamics of coastal zones, [Sande 2003] that assesses damages caused by flood, and [Lelong 2003] that applies RS technology to map the coffee crops in Uganda.

In ocean remote sensing, the most researched problem is the oil spill detection, whereas in case of remote sensing of forests, it is the detection of forest fires. [Fingas 1997] and [Brekke 2005] present state-of-the-art of the research on the oil spill detection by satellite remote sensing. The studies of the coastal marine pollution [DiGiacomo 2004] and the land-cover change [Chen 2005c] dominate the remote sensing of coastal areas. Mapping invasive plants in estuaries is also a popular application of remote sensing [Laba 2008]. For agricultural management, [Lelong 2003] considers the identification of crop covers the most challenging job because of the similar spectral characteristics of different crops. In addition, the spectral features also vary in the same crop with its health and growth level.

### **1.3 The remote sensing imagery**

The firsthand product of remote sensing is an image captured by a detector or a sensor onboard a platform high above in the atmosphere of Earth or even higher in the space. Each of the platforms and the sensors has its own advantages and limitations as discussed in the following sections.

#### **1.3.1 Platforms of remote sensing**

The remote sensing (RS) images are the images captured by various sensors onboard a spacecraft or an aircraft. Each RS platform has its own pros and cons. The images from airborne-sensors have many advantages over their space-borne counterparts and at the same time have many limitations and disadvantages as well. [Rees 2001] discusses pros and cons briefly but exhaustively. Besides others, the problem of the geometrical deformations in the images from the airborne sensors caused by turbulence in the aircraft is the most critical problem. Three kinds of aircraft movement called pitch, yaw and roll cause the geometrical deformations in the images. Figure 1.1, from [Rees 2001] depicts the same.

The satellite imagery now provides relatively cheaper, versatile and quite frequent detail than ever before to assist the researchers in many disciplines. The space-borne images also have the problem of the geometric distortions but less severe. However, for some applications, e.g. scouting over agricultural lands, the temporal resolution required is much higher than is available with the satellite borne platforms. In addition, acquiring multi-temporal data from space-borne platforms becomes prohibitively expensive. In addition, some applications like crop scouting [Jensen 2003] are not severely affected by geometrical deformations and do not require covering very large areas. In such cases, imaging equipment mounts over a radio-controlled model airplane to collect the data over the target area quickly and repeatedly as desired. The radio-controlled model airplanes can fly

at altitudes as low as 100 metres to gather the data with the resolutions of a few centimetres per pixel [Harman 2002].

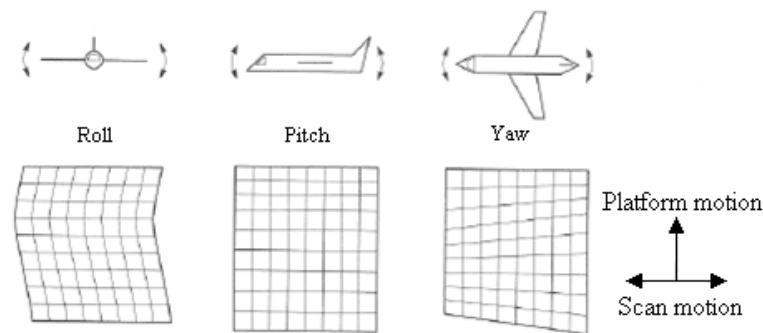


Figure 1.1: Geometric Distortions in an image captured by airborne sensor due to Pitch, Yaw and Roll motions of the platform

With each platform having its own pros and cons, the trade-off lies between the cost of platform, spatial, temporal, spectral, and radiometric resolution of the data that are displayed and interpreted as an image.

### 1.3.2 Types of RS imagery

The remote sensing images divide into three broad categories:

- Optical images
- Synthetic aperture radar (SAR) images
- Light detection and ranging (lidar) images

#### 1.3.2.1 Optical images

Optical sensors on board a spacecraft (satellite) or an aircraft capture the optical images. The optical images divide into two broad categories, Panchromatic (PAN) and Multi-spectral (MS). Most of the satellite platforms, all the prominent ones, have instruments to capture both, MS as well as PAN images. The SPOT, QuickBird, IKONOS, and Orbview are among well-known remote sensing satellites observing the Earth and providing the two kinds of optical images, whereas NOAA and METEOSAT are well-known series of satellites observing the Earth's atmosphere.

##### 1.3.2.1.1 Panchromatic images

These are black & white photographs taken by a single sensor of a wide spectral range covering visible as well as IR wavelengths. These usually appear as grey scale images, i.e. the displayed brightness of a particular pixel is proportional to the pixel digital number, which corresponds to the intensity of solar radiation reflected by the targets in the pixel and detected by the sensor. Thus, a panchromatic image appears to be a black-and-white photograph of the area as in picture 1.1 that

shows PAN image of Washington DC captured by IKONOS. Though it is also different in a way that unlike black and white photograph, PAN sensors of some satellites also cover the infrared wavelengths and in some satellites, these do not cover the blue wavelengths. The PAN images always have greater resolutions than the MS images from the same satellite. It is due to the much more energy per unit area gathered by a PAN sensor due to its wider bandwidth. Refer figure 1.2 that shows the responses of the PAN as well as MS sensors of QuickBird2 satellite to the visible and the infrared part of the electromagnetic spectrum.

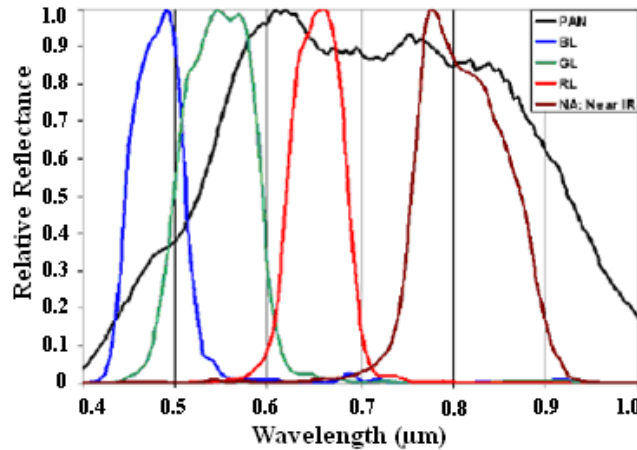


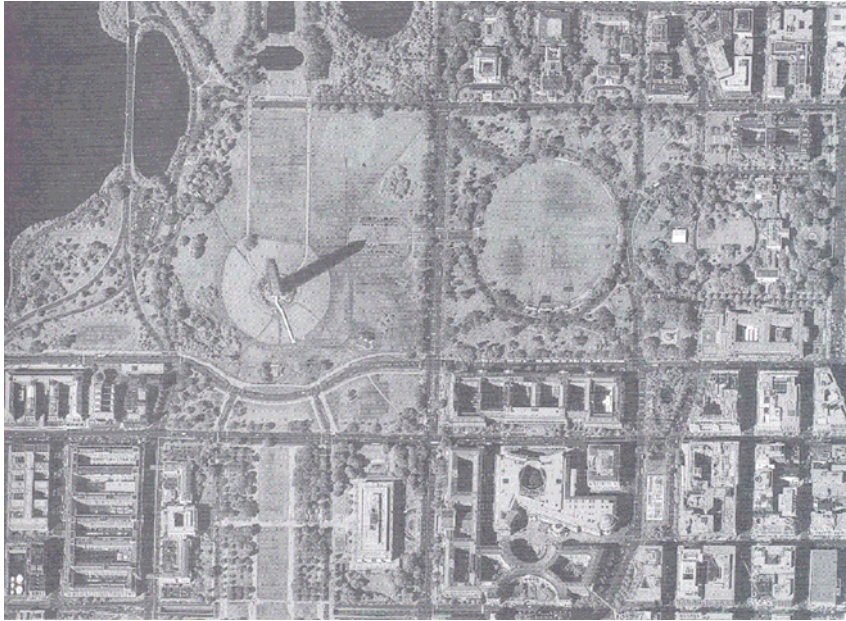
Figure 1.2: Sensor bands of QuickBird2 remote sensing satellite<sup>1</sup>

### 1.3.2.1.2 Multi-spectral (MS) image

An MS image is comparatively a low resolution image captured by more than one sensors simultaneously in different bands; collectively covering the visible as well as infrared range of wavelengths. Table 1.1 shows different bands of various satellites that provide optical image data. As shown in the table, some satellites like IKONOS and QuickBird have three bands in the visible spectrum, each corresponding to a basic colour i.e. red, green, and blue, which can be used to generate true colour images. Picture 1.2 depicts a scene from an urban area with a true colour image having 1 m resolution captured by IKONOS. The original resolution of the MS image is 4 m, but it is enhanced by its fusion with the corresponding panchromatic image having 1 m resolution.

In the satellites like SPOT-5, which do not have a sensor corresponding to blue colour band, the RS image analysts generate false-colour images by substituting the Blue colour band with the near infrared (NIR) band. Sometimes, they assign the NIR, Red and Green bands to red, green, and blue channel, respectively, of the display to generate false-colour images [Rees 2001].

<sup>1</sup> Source: <http://www.satimagingcorp.com/satellite-sensors/quickbird.html>



Picture 1.1: An IKONOS Panchromatic image of Washington DC [Jensen 2005]



Picture 1.2: A true colour IKONOS image<sup>2</sup>

Picture 1.3 from [Jensen 2005] shows a false colour composite image of Pearl harbour captured by TERA-A-ASTER. The image appears with following false colour combination:

---

<sup>2</sup> Source: <http://www.crisp.nus.edu.sg/~research/tutorial/image.htm>

Satellite	Instrument	Nation	Launch Date	Multispectral								PAN	Max Resolution
				Visible			Infrared						
				Band 1	Band 2	Band 3	Near IR		Thermal IR				
			Band 4	Band 5	Band 6	Band 7	Band 8						
SPOT 5	HRG	France	May 3, 2002	-	0.50 – 0.59	0.61 – 0.68	0.78 – 0.89	1.58 – 1.75	-	-	-	0.48 – 0.71	10 m 2.5 m
NOAA-N (POES-4)	AVHRR-3	USA	May 20, 2005	-	-	0.58 – 0.68	0.725 – 1.10	1.58 – 1.68	3.55 – 3.93	10.3 – 11.3	11.5 – 12.5	-	1.09 km
Landsat-7	ETM	USA	April 15, 1999	0.45 – 0.52	0.53 – 0.61	0.63 – 0.69	0.78 – 0.90	1.55 – 1.75	2.09 – 2.35	10.4 – 12.5		0.52 – 0.90	30 m 15 m
IKONOS	OSA	USA	September 24, 1999	0.45 – 0.52	0.52 – 0.60	0.63 – 0.69	0.76 – 0.90	-	-	-	-	0.45 – 0.90	4 m 0.82 m
QuickBird	BGIS2000	USA	October 18, 2001	0.45 – 0.52	0.52 – 0.60	0.63 – 0.69	0.76 – 0.90	-	-	-	-	0.45 – 0.90	2.44 m 0.61 m
TERRA	ASTER VNIR	USA & Japan	December 1999	-	0.52 – 0.60	0.63 – 0.69	0.76 – 0.86	1.6 – 1.7	2.145 – 2.43	10.25 – 11.65	-	-	15 m
IRS-1D	LISS III & PAN	India	September 29, 1997	-	0.52 – 0.59	0.62 – 0.68	0.77 – 0.86	1.55 – 1.7	-	-	-	-	23.5 m 5.8 m
ERS-2	GOME	Europe	-	-	0.54 – 0.56	0.66 – 0.68	0.86 – 0.88	1.45 – 1.75	3.55 – 3.85	10.3 – 11.3	11.5 – 12.5	-	
ALOS-1	PRISM AVNIR-2	Japan	January 24, 2006	0.42 – 0.50	0.52 – 0.60	0.61 – 0.69	0.76 – 0.89	-	-	-	-	0.52 – 0.77	10 m 2.5 m
Beijing-1 DMC	CMT ESIS	China	October 27, 2005	-	0.50 – 0.59		0.79 – 0.89					-	32 m 4 m
CBERS 2B	CCD IRMSS WFI-1	China & Brazil	September 10, 2007	0.45 – 0.52	0.52 – 0.59	0.63 – 0.69	0.79 – 0.89	0.50 – 1.10	1.55 – 1.75	2.08 – 2.35	1040 – 12.50	0.51 – 0.73	20 m 80 m
Orbview-3	OHRIS	USA	June 26, 2003	0.45 – 0.52	0.52 – 0.60	0.625 – 0.695	0.76 – 0.90	-	-	-	-	0.45 – 0.90	4 m 1 m
TopSat-1	RALCamPAN RALCamMS	UK	2005	0.40 – 0.50	0.555 – 0.60	0.60 – 0.70	-	-	-	-	-	0.5 – 0.7	5 m 2.5 m

Table 1.1: Sources of satellite-borne optical images from different nations (Note: All the measurements are in  $\mu$ meters)

Display channel	Satellite band
Red	Band 4 (invisible NIR)
Green	Band 3 (Visible Red)
Blue	Band 2 (Visible Green)



Picture 1.3: An image remotely sensed in two visible and one NIR band displayed and printed in one RGB colour space



Picture 1.4: Natural colour composite of a SPOT image<sup>3</sup>

---

<sup>3</sup> Source: [http://www.crisp.nus.edu.sg/~research/tutorial/sar\\_int.htm](http://www.crisp.nus.edu.sg/~research/tutorial/sar_int.htm)

The RS image analysts combine visible as well as non-visible MS bands in such a way that the appearance of the displayed image resembles a visible true colour photograph, i.e. vegetation in green, water in blue, soil in brown or grey, etc. Image generated using such a colour composite is called a natural colour image. Picture 1.4 is such an example generated with following combination:

<b>Display channel</b>	<b>Satellite band</b>
Red	Band 2 (Green)
Green	0.75 Band 2 + 0.25 Band 3 (Visible Red)
Blue	0.75 Band 2 - 0.25 Band 3

Sometimes only visible bands are displayed on an RGB display using different colour composites.

Table 1.2: Characteristics of AVHRR sensor of remote sensing satellites from NOAA series

Band No.	Spectrum ( $\mu\text{m}$ )			Band utility
	NOAA-6, 8, 10	NOAA-7, 9, 11, 12, 13, 14	NOAA-15, 16, 17	
1	0.58 – 0.68	0.58-0.68	0.58 – 0.68	Daytime cloud, snow, ice, and vegetation mapping
2	0.725 – 1.1	0.725 – 1.1	0.725 – 1.1	Land/water interface, snow, ice, and vegetation
3	3.55 - 3.93	1.1 – 3.93	A: 1.58 – 1.64	Monitor hot targets (volcanoes, forest fires), night-time cloud mapping
			B: 3.55 – 3.93	
4	10.5 – 11.5	10.3 – 11.30	10.3 – 11.30	Day/night cloud and surface-temperature mapping
5	Band 4 repeated	12.50	11.5 – 12.5	Cloud and surface temperature, day and night cloud mapping; removal of atmospheric water vapour path radiance

Since each object on Earth’s surface and atmosphere has its own reflectivity to electromagnetic waves of various frequencies, each band in a multi-spectral image is useful for imaging or measuring some specific object or quantity. For example, table 1.2 from [Jensen 2005] states the use of each band of the advance very-high resolution radiometer (AVHRR) sensor in the series of NOAA satellites. This fact establishes the importance of multi-spectral imagery and the fact that each one of the bands in a multi-spectral image collects different information. Extracting spectral information is not all about remote sensing. SAR images provide information on land-covers that is complimentary to that provided by the optical images

### 1.3.2.2 SAR images

These are the images synthesised from the radar signals captured by a moving radar onboard a spacecraft or an aircraft. There are several space-borne sources of SAR images. Tropical rainfall measuring mission (TRMM), RADARSAT-1, constellation of small satellites for Mediterranean basin observation (COSMO-SkyMed), ERS-2, ENVISAT and advanced land observing satellite (ALOS) are among the well known remote sensing satellites providing SAR images. ERS-2 and



ALOS satellites also provide the optical images besides SAR images. The SAR images have high resolution suitable for analysis of urban areas; however, researchers are harnessing the SAR images for environmental studies and military applications as well. [Matsuoka 2004] explains the use of SAR imagery in natural disaster assessment, [Di Giacomo 2004] describes the use of SAR images for observing coastal pollution and [Mathiew 1998] discusses the use of SAR imaging for military surveillance.

Radar instruments in different satellites have different polarizations. For example, some have HV polarization, whereas the letter on the left, i.e. H shows that the emitted wave is horizontally polarised and the letter on the right, i.e. V shows that the received wave is vertically polarised. Similarly, other instruments have VH, or more than one polarisations indicated by “&” sign in between. The radar instruments in various satellites also have different operating frequencies. Some satellites have multiple operating frequencies, providing observations of the same scene at different operating frequencies. This provides better discrimination among different vegetation types or the same vegetation type in different states of health or growth. Airborne-radar scanners also capture SAR images. Picture 1.5 shows a high-resolution airborne SAR image.



Picture 1.5: A high-resolution airborne SAR image of Washington DC<sup>4</sup>

An image analyst displays or prints the multi-frequency and/or the multi-polarisation SAR images in colour by assigning one of the RGB display channels to each frequency-polarisation combination. Such an example is illustrated in picture 1.6 showing SAR image of New York city metropolitan area in false colour composite of three bands LHH (displayed in red), LHV (displayed in green), and CHV (displayed in blue) of a space-borne radar imager [Rees 2001]. Here, the letter L represents the band of frequencies ranging from 40 to 60 GHz and letter C represents the band of frequencies ranging from 500 MHz to 1000 MHz.

---

<sup>4</sup> Source: [http://www.sandia.gov/radar/images/dc\\_big.jpg](http://www.sandia.gov/radar/images/dc_big.jpg)



Picture 1.6: Three-band SAR image of New York City metropolitan area [Rees 2001]

The SAR images can be captured any time, in presence or absence of sunlight, in any season or weather conditions. The SAR data is introduced to areas of no information on the optical data, i.e., areas covered by clouds and their shadows, providing useful information about the surface roughness or structure. Hence, the SAR images help differentiate the cover types that are otherwise indistinguishable in the optical images due to their similar spectral characteristics. However, SAR images are more susceptible to noise than their optical counterparts are. The SAR images usually suffer from severe levels of a non-Gaussian multiplicative speckle noise.

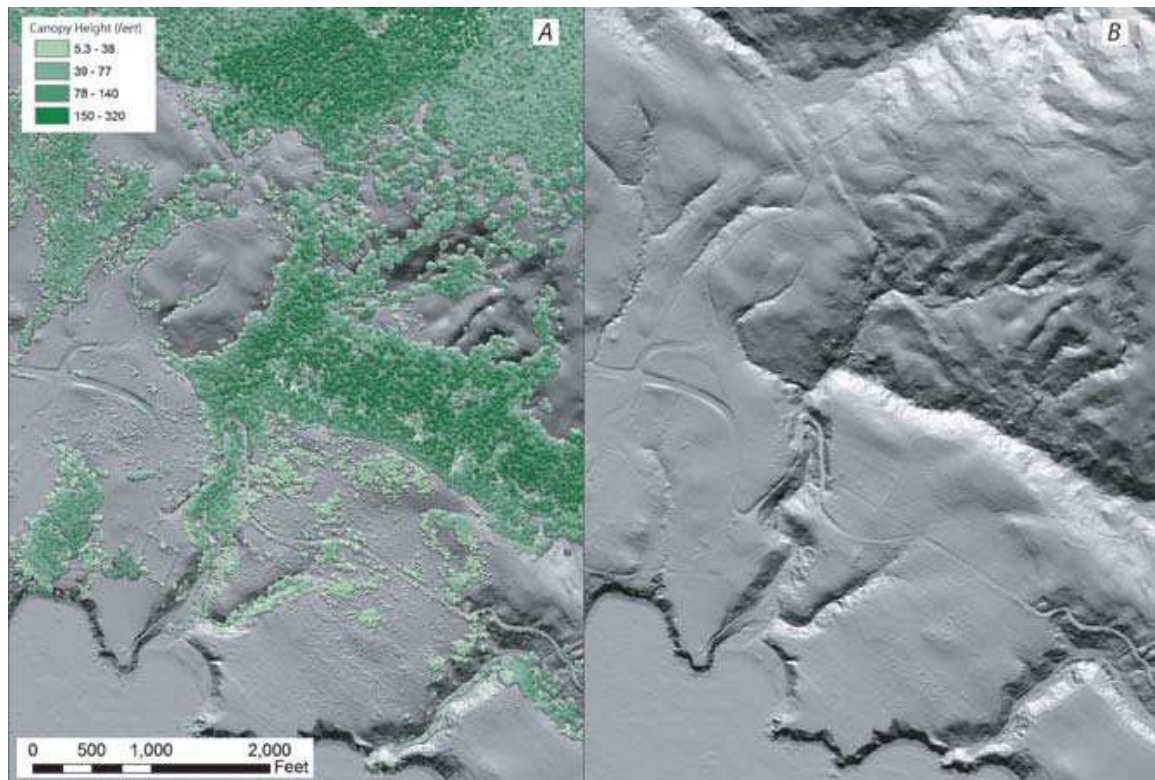
SAR provides the structural information in 2D. A more recent technology called lidar provides the same in 3D.

### **1.3.2.3 Light detection and ranging (LIDAR) images**

The lidar technology is relatively new and the satellite-borne lidar is a very recent development in the field of remote sensing. Lidar works on the same principle as radar, but uses laser light instead of radio waves. The lidar throws a pulse of laser light onto a target at the ground and receives back its echo reflected from the target. The echoed back pulse carries the information of time taken in the round trip, the intensity, and the phase of the echo. One estimates the distance to the target, i.e. the range, from the time it takes for the laser pulse to echo back. On the other hand, phase of the echoed back signal determines the velocity and temperature of the target. The frequency of the laser light used in lidar is usually in the ultra-violet, visible or near infrared range [Thomasson

2001]. In some remote sensing applications, multiple light-beams with different frequencies are used. One such example is [Ulrich 1998] that uses airborne multiple-beam lidar to detect oil spills and algae. Lidar system can also detect more than one echoes of the same pulse so that it can view through water, canopies and trees onto the ground to map surface terrain and to estimate the depth of the water body or plantation. See figure 1.3 showing such a lidar system used to generate surface terrain shown in picture 1.7.

NASA acquired the first remotely sensed lidar-data using a three-wavelength backscatter lidar called lidar in-space technology experiment (LITE). Developed by NASA Langley research centre, LITE flew on the space mission called Discovery in September 1994 and collected over 40 giga bytes of data during its 53 hours long operation, providing highly detailed global view of the vertical structure of cloud and aerosol from the Earth's surface through the middle stratosphere [Winker 2004]. This and other similar missions showed the importance of the data that a space-borne lidar can provide.



Picture 1.7: LIDAR images (a) shows the canopy heights in green shades, (b) shows the surface terrain<sup>5</sup>

After the success of LITE and similar missions, the Geoscience laser altimeter system (GLAS) flew aboard Ice-Sat in January 2002. Later, cloud aerosol lidar and infrared pathfinder observations (CALIPSO) was launched in April 2006, and atmospheric Doppler lidar called ALADIN is planned to be flown aboard European space agency's (ESA) atmospheric dynamics mission ADM-Aeolus

---

<sup>5</sup> Source: [http://www.interactions.org/sgtw/2006/0719/lidar\\_more.html](http://www.interactions.org/sgtw/2006/0719/lidar_more.html)

in June 2009. The lidar satellites are different from the conventional technologies of passive optical sensors and the active SAR sensor in the sense that it can observe both, the Earth and its atmosphere simultaneously. The primary goal of ICESat is to quantify ice sheet mass balance and understand how changes in the Earth's atmosphere and climate affect polar ice masses and global sea level. Its GLAS instrument has a lidar channel for height-resolved data and, therefore, is designed to make aerosol and cloud measurements including measuring ice-sheet topography, cloud and atmospheric properties, and the height and thickness of radiatively important cloud layers needed for accurate short term climate and weather prediction. However, since the lidar instrument of CALIPSO, called CALIOP has as coarse spatial resolution as 125m, it combines the active lidar instrument with passive infrared and visible imagers to probe the vertical structure and properties of thin clouds and aerosols. Similarly, the spatial resolution of GLAS is not sufficient for remote sensing of vegetation either; therefore, [Koetz 2007] combines lidar driven data with the spectral information to forest canopy characterisation.

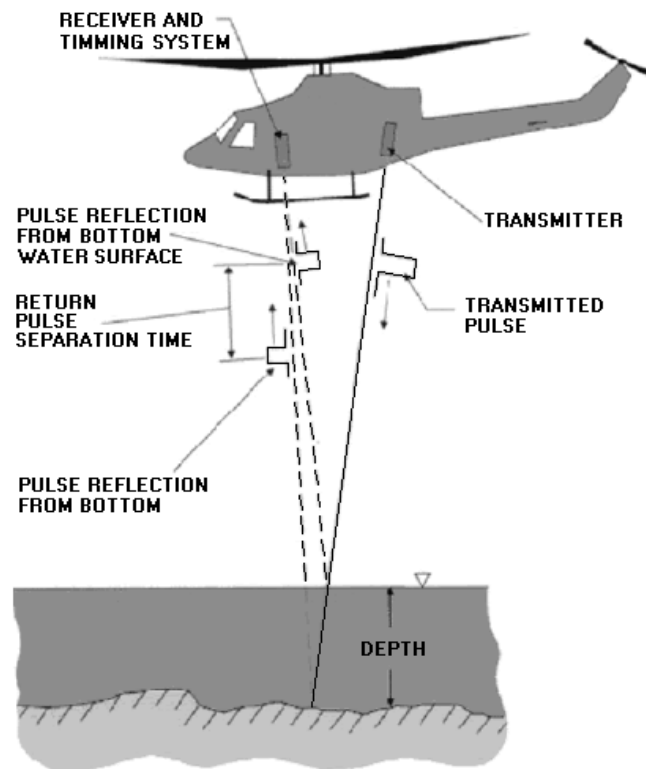


Figure 1.3: An airborne LIDAR remote sensing system<sup>6</sup>

Another space-borne lidar instrument called the Mercury laser altimeter (MLA) is on its way to Mercury, where it will make measurements of the topography of Mercury.

<sup>6</sup> Source: [http://www.hypernexussolutions.com/Projects/Examining\\_lidar\\_in\\_remote\\_sensing-%5Bslide\\_show%5D.pdf](http://www.hypernexussolutions.com/Projects/Examining_lidar_in_remote_sensing-%5Bslide_show%5D.pdf)

### 1.3.3 Very high resolution satellite imagery

The launch of American RS satellite IKONOS in the end of 1999 marked the age of very high-resolution satellite (VHRS) RS images. IKONOS provides the panchromatic images with the finest resolution of 82 cm at nadir and the multi-spectral images with the resolution of 4 m. Two years later followed another VHRS QuickBird that provides panchromatic images with the finest resolution of 61 cm at nadir. Again, with the gap of two years, OrbImage, a subsidiary of Orbital Sciences Corporation (OSC) launched another American VHRS called Orbview-3. Orbview-4 failed to orbit. OSC launched its next mission called Orbview-5, now renamed as GeoEye-1 on September 6, 2008. Orbview-5 provides imagery with the finest resolution of 41 cm. The optical imagery with this much high resolution carries sufficient detail to provide texture information of the land-covers and hence makes them distinguishable despite their similar spectral characteristics.

## 1.4 Typical land-covers

The discriminatory performance of any features extracted from the high-resolution RS images is subject to the number of unique classes to be discriminated within a single scene. Therefore I studied the work carried on land-cover classification using IKONOS images with the scene size of 121 km<sup>2</sup> and found that the minimum number of classes is generally 5 including, water (natural as well as artificial water bodies), greenery (trees, vegetation, grass, fields), bare soil, manmade structures (roads, houses, and other buildings), and ice or snow. Precise classification of land-covers in a large scene, specifically from a rural area, may result in as many as 12 classes including water plant, grass, deciduous tree, coniferous tree, bamboo, bush, bare land, orchard, vegetable field, rice field, open water, and manmade structure [Hirose 2004]. While classifying a scene from forest, [Tatiana 2006] identifies as many as 14 classes including dark/deep water, light/low deep water, water, non-vegetated area, herbaceous vegetation, sparse herbaceous vegetation, eucalyptus, maritime pine, stone pine, cork tree, broadleaved trees, new plantations of eucalyptus, burnt area, and shadow. From ecological point of view, NASA has pointed out nine land-covers, shown in figure 1.4 on the global scale. It should however be noted that these *land-covers* do not include water and ice/snow.

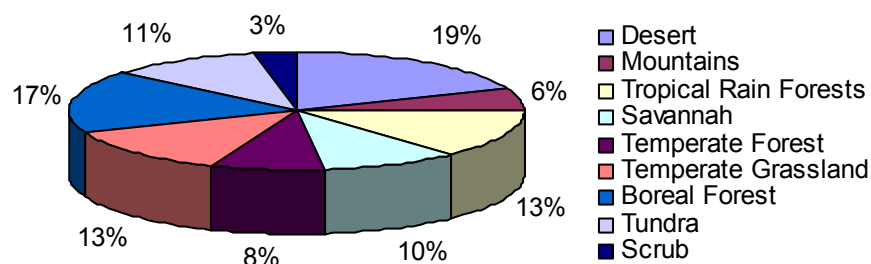


Figure 1.4: Land-cover types identified on global scale<sup>7</sup>

<sup>7</sup> Source: [http://rst.gsfc.nasa.gov/Sect3/Sect3\\_1.html](http://rst.gsfc.nasa.gov/Sect3/Sect3_1.html)

## 1.5 Spectral signatures of land-covers

The natural land-covers can be broadly categorised into three, i.e. water, bare non-vegetated land (soil) and vegetation. The three have distinct spectral characteristics that help discriminate them in the RS imagery. Figure 1.5 shows spectral response of five classes with water subdivided into two categories of clear water and turbid water, and the soil subdivided into two categories of silty clay and musk soil. It shows how the reflectance of the five classes varies with frequency. The five land-covers in figure 1.5 appear spectrally distinct and can be easily distinguished in satellite-borne RS imagery. However, each of these classes subdivides into several other subclasses as required by the underlying application or problem. For example, figure 1.4 shows several subclasses of greenery, i.e. grassland, forest, savannah, and scrub. Similarly, the following section discusses the classification of vegetation into several subclasses. In such cases of more specific subclasses within the class of vegetation, the classes are spectrally close and sometimes very close, making the distinction among them very difficult.

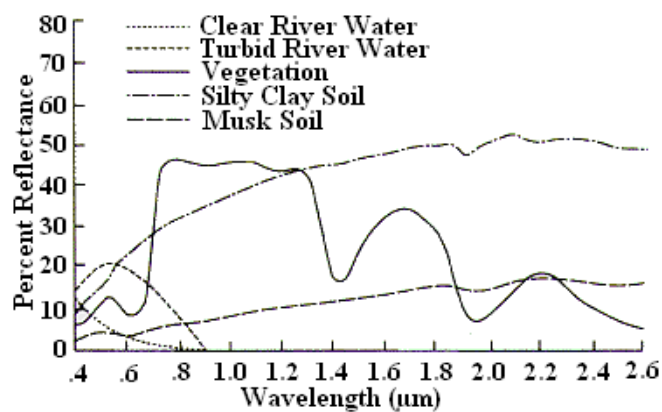


Figure 1.5: Spectral reflectance of vegetation, soil and water

## 1.6 Vegetation covers and their mapping

The vegetation covers have immense diversity. The vegetation covers further divide into the subclasses that subdivide further into smaller subclasses as per requirement and purpose of the classification. Mapping the vegetation-covers is useful in studying environmental degradation [Feng 2006], managing natural resources [Scott 2003], identification of fire-fuels in the forests for better fire fighting [Raymond 2006], and estimating crop yield and land use for agricultural management [Akbari 2006] besides many other applications. Therefore, States or regional governments maintain their own classifications and keeps statistics of the vegetation covers found in the region.

Table 1.3 lists the simplified classification of the vegetation in the Australian state of Victoria as given by the Australian department of sustainability and environment. It has as many as 300 vegetation classes with 20 major classes and 34 subclasses. Some of these classes appear spectrally

the same and discrimination among them becomes a challenging task. For example, [Cablk 2007] found that using satellite imagery could not spectrally distinguish riparian grass and the areas of dark soil from shrubs. Their mapping becomes impossible when these classes subdivide further into subclasses and the spectral difference among the classes becomes more and more subtle. Environment Canada finds as many as 23 subclasses of wetland in the region of St. Lawrence in Canada. Refer table 1.4.

Table 1.3: Simplified classification of vegetation in Victoria<sup>8</sup>

1	Coastal Scrubs Grasslands and Woodlands	11	Sub-alpine Grasslands, Shrublands or Woodlands
2	Heathy Woodlands	12	Plains Grasslands and Chenopod Shrublands
3	Lowland Forests	13	Plains Woodlands or Forests
4	Box Ironbark Forests or dry/lower fertility Woodlands	14	Riverine Grassy Woodlands or Forests
5	Lower Slopes or Hills Woodlands	15	Wetlands
6	Dry Forests	16	Heathlands
7	Wet or Damp Forests	17	Mallee
8	Riparian	18	Salt-tolerant and/or succulent Shrublands
9	Rainforests	19	Rocky Outcrop or Escarpment Scrubs
10	Montane Grasslands, Shrublands or Woodlands	20	Herb-rich Woodlands - Alluvial terraces and /or creek lines

## 1.7 Vegetation indices

In the process of photosynthesis, live vegetation absorbs part of the solar radiation in the frequencies called photo-synthetically active radiation (PAR) spectral region. The absorbed solar energy includes the visible light from wavelengths of 0.4 to 0.68 $\mu$ m. Refer figure 1.5. Leaf cells scatter, i.e., reflect and transmit solar radiation in the near-infrared spectral region, i.e. the light with the wavelengths from 0.68 to 1.3  $\mu$ m. Therefore, live vegetation has relatively low reflectance in the PAR and relatively high reflectance in the near-infrared region. However, the exact difference or ratio of the reflectance in the two regions, i.e. PAR and the non-PAR spectral regions, varies from one vegetation type to the other. This makes possible devising vegetation indices, a number generated by some combination of spectral reflectance of the target to remote sensing bands, that has some relationship to the amount and type of vegetation in a given image pixel. Using these indices helps identifying different vegetation types and estimating amount and condition of vegetation. However, it should be noted that these vegetation indices are generally based on empirical evidence and not on some analytical proof. Most of the indices used successfully for various applications are ratio based. Since these are ratios of the reflectance in the

<sup>8</sup> Source: <http://www.dpi.vic.gov.au/DSE/nrence.nsf/LinkView/>

bands of MS images, these do not have any dimension. Indices that are more complex are not important enough to be discussed here, since those have been almost abandoned.

Table 1.4: Subclasses of wetland in the region of St. Lawrence, Canada<sup>9</sup>

S. No.	Name of Species		
	English	French	Scientific
1	Shallow water dominated by submerged vegetation	Eau peu profonde à herbier submergé	-
2	Shallow water dominated by floating vegetation	Eau peu profonde à herbier flottant	-
3	Low marsh dominated by Common Three square	Bas marais à Scirpe d'Amérique	Schoenoplectus Pungens
4	Low marsh dominated by wild rice	Bas marais à zizanies	Zizania sp.
5	Low marsh dominated by arrowheads	Bas marais à sagittaires	Sagittaria sp.
6	Low marsh dominated by Great Bulrush	Bas marais à Scirpe des étangs	Scirpus Lacustris
7	Low marsh dominated by River Bulrush	Bas marais à Scirpe fluviatile	Bolboschoenus Fluviatilis
8	Low marsh dominated by Smooth Cord grass	Bas marais à Spartine alterniflore	Spartina Alterniflora
9	Low marsh dominated by Flowering Rush	Bas marais à Butome à ombelle	Butomus Umbellatus
10	Low marsh dominated by Pickerelweed	Bas marais à Pontéderie cordée	Pontederia Cordata
11	Low marsh dominated by Purple Loosestrife	Bas marais à Salicaire commune	Lythrum Salicaria
12	Low marsh dominated by cattails	Bas marais à typhas	Typha sp.
13	Low marsh dominated by Broad fruit Bur-reed	Bas marais à Rubanier à gros fruits	Sparganium Eurycarpum
14	Low marsh dominated by dense narrow-leaved vegetation (cattails and bulrushes)	Bas marais à végétation angustifoliée dense (typhas et scirpes)	Typha sp. Scirpus sp.
15	Low marsh dominated by River Bulrush and Purple Loosestrife	Bas marais à Scirpe fluviatile et à Salicaire commune	Bolboschoenus fluviatilis Lythrum salicaria
16	High marsh dominated by sedges	Haut marais à carex	Carex sp.
17	High marsh dominated by Common Reed	Haut marais à Phragmite commun	Phragmites Australis
18	High marsh dominated by Purple Loosestrife and sedges	Haut marais à Salicaire commune et à carex	Lythrum salicaria Carex sp.
19	High marsh dominated by Reed Canary grass	Haut marais à Phalaris roseau	Phalaris arundinacea
20	High marsh dominated by Blue joint	Haut marais à Calamagrostis du Canada	Calamagrostis canadensis
21	High marsh dominated by American Manna grass	Haut marais à Glycérie géante	Glyceria grandis
22	High marsh used for agriculture	Haut marais à affectation agricole	-
23	High salt marsh	Haut marais salé	-

<sup>9</sup> Source: [http://www.qc.ec.gc.ca/geo/mil/mil004\\_e.html](http://www.qc.ec.gc.ca/geo/mil/mil004_e.html)



Beginning from the simplest index called simple ratio, more complex indices have been proposed to consider other factors so that index becomes more robust undesired effects. These indices directly relate the sensor response to different biophysical parameters, such as percentage of vegetation cover, leaf area index, absorbed photo-synthetically active radiation, production rate of the biomass, etc.

### 1.7.1 Simple ratio (SR)

The simple ratio is the first ever and the simplest vegetation index introduced by [Birth 1968] to classify land-cover under observation as live green vegetation or otherwise. It is the ratio of red reflected radiant flux ( $\rho_{red}$ ) to the near-infra red flux ( $\rho_{nir}$ ). In (1.1), the numerator and the denominator represent the reflectance in red and infrared regions, respectively.

$$SR = \frac{\rho_{red}}{\rho_{nir}} \quad (1.1)$$

Similarly, inverse of this ratio called ratio vegetation index (RVI) is also used. Since this ratio is very simple, it is susceptible to many noise sources that render it limited in its use.

### 1.7.2 Normalized difference vegetation index (NDVI)

NDVI is a simple numerical indicator used to analyse remote sensing measurements, typically but not necessarily from a space platform. It is one of the most successful and well-known indices used to detect live green plant canopies in multi-spectral remote sensing data. This index is generally ascribed to [Rouse 1973], but [Kriegler 1969] first presented the concept of a normalised difference index. NDVI computes as in (1.2).

$$NDVI = \frac{\rho_{nir} - \rho_{red}}{\rho_{nir} + \rho_{red}}, \quad (1.2)$$

where  $\rho_{red}$  and  $\rho_{nir}$  stand for the spectral reflectance measurements acquired in the red and near-infrared regions, respectively. It is functionally equivalent and related to SR by the (1.3), i.e. a given SR value has a unique NDVI value and vice versa. While the SR ranges from 0 to infinity, NDVI has the advantage of varying between -1 and 1, where vegetated areas typically have values greater than zero, whereas values near or below zero show non-vegetated surfaces such as water, barren land, ice, snow, or clouds. Table 1.5 shows the typical NDVI values for various land-covers and its capacity to distinguish vegetated areas broadly from other surface types.

$$NDVI = \frac{1 - SR}{1 + SR} \quad (1.3)$$

The remote sensing experts have used NDVI and other similar indices in many diversified applications other than classification of the vegetation covers. For example, [Sellers 1985] and

[Myneni 1995] use NDVI to estimate the photosynthetic capacity of plant canopies, [Haboudane 2004] uses several vegetation indices including NDVI to estimate green leaf area index (LAI), [Ogawa 1999] uses NDVI to estimate evapotranspiration taking place in the vegetation, [Scott 2008] estimates biomass, [Jones 2007] uses NDVI to estimate chlorophyll concentration in spinach, [Pettorelli 2006] uses NDVI to estimate the plant productivity, and [Petja 2004] estimates accumulated rainfall from it.

Table 1.5: Typical NDVI values for various land-covers

S. No.	Land-cover type	Reflectance		NDVI value
		near-infrared	Red	
1	free standing water	very low	Low	negative ~-0.3
2	soils or rocks	somewhat larger than the red	somewhat smaller than the infrared	small positive from 0.1 to 0.2
3	live green vegetation	high	low	from 0.5 to ~1
4	snow and ice	low ~0.4	slightly lower than infrared	very slightly negative value

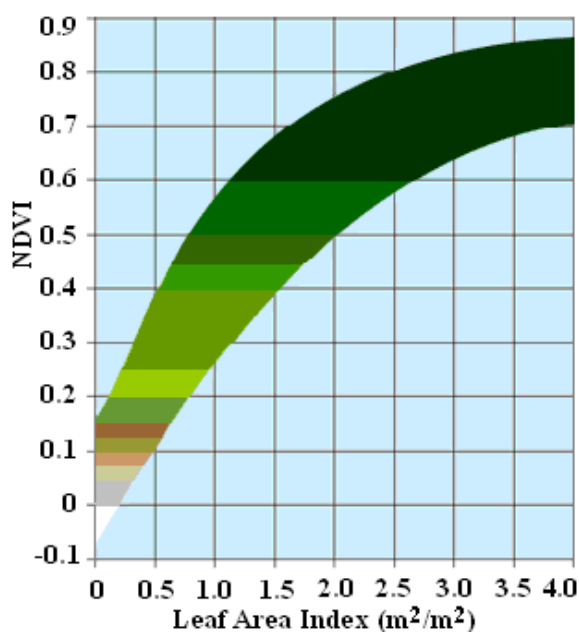


Figure 1.6: NDVI vs. LAI<sup>10</sup>

These diversified uses of NDVI point out that even for a given amount, type and/or condition of vegetation, NDVI changes depending on many other factors as well. For example, [Xiao 2003] presents the effects of atmospheric aerosols on the vegetation indices, [Adegoke 2002] studies the

<sup>10</sup> Source: <http://rangeview.arizona.edu/Tutorials/intro.asp>

effects of soil moisture, [Franke 2006] investigates the effects of sensor characteristics on NDVI, and [Trigg 2005] explores the effects of anisotropy, i.e. the effects of the viewing angle parameters on commonly used spectral indices. [Zoran 2006] presents findings on effects of climate on the performance of various vegetation indices. These perturbing factors can result in severe errors if not addressed properly before using the NDVI for quantitative assessments. Figure 1.6 shows how it varies with leaf area index (LAI). It illustrates that the NDVI has a non-linear response to the vegetation cover density and it saturates, as the vegetation gets denser.

### 1.7.3 Soil adjusted vegetation index (SAVI)

[Huete 1988] observes that the soil background of the vegetation affects the overall reflectance of the vegetation. Therefore, he proposes SAVI, expressed in (1.4), to compensate this undesired effect on the calculation of the vegetation index. The soil-adjusted index includes a new L factor, which represents height of the vegetation linearly and takes values from 0 representing very high vegetation to 1 representing very low vegetation. The most typical value is 0.5 that represents intermediate vegetation cover, but the RS expert chooses this value according to some a priori knowledge of the land-cover. As the multiplicative factor L tends to zero, the SAVI index approaches NDVI.

$$SAVI = \frac{(1+L)(\rho_{nir} - \rho_{red})}{\rho_{nir} + \rho_{red} + L} \quad (1.4)$$

The need for a priori knowledge of L leads to the circular problem of knowing the vegetation cover before calculating the vegetation index that one wants to use for knowing the vegetation cover. To overcome this problem, many researchers have proposed modified versions of SAVI, such as MSAVI, MSAVI2, etc.

### 1.7.4 Modified SAVI (MSAVI)

MSAVI is the Modified Soil Adjusted Vegetation Index introduced in [Qi 1994]. It proposes estimating a reasonable value of L from other indices that required the slope of soil line. The soil line is the hypothetical line [Kauth 1976] whose slope comes from the RS image itself. However, this process is also difficult, error prone, and highly sensitive to the climatic conditions. Therefore, another solution, [Qi 1994b] suggests MSAVI2 that does not need to know the slope of soil line or any other a priori knowledge to determine the value of L. He suggests substituting L in (1.4) with the value of SAVI itself that arrives at the recursive solution given in (1.5).

$$MSAVI2 = \rho_{nir} + 0.5 - 0.5\sqrt{(2\rho_{nir} + 1)^2 - 8(\rho_{nir} - \rho_{red})} \quad (1.5)$$

### **1.7.5 Transformed difference vegetation index (TDVI)**

[Bannari 2002] notes that unlike other perturbing factors that affect the vegetation cover classification, the problem of saturation and non-linearity is a weakness that results from the design and the analytical formulation of the vegetation indices. To overcome that weakness, [Bannari 2002] proposes TDVI, formulated in (1.6) and claims that it performs always better than NDVI, as good as SAVI and in some cases even better than SAVI. In addition to this, TDVI carries the advantages of being linear and non-saturating at a maximum limit. The TDVI was shown to have more linear relationship with percent cotton cover and Balsam Fir cover than NDVI and SAVI. However, the notion behind this particular algebraic formulation has never been justified theoretically.

$$TDVI = 1.5 \times \frac{(\rho_{nir} - \rho_{red})}{\sqrt{\rho_{nir}^2 + \rho_{red} + 0.5}} \quad (1.6)$$

Similarly, there are several other indices, not discussed here, that use trigonometric, logarithmic, or even complex formulations that have never been popular. There are also the indices that make use of the frequency bands other than red and near infrared. The following sections present two of those.

### **1.7.6 Normalized green-red difference index (NGRDI)**

This index was proposed by [Hunt 2005] for precision monitoring of crops using low altitude aerial photography with a digital camera. Since the red and green reflectance varies greatly with the exposure, [Hunt 2005] proposes the NGRDI, as in (1.7) to normalise the effects of different exposures and claims that the index relates to the crop biomass.

$$NGRDI = \frac{\rho_g - \rho_{red}}{\rho_g + \rho_{red}} \quad (1.7)$$

[Hunt 2005] finds this index to relate linearly with dry biomass at low amounts, and to reach a maximum value when biomass was greater than 120 g/m<sup>2</sup> for soybean and corn. This index does not directly relate to chlorophyll content of corn. For soybeans, alfalfa and corn, dry biomass from zero to 120 g/m<sup>2</sup> is linearly proportionate to the NGRDI, but for biomass greater than 120 g/m<sup>2</sup> in corn and soybean, the index does not increase further.

### **1.7.7 Atmospherically corrected vegetation indices**

The atmospherically resistant indices are a family of indices with built-in atmospheric corrections. The atmospheric correction is carried by replacing the red reflectance  $\rho_{red}$  with the term  $\rho_{rb}$  given by

$$\rho_{rb} = 2\rho_{red} - \rho_b \quad (1.8)$$

where  $\rho_b$  represents the blue reflectance. This was first proposed by [Kaufman 1992] for NDVI that resulted in atmospherically resistant vegetation index (ARVI). [Marshall 2004] finds ARVI four times less sensitive to the atmospheric effects than NDVI. Similarly, the same substitution in SAVI yields soil adjusted atmospherically resistant vegetation index (SARVI) and the same substitution in MSAVI2 yields atmosphere-soil-vegetation index (ASVI).

The vegetation indices driven from the multispectral data prove insufficient in many applications of remote sensing. This fact forced the remote sensing experts to use sources of information complementary to spectral data. Here comes the idea of fusing the structural information from SAR or textural information from the optical images with the spectral information.

## Part B: Image Fusion

In general, the image processing community describes image fusion as the process of producing a single image from two or more images from same or different sources/sensors. Broadly speaking, image fusion is a particular case of data fusion. [Ranchin 2000] defines image fusion as “combination of two or more images to create a new image containing more information by using a certain algorithm”. There are many other narrow-scope definitions of image fusion. However, in an attempt to summarize all those, I define it as:

**“the process of gathering all information from images and/or non-images of different nature, resolution, and source of origination, required for the application under consideration, into a single image or non-image representation”.**

According to [Phol 1998] and [Varshney 1997], image fusion can be performed at three different processing levels or at three levels of abstractions, i.e., *pixel level* at the lowest level of abstraction, *feature level*, and the *decision level* at the highest level of abstraction. Figure 1.7 illustrates this.

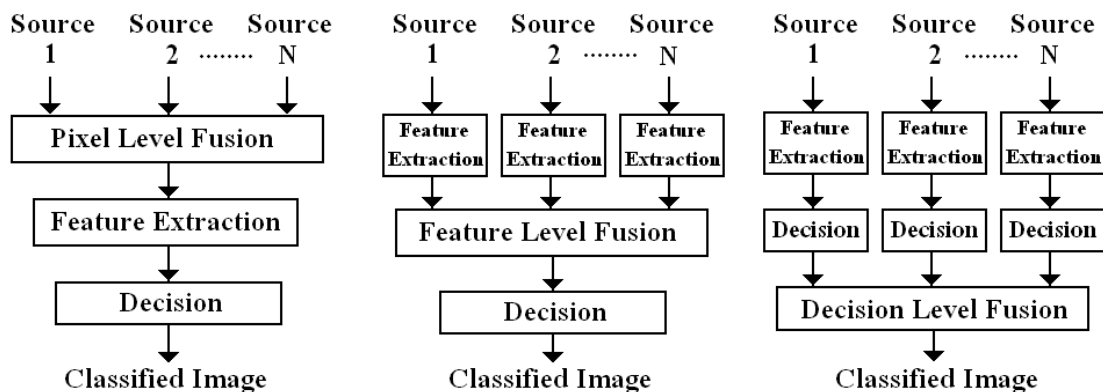


Figure 1.7: The fusion at three different levels of abstraction

Many others have asserted the same idea, but [Wald 1999] also argues that this categorization may be misleading and it may falsely imply that fusion processes do not deal simultaneously with these different levels. [Stubberud 2003] also points out that the boundaries between the three levels are unclear. So in some situations, it becomes controversial to decide the level of the fusion performed.

The fusion procedure at each level has its own methods, techniques, and pros and cons. The choice of a suitable fusion level depends on many different factors such as data sources, application and available tools. At the same time, the selection of the fusion level determines the necessary pre-processing involved. For instance, fusing data at pixel-level requires co-registered images at sub-pixel accuracy because the existing fusion methods are very sensitive to misregistration. Though so far the most researched domain is the data or pixel in the field of image fusion, one finds recent

examples of the research on all the three levels of fusion. The succeeding sections present a brief view of the image fusion at all the three levels of abstraction apart from multilevel fusion.

## 1.8 Pixel level fusion

This is the fusion at the lowest possible level of abstraction, in which the data from two different sources fuse directly. In image fusion, the data are the pixels of the images from different sources. Fusion at this level has the advantage that it uses the original data that is most possibly close to the reality. The images merge on the pixel-by-pixel basis, after the software co-registered them exactly at the same resolution level. Most of the times, the images are geo-coded as well before fusion since the fusion on pixel level requires accurate registration of the images to be merged. The accurate registration requires re-sampling and geometric correction. There are several methods of re-sampling and registration of the images. The geometric correction requires the knowledge of the sensor viewing parameters along with software that takes into account the image acquisition geometry and Ground Control Points (GCPs). The GCPs are the landscape features whose exact locations on the ground are known. The GCPs may be naturally occurring, e.g. road intersections and costal features; or may be intentionally introduced for the purpose of geometric corrections. In some cases, where the surfaces are highly uneven, a DEM is required. This is especially important for SAR data processing, whose sensor has the side-looking geometry, i.e. oblique view. The oblique radar waves strike a bump on the rough terrain instead of the targeted location on the surface. The image fusion at this level has the highest requirements of computer memory and processing power and it takes longer processing times. Figure 1.8 depicts a block diagram showing sequence of actions taking place in a system employing pixel-level-fusion.

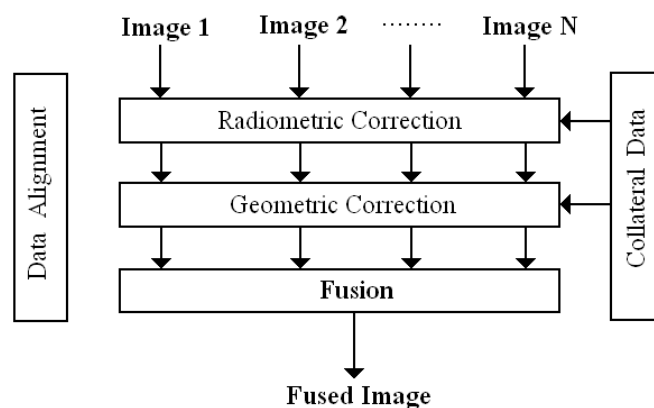


Figure 1.8: The pixel level fusion

The collateral data may include the sensor characteristics and imaging parameters required for radiometric correction, as well as the information regarding GCPs required for the geometric correction. The term data alignment refers to the registration of the images. [Geerling 2007] fuses spectral data from CASI sensors and lidar data for classification of floodplain vegetation. Apart

from remote sensing, pixel level image fusion finds its applications in defence [Xue 2003], computer vision [Goshtasby 2005], and medical imaging. In case of medical image fusion, the two source images might be coming from ultrasound and computer tomography scan [Wein 2005], or from x-rays and magnetic resonance imaging that often provide complimentary information [Qu 2001]. The experts in remote sensing fuse the images carrying complementary information for more accurate classification, better interpretability, and/or to develop digital elevation models (DEM) [Simone 2002]. However, the simplest and the most frequent example of RS image fusion is the fusion of multi-spectral (MS) and panchromatic (PAN) sensors from the aboard the same satellite. Several works, for example, [Zhang 2002] and [Tian 2004] propose and assess the techniques for fusing high-resolution PAN images with the low-resolution MS images. SAR and optical images also provide complimentary information, and sometimes fuse on the pixel level for enhanced interpretability [Westra 2004].

## **1.9 Feature level fusion**

This approach merges the datasets, i.e. images at an intermediate level of abstraction. It is suitable to opt for feature-level fusion only if the features extracted from various data sources, i.e. images can properly be associated with each other, for example the features like edges and segments can be extracted from both, the optical as well as SAR images and then can be merged together to work out joint features and classification. SAR images provide textural information that is complementary to the spectral information from the optical images. Therefore, texture features extracted from SAR images and spectral features extracted from MS images may fuse before a classifier classifies them. [Greiwe 2005] fuses hyper-spectral image with high-resolution image on the feature level. Some works propose fusing different kinds of features extracted from the same image before classifying the image. For example, [Zhao 2005] fuses texture features for classification of very high-resolution RS images and [Clausi 2001] fuses different texture features extracted from SAR images.

## **1.10 Decision level fusion**

The decision level fusion takes place after classification of the two or more sets of data. The different sets of data might come from the same source image using different features or from different source images. In case of different source, the two or more different source images are classified using the same or different classifiers and the resulting classifications are then fused to obtain a single classification that is often more accurate than any one of the two fused classifications alone. Sometimes, the same dataset is classified using two, or more, different classifiers and then the classifications are fused. There are several methods of classifier fusion, including those based on majority vote, Bayesian average and the maximum posterior probability as noted in [Bruzzone 2002]. Fusion at decision level results in concluding the results that were



impossible from the individual classifications. For example, fusing individual classification of a CT scan and an MRI may help in better diagnosis. The fused classification often helps have better interpretation of the input data, i.e. an observed scene in remote sensing.

### 1.11 Multilevel Fusion

It is not necessary to perform fusion at only one of the three levels. The fusion may take place at any two or all the three levels and there exist example of techniques that allow fusion of image and non-image data at multiple levels of inference [Hall 2003]. [Borghys 1998] applies multi-level fusion to multi-spectral image sequences for target detection. [Petrovic 2003] proposes the multi-level image fusion framework to perform image fusion at all the three levels and reports significantly better results with the image fusion simultaneously performed at the first two levels (i.e. pixel and feature level) than with the fusion performed at any one level alone. Figure 1.9 illustrates the multilevel fusion framework proposed by [Petrovic 2003]. However, multilevel fusion may take in several forms such as the one in the succeeding section.

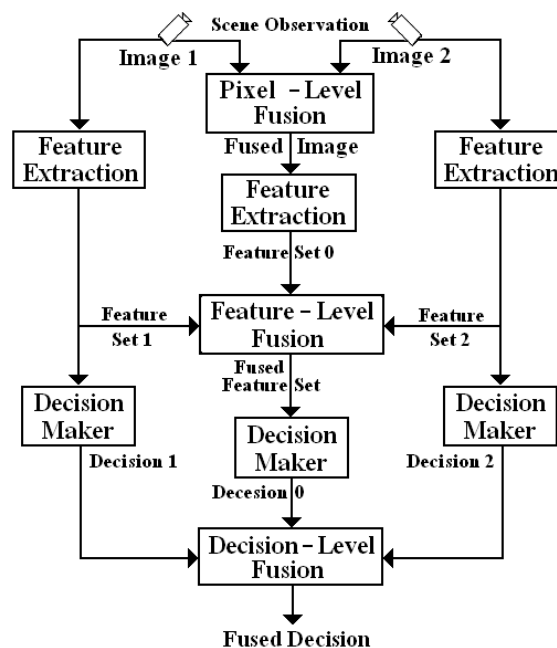


Figure 1.9: The multilevel fusion

### 1.12 Region based fusion

Some researchers proposed the fusion schemes that one cannot explain to be working on any of the three levels of abstraction. For example, [Zhang 1997] suggests what they call region-based fusion. It is in fact another scheme of multilevel fusion. One can also think of it as lying between the abstraction level of pixel and that of the features. [Piella 2003] is a more recent but similar work that proposes the pixel level fusion guided by feature analysis as depicted in figure 1.10.

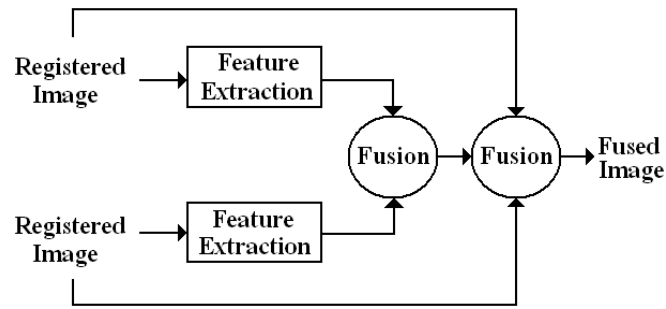


Figure 1.10: The region-based fusion

### 1.13 Conclusion

Traditionally, remote sensing of environment has depended highly on the vegetation indices driven by multi-spectral data from remote sensing satellites. However, all the vegetation indices designed for different applications remain susceptible to various climatic conditions and they remain vulnerable even to the change of data source, i.e. the sensors. The remote sensing experts often regard Multi-spectral data on a pixel as insufficient for land-cover classification, and emphasise on using the spatial context in which some spectral pixel value occurs; in other words, the texture. On the other hand, the RS satellites now provide the RS imagery that is both, affordable and high-resolution, making the identification of land-covers from remote sensing images easier than ever before. In the high-resolution imagery, every region covering a single land-cover type now comprises several image pixels, providing fine details and thereby making it possible to use texture information for identification of the land-covers apposed to the use of spectral information alone in the past. Therefore, the next chapter is dedicated to texture feature extraction and comparison of different texture features.

The image fusion techniques have been used to bring the resolution of the multispectral imagery at par with that of the panchromatic images. This utilises the image fusion at the lowest level of abstraction. Apart from this, the complimentary information provided by the radar and the optical images has traditionally encouraged the use of image fusion techniques in the remote sensing applications. Using the techniques of image fusion at higher levels of abstraction helps accomplish such tasks. Therefore, remote sensing applications often use image fusion techniques at one or more levels of abstraction. The land-cover mapping scheme presented in chapter 4 of this thesis also uses image fusion at two levels of abstraction.

# 2

## Texture features for classification of land-covers

Part A: Theoretical concepts .....	30
2.1 Image texture.....	30
2.2 Texture description.....	30
2.2.1 Ideal texture description.....	31
2.2.2 Practical texture description.....	31
2.2.3 Texture features based on Gabor filters .....	32
2.2.4 DFT-based texture signatures.....	34
2.2.4.1 Extraction of LFH based features .....	34
2.3 Modifications to LFH-based features .....	36
2.3.1 Circular neighbourhood .....	36
2.3.2 The phase features.....	37
2.3.2.1 Local image gradient .....	38
2.3.2.2 Rotation invariance of the phase features .....	40
2.3.3 Smaller neighbourhoods.....	41
2.3.4 Quantisation .....	42
2.4 Computational cost analysis.....	42
Part B: Experimental setup.....	44
2.5 Texture datasets.....	44
2.5.1 The training set .....	44
2.5.2 The test image sets .....	46
2.5.2.1 Test images without noise .....	46
2.5.2.1.1 Full set of test images.....	46
2.5.2.1.2 Subset of test images.....	47
2.5.2.2 The test images with Gaussian white noise .....	47
2.6 Performance evaluation.....	48
2.6.1 Recognition.....	49
2.6.2 Rotation invariant texture features .....	49
2.6.3 Retrieval.....	49
2.7 Results .....	50
2.7.1 The test images without noise .....	50
2.7.1.1 Recognition .....	50
2.7.1.1.1 Full set of test images.....	50
2.7.1.1.2 Subset of test images .....	52
2.7.1.2 Retrieval .....	53
2.7.1.2.1 Full set of test images.....	53
2.7.1.2.2 Subset of test images.....	55
2.7.2 The test images with Gaussian white noise.....	56
2.7.2.1 Recognition .....	57
2.7.2.2 Retrieval .....	57
2.8 Conclusion.....	58

Conventionally, the optical remote sensing is all about the extraction of spectral information on the land-cover and sometimes even precise information regarding objects on the surface of Earth, from multispectral remote sensing images. However, the constraint of such a classification lies in the fact that the reflectance of the pixel is not a function of only the part of land it corresponds to; instead, the pixel reflectance is a function of neighbouring regions also. It means that the spatial context of occurrence of a given spectral response is also important in the land-cover classification. This fact is even more important after the emergence of the high spatial resolution imagery, where a pixel represents a piece of land no bigger than 70cm×70cm. Therefore, it is logical to think of using

contextual information as well, i.e. the neighbourhood in which a particular reflectance occurs. The image texture is best at representing this contextual information. Nevertheless, extraction of texture characteristics from the remote sensing images has never been an easy task. [Schaale 2000] makes the same observations, noting that the standard practice of using only spectral information for the land-cover classification is because the extraction of texture features from land-covers is a challenging task. With the advent of very high-resolution satellite imagery, the remote sensing experts are now utilising texture information from the RS images to extract the required information.

This chapter divides into two parts. Part A discusses theoretical notions of texture and methods of extracting texture features. It also suggests improvements in a method of extracting the texture-features. Part B compares the performance of various texture features for image retrieval and recognition on noiseless and noisy images alike. Before presenting the results, it explains all the experimental conditions including training image set, test image sets, methods of performance evaluation. Part B also introduces a new performance parameter, called rotation variance, for the texture features.

## **Part A: Theoretical concepts**

### **2.1 Image texture**

Although one can find many definitions of texture in the contemporary literature, it has no definition that is uncontroversial and agreed upon unanimously by the image processing research community. Researchers have devised several definitions of texture comprising from a single sentence to several sentences. Nonetheless, one can describe texture as a pattern of gray level changes in an image or a local region therein. The image processing research community considers this property of image, called texture, an important image characteristic and an essential part of the content-based image retrieval (CBIR) systems and remote sensing alike.

Texture processing, including texture segmentation, recognition, classification, synthesis, and shape from texture, is now already a mature field of research. The texture segmentation and classification is performed on medical images for diagnostics, on natural images for computer vision, and on remote sensing images for land-cover classification besides several other applications.

### **2.2 Texture description**

Many a times, understanding images requires segmenting and classifying textures found therein and that in turn requires efficient texture descriptors. Following subsections explain the salient

characteristic features of an efficient texture description and the possible methods of efficient texture description.

### **2.2.1 Ideal texture description**

An ideal texture descriptor is one that represents image textures with the attributes invariant to image rotation, translation, scale change, gray-level change, and noise. Practically, the texture description requires invariance to one or more of these factors depending on the application. Classically, in remote sensing, the texture classification has been performed on SAR images, often to supplement the land-cover classification using spectral information from optical images. SAR images come with severe levels of noise. Therefore, working with SAR images requires highly noise-insensitive texture description. Recently the development of the very-high-resolution optical sensor, the optical RS images are source of both, the, spectral as well as the textural information. Luckily, noise is not a very serious issue in the optical remote sensing images. However, the optical RS images also suffer from the Gaussian additive noise. Therefore, the most important aspects of the texture description of the RS images remain that of rotation invariance, grey level insensitivity and noise insensitivity in the order of importance.

### **2.2.2 Practical texture description**

There are several kinds of texture descriptors, which are generally [Chen 1998, Tuceryan 1998] categorised as based on:

- Signal processing methods
- Statistical methods
- Model-based methods
- Geometrical methods

The first approach considers the spatial frequencies involved in the textures and analyses the textures in the spatial frequency domain using Gabor filters, discrete wavelet transforms, curvelet transforms and ridgelet transforms. These methods are very popular in texture analysis for computer vision, remote sensing [Schaale 2000] and other applications.

The second category includes the texture features that come from histogram-like representations like gray level co-occurrence matrix (GLCM), local binary patterns (LBP), and the local Fourier histograms (LFH). These features are known for their power of describing the textures with local neighbourhood properties.

The third category includes Markov random fields (MRF) and fractals [Mandelbrot 1982]. The methods based on MRF consider that each pixel depends statistically on the rest of the image. [Cicala 2004] is an example of using MRFs for texture analysis of a remote sensing image. These methods have not been very popular in the remote sensing applications. The methods based on

fractals have been used for compressing remote sensing images [Sankaragomathi 2007] and for the remote sensing of geological structures [Gloaguen 2007]. However, there does not appear any significant work on the use of texture attributes based on fractals for the classification of land-covers in a remote sensing image.

The two methods categorised in the model-based approaches can be recategorised. Those based on MRF can be considered as belonging to the category of statistical methods, because MRF is a statistical entity. Furthermore, methods based on fractals can be categorised as belonging to the geometrical methods, since these texture features are based on the principles of fractal geometry, despite the fact that it is non-Euclidean geometry.

The fourth category includes the texture descriptors based on geometric features. This type of texture description finds applications in texture synthesis and texture mapping for 3D rendering of the surfaces. Refer [Stamos 2002] and [Elber 2005]. These techniques have never been popular in the texture recognition in general and description of land-covers classification in particular.

The most popular methods of texture analysis belong to the first two categories. Methods from the first category are usually computationally expensive [Verma 2003], since these come from large filter banks, but these are moderately insensitive to noise [Ursani 2007]. On the other hand, those from the second category are computationally less expensive and have comparatively much higher descriptive power than the former, but are highly sensitive to the image noise [Ursani 2007]. [Verma 2003] observes that the texture features based on statistical methods and extracted from neighbourhoods as small as  $3 \times 3$  pixels can yield as good as or even better results than the features extracted from large filter banks, since texture information is contained in the local characteristics of the immediate pixel neighbourhood. Therefore, following sections present the texture descriptors from these two categories and part B of the chapter compares them on a large datasets of noiseless and noisy images.

### **2.2.3 Texture features based on Gabor filters**

The studies on human vision system analyses the perceived image in the spatial frequency domain. For this reason, the Gabor filters are popular among the image processing and computer vision experts who believe that the Gabor filters are best at replicating human vision system [Lee 1996]. Although there are several works reporting the use of texture descriptors based on Gabor filters, we present experimental results with the descriptor first presented in [Manjunath 1996], again in [Zhang 2000] along with a method of rotation invariant similarity measurement and compared in [Toyoda 2005]. These descriptors are designed quite carefully to avoid filter outputs carrying redundant information. Equations (2.2) through (2.10) are the expressions as they appear in [Manjunath 1996] and [Zhang 2000].

$$G_{mn}(x, y) = \sum_{s=-S/2}^{S/2} \sum_{t=-T/2}^{T/2} I(x-s, y-t) \psi_{m,n}^*(s, t, W), \quad (2.1)$$

where S+1 and T+1 specify the filter mask size, and \* represents operation of complex conjugate.

$$\psi(x, y, W) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right] \cdot \exp(j2\pi Wx), \quad (2.2)$$

where  $\psi$  is the mother Gabor wavelet,  $\sigma_x$  and  $\sigma_y$  are the functions of scale (m), described in (2.8) and (2.9), and  $W$  is the modulation frequency described in (2.7). The child Gabor wavelets are defined as

$$\psi_{m,n}(x, y, W_m) = a^{-m} \psi(\tilde{x}, \tilde{y}, W_m), \quad (2.3)$$

where  $\tilde{x}$  and  $\tilde{y}$  are given as

$$\tilde{x} = a^{-m} \left( \left(x - \frac{S}{2}\right) \cos \theta_n + \left(y - \frac{T}{2}\right) \sin \theta_n \right) \text{ and} \quad (2.4)$$

$$\tilde{y} = a^{-m} \left( \left(x - \frac{S}{2}\right) \sin \theta_n + \left(y - \frac{T}{2}\right) \cos \theta_n \right), \quad (2.5)$$

respectively, where  $\theta_n = n\pi/N$ , n specifies the orientation of the wavelet, N is the total number of orientations,  $a > 1$  and is given by

$$a = (U_h / U_l)^{\frac{1}{M-1}}, \quad (2.6)$$

where M is the number of scales,  $U_l$  and  $U_h$  are the lower and upper centre frequencies, whose values are proposed in [Manjunath 1996] and [Zhang 2000] as 0.05 and 0.4 respectively. The modulation frequency  $W_m$  is given as

$$W_m = a^m U_l. \quad (2.7)$$

The  $\sigma_x$  and  $\sigma_y$  of the children wavelets are given by

$$\sigma_{x,m} = \frac{(a+1)\sqrt{2\ln 2}}{2\pi a^m (a-1)U_l}, \quad (2.8)$$

$$\sigma_{y,m} = \frac{1}{2\pi \tan\left(\frac{\pi}{2N}\right) \sqrt{\frac{U_h^2}{2\ln 2} - \left(\frac{1}{2\pi\sigma_{x,m}}\right)^2}}. \quad (2.9)$$

The mean ( $\mu_{mn}$ ) and standard deviation ( $\sigma_{mn}$ ) of the magnitude of the filter outputs ( $G_{mn}$ ) from (2.1) are used as texture features. [Zhang 2000] proposed five scales ( $M=5$ ) and six orientations ( $N=6$ ), hence 30 filters in all. Therefore, the resulting texture descriptor has 60 values in all, as shown in (2.10).

$$F_{gabor} = \{\mu_{00}, \sigma_{00}, \mu_{01}, \sigma_{01}, \dots, \mu_{45}, \sigma_{45}\} \quad (2.10)$$

The rotation invariance is achieved by circularly shifting the feature vector so that the features (mean and variance) from the filter giving highest output energy are placed first [Zhang 2000]. Since our test images are only  $128 \times 128$  in size against  $200 \times 200$  in [Zhang 2000], we used the filter size of  $31 \times 31$  instead of  $61 \times 61$  proposed in [Manjunath 1996] and [Zhang 2000]. There are two reasons of using smaller filters. First, the larger filters cause computational load, and other that larger filter causes larger regions of no information near the image edges. The filter of  $61 \times 61$  pixels would be too large for processing an image of  $128 \times 128$  pixels.

Keeping in view the closeness of Gabor filters with the human vision system, computer vision community has extensively used Gabor filters for the extraction of texture features, but [Cutter 2003] observes that the human vision is deceptive for remote sensing applications. This is also true in the specific remote sensing application dealt here in this thesis, which processes a panchromatic image. This panchromatic image is a set of grey values corresponding to the response from a sensor having a very wide band that does not cover a part of frequencies visible to human vision system and covers a large part of infra-red frequencies invisible to human vision system. See figure 1.2 in chapter 1. Multispectral analysis in remote sensing also frequently uses the NIR band that is invisible to humans and at the same times is vital to the spectral discrimination of the land-covers.

## 2.2.4 DFT-based texture signatures

[Zhou 2001] proposed extracting texture features using discrete Fourier transform (DFT) in one dimension. Coefficients of the DFT of the 8-pixel sequence around each image pixel provide the texture signatures that are useful in texture segmentation, texture recognition, and retrieval. These coefficients are histogrammed, i.e. binned to compute local Fourier histogram (LFH). [Cutter 2003] notes that the LFH features are an accurate description of the local spatial distribution of the pixel grey-levels, and uses features extracted from local DFT from  $3 \times 3$ -pixel neighbourhoods for remote sensing application. In [Ursani 2007], we found these features more efficient and computationally less expensive than the features based on Gabor filters.

### 2.2.4.1 Extraction of LFH based features

Moving  $3 \times 3$ -pixel window across a texture image, 1D DFT of the digital numbers (pixel values) of sequence of 8 pixels,  $P_0$  through  $P_7$ , in the spatial domain, shown in figure 2.1 is computed as



$$X_k = \sum_{n=0}^7 x_n e^{-\frac{\pi i}{4}kn}, \quad (2.11)$$

where  $0 \leq k \leq 7$ ,  $X_k$  represents the  $k^{\text{th}}$  Fourier coefficient,  $x_0$  through  $x_7$  correspond to digital numbers of pixels P0 through P7. The symbol  $x$ , hereafter, represents the sequence of eight pixel values  $x_0$  through  $x_7$ .

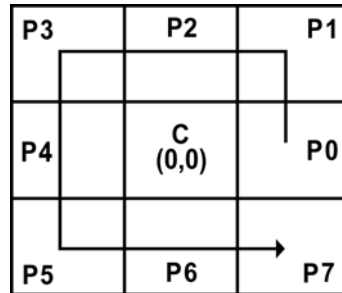


Figure 2.1: The 9-pixel neighbourhood in the spatial domain

From the computed DFT, the absolute values of the first five coefficients, i.e.  $|X_0|$  through  $|X_4|$  are used for the texture description. Altogether, the computed coefficients are normalised to take values from 0 to 255. After normalisation,  $|X_0|$  is linearly quantised into 8 bins and  $|X_1|$  through  $|X_4|$  are linearly quantized into 16 bins. For describing the textures, all the eight bins of  $|X_0|$  are made part of the resulting texture models, but only first eight bins of the remaining coefficients are used as features. This is because all the bins following bin 8 remain zero in all the experiments undertaken herein, which is evident from the histograms shown in figure 2.2. It can be seen that the histograms of the coefficients other than  $X_0$  do not go beyond 127 on the x-axis. In this way, we have  $8 \times 5$ , i.e. 40 features in the texture descriptor, hereafter referred to as LFH 40.

In (2.11), replacing  $k$  with 0, 1, 2, 3, and 4 yields the DFT coefficient  $X_0, X_1, X_2, X_3,$  and  $X_4$ , respectively.  $X_0$  and  $X_4$  are real, whereas others are complex. It is easy to show that the eight templates of table 2.1 yield the real and imaginary values of these DFT coefficients if convolved with the 9-pixel neighbourhood of figure 2.1.

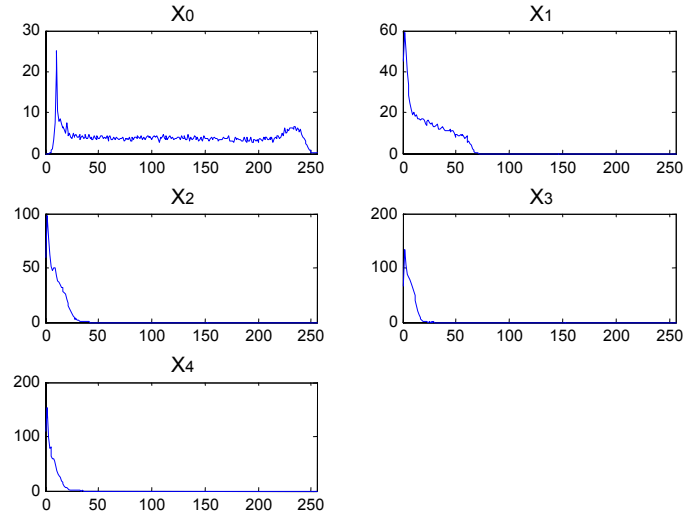

 Figure 2.2: Histograms of the coefficients  $X_0$  through  $X_4$  for image D2

Table 2.1: The templates to extract the DFT-based texture signatures

$\begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}$ $X_0$	$\begin{bmatrix} -1 & 1 & -1 \\ 1 & 0 & 1 \\ -1 & 1 & -1 \end{bmatrix}$ $X_4$	$\begin{bmatrix} -1 & 0 & 1 \\ -\sqrt{2} & 0 & \sqrt{2} \\ -1 & 0 & 1 \end{bmatrix} \times \frac{1}{\sqrt{2}}$ $X_1 \text{ (real)}$	$\begin{bmatrix} -1 & -\sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix} \times \frac{1}{\sqrt{2}}$ $X_1 \text{ (imaginary)}$
$\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$ $X_2 \text{ (real)}$	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$ $X_2 \text{ (imaginary)}$	$\begin{bmatrix} 1 & 0 & -1 \\ -\sqrt{2} & 0 & \sqrt{2} \\ 1 & 0 & -1 \end{bmatrix} \times \frac{1}{\sqrt{2}}$ $X_3 \text{ (real)}$	$\begin{bmatrix} -1 & \sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & -\sqrt{2} & 1 \end{bmatrix} \times \frac{1}{\sqrt{2}}$ $X_3 \text{ (imaginary)}$

Resulting from the unique combination of the eight pixels values, the DFT texture signatures provide 8<sup>th</sup> order statistics, against only 2<sup>nd</sup> order statistics provided by grey level co-occurrence matrices. There are ways to improve the LFH features further as described in the following sections.

## 2.3 Modifications to LFH-based features

There are several possibilities of variations in the LFH-based texture features introduced in [Zhou 2001]. For example, using circular neighbourhood [Arof 1998] instead of the rectangular one, also using phase features instead of the amplitudes only [Ursani 2008b], and using smaller neighbourhoods as well [Ursani 2008a], and using prior image quantisation. Following sections present these possibilities in detail.

### 2.3.1 Circular neighbourhood

The magnitudes of the DFT coefficients are also susceptible to changes if the image is rotated, since the pixel values in the 9-pixel neighbourhood get altered in the rotated image. Therefore, the magnitudes of the DFT coefficients are not fully rotation insensitive.

[Arof 1998] suggested a circular neighbourhood instead of the rectangular for extraction of similar texture features based on 1D DFT. It argues that because rotating the image alters the pixel values in the rectangular 9-pixel neighbourhood, and therefore, the features extracted from the rotated and non-rotated image are dissimilar. On the other hand, the pixel values in the corresponding 9-pixel circular neighbourhood remain unaltered in the wake of rotation.

Figure 2.3 shows the circular neighbourhood with the neighbours 1, 3, 5 and 7 (shown using empty circles) having non-integer pixel coordinates. [Arof 1998] interpolates these neighbours of non-integer coordinates using bilinear interpolation. For example, he interpolates neighbour  $x_1$  (having the coordinates  $X=0.707$  and  $Y=0.707$ ) between the four closest neighbours of integer coordinates, i.e. P0 (1, 0), P1 (1, 1), P2 (0, -1), and C (0, 0) of the rectangular neighbourhood. In addition, [Arof 1998] suggests using mean and standard deviation of the feature images of the DFT coefficients  $X_0$  through  $X_4$  instead of computing their histograms and using bins values as features as in [Zhou 2001].

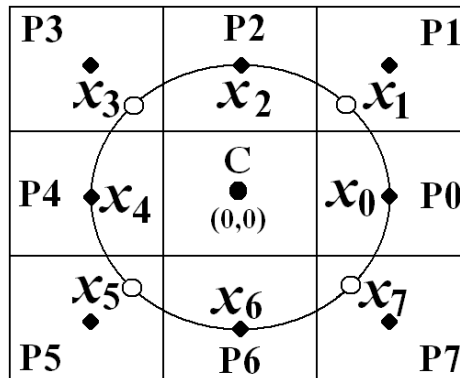


Figure 2.3: The circular neighbourhood as suggested in [Arof 1998]

Despite being computationally more expensive, the features extracted from the interpolated circular neighbourhood prove less efficient than those from rectangular neighbourhood. The reason of this failure is that there is a pitfall in the hypothesis of the circular neighbourhood itself. The interpolated neighbours also come from the same pixel values of the 9-pixel neighbourhood that is unstable to the rotation.

### 2.3.2 The phase features

The phases of DFT coefficients  $X_1$  through  $X_3$  were also proposed as features in [Zhou 2001] but only for those applications that do not deal with image-rotation. The phase features were otherwise excluded because, unlike amplitudes, the phases of the DFT coefficients are sensitive to image rotation.

Consider that an image rotates by an arbitrary angle, with the centre of rotation exactly in the middle of the image. The angle of rotation at any other point  $P_{xy}$  on the image would be different from what it is at the centre of rotation. Let the angle of rotation be  $\psi$  at point  $C_{00}$  (see figure 2.1),

corresponding to a shift in string  $x$  by  $m$  places. This shift in  $x$  causes nothing but the changes in the phases of the resulting DFT coefficients. Equation (2.12) states the shift property of DFT.

$$F(\{x_{n-m}\})_k = F(\{x_n\})_k e^{-\frac{\pi i}{4} km}, \quad (2.12)$$

where  $F(\{x_n\})_k$  represents the  $k^{\text{th}}$  coefficient of the DFT of  $\{x_n\}$ , and  $F(\{x_{n-m}\})_k$  represents the  $k^{\text{th}}$  coefficient of the DFT of the string  $\{x_{n-m}\}$  that is the same string  $\{x_n\}$  shifted by  $m$  places. Equation (2.13) shows that any displacement in time- or space-domain causes a phase shift in the coefficient  $X_k$  given by

$$\Delta\theta_k = -\frac{\pi}{4} km \quad (2.13)$$

in the Fourier domain. Replacing  $k$  with 1 in (2.13) gives the phase shift in  $\angle X_1$  as

$$\Delta\theta_1 = -\frac{\pi}{4} m = \psi. \quad (2.14)$$

Intuitively, the change in the local image-gradient angle ( $\delta$ ) is equal to the angle of rotation at point  $C_{00}$  ( $\psi$ ) that causes equal change in  $\angle X_1$ . Comparing (2.13) and (2.14) gives the phase shift in  $\angle X_k$  as

$$\Delta\theta_k = k \times \Delta\theta_1. \quad (2.15)$$

Therefore in [Ursani 2008b], we proposed using the histograms of the phases of  $X_2$  and  $X_3$  as well, but after compensating with the angle of local image gradient. We suggested adjusting phases  $\angle X_2$  and  $\angle X_3$  accordingly against the rotation by subtracting the angle of local image gradient ( $\delta$ ) as in (2.15). For  $k \in \{2, 3\}$

$$\phi_k = \angle X_k - k\delta, \quad (2.16)$$

where  $\phi_k$  represents the rotation compensated phase  $\angle X_k$ , and  $\delta$  represents the angle of local image gradient.

Further, in [Ursani 2008b] we showed that the phase of DFT coefficient  $X_1$ , i.e. is itself the local image gradient. Following section gives the mathematical evidence.

### 2.3.2.1 Local image gradient

Traditionally as a good compromise between cost and accuracy, the  $3 \times 3$ -pixel edge detection operators such as Sobel operator and Prewitt operators are often used to estimate local image gradient at a given pixel. Below are the general  $3 \times 3$  edge detection operators, in which the value of the parameter  $b$  varies from 1 as in Prewitt operator to 2 as in Sobel operator.

$$S_X = \begin{bmatrix} -1 & 0 & 1 \\ -b & 0 & b \\ -1 & 0 & 1 \end{bmatrix} \quad S_Y = \begin{bmatrix} -1 & -b & -1 \\ 0 & 0 & 0 \\ 1 & b & 1 \end{bmatrix} \quad (2.17)$$

$S_X$  and  $S_Y$  are convolved with a texture image to obtain two gradient images  $G_X$  and  $G_Y$  respectively. The local image gradient angle ( $\delta$ ) is calculated as

$$\delta = \tan^{-1}\left(\frac{G_Y}{G_X}\right). \quad (2.18)$$

Convolving the edge detection operators of (2.17) with the 3×3-pixel neighbourhood of figure 2.1 gives  $G_Y$  and  $G_X$  that are substituted in (2.18) giving

$$\tan \delta = \frac{-x_1 - bx_2 - x_3 + x_5 + bx_6 + x_7}{bx_0 + x_1 - x_3 - bx_4 - x_5 + x_7}. \quad (2.19)$$

By substituting  $k=1$  in (2.11), it can be shown for  $x$  that

$$\tan \angle X_1 = \frac{-x_1 - \sqrt{2}x_2 - x_3 + x_5 + \sqrt{2}x_6 + x_7}{\sqrt{2}x_0 + x_1 - x_3 - \sqrt{2}x_4 - x_5 + x_7}. \quad (2.20)$$

Expressions (2.19) and (2.20) happen to be exactly the same if  $b = \sqrt{2}$  and are very close otherwise, because the value  $\sqrt{2}$  falls between usual values, i.e. 1 as used in Prewitt operator and 2 as used in Sobel operator. To demonstrate this fact, histograms of the local image gradient angle were computed from  $\angle X_1$  and from Sobel operator ( $b=2$ ) for image D87 of Brodatz album [Brodatz 1966]. The two have a cross-correlation-coefficient (XCC) of 0.97 and a signal-to-noise ratio (SNR) of 69 dB verifying that the former is a very close approximation of the latter. All other images of the album were tested, and more or less similar values of SNR were found. Hence, instead of computing local image-gradient angle using any 2D edge detection operators, it suffices to use the value  $\angle X_1$  to compensate the phases of the two other DFT coefficients, i.e.  $\angle X_2$  and  $\angle X_3$  against the effects of image rotation. It can now be said that

$$\delta = \angle X_1. \quad (2.21)$$

Therefore for  $k \in \{2, 3\}$ , (2.16) resolves as

$$\phi_k = \angle X_k - k\angle X_1, \quad (2.22)$$

where  $\phi_k$  represents the rotation compensated phase  $\angle X_k$ , and  $\angle X_1$  replaces  $\delta$ . These phases are histogrammed into 12 bins each, yielding 24 values in total. Hence, there are 40+24, i.e. 64 values in the feature-set hereafter referred to as LFH 64. Before using these gradient-compensated phases as texture features, their invariance to rotation is subject to experimental verification.

### 2.3.2.2 Rotation invariance of the phase features

The images from Brodatz album were rotated to 30°, 45°, 60° and 90°, and histograms of  $\phi_2$  and  $\phi_3$  were computed at each orientation. Table 2.2 shows the XCC as a similarity measure between the histograms corresponding to 0° and those to 30°, 45°, 60°, 90° averaged over all the images from Brodatz album. As an example, figure 2.4 shows the histograms of  $\phi_2$  and  $\phi_3$ , for the image D87 from Brodatz album. All the histograms appear the same and do not exhibit any left- or right-shift, indicating that the two phases are highly rotation-invariant.

Table 2.2: XCC between the histograms of  $\phi_2$  and  $\phi_3$  corresponding to 0° and to 30°, 45°, 60°, 90° averaged over all the images from Brodatz album

	30°	45°	60°	90°
$\phi_2$	0.9974	0.9976	0.9954	0.9989
$\phi_3$	0.9953	0.9834	0.9855	0.9709

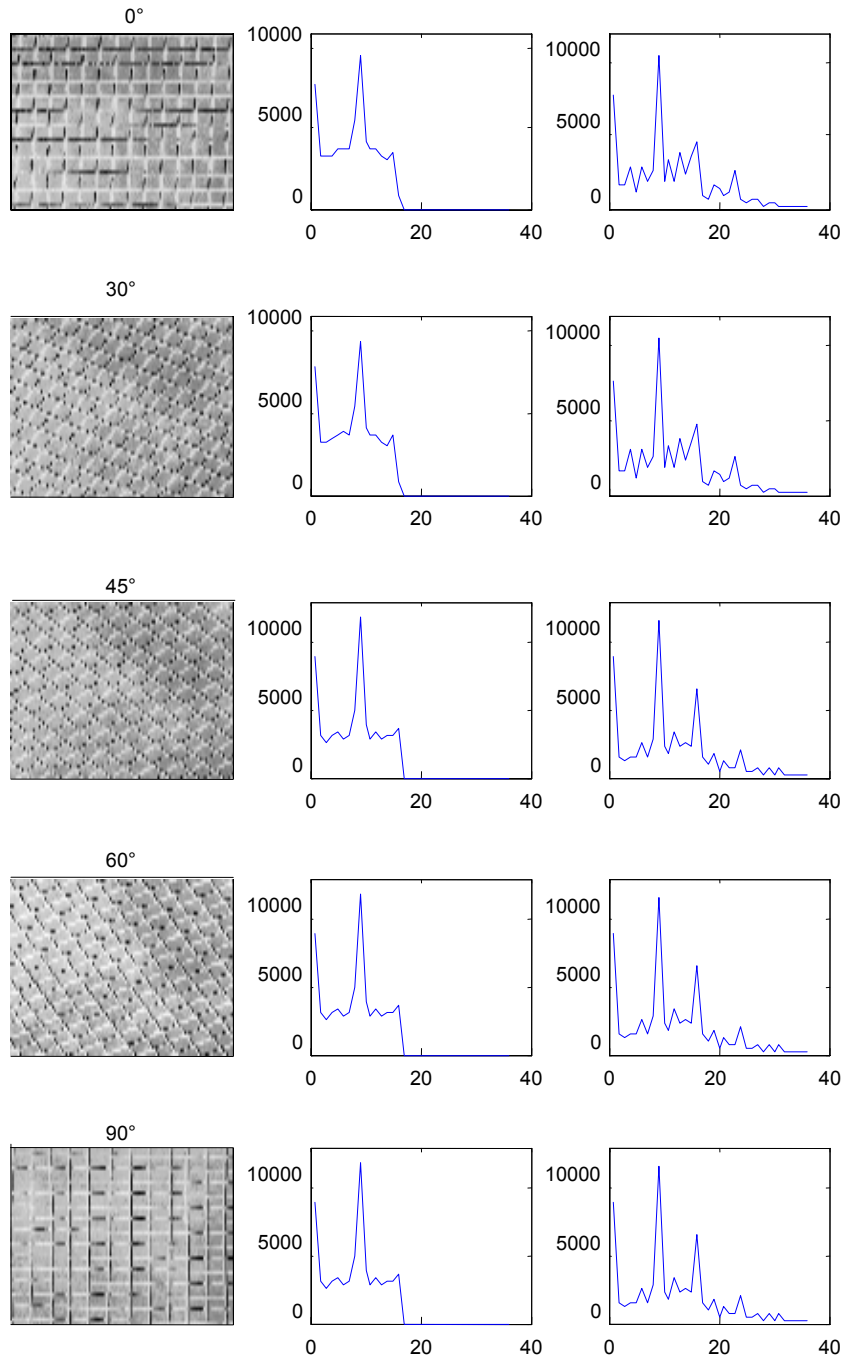


Figure 2.4: Histograms of  $\phi_2$  (middle column) and  $\phi_3$  (right most column) for image D87 at 4 angles of image orientations,  $\theta = 0^\circ, 30^\circ, 45^\circ, 60^\circ$ , and  $90^\circ$  (left column)

### 2.3.3 Smaller neighbourhoods

Sometimes, it is desirable to extract more and more information from the smallest possible neighbourhoods. In [Ursani 2008], we proposed improvements in the LFH-based features [Zhou 2001] by suggesting more features extracted from the four  $2 \times 2$ -pixel neighbourhoods within the same  $3 \times 3$ -pixel neighbourhood as shown in figure 2.5. The coefficients of the DFT of the four 4-pixel sequences provide 32 additional texture features. The magnitude of the second DFT coefficient, i.e.  $|X_1|$  from each of the four 4-pixel sequence quantises to 8 bins to provide 32 texture

features. In this way, the total number of values grows to 72 in the resulting feature-set hereafter referred to as LFH 72.

These were the three possible variations in the LFH-based features tested herein. Following paragraph presents a modification that applies to all the variations of the LFH-based feature set presented in the preceding sections.

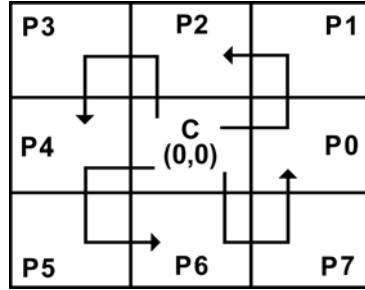


Figure 2.5: The four 4-pixel sequences

### 2.3.4 Quantisation

The LFH features perform even better when extracted from the images quantised to a number of grey levels much less than the usual value of 256 [Ursani 2008b]. The best value for the number of quantisation levels, found experimentally is 32. Therefore, this thesis proposes linearly quantising the test images, either noiseless or noisy, to 32 levels before the process of feature extraction.

## 2.4 Computational cost analysis

The computational complexity of performing convolution [Lu 1996] is as in (2.23).

$$Complexity = O(M^2) \tag{2.23}$$

where  $M \times M$  is the size of the template convolved with the image. Since one generally convolves a number of templates, say  $n$ , with an image for extracting multiple features, this complexity also becomes a function of the  $n$ . Therefore, the complexity comes to be as in (2.24).

$$Complexity = O(M^2 n) \tag{2.24}$$

In case of the DFT-based texture signatures, the size of the template, i.e.  $M=3$ , and the number of templates is  $n=8$ . Therefore, the computational complexity of the process of extracting DFT-based signatures is as in (2.25)

$$Complexity_{DFT} = O(3^2 \times 8) = O(72) \tag{2.25}$$

Similarly, in case of the features based on Gabor filters, the size of the template is 31 ( $M=31$ ), and the number of templates used is 30 ( $n=30$ ). Hence, the computational complexity of extracting these features becomes as in (2.26).



$$\text{Complexity}_{GBR} = O(31^2 \times 30) = O(28830) = O(72^{2.4}) = 2.4 \times O(72) \quad (2.26)$$

Equations (2.25) and (2.26) show that the computational complexity of the texture signatures based on Gabor filters is 2.4 times more than that of the DFT-based texture signatures.

Whatever be the computational complexity, part B of the chapter experimentally evaluates the features based on Gabor filters and the DFT.

## Part B: Experimental setup

This part of the chapter explains the datasets and the experimental conditions, before presenting the experimental results, so that the results are understandable and well interpretable.

### 2.5 Texture datasets

Evaluating a texture descriptor requires two separate datasets; one for training the classifier, and other for testing its performance. The following subsections explain separately the training image set and the test image sets used herein. The classification experiments use the training set to make models considered as class centres, and then label the test images as belonging to the nearest class. The experiments of retrieval use the class centres obtained from the training sets as queries to find the matches from within the test images. Figure 2.6 and figure 2.7 illustrate this intricate structure of the experimental setup.

#### 2.5.1 The training set

The 112 Brodatz texture images [Brodatz 1966] were downloaded from a webpage on the website of University of Stavanger<sup>11</sup> with the size of 640×640 pixels each. The training set comprises all but five, i.e. D14, D43, D44, D45, and D59. The excluded five texture-images are too irregular and chaotic to be considered as a single texture. Consequently, there are 107 images in the training set. These images serve as the training set for the nearest neighbour classifier used to produce the results presented in this chapter. Both, the nearest neighbour classifier and the k-nearest neighbour classifier are topics of the next chapter, but following is a brief explanation of the training procedure.

Feature vectors extracted from 500 square-windows randomly selected from each of the training images trained the classifier. These windows comprise  $X^2$  pixels, where  $X$  takes values from 71, 45 and 27 respectively. Each of the three training-window-sizes forms a different lookup table used one at a time by the nearest neighbour classifier. Every lookup table contains  $500 \times 107$ , i.e. 53500 feature vectors, carrying 500 representatives of each texture class and a class label at the end of every feature vector. This is a very big table, which becomes computationally expensive to use. Therefore, k-means clustering method introduced in the next chapter reduces i.e. edits these lookup tables, to have only 10 representatives of each class. Figure 2.6 elucidates all this process. For a given size of training window, the four methods of extracting texture features yield four different lookup tables with different number of columns for the kNN classifier. For example, the lookup tables corresponding to the LFH 40 features, LFH 64, and LFH 72 features have 41, 65, and 73

---

<sup>11</sup> <http://www.ux.uis.no/~tranden/brodatz.html>

columns respectively. The last column in all the tables carries the class indicator whom the feature vector represents. Similarly, the lookup table corresponding to the Gabor features has 31 columns.

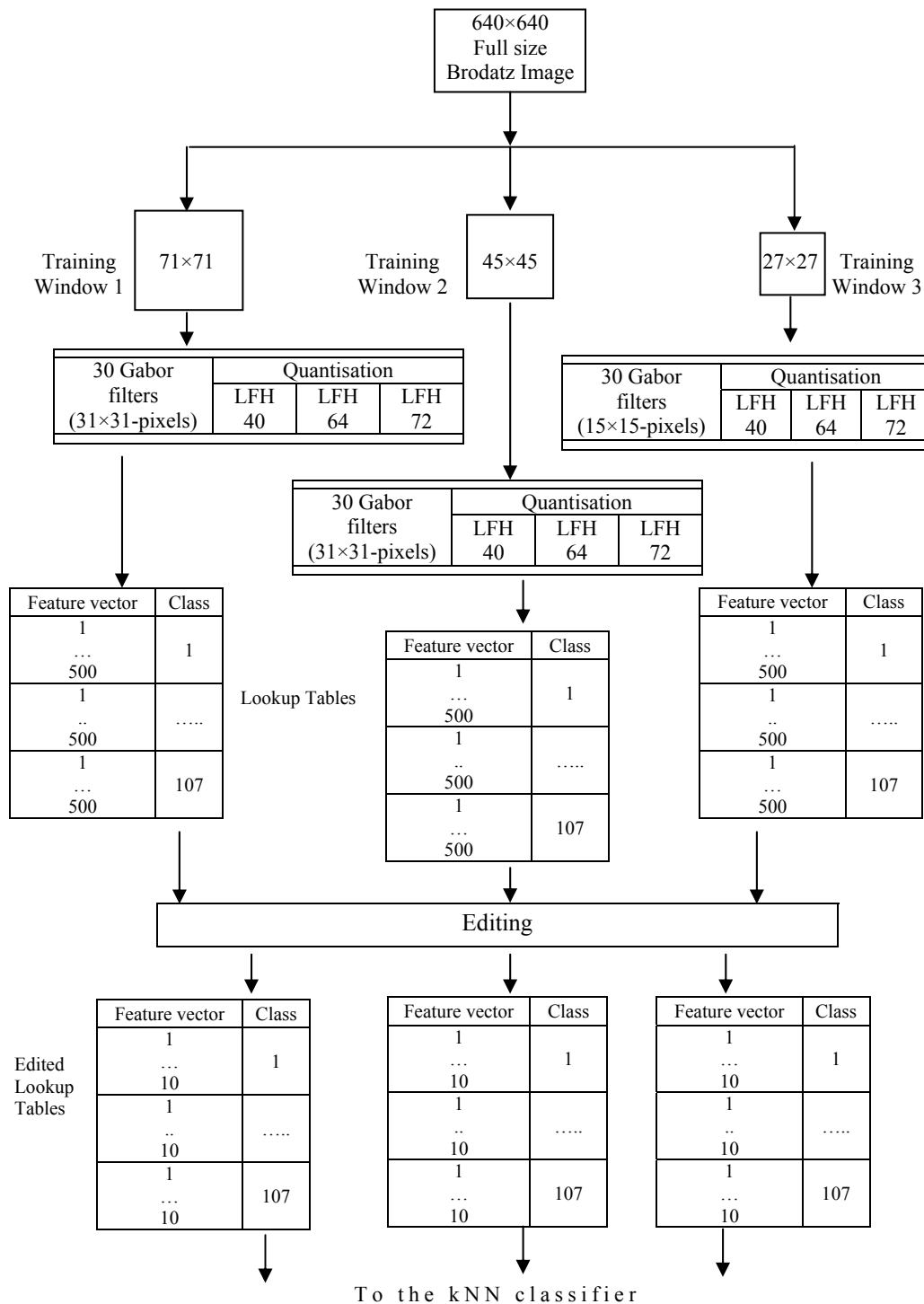


Figure 2.6: The training setup generating three lookup tables for a given feature-set, each resulting from a different training window size

This chapter presents the results using the 3 sizes of training window and the 4 methods of extracting texture features. As shall be seen later, the classification accuracy is function of the training-window size; larger the training window, higher the accuracy.

## 2.5.2 The test image sets

The texture features are tested on the set of texture images without noise and the set of texture images with Gaussian white noise.

### 2.5.2.1 Test images without noise

Two sets of noiseless test images were developed for performance evaluation. The first one consists of the variants of all the 107 Brodatz images in the training image set. The second one is a subset of the full test set.

#### 2.5.2.1.1 Full set of test images

Each one of the 107 training images was rotated to  $0^\circ$ ,  $30^\circ$ ,  $45^\circ$ ,  $60^\circ$  and  $90^\circ$ . This gave five variants of each of the training images. Sixteen images measuring  $128 \times 128$  pixels were cropped from each one of the rotated images. This gave 8560 test images in all comprising  $5 \times 16$ , i.e. 80 siblings of each of the training images. Figure 2.7 depicts this process of generating the test image sets, whence the texture features are extracted and fed to the kNN classifier for recognition and retrieval.

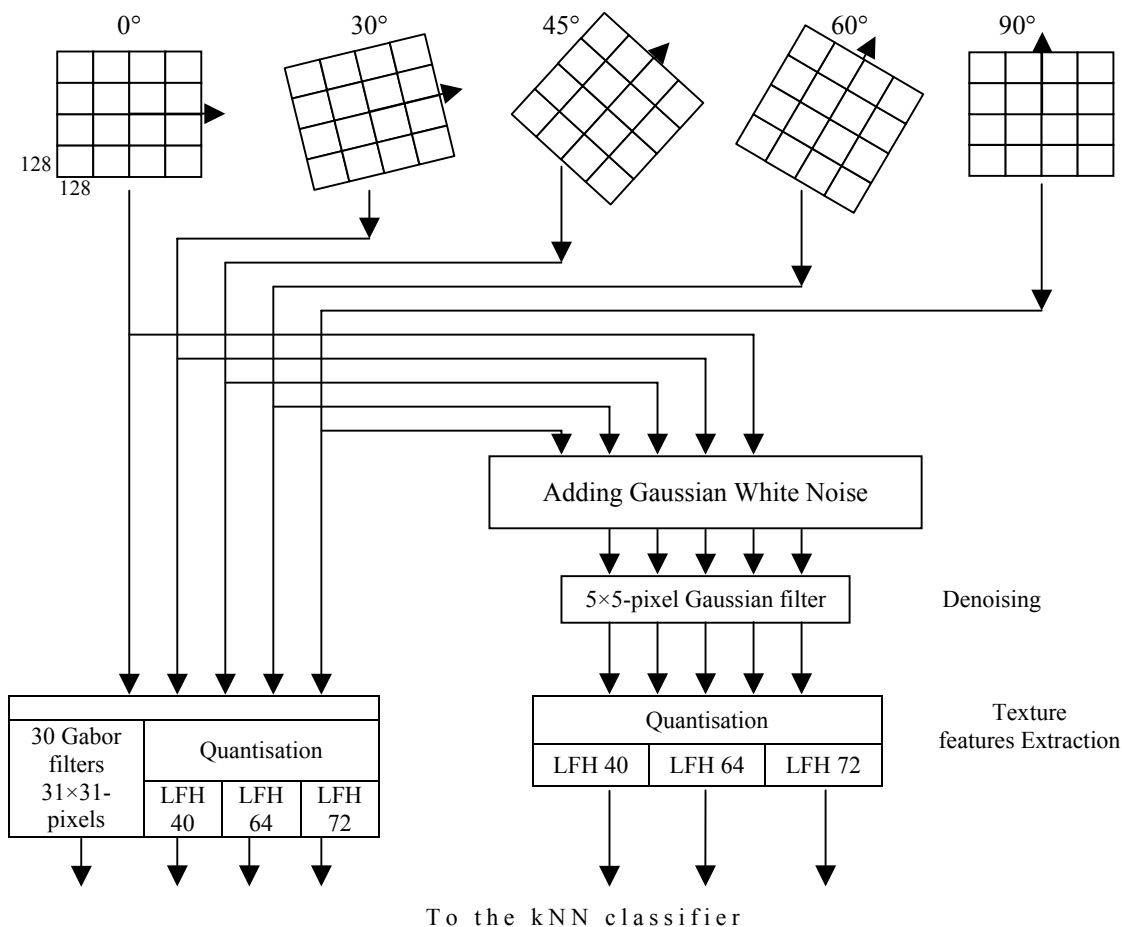


Figure 2.7: The testing setup, generating the noised and noiseless test image sets

### 2.5.2.1.2 Subset of test images

Realising that the Brodatz album indeed contains many images of a given type (paper, woven wire, canvas, stone, cloth, brick wall, flower, water, skin, etc) with different lighting conditions, zoom, etc., only one image of each type has been included in the reduced test image set, resulting in a subset comprising 32 texture classes shown in figure 2.8. Each class contains 80 siblings resulting from five orientations and 16 cropped parts, as explained in figure 2.7. In this way, there are 2560 test images in all.

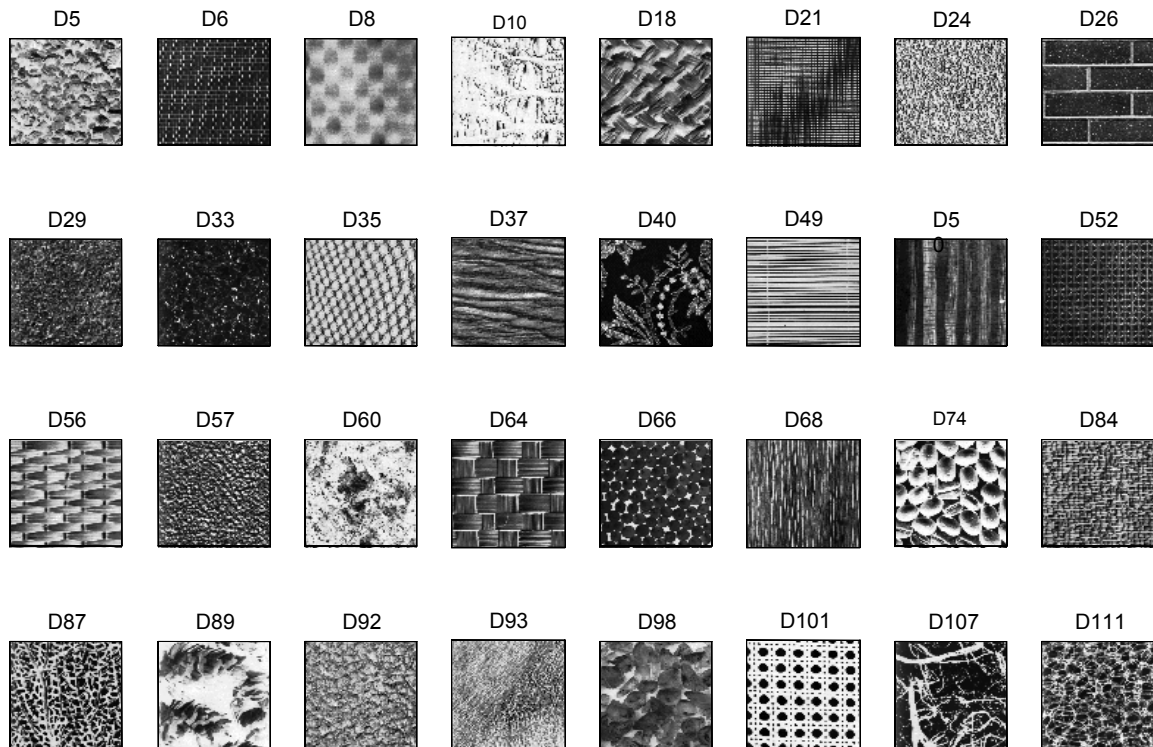


Figure 2.8: The subset of 32 texture images from Brodatz collection selected for experimentation

### 2.5.2.2 The test images with Gaussian white noise

This chapter also presents results on a set of images carrying Gaussian white noise. Generally, to exploit the RS images, these are pre-processed against noise before extracting any features thence. Therefore, Gaussian noise is added to the subset of test images comprising 32 classes and 2560 images. The added noise is quite severe and results in an average signal to noise ratio (SNR) of 2.9 dB. Table 2.3 shows SNR values for all the 32 images. Equation (2.27) estimates the SNR of the images with Gaussian white noise. Figure 2.9 shows image D87 with and without Gaussian white noise. Later, a 5×5-pixel Gaussian filter denoises these images before extracting any features from these test images. This replicates the experimental conditions similar to practical remote-sensing applications, where an RS image is received carrying Gaussian noise and is pre-processed against the same.

$$SNR = 20 \times \log_{10} \sqrt{\frac{\sum_{i=1}^X \sum_{j=1}^Y |I|^2}{\sum_{i=1}^X \sum_{j=1}^Y |I - In|^2}} \quad (2.27)$$

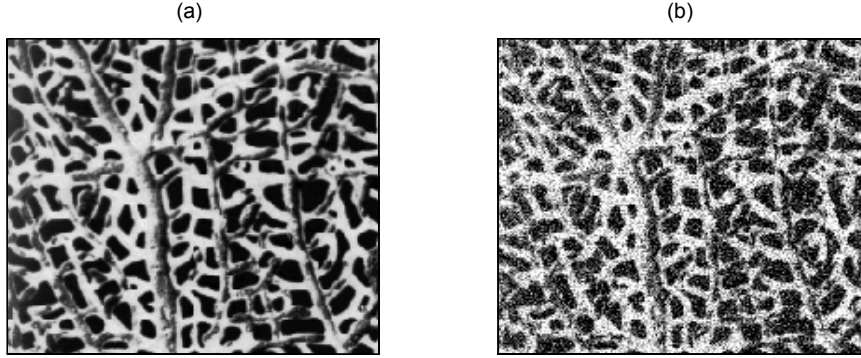


Figure 2.9: (a) the noiseless texture image D87, (b) the same image with Gaussian white noise

Table 2.3: The SNR values in decibels for the images with Gaussian white noise

S. No.	Image	SNR (dB)
1	D5	2.90
2	D6	2.87
3	D8	2.89
4	D10	4.14
5	D18	2.87
6	D21	2.80
7	D24	2.91
8	D26	2.88
9	D29	2.87
10	D33	2.06
11	D35	2.91
12	D37	2.88
13	D40	1.61
14	D49	2.99
15	D50	2.86
16	D52	2.51
S. No.	Image	SNR (dB)
17	D56	2.88
18	D57	2.81
19	D60	3.54
20	D64	2.85
21	D66	2.88
22	D68	2.84
23	D74	3.39
24	D84	2.85
25	D87	2.31
26	D89	3.61
27	D92	2.88
28	D93	2.89
29	D98	2.89
30	D101	3.68
31	D107	2.28
32	D111	2.64

## 2.6 Performance evaluation

To evaluate the performance of the various texture feature sets, following sections present the results of experiments performing image recognition and image retrieval on the test image sets explained in the preceding section.

Following sections explain the process of recognition and retrieval as well as the performance measures used to evaluate the texture feature sets. The four texture feature sets include the features extracted from Gabor filters as proposed in [Zhang 2000], the LFH 40 features from the rectangular neighbourhood as proposed by [Zhou 2001] (explained in section 2.2.1), LFH 64 features from the rectangular neighbourhood as we proposed in [Ursani 2008b] (explained in section 2.3.2), and the LFH 72 features from the rectangular neighbourhood as we proposed in [Ursani 2008a] (explained in section 2.3.3).

### 2.6.1 Recognition

In these experiments, the k-nearest neighbour classifier, explained in the next chapter assigns each of the test images, one of the 107 or 32 classes, as the case may be. The overall recognition performance is estimated as percentage of the total number of images correctly classified from the total number of test images. In addition, individual results of recognition are also presented for each orientation of the test images.

### 2.6.2 Rotation invariant texture features

Although many works on rotation invariant texture features have been published, including [Zhang 2000], [Zhou 2001], [Pun 2003], [Miguel 1994], and [Do 2002], practically there has not been any effort of numerically estimating the claimed rotation invariance or otherwise of the texture features. Therefore, this thesis also proposes a quantifier for rotation variance of the texture features. Since the test images are oriented at  $0^\circ$ ,  $30^\circ$ ,  $45^\circ$ ,  $60^\circ$  and  $90^\circ$ , the results are also presented for each orientation separately apart from the overall accuracy. This allows us estimating the rotation variance of the feature sets. Equation (2.28) analytically defines the parameter called *rotation variance (RV)*.

$$RV = \frac{std(\{A_0, A_{30}, A_{45}, A_{60}, A_{90}\})}{OA} \times 100, \quad (2.28)$$

where *std* represents standard deviation, OA represents overall percent accuracy of recognition, and  $A_0$  through  $A_{90}$  represent the percent recognition accuracies in the respective orientations.

### 2.6.3 Retrieval

The process of retrieval sorts out the images from a database, which are relevant to a query image in the order of relevance. In the retrieval experiments reported herein, each of the models from the training images presents itself as a query, and the classifier finds the closest  $N$  matches from the database that contains 80 siblings of each of the query images. The  $N$  takes 20 values from 16 to 320, with the step of 16. Often, the image processing community presents the results of the content based image retrieval using precision and recall curves. Equation (2.29) and (2.30) mathematically explain the performance parameters called precision and recall, respectively.

$$recall = 100 \times \frac{A}{A + B}, \quad (2.29)$$

where A represents the number of relevant images retrieved and B represents the number of relevant images not retrieved. The precision is calculated as

$$precision = 100 \times \frac{A}{A + C}, \quad (2.30)$$

where A represents the number of relevant images retrieved and C is the number of irrelevant images retrieved. It should be noted that the precision and recall curves show the consolidated retrieval results for the complete sets rather than for individual images.

In case of the full set test images, the retrieval tests reported in the following sections take each of the 107 training images as a query image and searches for its matches among 8560 test images. In case of the subset of test images, these take each of the 32 images training as a query image and looks for its matches among 2560 test images. The average precision is plotted against average recall for all the values of N.

## 2.7 Results

This section presents the results of texture recognition and texture retrieval using the texture features explained in the preceding sections. The k-nearest neighbour classifiers using the training window sizes of  $71 \times 71$  pixels,  $45 \times 45$  pixels, and  $27 \times 27$  pixels, respectively, produce the results of texture recognition, whereas the nearest neighbour classifiers with the same training window sizes produce the results of texture retrieval. The results are presented successively on the test images without and with noise.

### 2.7.1 The test images without noise

Following sections present the recognition and retrieval results on the full set as well as subset of the noiseless test images.

#### 2.7.1.1 Recognition

The following subsections present the results of texture recognition. In the results presented in the tables, the bracketed value shows the value of k that yields the best results while using the kNN classifier.

##### 2.7.1.1.1 Full set of test images

The classifier takes each one of the 8560 test images as a query image, recognises it as belonging to any one of the families corresponding to the 107 images. Table 2.4 through table 2.6 show the results obtained using the training window size of  $71 \times 71$  pixels,  $45 \times 45$  pixels, and  $27 \times 27$  pixels respectively. The numbers in bold indicate the best results.

The LFH-base features sets exhibit less rotation variance as compared to the feature sets based on Gabor filters. This shows that the texture features proposed herein, i.e. LFH 64 and LFH 72; perform better not only in terms of percent accuracy but also in terms of the rotation invariance. Among all, the LFH 64 features perform the best.



Table 2.4: Recognition results using the training window size of 71×71 pixels

A total of (107×16×5) 8560 images recognized		Orientation-wise Percent Accuracy			
		Gabor (k=1)	LFH 40 (k=1)	LFH 64 (k=1)	LFH 72 (k=1)
Orientation	0°	67.46	<b>84.35</b>	84.23	80.20
	30°	47.78	65.95	<b>75.00</b>	72.08
	45°	38.90	60.16	<b>74.53</b>	68.87
	60°	48.25	57.77	70.74	<b>70.97</b>
	90°	66.30	69.63	79.09	<b>81.31</b>
Overall % Accuracy		53.74	67.57	<b>76.72</b>	74.68
Rotation Variance		23.39	15.51	6.70	7.60

Table 2.5: Overall and orientation-wise recognition results using the training window size of 45×45 pixels

A total of (107×16×5) 8560 images recognized		Orientation-wise Percent Accuracy			
		Gabor (k=1)	LFH 40 (k=1)	LFH 64 (k=3)	LFH 72 (k=1)
Orientation	0°	55.08	70.85	<b>76.46</b>	70.56
	30°	43.16	63.32	<b>67.35</b>	62.91
	45°	35.35	62.38	<b>66.82</b>	58.00
	60°	40.43	61.04	<b>62.32</b>	61.33
	90°	53.13	69.92	70.04	<b>72.55</b>
Overall % Accuracy		45.43	65.50	<b>68.60</b>	65.07
Rotation Variance		18.55	6.94	7.57	9.56

Table 2.5 shows the results using the training window size of 45×45 pixels. Again, LFH 64 features perform the best. However, LFH 72 features could not perform better than LFH 40.

Table 2.6: Overall and orientation-wise recognition results using the training window size of 27×27 pixels

A total of (107×16×5) 8560 images recognized		Orientation-wise Percent Accuracy			
		Gabor (k=250)	LFH 40 (k=1)	LFH 64 (k=5)	LFH 72 (k=5)
Orientation	0°	2.80	53.33	<b>58.18</b>	56.72
	30°	2.80	46.90	<b>50.47</b>	48.31
	45°	2.80	46.09	<b>50.12</b>	45.74
	60°	2.80	46.26	<b>48.60</b>	47.08
	90°	2.80	53.33	53.45	<b>56.95</b>
Overall % Accuracy		2.80	49.18	<b>52.16</b>	50.96
Rotation Variance		0	7.72	7.27	10.68

Once again, the Gabor features are the worst. Table 2.6 shows the results using the training window size of 27×27 pixels. In this case, also, LFH 64 feature-set perform the best. This time, LFH 72 features outperform LFH 40 features.

The results are quite low obviously, because there are many variants of the same texture in the full set of 107 images that the classifier does not consider as belonging to the same class in this

experiment. It is also noticeable that the classification accuracy decreases phenomenally with the decreasing size of the training-window.

**2.7.1.1.2 Subset of test images**

Table 2.7 through table 2.9 show the overall accuracy, the orientation-wise accuracies and the rotation variance for the test images without noise, with the training window size of 71×71 pixels, 45×45 pixels, and 27×27 pixels respectively.

In table 2.7, the best results were obtained with the value of k being 1. In this case, LFH 72 features yield the best overall accuracy, but for the test images oriented at 0°, 30°, 45° and 60°, the best accuracy come with LFH 64 features. LFH 72 features outperform LFH 64 features only when the test images are oriented at 90°. The performance of LFH 72 features is slightly better in overall classification, but has higher rotation invariance than LFH 64 features have.

Table 2.7: Overall and orientation-wise recognition results using the training window size of 71×71 pixels

A total of (32×16×5) 2560 images recognized		Orientation-wise Percent Accuracy			
		Gabor (k=1)	LFH 40 (k=1)	LFH 64 (k=1)	LFH 72 (k=1)
Orientation	0°	87.11	93.75	<b>96.09</b>	95.51
	30°	71.09	91.41	<b>93.75</b>	92.77
	45°	57.42	92.38	<b>93.16</b>	91.02
	60°	67.97	90.82	91.21	<b>91.60</b>
	90°	85.94	94.73	91.80	<b>96.29</b>
Overall % Accuracy		73.91	92.62	93.20	<b>93.44</b>
Rotation Variance		17.03	1.75	2.05	2.51

In table 2.8, the orientation-wise result remains the same as before, but this time LFH 64 features outperform LFH 72 features in the overall accuracy as well. LFH 72 features yield their best results with k=3, as compared to other that perform at their best with k=1.

Table 2.8: Overall and orientation-wise recognition results on the smaller set of test images using the training window size of 45×45 pixels

A total of (32×16×5) 2560 images recognized		Orientation-wise Percent Accuracy			
		Gabor (k=1)	LFH 40 (k=1)	LFH 64 (k=1)	LFH 72 (k=3)
Orientation	0°	55.08	89.26	<b>93.16</b>	90.23
	30°	43.16	85.74	<b>91.02</b>	84.96
	45°	35.35	83.79	<b>88.09</b>	84.38
	60°	40.43	83.79	<b>86.33</b>	85.55
	90°	53.13	90.04	87.50	<b>91.02</b>
Overall % Accuracy		45.43	86.52	<b>89.22</b>	87.23
Rotation Variance		18.55	3.44	3.14	3.60

In table 2.9, LFH 64 features outperform all others. The best results come with  $k=10$ . Once again, LFH 72 features outperform LFH 64 features when the test images are oriented at  $90^\circ$ . The difference in overall accuracies of the two is not very big but LFH 64 features have considerably lower rotation variance than LFH 72 features have.

Table 2.9: Overall and orientation-wise recognition results using the training window size of  $27 \times 27$  pixels

A total of $(32 \times 16 \times 5)$ 2560 images recognized		Orientation-wise Percent Accuracy			
		Gabor ( $k=25$ )	LFH 40 ( $k=10$ )	LFH 64 ( $k=10$ )	LFH 72 ( $k=3$ )
Orientation	$0^\circ$	9.38	76.37	<b>82.23</b>	81.84
	$30^\circ$	9.38	72.85	<b>78.52</b>	74.80
	$45^\circ$	9.38	73.63	<b>79.49</b>	73.05
	$60^\circ$	9.38	71.48	<b>76.76</b>	75.00
	$90^\circ$	9.38	74.41	74.61	<b>83.20</b>
Overall % Accuracy		9.38	73.75	<b>78.32</b>	77.58
Rotation Variance		0	2.47	3.66	5.93

The results of texture recognition show that the LFH-based features have higher description power than the features based on Gabor filters for all the sizes of the training window. In addition, performance of all the texture descriptors decreases phenomenally with the decreasing size of the training window. However, the size of training window affects the performance of Gabor-based features the most drastically. The Gabor features 9.375% accuracy, because they correctly recognise the test images belonging to only three classes, i.e. D10, D21, and D49. They recognise all other test images incorrectly as belonging to the class D10.

Among the LFH-based features, LFH 64 features perform the best, but yield accuracy lower than that of LFH 72 features when the test image is oriented at  $90^\circ$ .

### 2.7.1.2 Retrieval

Following sections present the results of image retrieval using the four texture feature-sets on the two datasets. In these experiments, each of the training images presents itself as query, and the nearest neighbour classifier sorts out the closest matches among all the test images.

#### 2.7.1.2.1 Full set of test images

This section presents the results of retrieval on the full set of test images consisting of 107 classes. The graph of figure 2.10 illustrates the results of retrieval obtained with the four feature sets, while using the training-window size of  $71 \times 71$  pixels.

The results reveal that for a given value of precision, the LFH-based features provide higher recall than the features based on Gabor filters do. The Gabor features perform far below the LFH-based features. Similarly, for a given value of recall, LFH features attain higher precision than others do. Among the LFH-base feature-sets, LFH 64 feature-set outperforms the two others and

always remains on the upper-right side of the curves representing other feature-sets. Similarly, figure 2.11 presents the results of retrieval while using the training-window of  $45 \times 45$  pixels. All the features perform worse than while using the larger training window. However, the performance of LFH-based feature-sets is affected much less severely than that of the Gabor-based features.

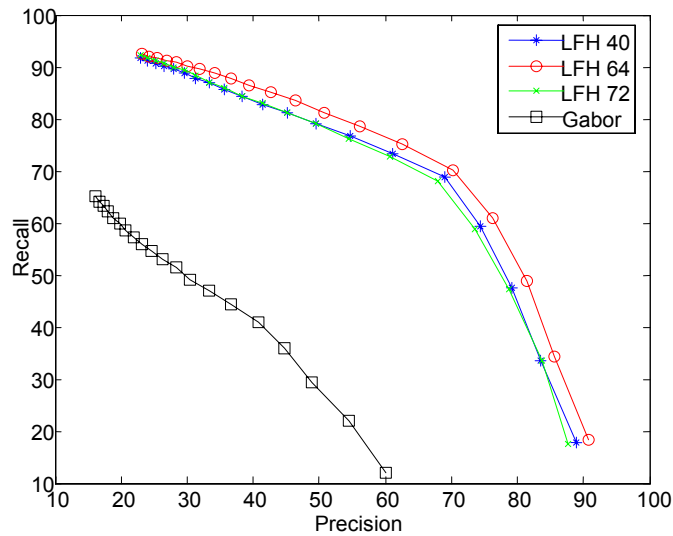


Figure 2.10: Performance of texture features while retrieving the images from the database comprising 107 classes and 8560 test-images using the classifier trained with window size of  $71 \times 71$  pixels

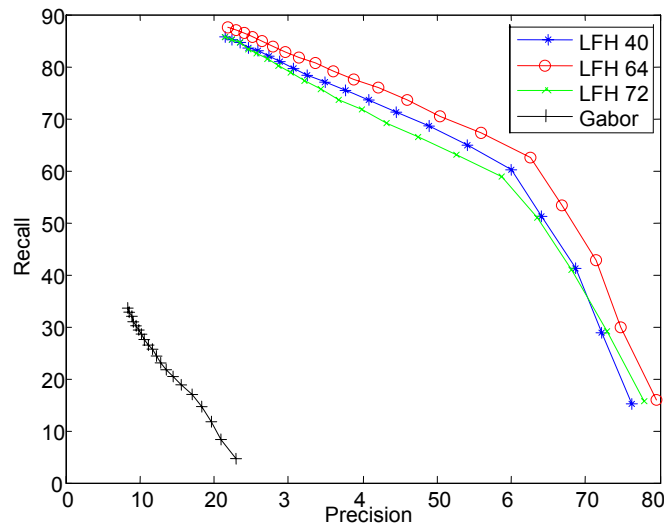


Figure 2.11: Performance of texture features while retrieving the images from the database comprising 107 classes and 8560 test-images using the classifier trained with window size of  $45 \times 45$  pixels

Figure 2.12 shows the retrieval results while using the smallest size of the training-window. In this case, LFH 64 features have the clear edge over all others. However, LFH 72 features seem slightly better than LFH 40 features. The features based on Gabor filters perform the worst. Its curve appears at the bottom-left corner of the plot.

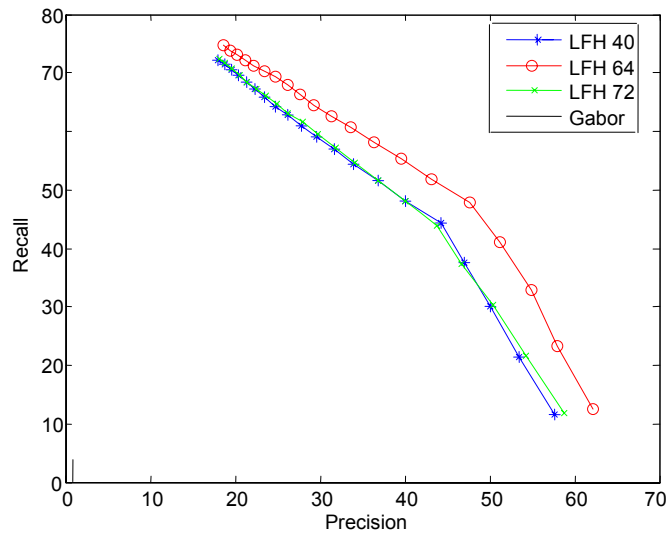


Figure 2.12: Performance of texture features while retrieving the images from the database comprising 107 classes and 8560 test-images using the classifier trained with window size of  $27 \times 27$  pixels

### 2.7.1.2.2 Subset of test images

Figure 2.13 through figure 2.15 show the retrieval results with the subset of test images (comprising 32 classes), using the window size of  $71 \times 71$  pixels,  $45 \times 45$  pixels, and  $27 \times 27$  pixels respectively. The results show that the overall performance becomes worse with the decreasing size of the training window size. One can also conclude that the features based on Gabor filters perform much poorly as compared to the performance of LFH-based texture features.

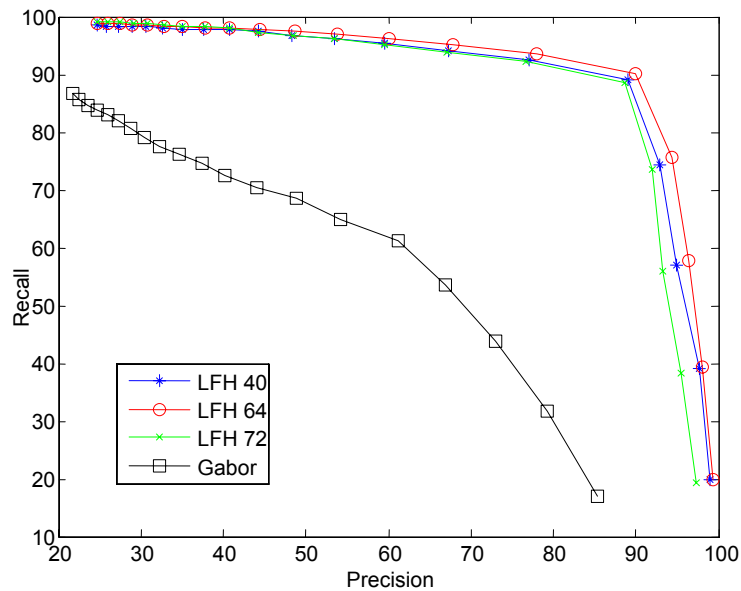


Figure 2.13: Retrieval results on the subset of 32 test images using the training window of  $71 \times 71$  pixels

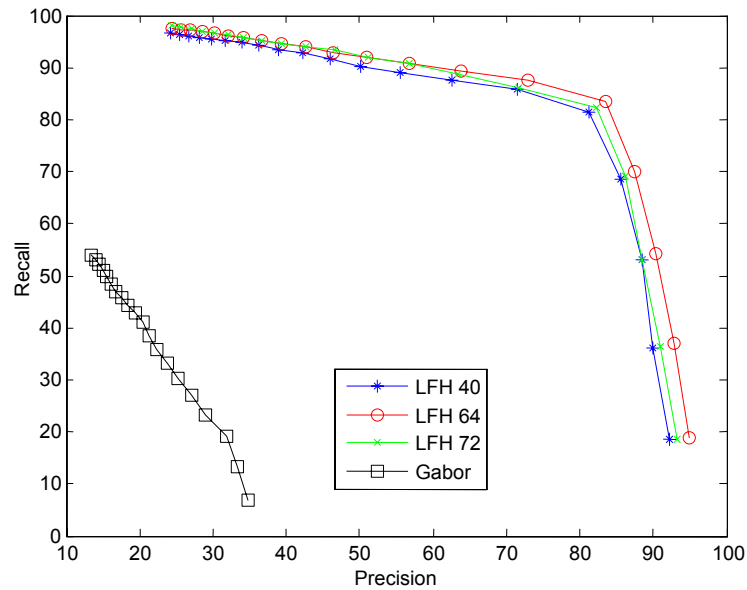


Figure 2.14: Retrieval results on the subset with 32 test images using the training-window of  $45 \times 45$  pixels

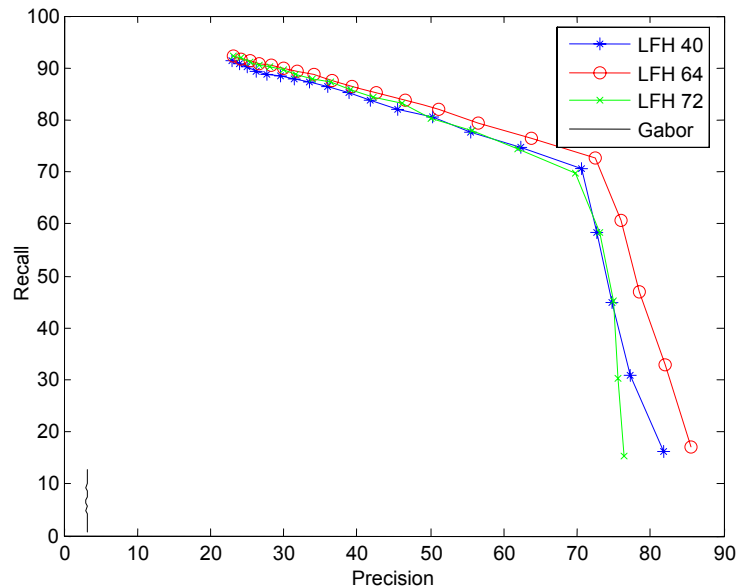


Figure 2.15: The retrieval results on the subset of 32 test-images using the training-window of  $27 \times 27$  pixels

The results show that the smallest training window gives the worst precision and recall. However, the LFH 64 features consistently prove the best among all and the Gabor-based features consistently prove the worst among all. In case of the smallest training window, the features based on Gabor filters simply fail to perform at all.

### 2.7.2 The test images with Gaussian white noise

Since the remote sensing images contain Gaussian white noise, this section presents results on the images contaminated with the Gaussian noise. As will be observed in chapter 4, the texture-based classification of the remote-sensing image under process cannot use the training window sizes larger than  $27 \times 27$  pixels. Therefore, following sections present the results with the smallest training

window size, i.e.  $27 \times 27$  pixels only. With this size of the training window, texture features based on Gabor filters failed in the noiseless conditions. Even the results on the images without noise with the Gabor features are worse than the results of the LFH-based features in the noisy conditions. Therefore, following experiments do not consider the texture features from Gabor filters any more.

### 2.7.2.1 Recognition

Table 2.10 shows the recognition results. In the noisy conditions as well, the LFH 64 features perform the best among all. This time, best results are obtained with much higher values of  $k$ . LFH 64 and LFH 72 features yield their best results with  $k=15$ , whereas LFH 40 features yield their best results with  $k=25$ . LFH 72 features perform the worst and LFH 64 features perform the best. Although LFH 72 features exhibit less rotation variance than LFH 64 features, the latter outperforms the former at every orientation of the test images. Contrary to the results in the noiseless conditions, the LFH 64 features are at their best with the test images oriented at  $90^\circ$ .

Table 2.10: Recognition results on the smaller set of test images with Gaussian noise

A total of (32×16×5) 2560 images recognized		Orientation-wise Percent Accuracy		
		LFH 40 (k=25)	LFH 64 (k=15)	LFH 72 (k=15)
Orientation	0°	19.34	<b>21.88</b>	11.52
	30°	20.51	<b>22.27</b>	10.55
	45°	20.31	<b>21.68</b>	10.55
	60°	20.51	<b>22.46</b>	11.33
	90°	19.14	<b>24.41</b>	10.55
Overall % Accuracy		19.96	<b>22.54</b>	10.90
Rotation Variance		3.35	4.85	4.46

### 2.7.2.2 Retrieval

Figure 2.16 shows the results of retrieving noisy test images when a noiseless training image presents itself as a query. The LFH 64 features perform the best among all. In this case, the LFH 72 features also fail to respond in a normal fashion of the decreasing precision with the increasing recall; the precision remains somewhat constant for a big range of recall values. It can be seen that for a number of retrieved matches higher than 196, LFH 72 feature-set outperforms other LFH feature-sets. On the other hand, if the number of retrieved matches is less than 192, the other two variants of the LFH features outperform the LFH 72 feature-set.

Since the purpose of using texture features in the research reported herein is to recognise the land-covers, it is the requirement is the high precision. Therefore, LHF 64 texture feature-set is the best among all feature-sets tested herein for the applications under consideration.

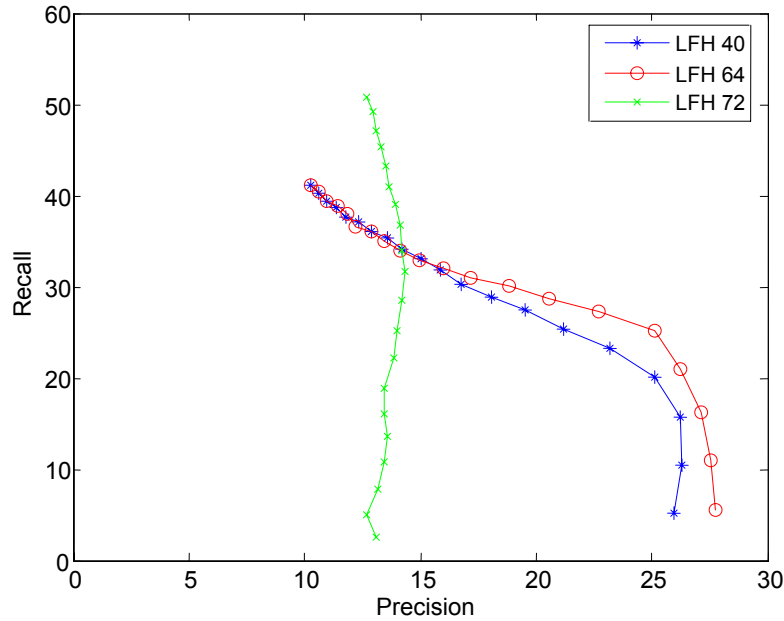


Figure 2.16: The retrieval results on subset comprising 32 test images using the training window size of  $27 \times 27$  pixels

## 2.8 Conclusion

This chapter dealt with the extraction of texture features from the images containing a single texture and their recognition as belonging to a class or family of textures for evaluating the texture features. The experiments conducted in this chapter also include retrieval tests.

The texture features based on LFH exhibit greater description power than the features based on Gabor filters. LFH-based features also exhibit low values for the parameter of rotation variance introduced herein, which verifies its better robustness against rotation as compared to the features based on Gabor filters. The experimental results also help conclude that among the LFH-based features, LFH 64 features set that we proposed in [Ursani 2008b] performs the best, but yields lower accuracy than that of LFH 72 when the test images are oriented at  $90^\circ$ . This shows that there is a room for improving the implementation of the LFH 64 features.

This chapter concludes with the remark that the texture features based on LFH suit the application of remote sensing, since they are more objective and independent of human perception unlike Gabor features [Cutter 2003]. As the remote sensing images are captured using the electromagnetic energy visible as well as invisible to human vision system, they may contain the texture that are apparently similar to humans.

[Julesz 1973] found that textures with similar  $2^{\text{nd}}$  order statistics appear indistinguishable to the human vision system. Therefore, the texture features based on grey level co-occurrence matrices (GLCM) features that are  $2^{\text{nd}}$  orders statistics, suit to the applications that aim at mimicking human vision. However, in the applications like remote sensing requiring more objective assessment and



hence higher order statistical analysis of texture, GLCM do not yield good results [Schröder 1998]. On the other hand, LFH-based features provide 8<sup>th</sup> order statistics of textures. As all coefficients of Fourier transform except the first one are invariant to the grey-level shift, the LFH-based features inherit the property of grey level invariance if used without the first coefficient.

The window-size in the training phase affects the accuracy of both, recognition as well as the retrieval. Larger the window size, higher the performance. The features based on Gabor filters are impractical in training the kNN classifier with the window sizes smaller than 45×45 pixels. In the situations where the training samples are even smaller than 45×45 pixels, the texture features based on signal processing approaches become impractical. Such is the case while training the land-covers in an RS image to the kNN classifier using texture features based on Gabor filters.

Classifying a remote-sensing image, using these or any other features, involves the processes of segmentation and classification for delineating and recognising the regions of homogeneous land-cover, the topics of the next chapter.



# 3

## Image Segmentation and classification

3.1	Segmentation and classification .....	62
3.2	Unsupervised classification .....	62
3.2.1	Single-run k-means algorithm .....	64
3.2.1.1	Methods of initialisation .....	64
3.2.1.2	K-means with multiple restarts .....	65
3.2.1.3	Gene Cluster 3.0 .....	65
3.2.1.4	Problems with k-means .....	65
3.2.2	Notion behind FOOS .....	66
3.2.3	Fusion of over-segmentations (FOOS) .....	67
3.2.3.1	Over-segmentations .....	67
3.2.3.2	Fusion .....	68
3.2.4	Computational complexity .....	69
3.2.5	Comparison .....	70
3.2.5.1	Synthetic texture images .....	71
3.2.5.1.1	Natural digital photos .....	72
3.2.5.1.2	The remote-sensing images .....	80
3.2.5.2	Non-image datasets .....	82
3.2.5.2.1	A dataset with 2 classes .....	82
3.2.5.2.2	A dataset with 10 classes .....	82
3.3	Supervised classification .....	83
3.3.1	The nearest neighbour classification .....	83
3.3.2	K nearest neighbours classification .....	84
3.3.3	Similarity criterion .....	84
3.3.4	Editing the lookup table .....	84
3.4	Conclusion .....	85

Procedures in remote sensing of environment include segmentation and classification of the RS images captured by the sensors aboard a distant platform like an Earth observation satellite. Image segmentation is a fundamental step in most of the applications of image processing by large [Lucchese 2001], including computer vision, apart from remote sensing. The region-based techniques of classifying an RS image consist of the image segmentation process before the classification of segments or regions takes place [Hirose 2004] [Liu 2006]. The image segmentation involves extraction of some image attributes, such as texture and then using a data clustering method. Therefore, this chapter suggests improvements in the clustering method called k-means, and presents results of the new algorithm called FOOS, on remote sensing images, non-remote sensing images as well as some non-image datasets. The image characteristics used for image segmentation are the texture features introduced in the previous chapter.

The following sections introduce the problem of segmentation and classification, as well as the techniques of unsupervised clustering and supervised classification. The subsections present k-means clustering method and analysis of the problems associated with it and causes of its possible failure, before suggesting improvements. This chapter also explains the k-nearest neighbour classifier that produced the results presented in the previous chapter.

### 3.1 Segmentation and classification

In the context of remote sensing, segmentation is the process of demarcating individual regions of homogenous land-cover in an RS image, whereas classification is the subsequent process of identifying the delineated regions as belonging to a specific land-cover.

The segmentation may or may not be a supervised process. Depending on the application, different levels of supervision have been proposed. Some researchers have described their work on segmentation even as semi-supervised [Xia 2005] and weakly supervised [Vasconcelos 2006]. However, the process involves only delineating different regions that are homogeneous with regard to some characteristics. Therefore, in its simplest way, it consists of the feature extraction and the subsequent unsupervised clustering process, and does not necessarily require any prior learning. Since the segmentation process only requires distinguishing among the homogenous regions present in the image without needing to know the region- or class-specific characteristics a priori, it is generally an unsupervised process, more formally known as unsupervised classification, apart from the usual names of clustering and segmentation. In this thesis, an unsupervised classification, i.e. a clustering method segments the remote sensing images into the regions of homogeneous texture using texture features.

In the context of remote sensing, classification is the process of land-cover mapping, i.e. assigning each image-region of homogeneous characteristics, called as land-cover hereafter, a specific class name such as water, barren land, vegetation, ice, etc. It is essentially a supervised process that requires knowing a priori, the characteristics of the specific land-covers to be recognised or identified. This needs the human-supervision and requires a method to model the a priori knowledge of the land-covers or classes at hand. Usually, one refers to the process of supervised classification simply as the classification.

### 3.2 Unsupervised classification

The clustering is a process of dividing a dataset into a number of disjoint clusters by putting the member data elements, hereafter called as instances, similar in some respects into a single cluster. Clustering methods come to use when there is no information available a priori regarding the dataset to be classified, like the clusters/classes themselves, their probability of occurrence, mean and/or variance in the feature space. These unsupervised algorithms segment the datasets without requiring any a priori knowledge regarding the dataset itself or the expected classes therein. These methods decide upon the similarity and dissimilarity among the instances of the dataset on the basis of some features and a distance measure in the feature space. The most common distance measure is Euclidean distance, a specific case of Minkowsky distance expressed in (3.1).

$$D_{ij} = r \sqrt{\left( \sum_{n=1}^L |v_{i,n} - v_{j,n}|^r \right)} \quad (3.1)$$

where  $D_{ij}$  represents the distance between  $i^{\text{th}}$  and  $j^{\text{th}}$  instances,  $L$  is length of the feature vector,  $v_i$  and  $v_j$  are the feature vectors of the  $i^{\text{th}}$  and the  $j^{\text{th}}$  instance, respectively. In case of Euclidean distance,  $r$  equals 2. Similarly, in case of Manhattan distance,  $r$  equals 1.

Different tradeoffs in the problem of clustering include the number of clusters, the minimum distance between the cluster-centres, maximum number of instances in a cluster, etc. The user must specify some of these parameters to initiate the execution of the clustering process. For example, user must specify any one of the minimum distance between the cluster-centres, the maximum number of instances in a cluster, or the expected number of clusters.

Despite continued research by many researchers engaged for many long years, data clustering remains one of the most non-trivial and one of the most challenging tasks in the fields of data mining [Raymond 1994], bioinformatics [Alon 1999], and image segmentation [Shi 2000] alike. Many a times, the correct classification of the remote sensing (RS) images also depends on the correct segmentation of the image. Data mining applications also need to dig out the relevant information from the volumes of data that have never been as huge as today. This requires a clustering technique that is as quick as possible. ISODATA [Verbyla 1995], k-means [McQueen 1967], quality threshold clustering [Heyer 1999], among others are the data clustering methods equally applied to image and non-image data.

Although there are many clustering techniques, so far k-means remains the most popular among unsupervised techniques of data clustering due to its simplicity and faster convergence. In addition, k-means algorithm suits the requirements of the application dealt in this thesis. All the algorithms of clustering require some prior information. The k-means algorithm requires the number of clusters to find. Other algorithms require for example, the maximum population of a cluster, the minimum population of a cluster, the maximum variance of a cluster, the maximum diameter of a cluster in the feature space, etc. All these are dependent on the type of features and the feature space of the dataset, and are difficult to find. On the other hand, the number of clusters is pre-decided in our application described later in chapter 4. Therefore, k-means becomes the most suitable among all other methods.

Nevertheless, often falling into a local minimum remains a problem *semper instans*. This thesis puts insight into and analyses the problems of the k-means clustering, presents solution to the problem in the light of that analysis, by suggesting improved way of initialising the iterative process of k-means clustering.

### 3.2.1 Single-run k-means algorithm

The k-means algorithm, also called Lloyd's algorithm is an iterative partitioning algorithm that starts with a known number of disjoint clusters. In the simplest case, the algorithm chooses the cluster centres arbitrarily from the dataset. Afterwards, it associates each instance in the dataset to the cluster whose centre lies closest to the instance as explained in (3.2).

$$x_j \in X_n \leftrightarrow d_{j,n} = \min_{i=1}^k (d_{j,i}) \quad (3.2)$$

where  $X_n$  is the  $n^{\text{th}}$  cluster,  $d_{j,n}$  is the Euclidian distance of  $x_j$  from  $n^{\text{th}}$  cluster-centre, and  $d_{j,i}$  is the Euclidian distance of  $x_j$  from  $i^{\text{th}}$  cluster-centre. The first iteration ends here and before starting next iteration, k-means updates the cluster centres by replacing older ones with the mean of the cluster members in the feature space. The new cluster centres are the means vectors as in (3.3).

$$\bar{\mu}_k = \frac{1}{N_k} \sum_{i=1}^{N_k} \bar{x}_{k,i} \quad \forall k | k=1, 2, \dots, K, \quad (3.3)$$

where  $N_k$  represents cardinality of  $k^{\text{th}}$  cluster,  $\bar{\mu}_k$  is the centre of  $k^{\text{th}}$  cluster,  $\bar{x}_{k,i}$  is the feature vector of  $i^{\text{th}}$  instance in  $k^{\text{th}}$  cluster, and  $K$  is the number of clusters to be found. Sometimes, statisticians use medians to represent the cluster-centres instead of means.

This iterative process continues until it reaches the termination condition. The condition of termination may be set as no change in the old and the new cluster centres, or as not a single change in the cluster associations of the instances or less than 1% change in the class centres, the maximum number of iterations, and so on. There are several variants of k-means mainly differing in the way of initialisation and termination.

#### 3.2.1.1 Methods of initialisation

Several methods of initialising the k-means clustering process have been proposed and been under use. The most simple is the McQueen's method that initialises the initial cluster-centres with the samples randomly selected from the dataset under segmentation. Others are more complicated involving the determination of concentrations of population in the feature space. Reference [Lozano et al. 1999] presents a good comparison of the initialisation methods for k-means and discusses the problems associated with the k-means clustering.

In addition, k-means procedure can progress in two ways, one that gives the results that are invariant to the instance order and other that give different results for the same dataset and initialisation if the instance order is changed. The k-means, more specifically k-medians invariant to instance order is the subject of interest herein, and the term k-means refers to k-medians hereafter.

Although the simplest, due to random initialisation, the McQueen's k-means algorithm has a poor repeatability, i.e. it may result in amazingly different clusters each time it executes. Here comes the k-means with multiple restarts to help.

### 3.2.1.2 K-means with multiple restarts

The k-means is generally performed several times (usually hundreds of times) before choosing best solution among the several ones found therein. The number of solutions found is generally a function of cardinality of the dataset under process. The best solution is the one that minimises intra-cluster distance, also called total classification error, defined as

$$E = \sum_{k=1}^K \sum_{n=1}^{N_k} |\mu_k - x_n|, \quad (3.4)$$

where  $K$  is the total number of clusters,  $N_k$  is cardinality of the  $k^{\text{th}}$  cluster,  $\mu_k$  is the mean feature vector of  $k^{\text{th}}$  cluster, and  $x_n$  is the feature vector of  $n^{\text{th}}$  data member of a running cluster  $k$ . This approach, called k-means with multiple restarts, is usually successful and many professional computer programs such as Gene Cluster 3.0 introduced in a later section employs this method. However, one is never sure that the best solution found is the optimal one.

### 3.2.1.3 Gene Cluster 3.0

Gene Cluster 3.0 is a computer program [Hoon 2004] based on the original Cluster program written by Mike Eisen of Berkley Lab. Gene Cluster uses a variant of k-means with multiple restarts. It works out a maximum acceptable value of intra-cluster distance [Lozano 1999] and computes it as defined in (3.4) after finding every clustering solution. If the intra-cluster distance is lower than the maximum acceptable value, it stops finding clustering solution any further, or it continues otherwise until it finds the user-defined number of solutions. In the later case, it selects the solution with the least intra-cluster distance.

### 3.2.1.4 Problems with k-means

K-means method of clustering assumes that the desired clusters are populated in spherical Gaussian distributions; and that any deviation from this situation causes k-means algorithm to fall into a sub-optimal point. [Chen 2005] is a recent work that points out three established problems/disadvantages with the k-means algorithm, which many earlier works have already pointed out. First that it requires the number of clusters beforehand, second that it is sensitive to the outliers, and third that any two randomly chosen initial cluster-centres might be too close to be considered as centroids of two distinct clusters. Reference [Lozano 1999] notes that despite having all the advantages of convergence and computational simplicity, the k-means is highly sensitive to the choice of initial cluster-centres and may easily converge into a local optimum due to its assumption that the clusters it is trying to find lie in a spherical Gaussian distribution. Fusion of over-

segmentations (FOOS) [Ursani 2007] is one of the possible solutions to these problems. Following sections present the method called FOOS and the theory behind its conception.

### 3.2.2 Notion behind FOOS

Let us have deeper look at into and analyse the three problems of k-means pointed out by [Chen 2005] and others. Requiring the number of clusters beforehand is not always a problem. Often, the user prefers to decide upon it. Even the results of manual clustering of any given dataset yield different outcomes since the number of clusters is almost always subjective depending on the required rigour of segmentation. Therefore, in many situations, the number of clusters (K) is a user-defined parameter. It is hence rather an advantage in many situations. The random selection of a cluster-centre may repeat if it does not lie at a considerable distance to all other selected cluster-centres. Rest of the problems, i.e. its assumption of spherical Gaussian distribution, sensitivity to outliers, and dependence on the initial cluster-centres, all are inter-related. Keeping in view the determination of the number of clusters using Gaussian separation in [Chang 2002], one can summarize these problems into a single problem statement as follows:

*A given dataset usually has a different number of clusters from the Gaussian analysis [Chang et al. 2002] viewpoint than the number of clusters determined by the application, the requirement, and/or the subjective assessment. This disagreement makes distributions of the desired clusters look non-Gaussian and some of their members as outliers to the k-means algorithm, making it sensitive to the choice of initial cluster-centres.*

The problem of so-called *outliers* is the same problem of non-Gaussian distribution in the different words. For a given dataset, segmenting into a larger number of clusters relieves the problem of *outliers* or that of skewness in the cluster distributions; since more and smaller/finer Gaussian classes can more closely approximate non-Gaussian distributions. [Steinwolf 1993, 1996, 2006] explain how piecewise Gaussian distributions approximate a non-Gaussian distribution. Therefore, following section proposes over-segmentation as a computationally simple way of resolving the problem of non-Gaussian cluster distributions.

The poor repeatability of k-means results because all the dataset is open for random selection of the initial cluster-centres. K-means generally succeeds in reaching the optimal solution, with the exception of failure caused by the selection of an *outlier* as one of the K initial cluster-centres. A possible solution of the poor repeatability can be determining two over-segmentations and fusing the clusters from the two clustering solutions to determine the K initial cluster centres for performing the third and the decisive k-means clustering. Over-segmenting the dataset keeps the *outliers* in separate but insignificantly smaller Gaussian clusters away from the random selection of the K initial cluster-centres and the process of fusion makes the choice of initial cluster-centres



stable, i.e. repeatable. Consequently, one reaches to the optimal or a near-optimal solution with a high repeatability.

### 3.2.3 Fusion of over-segmentations (FOOS)

FOOS is a means of finding the initial means that, unlike other initialisations methods, leads to the optimal or a near-optimal solution even if the desired clusters do not lie in the spherical Gaussian distribution. FOOS performs the k-means procedure only 3 times: it initialises 3<sup>rd</sup> and the last run of k-means with the cluster-centres found from the two preceding runs of the algorithm. The algorithm suggests finding out two over-segmentations and then performing a kind of fusion of the clusters found therein. Refer figure 3.1.

#### 3.2.3.1 Over-segmentations

The first step is to over-segment the dataset into P and Q clusters, respectively, using the Mac Queen k-means algorithm, where K is the desired number of clusters,  $P > K$  and  $Q = P + 1$ . The initial cluster-centres come from the successively randomly sub-sampled dataset, assuring a minimum distance between any two randomly selected initial cluster-centres. The condition of terminating the iterative process of the k-means algorithm is the sum of distance between present and previous cluster-centres to be less than the preset minimum value (0.01 in this study). This termination condition seems working well in all the experiments presented herein. The two over-segmentations then fuse, as described in the next sub-section, to determine good initial cluster-centres to perform the same process of k-means clustering for the third and the last time.

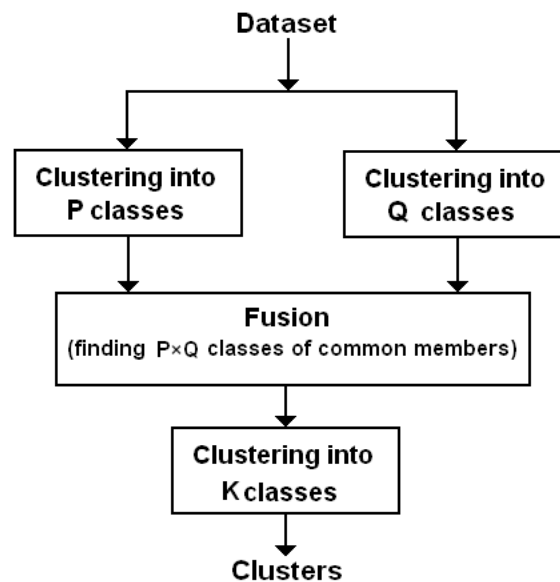


Figure 3.1: The fusion of over-segmentations

Least possible over-segmentation is suggested for the datasets with  $K \leq 5$ . In this case,  $P=K+1$ . For the datasets having more classes, i.e.  $K > 5$ , the least possible over-segmentation doesn't seem to be the best solution, as shown in the results in section 3.2.5.2.2. In case where  $K \leq 5$ , FOOS performs least possible over-segmentation leading to  $P=K+1$ , and  $Q=K+2$ . In case when  $K=5$ ,  $P$  becomes 6 and  $Q$  becomes 7; i.e. two more on five or 40% over-segmentation. If this idea of 40% over-segmentation extends to a dataset with  $K=10$ , the two proposed over-segmentations become  $P=13$  and  $Q=14$ .

### 3.2.3.2 Fusion

The process of fusion determines the  $P \times Q$  clusters of instances, which are in fact the sets of common instances. New  $K$  initial cluster-centres for clustering the dataset into  $K$  segments come from the medians of the largest sets of the common instances. In this way, FOOS rejects the smallest sets, selecting only the  $K$  largest clusters among  $P \times Q$  sets of common data points, i.e.  $Z_{p,q}$ . FOOS algorithm then computes the corresponding  $K$  medians from them to be considered as initial cluster-centres to perform the following and decisive iterative process of k-means over the dataset as in (3.2). Equations (3.5) and (3.6) explain the process of fusion.

$$\bigcup_{p=1}^P X_p = \bigcup_{q=1}^Q Y_q = I, \quad (3.5)$$

where  $P=K+1$ ,  $Q=K+2$ ,  $I$  is the set of all the instances in the dataset,  $X_p$  is one of the  $P$  clusters found in the first over-segmentation and  $Y_q$  is one of the  $Q$  clusters found in the 2<sup>nd</sup> over-segmentation.

$$\forall p, q, \exists Z_{p,q} = X_p \cap Y_q, \quad (3.6)$$

where  $1 \leq p \leq P$  and  $1 \leq q \leq Q$ . FOOS uses medians of the  $K$  largest sets among  $Z_{p,q}$  sets as the initial cluster-centres for the third and the last run of k-means.

As an example, figure 3.2(a) shows a synthetic image with  $256 \times 256$  pixels having four ( $K=4$ ) textures, D92, D55, D4, and D21 from the Brodatz album. Figure 3.2(b) shows its over-segmentation into five ( $P=5$ ) clusters and figure 3.2(c) shows its over-segmentation into six ( $Q=6$ ) clusters. Pixels in white and three grey levels of figure 3.2(d) show the four ( $K=4$ ) largest sets of common members, whereas the pixels in black form the smaller sets of common members among the  $P$ -cluster and  $Q$ -cluster over-segmentations.

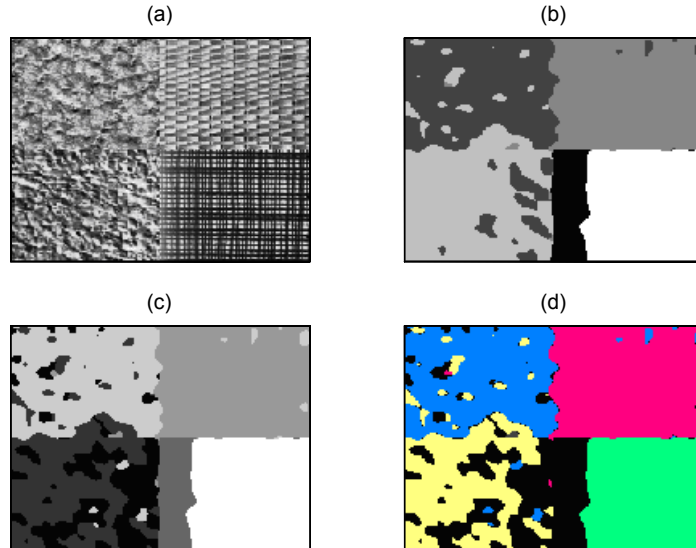


Figure 3.2: (a) The image with four textures from Brodatz collection, Segmentation into (b) 5 clusters (c) 6 clusters (d) The 4 largest classes of common member pixels

To segment the shown image into four clusters, the four initial cluster-centres come from the four largest sets of common members. Table 3.1 shows the statistics of the  $P \times Q$  sets, i.e. 30 sets of common members. The numbers in bold indicate the largest sets of common members. The largest one is the intersection of class 3 from P-cluster segmentation and class 4 from Q-cluster segmentation.

Table 3.1: Statistics of the clusters formed after over-segmentation of the image in 5 clusters (left column) and 6 clusters (top row). The other cells in the table show the number of common pixels ( $N_{pq}$ ) found in the 2 over-segmentation, with the highest 4 numbers in bold

		Segmentation into Q (k+2) 6 clusters						
		Class #	q	1	2	3	4	5
Segmentation Into P (k+1) 5 clusters	p	$\ Y_q\ $	7395	11245	3656	16388	14075	12777
		$\ X_p\ $						
	1	3660	13	0	3647	0	0	0
	2	16710	2598	96	0	52	<b>13964</b>	0
	3	16405	38	10	0	<b>16324</b>	33	0
	4	15975	4746	<b>11135</b>	5	12	77	0
5	12786	0	4	4	0	1	<b>12777</b>	

### 3.2.4 Computational complexity

The computational complexity of k-means happens to be

$$C_{kmeans} = O(KtN), \quad (3.7)$$

where  $K$  represents the number of clusters to be found,  $t$  is the number of iterations performed, and  $N$  is the cardinality of the dataset to be clustered. [Eduardo 2005] argues that since  $\{K, t\} \ll N$ , the computational complexity in (3.7) can be approximated by

$$C_{kmeans} = O(N). \quad (3.8)$$

Since FOOS is equivalent to three successive executions of k-means to perform clustering into  $K+1$ ,  $K+2$ , and  $K$  clusters respectively, its computational complexity becomes

$$C_{FOOS} = O((K + (K + 1) + (K + 2))tN) \quad (3.9)$$

and considering that  $3Kt \ll N$ , approximates to

$$C_{FOOS} \cong O(3KtN) \approx O(N), \quad (3.10)$$

which tells that the computational complexity of the two methods happens to be the same. Computationally, FOOS takes fewer iterations than 3 times those taken by a single run of k-means, ensuring that it is almost never more expensive than three runs of k-means.

One finds that the contemporary methods devised for clustering problem have much higher computational complexity. For example, (3.11) estimates the computational complexity of the global k-means [Likas 2003] that suggests clustering the dataset into from 1 to  $N$  clusters.

$$C_{gkm} = O\left(\sum_{k=1}^N KtN\right) \quad (3.11)$$

When  $K$  approaches  $N$ , (3.12) approximates the computational complexity for a single iteration, and predicts the *curse of cardinality*. Moreover, for a number of classes/clusters that is close to reality, the k-means converges quickly. On the other hand if executed to workout incorrect number of clusters (much higher than the actual number of classes), it becomes less probable and more difficult for k-means to converge that usually requires unusually higher number of iterations.

$$C_{gkm} \cong O(N^2) \text{ as } k \rightarrow N \quad (3.12)$$

The conventional k-means with multiple restarts too is obviously computationally more demanding than FOOS, because the number of trials is known, i.e. only three, in case of the later, whereas it is a function of cardinality in case of the former.

### 3.2.5 Comparison

Following sections present the results of segmenting various image and non-image datasets. All the three methods, the single-run k-means implemented in the author-written matlab® program, k-means with multiple restarts implemented in Gene Cluster, and FOOS segment the datasets. In the following sections, unless specified otherwise, the three algorithms segment the images in the feature space of texture signatures from 30 Gabor filters [Manjunath 1996].

### 3.2.5.1 Synthetic texture images

Since it is easy to numerically evaluate the results of segmenting the synthetic images, table 3.2 shows seven results on the collages of texture images from Brodatz album. Each collage contains four textures and has the size of  $256 \times 256$  pixels. The three clustering algorithms segment the images into four clusters. Figure 3.3 shows the first one of these collages; along with the segmentation-results obtained using a single-run of k-means algorithm, the Gene Cluster, and the FOOS. In case of Gene Cluster, apart from accuracy, the table also lists the number of times it finds the clustering solution before reaching an acceptable solution.

The overall accuracy averaged over the seven results show that the FOOS outperforms both, the single-run k-means and the Gene Clusters that takes on 172 runs on the average against 3 runs of the FOOS.

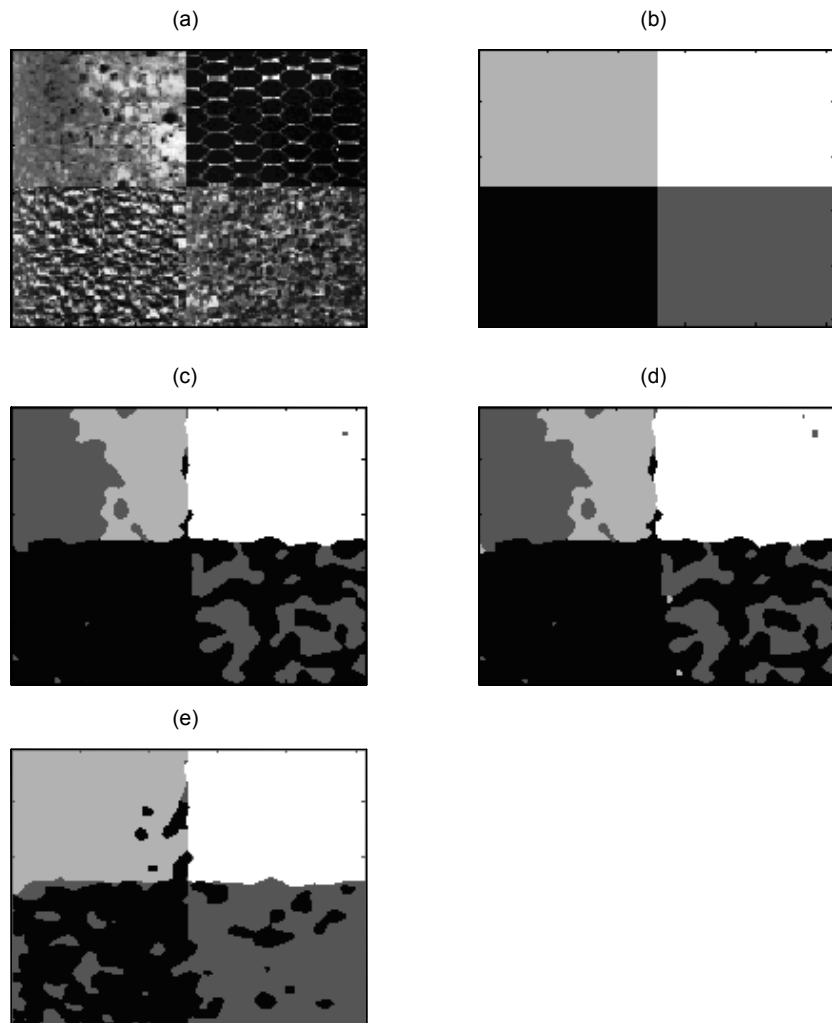


Figure 3.3: (a) The image with 4 textures from Brodatz collection, (b) its ideal segmentation, segmentation result with (c) k-means, (d) Gene Cluster, (e) FOOS algorithm

Table 3.2: The list and details of the images with a collage of four Brodatz textures each: The last column shows how many times the Gene Cluster found clusters using k-means.

S. No.	Textures Selected (From top left to bottom right)	Segmentation Accuracy			
		K-Means	FOOS (3 runs of k-means)	Gene Cluster 3.0 Accuracy	No. of Runs
1	D73, D34, D57, D29	69.5%	88.8%	69.2%	274
2	D4, D55, D9, D21	67.6%	97.0%	96.9%	178
3	D24, D84, D4, D21	67.8%	95.4%	95.4%	453
4	D92, D55, D4, D21	67.5%	81.1%	67.2%	029
5	D3, D22, D112, D80	60.4%	91.5%	91.4%	192
6	D54, D84, D57, D100	52.3%	69.8%	69.5%	028
7	D73, D37, D57, D29	55.0%	73.2%	55.0%	050
Overall Average Accuracy		62.8%	85.3%	77.8%	172

### 3.2.5.1.1 Natural digital photos

Apart from the synthetic images, here are the results on natural photos to allow subjective assessment of the clustering algorithms. The natural photos come from two online image databases, i.e. Berkley Segmentation Database<sup>12</sup> (BSD) [Martin 2001] and the Defence Image Database<sup>13</sup>. The author sought the required prior explicit permission of publishing the images from the intellectual property rights section of Defence Image Database. Refer appendix A.

All the images from BSD measure 481×321 pixels. These images come with segregation by more than one human subjects, thereby called users. These segmentations by human subjects help compare the results from the three clustering methods.

<sup>12</sup> Source: <http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/>

<sup>13</sup> Photograph by: .....; © Crown Copyright/MOD, image from [www.photos.mod.uk](http://www.photos.mod.uk).  
Reproduced with the permission of the Controller of Her Majesty's Stationery Office

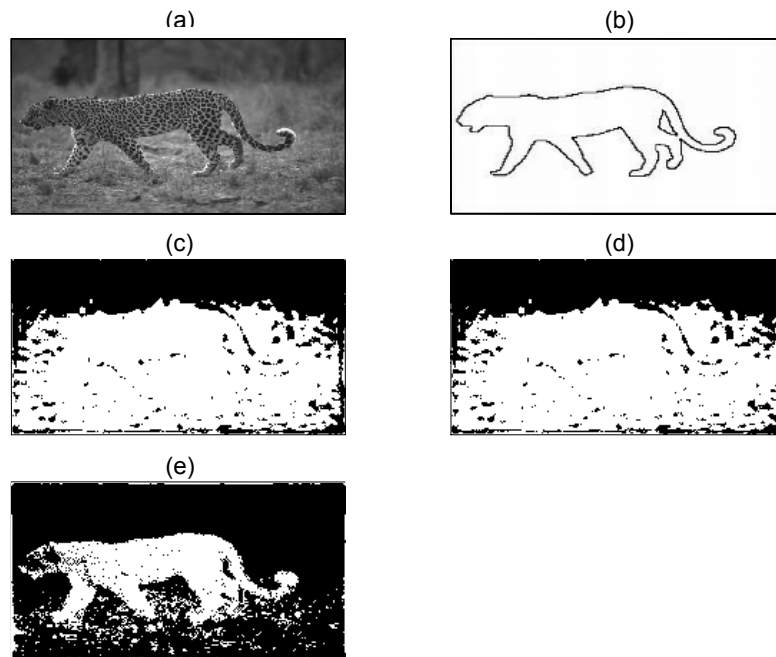


Figure 3.4: (a) A natural photo of leopard from BSD, its segmentation by (b) a human subject, (c) k-means, (d) Gene Cluster, (e) FOOS

Figure 3.4(a) shows a photo of leopard from BSD. Figure 3.4(b) shows its segmentation by a human subject (user# 1124), outlining the leopard. K-means, Gene Cluster, and FOOS go for finding two clusters from the 30 filter outputs with the aim of delineating the leopard from the rest of the image. Figure 3.4(c) and figure 3.4(d) show that k-means and Gene Cluster fail in accomplishing the task, whereas FOOS succeeds to a high extent.

Figure 3.5(a) shows another natural photo from the same database (BSD). As shown, the image apparently has three textures, 1<sup>st</sup> one that of the grass, 2<sup>nd</sup> one that of the zebras, and 3<sup>rd</sup> in the rest of the image, mainly the non-grass background. Figure 3.5(b) illustrates its segmentation (into 36 segments) by a human subject (user #1107) as given in the BSD benchmark, clearly outlining the zebras, the grass, and rest of the image (the non-grass background). It should be noted however that the segmentation by human subjects or otherwise provided at BSD website are with regard to the edge detection only, and not with regard to the texture.

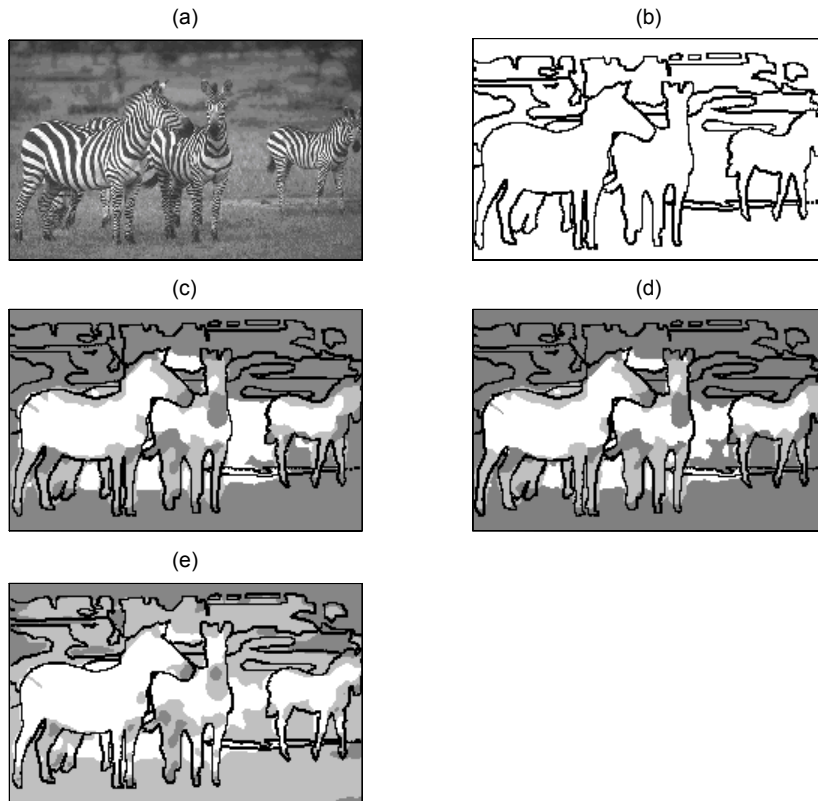



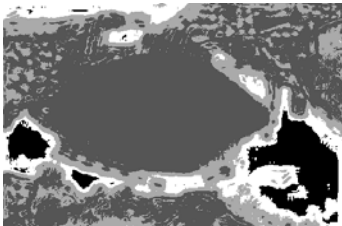
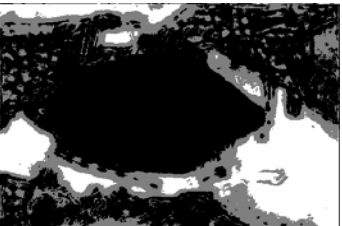
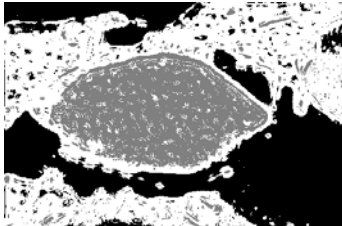
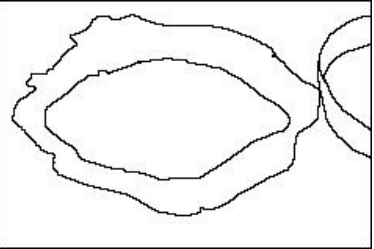




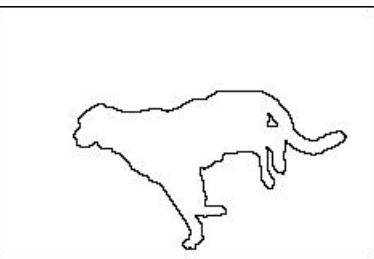



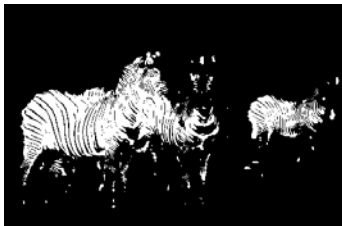
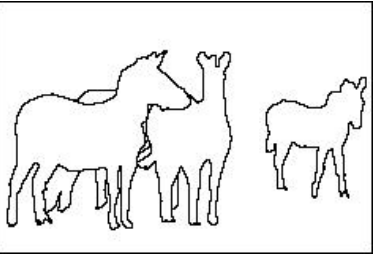
Figure 3.5: (a) The zebra image from BSD, its segmentation (b) as provided by BSD, (c) with k-means, (d) with Gene Cluster, (e) with FOOS

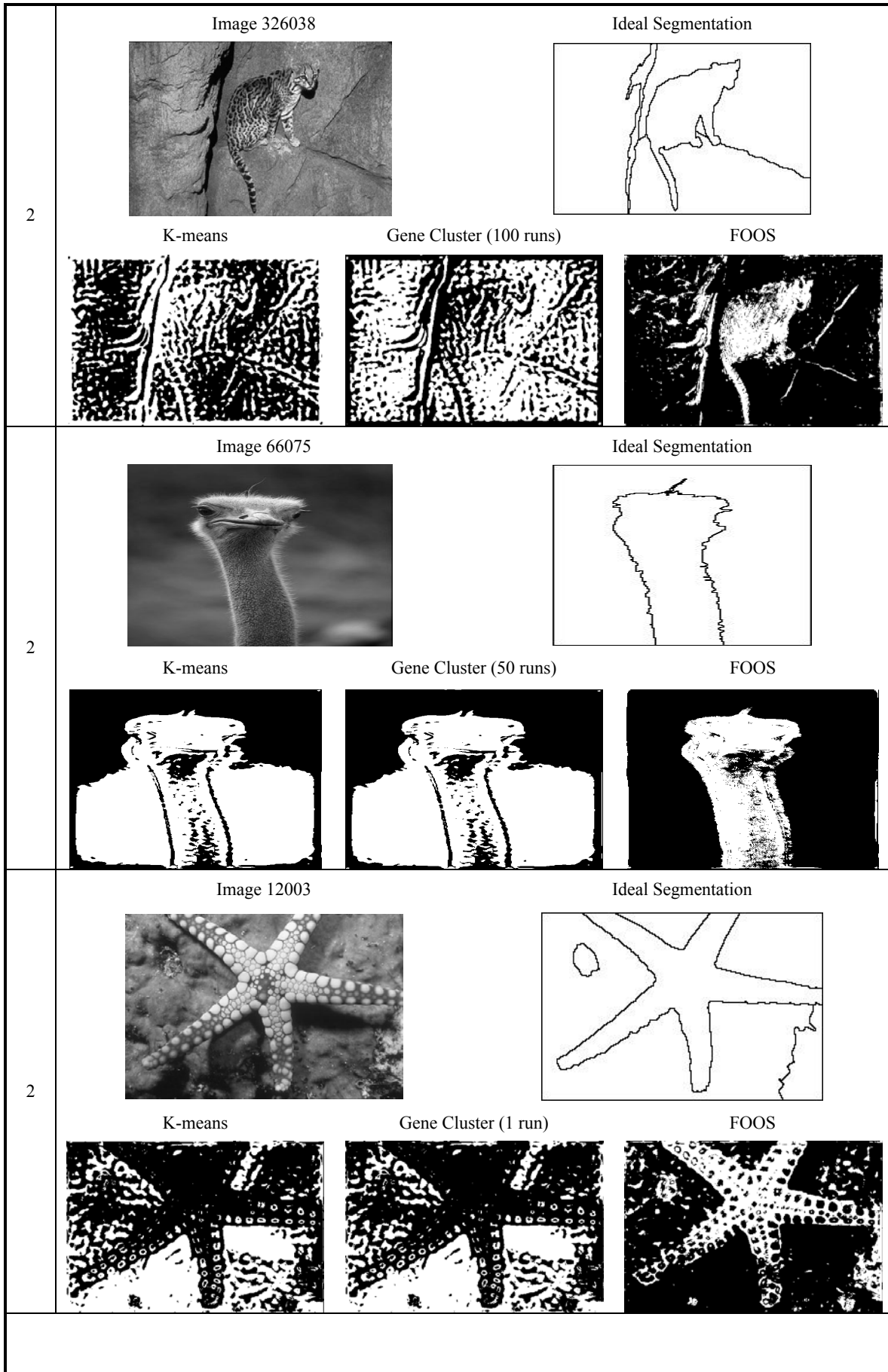
Figure 3.5(c) and 3.5(d) show the result obtained with k-means and Gene Cluster, respectively, with the outline of figure 3.5(b) overlaid on them. Figure 3.5(e) shows the result obtained with FOOS. The Gene Cluster performed 472 k-means processes and took hours to complete. The FOOS seems to have outperformed the other two methods by successfully separating the zebras, the grass, and the non-grass background. Having no texture, the mouths and the feet of the zebras also appear different from the zebra skin. The remaining misclassifications apparently point out the reduced discrimination ability of the extracted features. On the other hand, the k-means and Gene Cluster fail in distinguishing between the grass and the non-grass background, forcefully creating a class within the zebra texture and classifying legs as different from zebra's main body. The k-means and Gene Cluster also misclassify the non-occluded part of the otherwise occluded zebra's back.



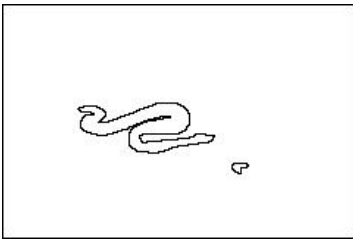
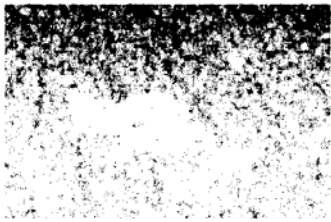



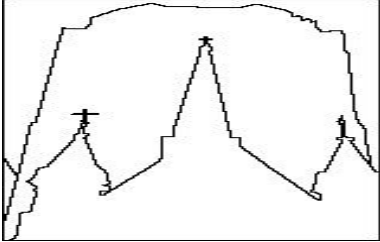
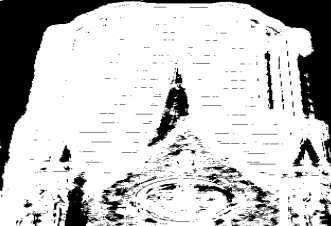
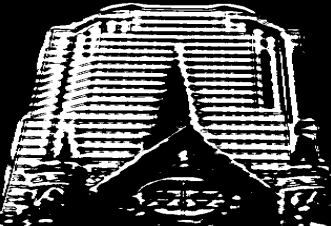


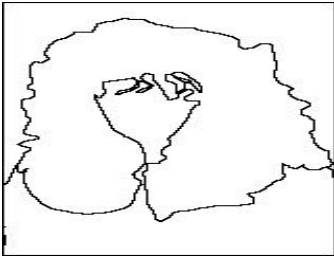


Table 3.3 shows the results on ten more images from Berkley segmentation database. The first column of the table indicates the number of clusters found in the corresponding image.

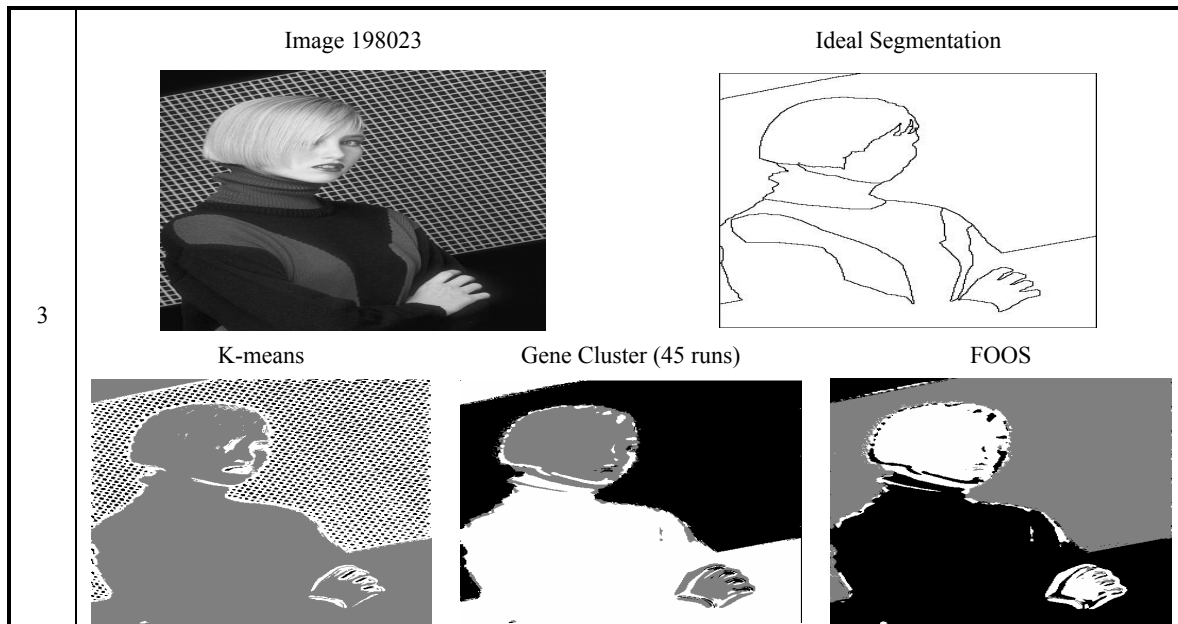


Table 3.3: Ten more segmentation results on the images from Berkley segmentation database

NO C	Image 58060	Ideal Segmentation
3	 <p data-bbox="435 544 531 573">K-means</p>  <p data-bbox="738 544 962 573">Gene Cluster (45 runs)</p>  <p data-bbox="1185 544 1249 573">FOOS</p> 	
2	 <p data-bbox="435 1137 531 1167">K-means</p>  <p data-bbox="738 1137 962 1167">Gene Cluster (20 runs)</p>  <p data-bbox="1185 1137 1249 1167">FOOS</p> 	
2	 <p data-bbox="435 1738 531 1767">K-means</p>  <p data-bbox="738 1738 962 1767">Gene Cluster (49 runs)</p>  <p data-bbox="1185 1738 1249 1767">FOOS</p> 	



2	<p style="text-align: center;">Image 196073</p> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;">  <p>K-means</p>  </div> <div style="text-align: center;"> <p>Ideal Segmentation</p>  <p>Gene Cluster (39 runs)</p>  <p>FOOS</p>  </div> </div>
2	<p style="text-align: center;">Image 277095</p> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;">  <p>K-means</p>  </div> <div style="text-align: center;"> <p>Ideal Segmentation</p>  <p>Gene Cluster (13 runs)</p>  <p>FOOS</p>  </div> </div>
2	<p style="text-align: center;">Image 198054</p> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;">  <p>K-means</p>  </div> <div style="text-align: center;"> <p>Ideal Segmentation</p>  <p>Gene Cluster (50 runs)</p>  <p>FOOS</p>  </div> </div>










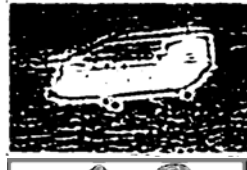














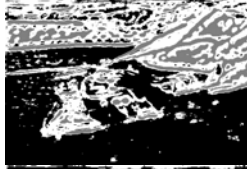

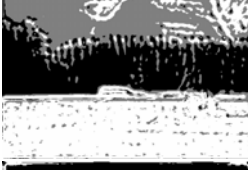

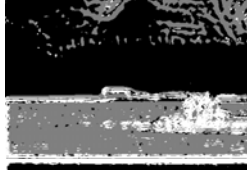








In case of image 58060, image 134008, image 253027, image 326038, image 66075, and image 12003, FOOS clearly outperforms single-run k-means and Gene Cluster. In the first one, only FOOS succeeds in separating the grain from the background. In the second one, only FOOS succeeds in delineating the leopard from the background. In the third one, only FOOS succeeds in outlining the cat from the background. In the fourth image again, only FOOS succeeds in demarcating the bird from the background. In case of image 196073, FOOS performs equally well as single-run k-means but Gene Cluster fails miserably. Interpretation of the results for images 198054, 277095, and 198023 depends on the subjective assessment.

Table 3.4 shows the results on nine images from the Defence image database. These images have varying sizes. Therefore, all the images are resized/re-sampled to  $321 \times 481$  pixels. The images were also converted to grey-scale images before extracting the texture features. The first column of table 3.4 indicates the number of clusters found and in the braces, the number of runs Gene Cluster goes for. The Defence image database did not grant us the permission to reproduce the third image; therefore, the table lists its reference number instead. Refer appendix A on page 113.

For the first three images, all the three methods yield the same outcome. However, Gene Cluster took 42 iterations before arriving to the clustering solution of the first image, whereas FOOS assures the solution always in only three runs of k-means. Interpretation of results of the next four images depends on the subjective assessment and/or application. In the third image from the bottom, FOOS fails in discriminating the trees from the hills, whereas the single-run k-means and Gene Cluster succeed in doing so. On the contrary, FOOS succeeds in separating two cyclists from the background, whereas the single-run k-means and the Gene cluster fail to do so. The results of the last two images demonstrate that the FOOS clearly outperforms the single-run k-means and the

Gene Cluster. In case of the last image, even the clustering solution of single-run k-means is better than that of the Gene Cluster.

Table 3.4: The segmentation results on five images from Defence Image database

NO C	Image	K-means	Gene Cluster	FOOS
2 (42)				
2 (2)				
3 (5)	<a href="#">45147231</a>			
2 (5)				
3 (30)				
3 (1)				
3 (4)				
2 (28)				
2 (22)				

### 3.2.5.1.2 The remote-sensing images

Here are some results on the remote sensing (RS) image acquired from the QuickBird satellite having the ground resolution of 70 cm per pixel. Chapter 4 gives more details on this RS image. Gabor filter-based texture features used here come from the same filters except the change of filter size that is  $15 \times 15$  pixels instead of  $31 \times 31$  pixels. This is because of the small size of the individual regions of the homogeneous texture in the RS image. The three clustering algorithms, i.e. single-run k-means, Gene Cluster, and FOOS now cluster different regions cropped into sub-images measuring  $256 \times 256$  pixels and having different number of classes.

Figure 3.6 (a) shows a sub-image from the RS image, and figure 3.6(b) shows its ground truth. It comprises three land-covers, namely, truck crops, wine, and old orchards. The ground truth also shows the uncultivated parts on the frontiers between any two land-covers.

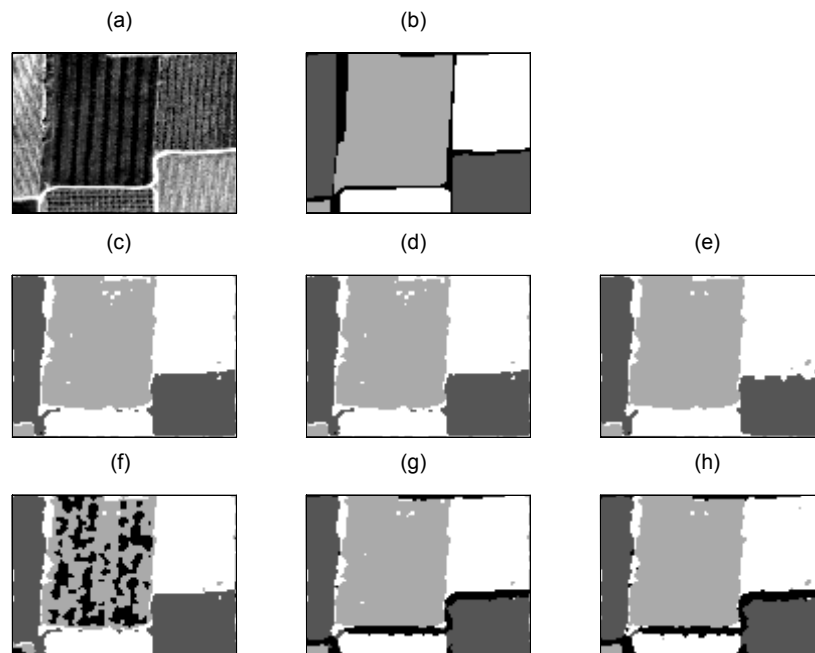


Figure 3.6: The results of segmenting (a) remote sensing image with (b) ground truth into 3 classes using (c) k-means (d) Gene Cluster, (e) FOOS, and into 4 classes using (f) k-means (g) Gene Cluster, (h) FOOS

Since the ground truth has three classes, the single run k-means algorithm, the Gene Cluster, and the FOOS segment this image in three clusters. As can be seen, apart from some misclassifications around the borders, more or less all the algorithms perform the same. However, result from FOOS appears a little better, since it has more errors only on the border between the two land-covers on the right, but has fewer errors in the regions of homogeneous land-covers. In case the user is not sure about the exact number of the clusters, and he segments into four clusters, even then the Gene Cluster and the FOOS find a meaningful segmentation, with the uncultivated borders between the land-covers as the fourth cluster. On the other hand, single run k-means fails to do so. Gene Cluster performs the k-means for five times.

Figure 3.7 shows another sub-image from the same remote sensing image. It comprises three land-covers, namely young orchards in the top-right, wine in the bottom-left and old orchards, but the image has two possible ground truths. In the first one, shown in figure 3.7(b), there is a single class of old orchards in the top-left and the bottom-right corners. However, in the second possible ground truth, shown in figure 3.7(c), the class of old orchards divides into two. The ones on the bottom-right have grassy ground and the ones on the top-left corner have the bare ground, hence the second ground truth has four classes instead of three. The two parts of the old orchards are also different in terms of vegetation density and the size of the trees. The middle row of figure 3.7 shows the segmentation into three classes, whereas the bottom row of figure 3.7 shows the segmentation results into four classes.

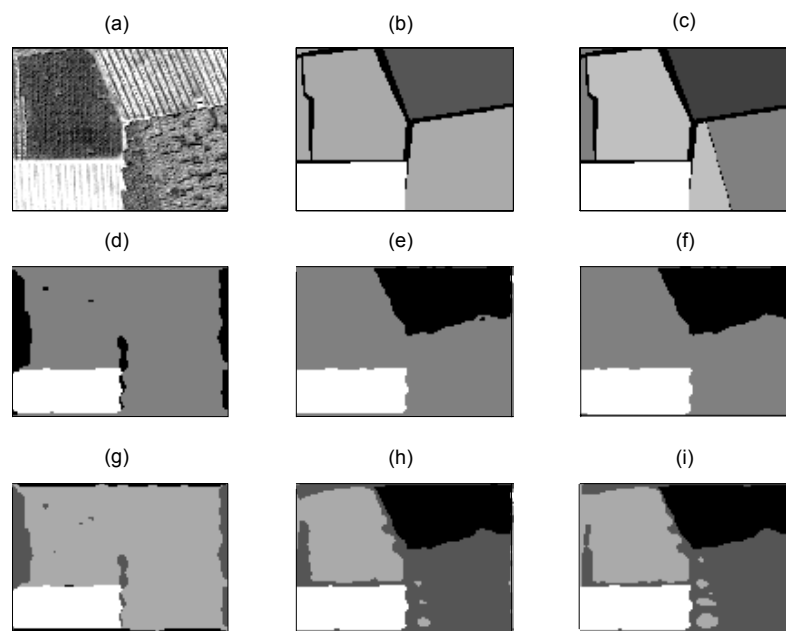


Figure 3.7: (a) an RS image (b) ground truth with 3 land-covers, (c) ground truth with 4 land-covers, segmentation into 3 clusters using (d) k-means (e) Gene Cluster, (f) FOOS, segmentation into 4 clusters using (g) k-means, (h) Gene Cluster, (i) FOOS

In case of three classes, k-means fails in discriminating between the old orchards and the young orchards as can be seen in figure 3.7(d), and achieves only 70.6% accuracy. On the other hand, FOOS achieves 97.1% accuracy. Gene Cluster also performed marginally worse than the FOOS, achieving 97.0% accuracy after 446 runs of k-means.

In the case of four classes, the k-means once again fails in discriminating the young orchards from old ones and achieves on 52% accuracy. On the other hand, FOOS not only successfully discriminates between the old and the young orchards but also between the two kinds of the old orchards as well, thereby achieving 85.0% accuracy. Refer figure 3.7(i). The result of Gene Cluster, shown in figure 3.7(h) is slightly worse than FOOS, achieving 84.7% accuracy after 366 runs of k-

means. In this example, k-means (single run) took 43 iterations, whereas FOOS took 93 iterations, i.e. much less than three times those of single-run k-means.

### 3.2.5.2 Non-image datasets

Following are the results on two datasets from UCI machine learning repository<sup>14</sup> [Murphy 1994]. As seen in the results on the image data, the single-run k-means may result in a very good or a very bad clustering solution depending on its random initialisation. Hence, following results no more consider the single-run k-means.

#### 3.2.5.2.1 A dataset with 2 classes

This is a dataset of human body dimensions, whose original source is [Heinz 2003]. The dataset includes 507 observations of 21 body dimensions as well as age, weight, height, and sex. The subjects include 247 male and 260 female individuals. The feature of sex was removed from this dataset. Further, all the feature values were rescaled from -250 to 250 to ensure that all the feature values receive the equal weight in the distance measurement. The dataset was segmented into 2 clusters corresponding to the 2 sexes. Evaluation phase uses the removed column of sex as the ground truth. The results from Gene Cluster and FOOS are exactly the same, i.e. 87.4%, with 443 instances classified correctly. Very small number of classes makes the task quite easy; therefore, results ought to be the same. However, Gene Cluster performed 9216 runs of the k-means before reaching the solution, against three runs by FOOS.

#### 3.2.5.2.2 A dataset with 10 classes

Originally, E. Alpaydin and C. Kayna provided this dataset. The dataset contains 2000 instances corresponding to hand-written digits “0” to “9”. The class distribution is uniform, viz. 200 instances of each class. The number of dimensions is 649. Gene Cluster stopped after performing just a single run of k-means, and as suggested, it was therefore run repeatedly four times. Each time, it stopped after performing a single run of k-means, yielding accuracies of 80.45%, 80.15%, 80.55%, and 80.60%, respectively. FOOS ran four times, each time with a different value of Q, as illustrated in table 3.

Table 3.5: FOOS Results with different over-segmentations

P	Q	Over-segmentation	Accuracy
11	12	20%	75.95%
12	13	30%	83.15%
13	14	40%	91.70%
14	15	50%	91.15%

<sup>14</sup> <http://mllearn.ics.uci.edu/MLSummary.html>



The last three results, i.e. corresponding to 30-50% over-segmentations are better than that of Gene Cluster. The best result is the one corresponding to 40% over-segmentation, with an accuracy of 91.7%, i.e. 11% more than that of Gene Cluster.

[Breukelen 1998] reports that using neural network classifiers (NNC), even the learning error was not less than 10%, which corresponds to accuracies of below 90%! It further states that the combined NNC failed in distinguishing between class “6” and class “9”. The clusters resulting from FOOS show that class “6” is well separated from class “9” with not a single member of class “6” classified as belonging to class “9” or vice versa; though both the clusters, i.e. cluster “6” and cluster “9”, have some 10% misclassifications.

### **3.3 Supervised classification**

The process of supervised classification has two major phases. The first one is *training*, in which one collects and organises the a priori knowledge of the classes to be identified; and the second one, in which the classifier uses that a priori knowledge to classify an instance as belonging to a specific class. The maximum likelihood, naïve Bayes, neural networks, and the nearest neighbour are among the most famous methods of supervised classification. Among these, the nearest neighbour classification is the simplest, and is used herein. The following sections explain the nearest neighbour classification and its variants.

#### **3.3.1 The nearest neighbour classification**

The NN classifiers are an example of instance-based learning [Kotsiantis 2007]. Unlike simple nearest-to-mean classification, an NN classifier keeps several representative samples from each class and the NN classification process compares each of the queries with every representative of all the classes. Depending on the application and nature of data, the number of samples representing a class may vary from class to class or all the classes may have equal representation. In case a class has much more variance in the feature space than the others, the NN classifier may choose to represent the class with more representatives than it chooses for the rest of the classes. Similarly, if a class has much less variance in the feature space than the others, it may get smaller representation accordingly.

The training phase of nearest neighbour classification consists only of building a lookup table that lists the instances from training data and their corresponding classes. In the testing/classification phase, for each query, similarity is determined with every entry in the lookup table resulting from the training phase. The class corresponding to the nearest entry from the lookup table is assigned to the query. Since the similarity of query must be measured with all the entries in the lookup table, the time classification takes is a function of the size of the lookup table.

The nearest neighbour classification is in fact a specific case of the k nearest neighbour (kNN) classification, in which k equals 1.

### 3.3.2 K nearest neighbours classification

Under this scheme of classification, from the lookup table of S entries, k nearest matches instead of one, are selected to decide upon the class of the query, where  $k > 2$  and  $k \ll S$ . When  $k=1$ , it becomes the nearest neighbour classification as discussed above. The simplest way to classify the query is the voting, i.e., to assign the class most frequent in the k nearest neighbours to the query. However, this causes undue favour to the most populous class in the lookup table that might be containing uneven number of instances for different classes. To alleviate this problem, each occurrence of a class among k neighbours is weighted by its corresponding degree of similarity. Hence, the similarity values of the closest neighbours belonging to a class simply add up and the classifier assigns the class with highest similarity value to the query.

### 3.3.3 Similarity criterion

Normally, the k-nearest neighbours (kNN) classification method uses the inverse Euclidean distance as the most common criterion of similarity. Inverse of Manhattan, Chebyshev, Camberra, distances are also used apart from Kendall's Rank Correlation and variations of the Minkowsky distance as noted by [Kotsiantis 2007]. However, repeated experiments with the LFH-based features show that the best accuracies result while using the similarity measure of the cross-correlation as follows:

$$\rho_{xy} = \frac{\sum_{n=1}^L (x_n - \mu_x)(y_n - \mu_y)}{(L - 1)\sigma_x\sigma_y}, \quad (3.13)$$

where  $x$  represents the feature vector of the instance to be classified,  $y$  represents the feature vector of the median of a cluster,  $\rho_{xy}$  represents the coefficient of cross correlation between  $x$  and  $y$ ,  $\sigma$  represents standard deviation,  $\mu$  represents the mean or the expected value, and  $L$  represents the length of the feature vector.

### 3.3.4 Editing the lookup table

It is known that the kNN classifier performs very close to much more sophisticated classifiers such as Naïve Bayes' classifier. To keep the classification time to acceptable limits, the training data is either minimised or the larger lookup table from full-length training set is edited. The process of editing selects only those points in the feature space that are crucial in decision making, i.e. which define the very frontiers between the classes. If the total number of classes is quite limited and the lookup table is reasonably large, the process of editing is skipped. However, it is usually preferable to have larger amount of data to train the classifier and then to have a smaller lookup table after

editing. The most popular method of editing the kNN classifier is the Voronoi diagram [Inaba 1994]. However, [Belongie 2002] proposed using k-medoid, a variant of k-means clustering for editing the lookup table of the nearest neighbour classifiers. Texture recognition and retrieval experiments with the dataset made from Brodatz album presented in chapter 2 use FOOS clustering for editing the lookup table of the kNN classifier. However, instead of using the usual similarity criterion of inverse Euclidian distance, it uses the similarity criterion based on cross correlation coefficient, as explained in the previous section. The experiments of kNN classification in chapter 2 use 100 training points per class and 10 centroids per class after editing with FOOS, whereas  $k=1$ .

### **3.4 Conclusion**

This chapter studied the methods of supervised and unsupervised classifications of images with the texture signatures. The supervised classification comes to the use when the goal is to look for the known classes. In this case, the classifier needs training of the known classes before it can recognise the same classes as they appear in an image earlier unseen by the classifier. On the other hand, unsupervised classification is only a process of clustering an image in a given number of segments. In this case, the classifier does not know either the classes or the number of classes beforehand. The classifier goes for finding an approximate number of classes, generally bigger than the actual number of classes, in the image. The result is a number of clusters that comprise the pixels with similar characteristics.

This chapter also introduced an improved method of k-clustering, called FOOS. In most of the cases, FOOS outperforms the single-run McQueen k-means algorithm and the k-means with multiple restarts implemented in the computer program called Gene Cluster. It is also noticeable that while segmenting some of the natural digital photos, even the single-run McQueen k-means clustering implemented by the author performed as good as or outperformed Gene Cluster in clustering the natural images. This shows that either the maximum allowable intra-cluster distance worked out by Gene Cluster is over-estimated or the criterion of intra-cluster distance, also called Classification Error does not necessarily lead to the optimal solution. However, the unsupervised classification does not resolve the class identification of the clusters. Moreover, any two clusters might belong to the same class/land-cover type. Usually, the most interesting information in the unsupervised classification is the class delineation.

The succeeding chapter applies these methods of supervised and unsupervised classifications on the remote sensing images and merges the two outputs to obtain the land-cover classification that is much more accurate than the one obtained from supervised classification alone.



# 4

## Fusion of region and class information for classifying remote sensing images

4.1	The source RS images .....	88
4.1.1	Panchromatic image .....	88
4.1.2	Multi-spectral (MS) image .....	88
4.1.3	The land-covers .....	90
4.1.4	The training regions .....	91
4.1.5	The test regions .....	92
4.2	Segmentation .....	95
4.2.1	Template matching .....	95
4.2.2	Texture signatures .....	97
4.2.3	Spectral signatures .....	97
4.3	Classification .....	97
4.3.1	The training phase .....	98
4.3.2	The testing phase .....	99
4.3.2.1	Region classification .....	99
4.3.2.2	Grid classification .....	99
4.4	Fusion of the region and the class information .....	100
4.4.1	Scheme 1 .....	101
4.4.2	Scheme 2 .....	101
4.4.3	Discussion .....	102
4.5	Performance evaluation .....	103
4.5.1	Kappa statistics .....	103
4.5.1.1	Error matrix (EM) .....	103
4.5.1.2	User's accuracy (UA) .....	104
4.5.1.3	Producer's accuracy (PA) .....	104
4.5.1.4	Overall classification accuracy (OCA) .....	104
4.5.1.5	Product matrix (PM) .....	105
4.5.1.6	Expected classification accuracy (ECA) .....	105
4.5.1.7	Kappa .....	106
4.5.2	The resulting land-cover maps and their accuracy .....	106
4.6	Further improvements .....	112
4.6.1	Polygon filling .....	112
4.6.2	Two block-sizes .....	112
4.6.3	Feature-level fusion .....	113
4.7	Conclusion .....	113

Today, the remote sensing experts believe that no single source of information is sufficient for recognition of land-covers in a remote sensing image. This has made researchers propose different ways of fusing multiple sources of information on land-cover, such as spectral information and the textural information. Therefore, this chapter is about the fusion of spectral and textural information for land-cover mapping. The chapter presents results of land-cover mapping in a very high-resolution satellite-borne RS image using the methods of segmentation and classification presented in the preceding chapters. Spectral signatures from the vegetation indices drive the process of segmentation, whereas the texture signatures drive the process of classification. Finally, the chapter presents a novel way of fusing the two different pieces of information resulting from image classification and image segmentation, respectively.

The following sections describe the source RS images and the eight land-covers found therein. The succeeding sections explain the process of image segmentation, block-wise classification of

image, and the method of fusing the segmentation and the classification results. The last sections of the chapter propose some future work for further improving the results before concluding the chapter.

## **4.1 The source RS images**

The land-cover maps presented herein result from the fusion of information extracted from a panchromatic image and a three-band multi-spectral (MS) image remotely sensed by the sensor on board QuickBird satellite captured simultaneously in July 2006 over the region near Nîmes in France. The geographical coordinates of the top-left, i.e. NW corner of the image are  $4^{\circ} 24' 44.687''$  E (longitude), and  $43^{\circ} 42' 33.22''$  N (latitude); or Universal Transverse Mercator (UTM) coordinates: 613790.8/4840547.6. The coordinates of the centre of the imaged area are approximately  $4^{\circ} 25'$  E and  $43^{\circ} 42'$  N. The images have the radiometric resolution of 8 bits, and have been radiometrically corrected as per coefficients of the calibration provided in the metadata accompanying the images, but no orthorectification has been carried out on the images.

Mrs. Camille Lelong of CIRAD<sup>15</sup> generously and graciously provided not only these RS images including the panchromatic and the multispectral images for this research, but also the corresponding ground truth collected through field survey.

### **4.1.1 Panchromatic image**

The spatial resolution of the panchromatic (PAN) image is 70 cm/pixel, and has the dimensions of  $2122 \times 2411$  pixels, i.e. an area of 5116142 pixels corresponding roughly to a surface area of 2.3 km<sup>2</sup>. However, this area includes a triangular region of no information as well. PAN image is source of textural information in this research work, but it has some places of saturation despite radiometric correction. The panchromatic image, as received in tif format after the pre-processing, contains only 43 grey levels. This has possibly removed the information content that could be useful in discrimination of the land-covers. Picture 4.1 shows the PAN image.

### **4.1.2 Multi-spectral (MS) image**

The MS image has three bands including near infrared, red and green. Originally, the MS image-bands have the spatial resolution of 2.8 meters per pixel, but the pixel-level fusion of these MS bands with the panchromatic image using Brovey transform [Wang 2005] brings the resolution of the former up to that of the later. Therefore, both of the RS images used herein have the identical spatial resolution of 70 cm per pixel. Picture 4.2 shows the MS image using the false colour composite, assigning NIR, red, and green bands to R, G, and B channels, respectively, of the

---

<sup>15</sup> CIRAD – maison de télédétection – 500 Rue JF Breton Montpellier, France

display. Therefore, the redder parts in the MS image correspond to more densely vegetated areas that usually have higher spectral response to the NIR frequencies.



Picture 4.1: The panchromatic remote sensing image captured by QuickBird



Picture 4.2: The MS image in false colour composite

### 4.1.3 The land-covers

The eight known classes and a non-class shown in table 4.1 cover this remotely sensed area. As can be seen, some classes are bigger in population than the others are, with the largest class being class 3, *the crop fields*, with the total population of 1090856 pixels and the smallest class being class 6, *the apple trees* with the total population of 44194 pixels. Figure 4.1 shows the ground truth of the remotely sensed area of picture 4.2. The non-class covers the triangular area of no information on the bottom right part of the image, the civil structures like roads, chambers, and the borders between the regions of the eight known land-covers. The pie chart of figure 4.2 shows the percentile population size of each of the eight classes and the non-class. The land-cover of truck crops also includes the greenhouses as well. Two greenhouses are clearly visible close to the triangular region of no-information near the bottom right corner of image. The third greenhouse is situated just below the top-left corner of the image. The manmade structures in the image also include a principal road that passes from top to bottom of the image, and along the twin greenhouses, dividing it into two approximately equal parts.

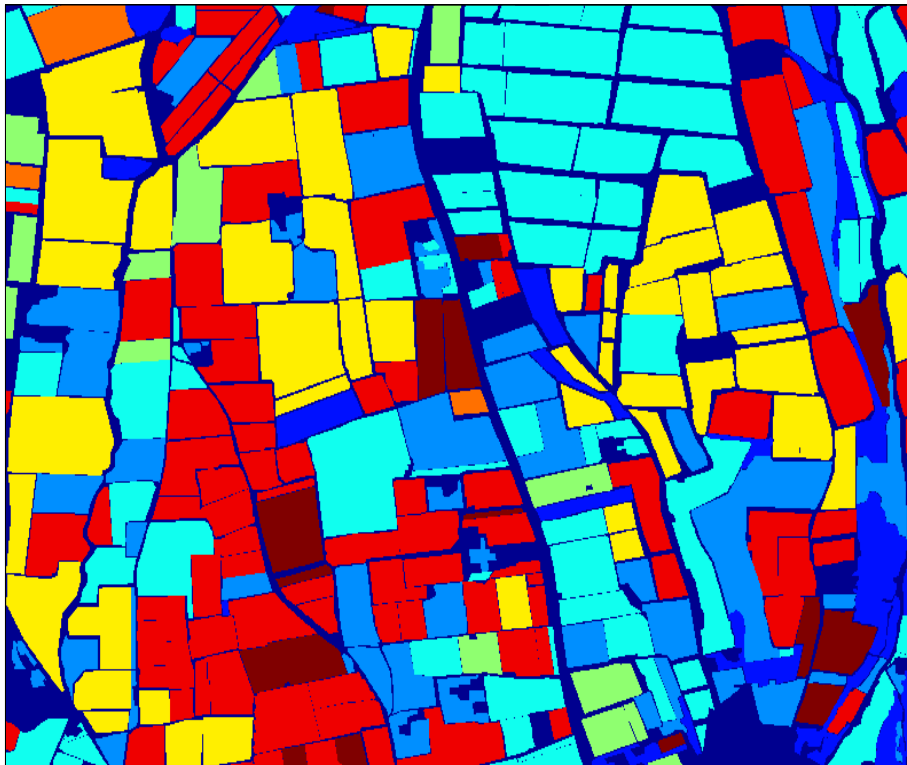


Figure 4.1: The ground truth of the image



Table 4.1: The land-covers in the source RS image

S. No.	Class		Colour	Population
	English	French		
1	Forests	Forêts	Navy blue	238097
2	Fallow	Friche	Blue	650099
3	Crop fields	Champ	Cyan	1013992
4	Truck crops	Cultures maraîchères	Green	155475
5	Vineyards	Vignes	Yellow	896382
6	Apple trees	Pommiers	Orange	44194
7	Old Orchards	Vieux Vergers	Red	1052101
8	Young Orchards	Jeunes Vergers	Brown	181810
9	Non-class	Non-classe	Black	883992
Total				5116142

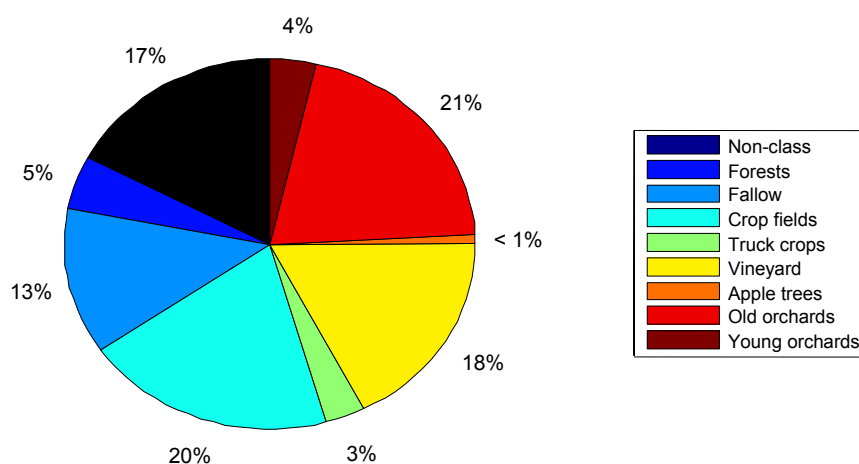


Figure 4.2: Percentile population of the eight classes and the non-class

#### 4.1.4 The training regions

With the availability of only one image containing the eight classes, there does not remain a lot to have separate images for training and testing. Therefore, test images and the training images come from the same source image. Since a given land-cover appears apparently different from one region of its occurrence to other, the training from one region only becomes insufficient. Therefore, the classifier trains; i.e. learns the eight known classes from the training regions scattered throughout the image, because each of the known classes varies not only in grey scale and texture, but also from the agricultural point of view. For example, there can be several sub-classes within the class of the fields. Therefore, the training regions are quite small and are selected carefully to cover every variation of each class. One of the training regions is as narrow as 28 pixels wide. Since the size of the selected training regions is quite small, the training area covers a little proportion of the total area of the RS image. The filled rectangles in figure 4.3 indicate the location of the training regions as scattered throughout the remotely sensed area. The colour of the regions represents its corresponding class/land-cover.

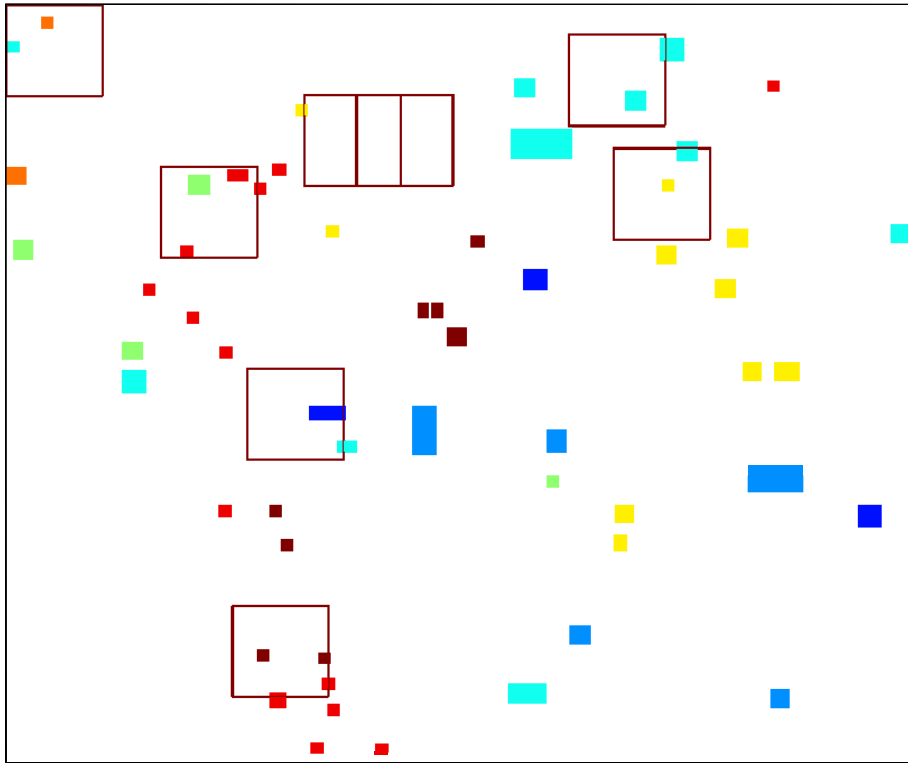


Figure 4.3: The regions selected for classifier’s training (filled rectangles) and the square test regions (hollow squares)

### 4.1.5 The test regions

To have several results and to perform several tests on the proposed scheme of land-cover mapping, several sub-images have been cropped from the source RS image. Table 4.2 gives a list and complete details of the eight sub-images cropped from the RS image and used as test images. All the selected regions measure 256×256 pixels. The large squares in figure 4.3 outline the locations of all the test regions in the RS image. The test images have from one to four land-covers, and have been selected so as cover all the eight classes. Figure 4.4 through figure 4.11 show the eight test images, corresponding to test region 1 through test region 8, respectively, of table 4.2, and their corresponding ground truths.

Table 4.2: Description of the test regions

Region No.	Coordinates				No. of Classes	Classes
	Up	Down	Left	Right		
1	84	339	1491	1746	1	3
2	400	655	1610	1665	2	3, 5
3	1678	1933	598	853	2	7, 8
4	452	707	410	665	3	4, 5, 7
5	1	256	1	256	3	3, 5, 6
6	253	508	791	1046	3	2, 5, 7
7	251	506	928	1183	3	2, 3, 7
8	1018	1273	638	893	4	1, 3, 5, 7

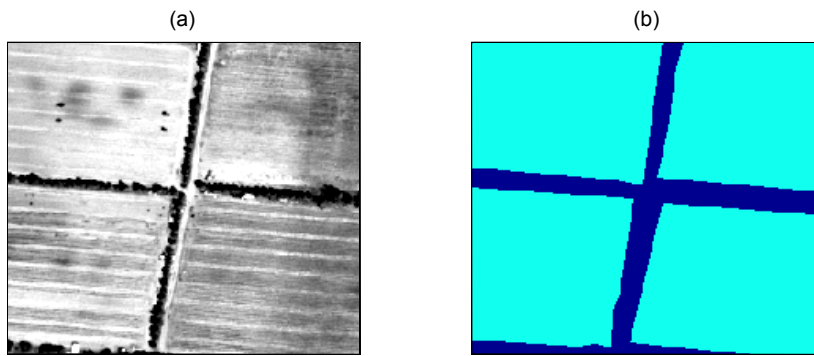


Figure 4.4: test region 1 and (b) its ground truth

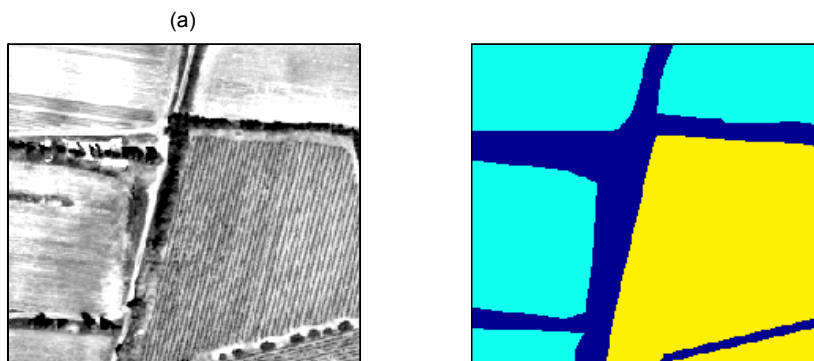


Figure 4.5: (a) The test region 2 and (b) its ground truth

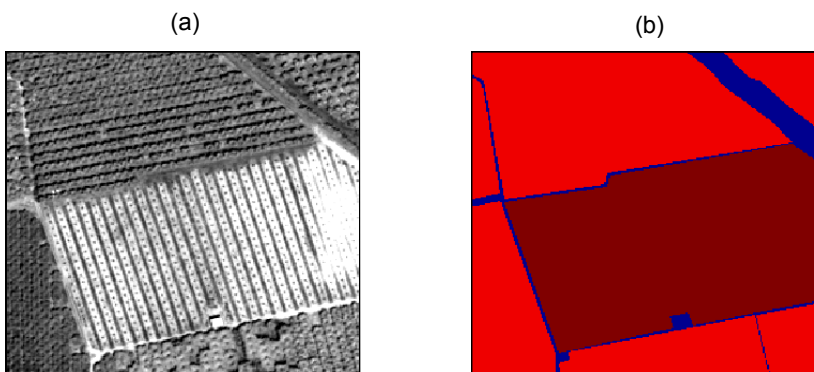


Figure 4.6: (a) The test region 3 and (b) its ground truth

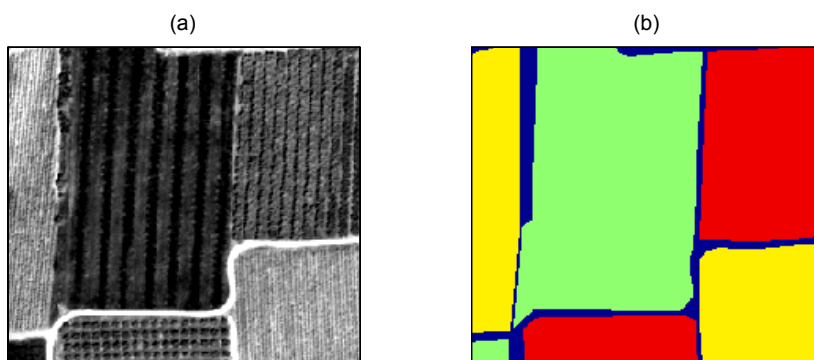


Figure 4.7: (a) The test region 4 and (b) its ground truth

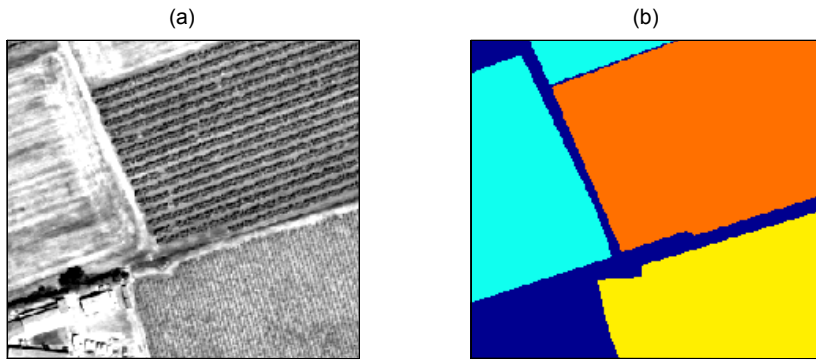


Figure 4.8: (a) The test region 5 and (b) its ground truth



Figure 4.9: (a) The test region 6 and (b) its ground truth



Figure 4.10: (a) The test region 7 and (b) its ground truth

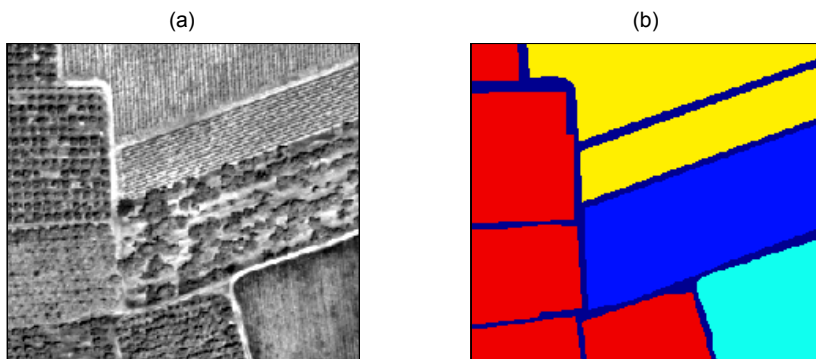


Figure 4.11: (a) The test region 8 and (b) its ground truth

## 4.2 Segmentation

Segmentation is the process of clustering the pixels of a test image into regions of homogeneous class/land-cover, whereas the number of regions can be greater than the number of classes found therein. The purpose of segmenting the test image is to identify major regions of homogeneous classes in the image so that any two regions with dissimilar land-covers become disjoint. To assure this, the proposed approach segments the images into a pre-decided number of clusters that is always more than actual number of classes. The exact number of classes, in a test image is unknown to the classification process, but the human expert always has an idea on what is the maximum possible number of classes in the area under remote sensing. The test regions selected herein always have four land-covers at the most; though a region of this much area may possibly have five land-covers at the most. Apart from this, a test region may have considerable size of the non-class as well. In some case, it may have two variations of a single land-cover. Therefore, all the experiments reported hereunder segment all the test regions into six clusters using the FOOS, clustering method, introduced in the preceding chapter.

This chapter presents three ways of finding the regions of homogenous classes, i.e. using template matching, texture signatures, and spectral signatures. All the three methods extract the multivariate attributes for each pixel, which a clustering algorithm exploits to segment the test images. The multivariate attributes from the last two methods are quite grainy and therefore need smoothing by a Gaussian filter before the clustering algorithm can process them. The land-covers of “*truck crops*” has the coarsest texture with 17 pixels wide texton. Therefore, the 17×17-pixel Gaussian filter smoothens the multivariate data resulting from the extraction of texture signatures. The third method uses the spectral signatures. The varying filter-widths provide a look at the spectral features on different scales. The larger is the filter, coarser is this scale and more is the border effect in the segmentation results; and smaller the Gaussian-averaging window, finer is the scale and grainier is the segmentation result. The window size of 9×9 pixels proves experimentally an acceptable and the optimal one in this trade-off. The standard deviation of the Gaussian filter used is  $\sigma=x/5$ , where  $x$  is the filter’s width. This standard deviation is found to give the optimal results.

### 4.2.1 Template matching

As there are only eight possible classes known a priori in a test image, the samples from the training regions can serve as templates to segment a test-image. Using the correlation found between the test image and each of these templates, FOOS segments a test image into the required number of clusters. The templates are the 17×17-pixel samples of the classes. This approach uses two samples per class. Because there are eight classes, the total number of templates becomes 16. Figure 4.12(a) shows the test region 3 and figure 4.12(b) shows its segmentation into six clusters

using template matching. Similarly, figure 4.13(a) shows the test region 4 and figure 4.13(b) shows its segmentation into six clusters using template matching. It takes little longer in segmenting this dataset. Therefore, this proves a little more expensive than using texture features for segmenting a test region. Although the template matching seems to work well, it does not guarantee a reasonable outcome with the rotated test images or if it encounters another image captured at another date. In addition, it exhibits severe border effects in some cases.

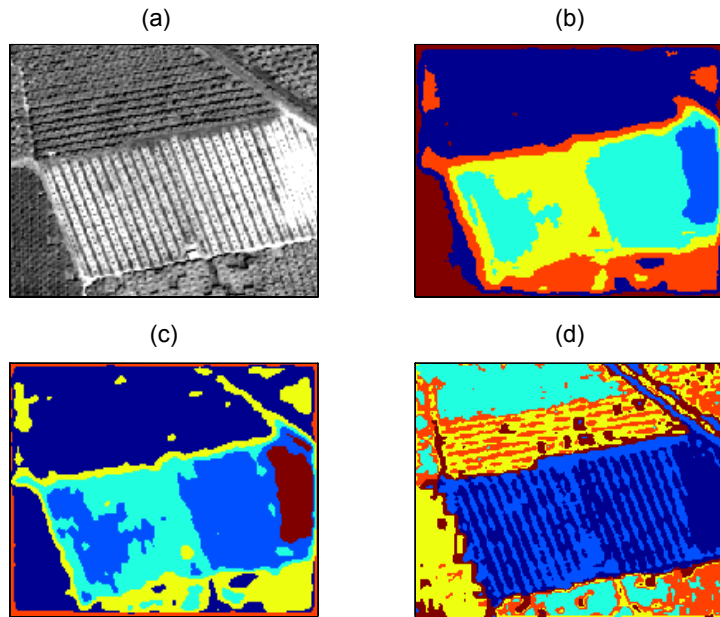


Figure 4.12: (a) The test region 3, its segmentation using (b) templates, (c) texture signatures, and (d) spectral signatures

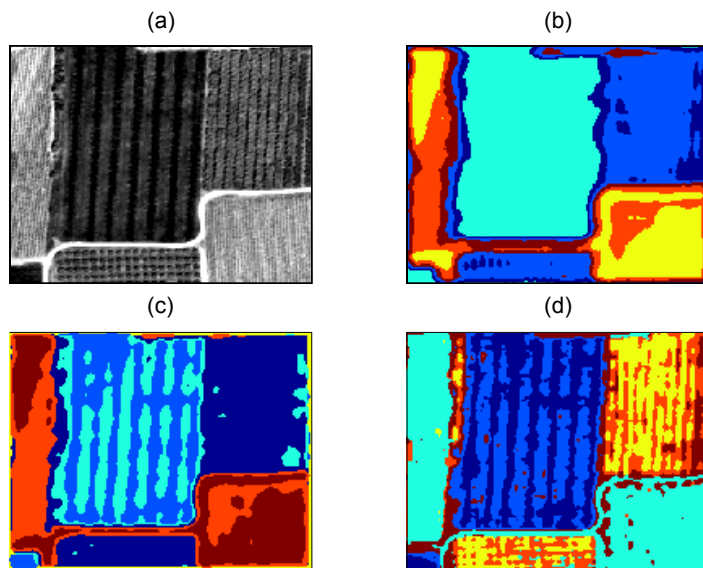


Figure 4.13: The test region 4, its segmentation using (b) templates, (c) texture signatures, and (d) spectral signatures

### 4.2.2 Texture signatures

The texture signatures based on DFT [Ursani 2008b], introduced in chapter 2, can also help segment a test image. The magnitudes of the DFT coefficients, i.e.  $|X_0|$ ,  $|X_1|$ ,  $|X_2|$ ,  $|X_3|$ , and  $|X_4|$  and the two phases  $\angle X_2$  and  $\angle X_3$  of the 8-pixel sequence around each pixel in the test image result in 7 images carrying DFT-based texture signatures. Figure 4.12(c) and figure 4.13(c) show the segmentation of test region 3 and test region 4, respectively, using the DFT-based texture signatures. It takes much less time to segment this dataset having 7 dimensions as compared to the time it takes to segment the 16-dimensions dataset generated using template matching. In addition, generating this dataset is comparatively cheaper than the template matching as well.

### 4.2.3 Spectral signatures

The vegetation indices NDVI, SAVI, MSAVI2, and NGRDI, introduced in chapter 1 are computed for each pixel of the test image from the NIR, R, and G bands of multi-spectral images. This results in four images carrying the spectral signatures. These four spectral signatures then segment the test image into six clusters using FOOS that takes lesser time as compared to the time taken in the two preceding methods. Moreover, generating this dataset is computationally the simplest one, since it only involves the basic arithmetic operations. Figure 4.12(d) and figure 4.13(d) show the segmentation of test region 3 and test region 4, respectively, using the spectral signatures. It should be noted that it was also tried segmenting the test regions using PAN image and the three bands of the MS image altogether, but it did not yield acceptable results.

Up to this point, a test image has only divided into the segments or regions. In order to have a complete land-cover map, it yet requires classifying each of the delineated regions as belonging to a specific class of a land-cover. Therefore, following sections explain the details of the classification process.

## 4.3 Classification

The k-nearest neighbour (kNN) classifier explained in the previous chapter assigns a specific land-cover to each of the delineated regions in a segmented test images. To determine the class of a region, the classifier uses LFH 64 texture-features extracted from the PAN image [Ursani 2008b]. Experiments show that the LFH 64 texture features described in section 2.3 of chapter 2 perform best with 32 grey levels [Ursani 2008]; therefore, the PAN image quantises to 32 levels. There are two phases of the process of classification, i.e. the training and the testing. The following sub-sections explain the way the two phases take place.

### 4.3.1 The training phase

The training of a k-nearest-neighbour classifier involves building a lookup table. The look table keeps the representative feature vectors of all the known classes to be learnt. These representative feature vectors come from the training regions located in figure 4.3. Table in appendix B gives more details including coordinates and area (in pixels) of the training regions for each class. Table 4.3 gives statistics on the percent area used for training each class, which shows that none of the classes, i.e. land-covers is trained unreasonably heavily.

Table 4.3: The training regions

Class	No. of Training Regions	Total Training Area	Total area of the class	Percentage of the Training Area
1	3	10803	238097	4.54%
2	5	27007	650099	4.15%
3	8	39209	1013992	3.87%
4	3	9121	155475	5.87%
5	7	19819	896382	2.21%
6	3	3511	44194	7.94%
7	13	14794	1052101	1.41%
8	8	9702	181810	5.34%
Total	50	133966	4232150	3.17%

The classifier-training phase extracts at least a single vector from each training region. The exact number of feature-vectors extracted from a test region depends on both, size of the training region and size of the training window. Table 4.4 shows details of the lookup tables of the kNN classifier, corresponding to various window sizes.

Texton size of the coarsest texture puts the lower limit to the training-window-size at  $17 \times 17$  pixels. On the other hand, the narrowest training region (See appendix B) puts the upper limit to the training-window-size at  $28 \times 28$  pixels. Nevertheless, the maximum window size that seems useful in classification is  $21 \times 21$  pixels. The windows larger than this simply yield the results that are always worse than the smaller windows. Therefore, the training window-sizes ranging from  $17 \times 17$  to  $21 \times 21$  pixels were tested to classify the test images.

The texture-features extracted from the windows measuring from  $17 \times 17$  pixel to  $21 \times 21$  pixel become part of the kNN classifier's respective lookup table. This means that there are five lookup tables, which all come from a different window size. Therefore, the parameter of training-window-size passes to the classifier while classifying a test image and the classifier uses the corresponding lookup table accordingly.



Table 4.4: Details of training the kNN classifiers – number of feature vectors representing each class for a given window-size

Size of the training window	Class								Total number of feature vectors
	1	2	3	4	5	6	7	8	
17	50	114	154	37	75	13	63	41	547
18	50	97	147	37	75	13	63	41	523
19	43	94	141	31	75	13	63	41	501
20	34	90	141	31	75	13	63	41	474
21	28	76	111	31	70	13	58	37	424

### 4.3.2 The testing phase

There are two parameters governing the kNN classifier, i.e. the training-window-size and the number of neighbours (k) to consider. The kNN classifier uses the lookup table corresponding to the window-size passed to it. Following sections explore two ways of classifying the test images, i.e. region classification and the grid classification.

#### 4.3.2.1 Region classification

From the segmentation results shown in figure 4.12 and figure 4.13 it is evident that most of the delineated regions include pixels from more than a single land-cover. The delineation is never accurate. This makes the texture features extracted from such delineated regions misleading causing incorrect classification of the whole region. Therefore, the region based classification fails.

Moreover, even if the regions delineate with utmost perfection, and thereby contain only a single land-cover, even then the intra-class variations across the region tend to cause misclassifications, because the extracted features are averaged over the entire region and are more representative of the region that is a unique occurrence of the given land-cover. The LFH-base texture features come from the histogram of the DFT-based texture signatures of an entire region. Therefore, LFH-based feature vector representing a large region simply fails to resemble with the features vectors extracted from the comparatively much smaller training regions. Contrarily, the features extracted from several blocks of much smaller area allow modelling the intra-class variations more effectively.

#### 4.3.2.2 Grid classification

This approach divides the test image into a grid of  $X \times X$  pixel square blocks, and recognises each block as belonging to any one of the eight classes. The value of X varies from 17 to 21, corresponding to the training-window-size. The experiments of grid-classification performed on the eight test images show that for each test image, a different window-size results in the best classification accuracy. However, as shown later, only two window sizes, i.e. 18 and 20 suffice to classify all the test images.

Figure 4.14 and figure 4.15 show the test region 3 and test region 6, respectively, and their corresponding ground truths and the outcomes of the grid classification. The grid classification is performed over the block size of  $17 \times 17$  pixels.

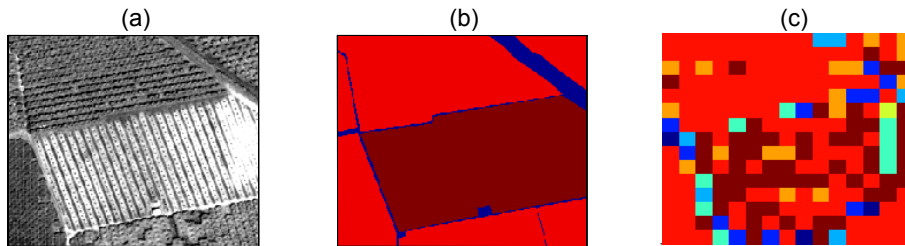


Figure 4.14: (a) the test region 3, (b) its ground truth, (c) its grid classification with the block size of  $17 \times 17$  pixels

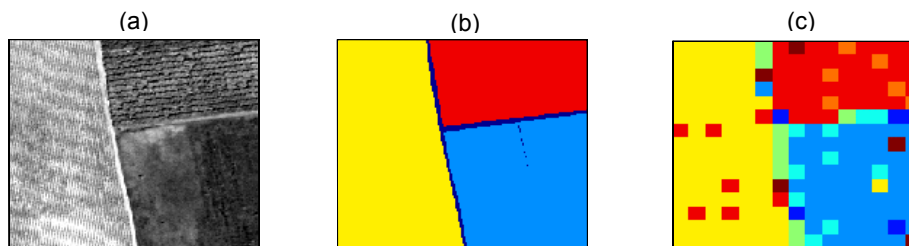


Figure 4.15: (a) the test region 6, (b) its ground truth, (c) its grid classification with block size of  $17 \times 17$  pixels

In the result of region 3, the class 7 (old orchards in red) is classified with a high accuracy. However, class 8 (young orchards in brown) remains mostly misclassified. Class 7 is confused with class 8 and others. The blocks on the borders of the two classes and on the edges are also misclassified as belonging to a third class. In the result of region 6, there are relatively fewer errors. The blocks in all the three classes are classified mostly correctly, with the classification accuracy of 82.3%.

#### 4.4 Fusion of the region and the class information

The grid classification accuracy is highly dependent on the block size. Generally, larger blocks have more description power, i.e. the discrimination power than the smaller ones, but the larger blocks tend to increase the errors around the class borders. On the contrary, the smaller blocks usually have less description power but result in fewer errors around the borders. However, whatever be the size of the block, there appear some misclassified blocks even in the regions of homogenous classes, in addition to the fact that the grid classification lacks the important information on the accurate borders or the frontiers between the land-covers.

Since demarcation of regions in the test image representing similar or dissimilar land-covers provides the lacking contour information, it is useful to merge this information with the image carrying the class or the land-cover information from the grid classification. Hence, the process of

fusion merges the two images, i.e. one resulting from the grid classification and the other resulting from the segmentation into regions, to obtain the final and more accurate land-cover classification. This not only adds the lacking contour information to the classification, but also removes the errors within the regions of homogenous classes/land-covers. There are two possible schemes of such a fusion. Following subsections present the two.

#### 4.4.1 Scheme 1

Figure 4.16 illustrates the first possibility, where panchromatic RS image provides both, the class information and region information. FOOS segments the RS image using the texture signatures, and kNN classifier classifies each block of the image discretised into blocks. Finally, the results of the segmentation and the classification fuse together to produce the final classification map.

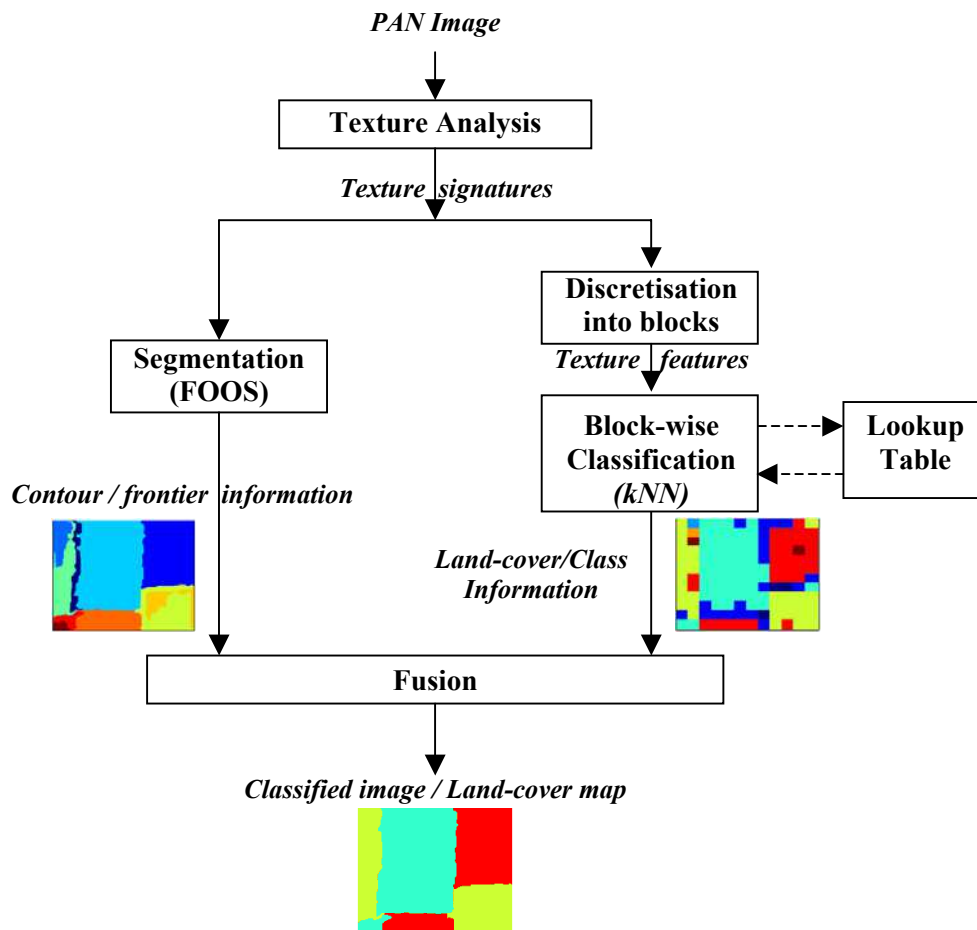


Figure 4.16: The process of fusing the region/frontier and the land-cover information

#### 4.4.2 Scheme 2

Figure 4.17 depicts the other possibility, where the region information comes from the multi-spectral images and the class information comes from the PAN image. It should be noted that the resolution of the MS image has been enhanced using the pixel-level fusion with the panchromatic image. FOOS segments the test image using spectral signatures and kNN classifies the test image

discretised into blocks using texture features. Finally, the results of segmentation driven by spectral features and the classification driven by the textural information fuse together to give the final land-cover map.

To fuse the class and the region information, each of the pixels within a region vote in favour of the corresponding class or land-cover in the grid classification. Then, the highest vote count decides upon the class of each region.

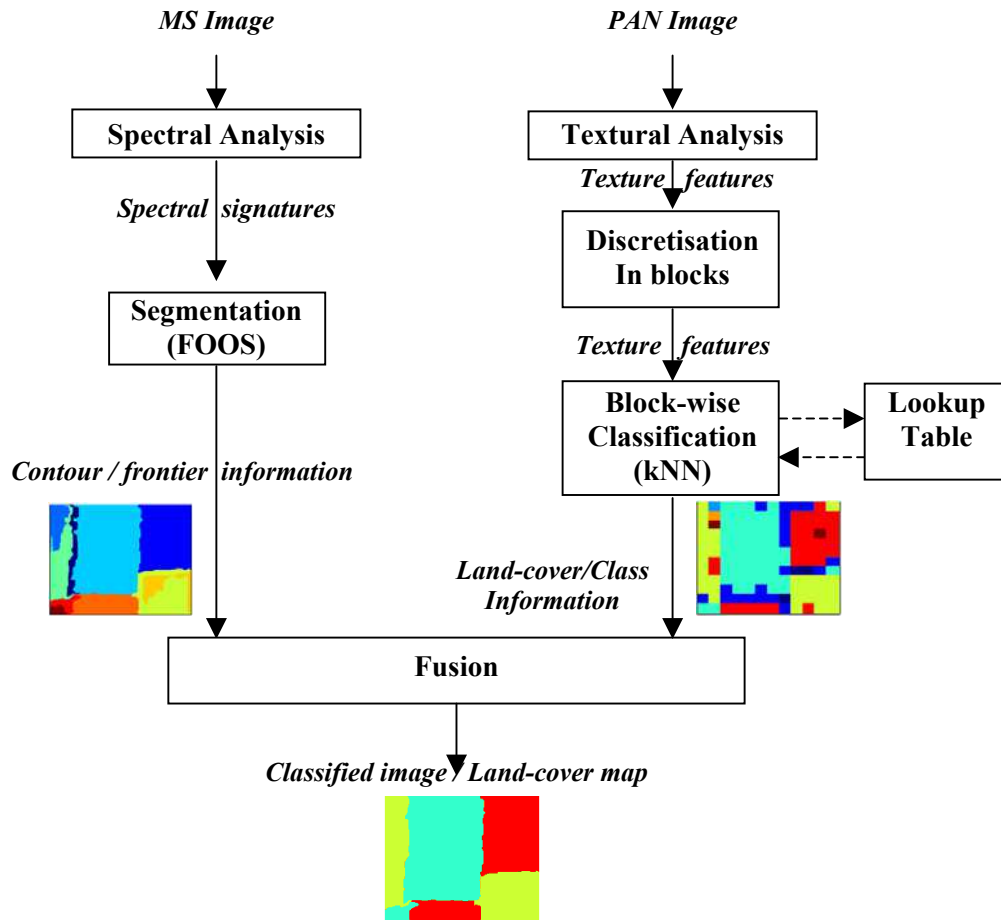


Figure 4.17: Scheme of fusing the spectral information and the textural information

#### 4.4.3 Discussion

The preceding sections presented two possible ways of fusing the contour information with the class information. The first one has the advantage that it does not need the multispectral data and only panchromatic image from the target area suffices. Its disadvantage is that it is computationally expensive. FOOS requires more time to segment the dataset comprising texture signatures than it takes to segment the dataset of spectral signatures. The advantage of the second scheme is that it is computationally less demanding and that it utilises complimentary information of MS images and provides a means of fusing the spectral information with the textural information. The later reason is the most important aspect in the contemporary techniques of remote sensing. Due to availability

of the MS imagery, the results presented in the succeeding section use the scheme 2 that fuses region information obtained using the spectral signatures with the class information obtained using the texture features.

## 4.5 Performance evaluation

This section presents the results of classifying the 8 test regions introduced in the section 4.1.5. The results include both, before fusion and after fusion classification. The remote sensing community often carries out performance evaluation in terms of Kappa statistics and a measure called Kappa.

### 4.5.1 Kappa statistics

The Kappa value is a function of user's accuracy (UA), producer's accuracy (PA), overall accuracy (OA), and the expected classification accuracy (ECA) that are computed from an error matrix (EM) and a power matrix. Therefore, following subsections explain the Kappa statistics.

#### 4.5.1.1 Error matrix (EM)

An EM, also called confusion matrix or contingency table, is the statistics of ground truth and the classification results presented in rows and columns. Table 4.5 shows the EM resulting from the grid classification of the test region 3 shown in figure 4.14 and table 4.6 shows the EM resulting from the grid classification of the test region 6 shown in figure 4.15.

The eight cover types appear row-wise as well as column-wise. The last column shows the total number of pixels classified as belonging to each class and the last row shows the number of pixels actually belonging to each class. Since there are only two land-covers in the test region 3, only two corresponding columns are non-zeros in its EM of table 4.5. Similarly, since there are only three land-covers in the test region 6, only three corresponding columns are non-zeros in its EM of table 4.6.

Table 4.5: The EM for the grid classification of the test region 3 shown in figure 4.10

		Reference Data								Rows Total
		1	2	3	4	5	6	7	8	
Cover Type		Forests	Fallow	Fields	Truck crops	Vine-yard	Apple	Old Orchards	Young Orchards	
1	Forests	0	0	0	0	0	0	1656	849	2505
2	Fallow	0	0	0	0	0	0	2453	438	2891
3	Fields	0	0	0	0	0	0	578	269	847
4	Trck crops	0	0	0	0	0	0	835	1585	2420
5	Vineyards	0	0	0	0	0	0	149	289	438
6	Apple	0	0	0	0	0	0	3823	4709	8532
7	Old Orch	0	0	0	0	0	0	22026	4347	26373
8	Yng Orch	0	0	0	0	0	0	3515	14195	17710
Column Total		0	0	0	0	0	0	35035	26681	61716

Table 4.6: The EM for the grid classification for test region 6 shown in figure 4.11

		Reference Data								Rows Total
		1	2	3	4	5	6	7	8	
Cover Type		Forests	Fallow	Fields	Truck crops	Vine-yard	Apple	Old Orchards	Young Orchards	
1	Forests	0	617	0	0	237	0	276	0	<b>1130</b>
2	Fallow	0	14144	0	0	7	0	486	0	<b>14637</b>
3	Fields	0	4678	0	0	644	0	0	0	<b>5322</b>
4	T. crops	0	40	0	0	734	0	299	0	<b>1073</b>
5	Vineyards	0	966	0	0	24264	0	0	0	<b>25230</b>
6	Apple	0	0	0	0	0	0	399	0	<b>399</b>
7	O. Orch	0	0	0	0	1323	0	12994	0	<b>14317</b>
8	Y. Orch	0	441	0	0	517	0	621	0	<b>1579</b>
Column Total		<b>0</b>	<b>20886</b>	<b>0</b>	<b>0</b>	<b>27726</b>	<b>0</b>	<b>15075</b>	<b>0</b>	<b>63687</b>

#### 4.5.1.2 User's accuracy (UA)

The term UA refers to the ratio of the number of pixels correctly classified as the cover type of interest to the total number of pixels that are classified in the cover type of interest.

$$UA_i = 100 \times \frac{EM(i, i)}{\sum_{j=1}^N EM(i, j)}, \quad (4.1)$$

where  $UA_i$  represents user's accuracy for  $i^{\text{th}}$  class,  $N$  represents the number of classes, and  $EM(i, j)$  represents value in the  $i^{\text{th}}$  row and  $j^{\text{th}}$  column of the error matrix. For example, for a user interested in cover type *forest*, the UA is the ratio (percentage) of pixels correctly classified as *forest* to the total number of pixels classified as *forest*.

#### 4.5.1.3 Producer's accuracy (PA)

The term PA refers to the ratio of the number of pixels correctly classified as the cover type of interest to the number of pixels actually belonging to that class.

$$PA_j = 100 \times \frac{EM(j, j)}{\sum_{i=1}^N EM(i, j)}, \quad (4.2)$$

where  $PA_j$  represents producer's accuracy for  $j^{\text{th}}$  class. For example, the PA for the class *forest* is ratio (percentage) of the no. of pixels correctly classified as *forest* to the no. of pixels that actually belong to the class *forest*.

#### 4.5.1.4 Overall classification accuracy (OCA)

The OCA is the ratio of the number of correctly classified pixels to the total number of the pixels in the test image. The number of pixel correctly classified comes from the diagonal (in bold) of the EM. The OCA is the ratio of the sum of the values in diagonal (with colour background) of EM to the sum of last column (bottom-right value of the EM).

$$OCA = \frac{\sum_{i=1}^N EM(i, i)}{\sum_{i=1}^N \sum_{j=1}^N EM(i, j)}. \quad (4.3)$$

#### 4.5.1.5 Product matrix (PM)

The product matrix derives from the error matrix.  $i^{\text{th}}$  row and  $j^{\text{th}}$  column of the PM equals the product of the EM (last row,  $j^{\text{th}}$  column), and EM ( $i^{\text{th}}$  row, last column). As an example, table 4.7 and table 4.8 give the PMs corresponding to the EMs of table 4.5 and table 4.6, respectively. Equation (4.4) explains the same analytically.

$$PM(i, j) = \sum_{i=1}^N EM(i, j) \times \sum_{j=1}^N EM(i, j). \quad (4.4)$$

Table 4.7: The PM corresponding to the EM of table 4.5

		Reference Data							
		1	2	3	4	5	6	7	8
Cover Type		Forests	Fallow	Fields	Truck crops	Vineyard	Apple	Old Orchards	Young Orchards
1	Forests	0	0	0	0	0	0	13943930	10619038
2	Fallow	0	0	0	0	0	0	30795765	23452599
3	Fields	0	0	0	0	0	0	206321115	157124409
4	Truck crops	0	0	0	0	0	0	29569540	22518764
5	Vineyards	0	0	0	0	0	0	13488475	10272185
6	Apple	0	0	0	0	0	0	312477165	237967839
7	Old Orchard	0	0	0	0	0	0	994258265	757180099
8	Y. Orchard	0	0	0	0	0	0	561365805	427509663

Table 4.8: The PM corresponding to the EM of table 4.6

		Reference Data							
		1	2	3	4	5	6	7	8
Cover Type		Forests	Fallow	Fields	Truck crops	Vineyard	Apple	Old Orchards	Young Orchards
1	Forests	0	23601180	0	0	31330380	0	17034750	0
2	Fallow	0	305708382	0	0	405825462	0	220652775	0
3	Fields	0	111155292	0	0	147557772	0	80229150	0
4	Truck crops	0	22410678	0	0	29749998	0	16175475	0
5	Vineyards	0	526953780	0	0	699526980	0	380342250	0
6	Apple	0	8333514	0	0	11062674	0	6014925	0
7	Old Orchard	0	299024862	0	0	396953142	0	215828775	0
8	Y. Orchards	0	32978994	0	0	43779354	0	23803425	0

#### 4.5.1.6 Expected classification accuracy (ECA)

The ECA is the accuracy by chance, i.e. the accuracy resulting from random assignment of a class to each image pixel. It comes from the PM described in the preceding sub-section. The ECA equals the ratio of the sum of all the diagonal values (with colour background) to the sum of all the values in the PM. Equation (4.5) expresses the ECA mathematically.

$$ECA = \frac{\sum_{i=1}^N PM(i,i)}{\sum_{i=1}^N \sum_{j=1}^N PM(i,j)}, \quad (4.5)$$

where N is the number of classes, and in the experiments presented herein, N=8.

#### 4.5.1.7 Kappa

The kappa is a measure to evaluate the performance of automatic classification especially in case of the land-cover classification in the remote sensing applications. Kappa, often represented as KHAT, is a performance measure of the automatic classification process as compared to the random classification of an image. Equation (4.6) defines it:

$$KHAT = \frac{OCA - ECA}{1 - ECA} \quad (4.6)$$

Table 4.9 shows performance of classification in terms of kappa statistics including UA, PA, and KHAT obtained from the error matrix of table 4.5 and power matrix of table 4.7, corresponding to the test region 3. Similarly, table 4.10 shows the results obtained from error matrix of table 4.6 and power matrix of table 4.8, corresponding to test region 6. Higher values of KHAT reflect better classification performance, whereas lower KHAT values reflect poorer classification performance.

Table 4.9: The statistics derived from EM of table 4.5 and PM of table 4.7

Grid Classification	UA		PA		ECA	OCA	KHAT
	Class 7	Class 8	Class 7	Class 8			
	Young Orchards	Old Orchards	Young Orchards	Old Orchards			
Accuracy	83.6%	79.2%	63.7%	49.9%	36.4%	57.8%	0.3362

Table 4.10: The statistics derived from EM of table 4.6 and PM of table 4.8

Grid Classification	UA			PA			ECA	OCA	KHAT
	Class 4	Class 5	Class 7	Class 4	Class 5	Class 7			
	Truck Crops	Vine-yard	Young Orchard	Truck Crops	Vine-yard	Young Orchard			
Accuracy	91.2%	100.0%	73.8%	77.7%	52.7%	67.6%	27.9%	68.0%	0.5556

#### 4.5.2 The resulting land-cover maps and their accuracy

Here are some results of the land-cover classifications before and after the fusion performed as per scheme 2 of figure 4.17. Table 4.11 lists the results in terms of Kappa statistics for eight regions. The table shows results of the grid classification, i.e. before fusion, and after fusion with the contour information. The value of k (the number of neighbours) is 1 in all the results, i.e. the



nearest neighbour classification has been used rather. Different values of k result in different accuracies in the grid classification. Values other than 1 may result in better accuracies in the grid classification, but will not change the post-fusion classification accuracies that are highly dependent on the accuracy of the segmentation. Therefore, results are only given for k=1, since the interest is to find a single value that can be used in all cases and for all the test images. Similarly, window/block sizes other than 18×18 pixels and 20×20 pixels perform better in some images, but the interest was to find a single value that performs acceptably well for all the test images. Window sizes of 18×18 pixels and 20×20 pixels work acceptably for almost all the test images. The two window sizes result in the same land-cover map after the fusion, with the exceptions of test region 3, 5 and 8, for which the results are little worse with one or the other.

Table 4.11: The classification results in terms of Kappa statistics for eight test regions

Test Region	Classes	Accuracy						k	Block/ window size
		Before Fusion			After Fusion				
		ECA	OCA	KHAT	ECA	OCA	KHAT		
1	3	69.4%	69.4%	0	99.9%	100%	1	18, 20	
2	3, 5	36.1%	67.8%	0.4950	50.2%	91.3%	0.8250	18, 20	
3	7, 8	36.1%	55.2%	0.2999	50.8%	98.9%	0.9774	20	
4	4, 5, 7	28.4%	72.0%	0.6084	34.5%	93.9%	0.9074	18, 20	
5	3, 5, 6	22.3%	53.5%	0.4010	35.3%	92.7%	0.8878	18	
6	2, 5, 7	30.1%	80.7%	0.7240	35.1%	94.1%	0.9098	18, 20	
7	2, 3, 7	31.9%	77.3%	0.6662	37.6%	90.2%	0.8423	18, 20	
8	1, 3, 5, 7	18.1%	46.2%	0.3430	26.2%	79.0%	0.7161	18	

Figure 4.18 through figure 4.25 show the land-cover maps resulting after fusion of the class and the contour information for test region 1 through 8, respectively. The maximum classification accuracy achieved is 100%, corresponding to the first test region and the least classification accuracy achieved is 79%, corresponding to the last test region, i.e. test region 8.

Figure 4.18(a) shows the results on the test region 1. As shown in figure 4.18 (b), this test image contains only one class, i.e. *crop fields* in four disjoint regions separated by the strips of non-class. Part (c) of the figure shows the test image as segmented using the spectral indices. Therefore, the pixels/regions having the same colour are spectrally similar to each other and spectrally dissimilar to the pixels/regions having different colour. However, the regions in figure 4.18(c) do not correspond to any specific class or land-cover type, since it is the result of the unsupervised segmentation process. Part (d) of the figure shows the grid segmentation of the test image performed as explained in section 4.3.2.2. Part (e) of the figure shows the improved classification using the process of fusion as explained in the section 4.4. Part (f) of the figure shows the binary error image, indicating the misclassified pixels in black. The calculation of the accuracies and the error image does not take into account the pixels in the non-class that always appear white in the error image. Since there appear no black pixels in the error image of figure 4.18 (e), therefore, the

overall classification accuracy is 100% and the corresponding KHAT value is 1 as shown in table 4.11.

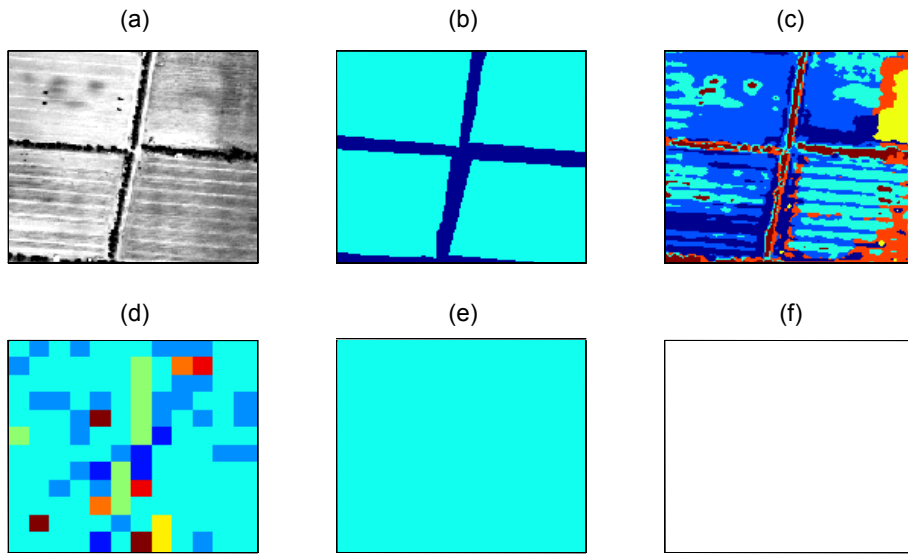


Figure 4.18: (a) the test region 1, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image

In case of test region 2, the accuracy in the grid classification is not more than 70%, but the process of fusion considerably improves the classification accuracy. There still appear some errors in the regions of homogeneous land-cover due to the imperfections in the segmentations. A process of polygon filling as discussed in a later section can help remove these errors largely.

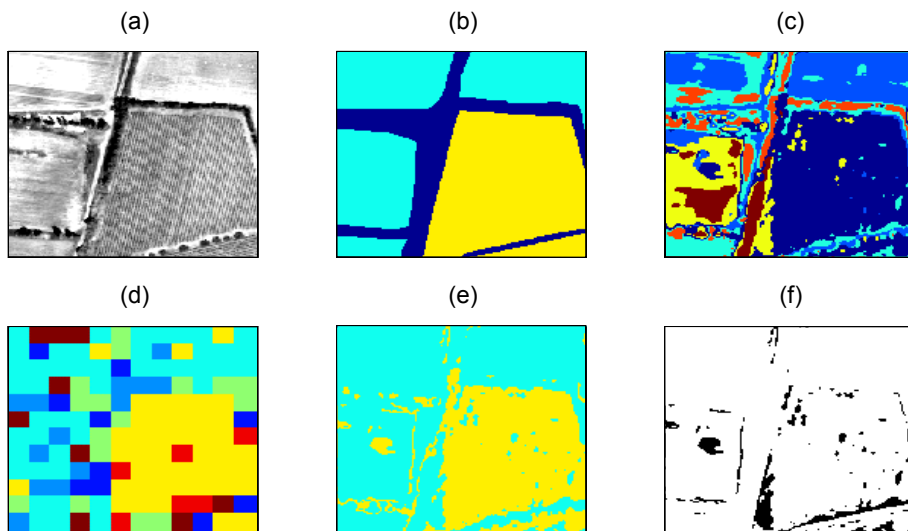


Figure 4.19: (a) the test region 2, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image

In case of test region 3 as well, the process of fusion brings in a big gain in classification accuracy. In the final land-cover map, the errors remain only around the borders. Class 8 is misclassified very heavily. Mostly, it is confused with class 6 and 7 and others. As the misclassifications divide into several classes, the highest vote count goes in favour of the correct

class, i.e. class 8. Consequently, the following step of fusion results in the correct classification of class 8. However, this land-cover map results only with the window-size of  $20 \times 20$  pixels. With the window-size of  $18 \times 18$ , there remain some errors that can be removed with improved segmentation.

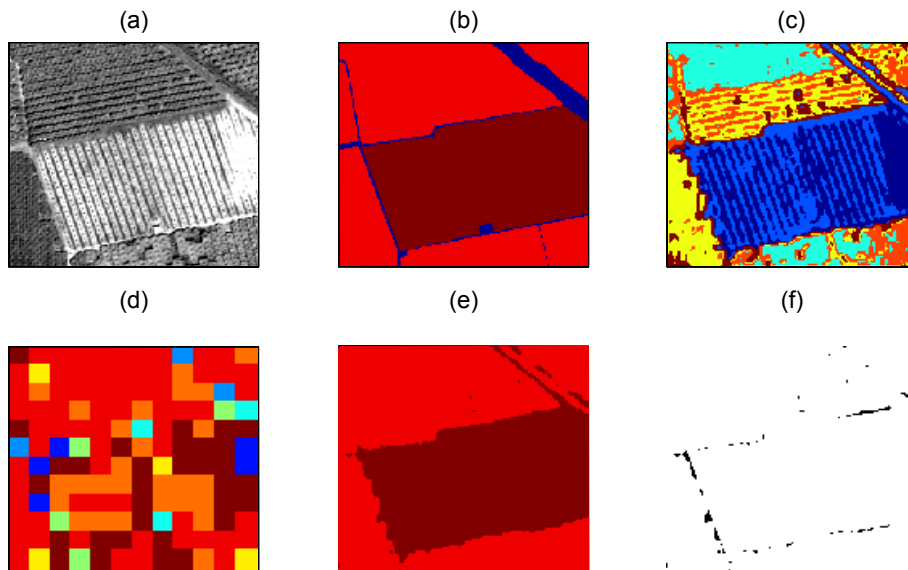


Figure 4.20: (a) the test region 3, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image

In the grid classification of region 4, the misclassified blocks are mainly on the borders. The misclassified block in the bottom-left corner is one of the few exceptions. The process of fusion rectifies this misclassification and all other misclassifications as well. In the result after fusion, the errors appear mostly around the class borders. The process of polygon filling explained in a later section can remove the small patches of errors and hence can improve the accuracy.

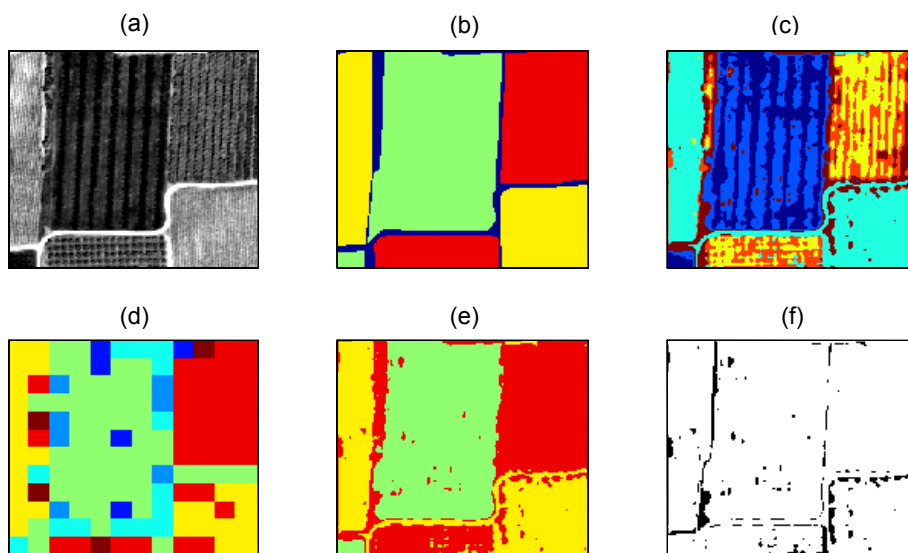


Figure 4.21: (a) the test region 4, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image

There are several misclassifications in the grid classification of region 5. However, after fusion, the classification accuracy becomes quite high and the misclassification are either on the borders or are granular in the region 5.

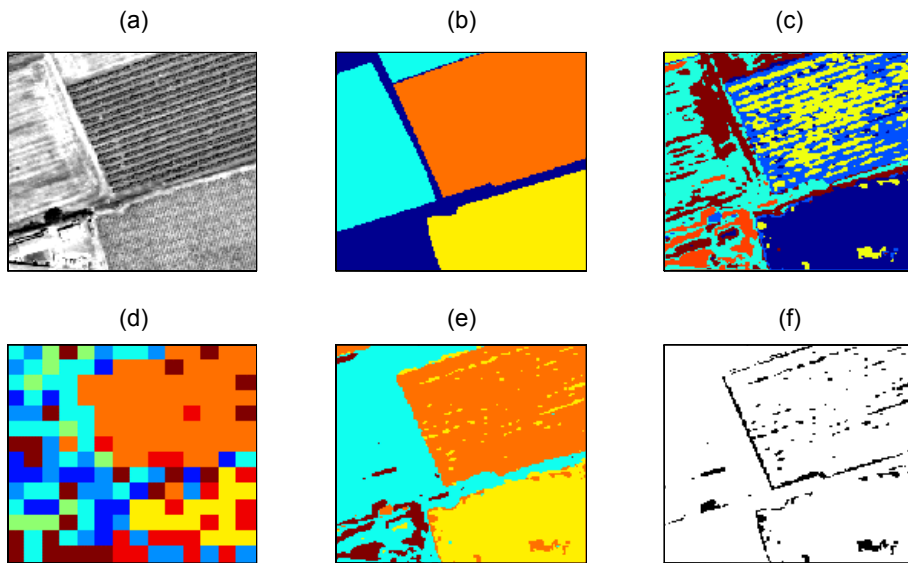


Figure 4.22: (a) the test region 5, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image

The grid classification of region 6 is quite successful. There are not too many misclassified blocks in class 5 and class 7. The incorrectly classified parts of class 2 are rectified after fusion. On the other hand, a part of class 2 that was correctly classified in the grid classification is misclassified after fusion and appears as a black patch in the error image.

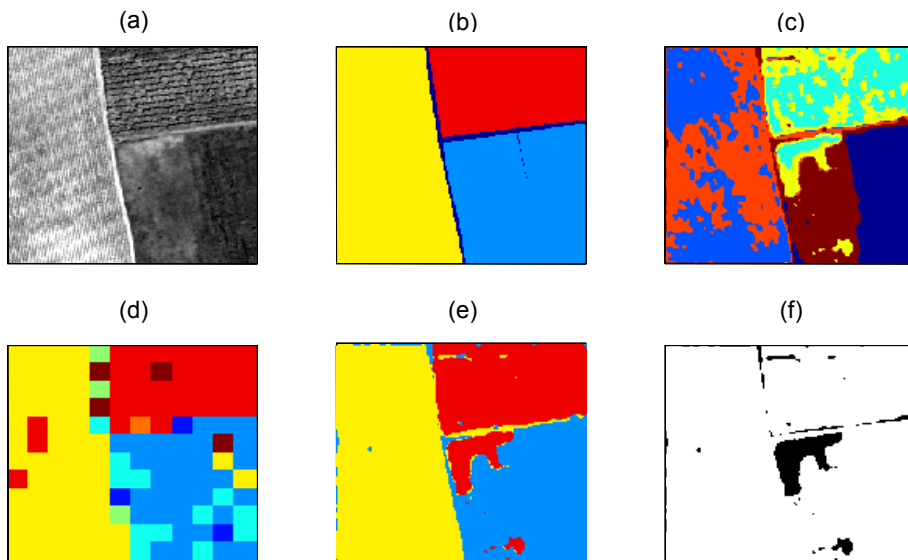


Figure 4.23: (a) the test region 6, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image

The test region 7 contains quite large area of the non-class comprising the manmade structures. This generally makes the process of classification more difficult and becomes the cause of

misclassifications. However, the block classification is quite successful and the following process of fusion further improves the classification accuracy.

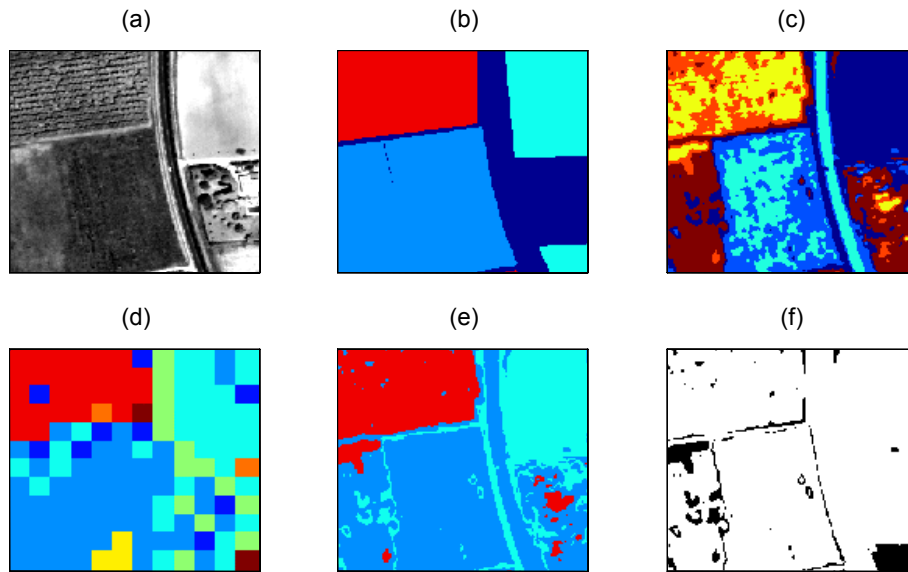


Figure 4.24: (a) the test region 7, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image

In case of test region 8, a part of class 7, i.e. old orchards has been misclassified as class 1, i.e. forest. This is because the misclassified part of old orchards is spectrally different from other parts of the same class and is more close to the forest. In addition, the misclassified part of the old orchards is different from other parts of the same class with regard to the texture as well. The error images show that there remain errors within regions of homogenous land-cover, due to imperfect demarcation. Most of the times, these errors can be eliminated using the polygon filling, a post-processing on FOOS-generated segmentation.

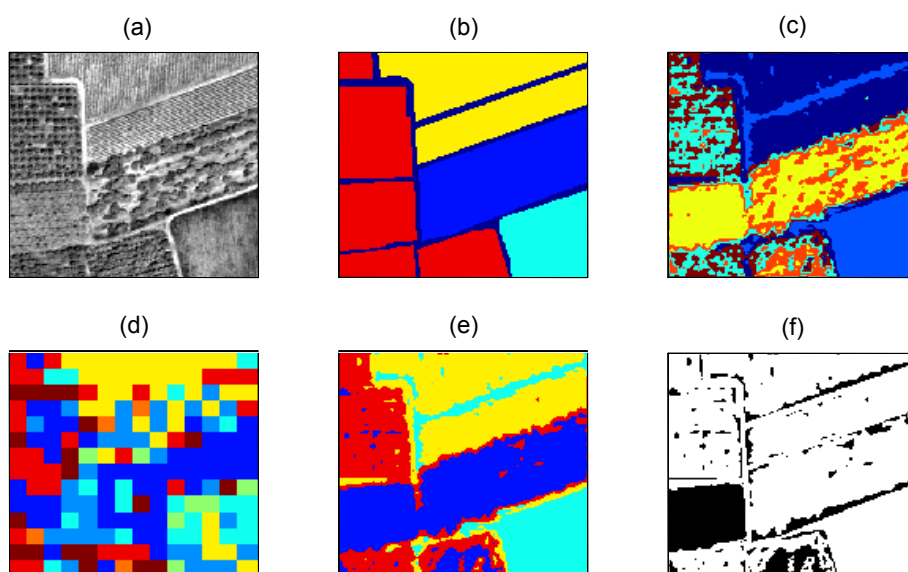


Figure 4.25: (a) the test region 8, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) fused classification map, and (f) the error image

## 4.6 Further improvements

There are certain possibilities of further improvement in the classification performance. Following sections present the ideas behind the possibilities and the limited results available on one of those possibilities of improvement.

### 4.6.1 Polygon filling

Sometimes, the FOOS-segmented image has some clusters comprising the pixels from dissimilar land-covers. Such clusters are often spatially disjoint and are the cause of misclassification. Usually, these patches are numerous and small in area. Similar is the situation in the segmentation result of figure 4.23(c) that contains several small patches as apposed to only three large regions in the corresponding ground truth. The process of polygon filling refines the FOOS-generated segmentation by:

- dissociating the patches with the same spectral class but different land-covers and
- systematically removing the patches that are too small

The first step finds the contours, removes the spectral information, and gives each polygon a unique label. The second step removes the patches that are too small. The area-to-perimeter ratio is a good criterion for a region being or not being too small. The critical value of this parameter is chosen to be 5. A too small patch gets replaced with the most frequent label in its surroundings. It starts with the patch having the smallest area-to-perimeter ratio. The process continues until there is no patch having the area-to-perimeter ration smaller than 5. The process of polygon filling usually improves the results but is computationally expensive.

Figure 4.26 shows the classification results for region 6 after polygon filling. In this case, the errors are limited to places near borders, and the overall percent accuracy of the classification improves considerably.

The result of figure 4.26(e) has the overall classification accuracy of 99.3%. Table 4.12 shows the corresponding Kappa statistics. Similarly, all the results presented in table 4.11 can improve largely with polygon filling.

### 4.6.2 Two block-sizes

The MS image used herein is a fused product, resulting from the pixel level fusion. The fusion of region and class information proposed in the preceding sections is a form of fusion at the decision level. However, another fusion is also possible at the decision-level. Experimental observations show that the block/window size of 18×18 pixels is preferable for successful classification of certain land-covers such as *young orchards* whereas the block/window size of 20×20 pixels for the

*old orchards*. Hence, it is interesting to perform the grid classification twice, once with each of the two preferred block/window sizes, and then perform the decision level fusion. In this case, every pixel votes two times, either for two different classes or for the same class twice, but the method of fusion remains the same.

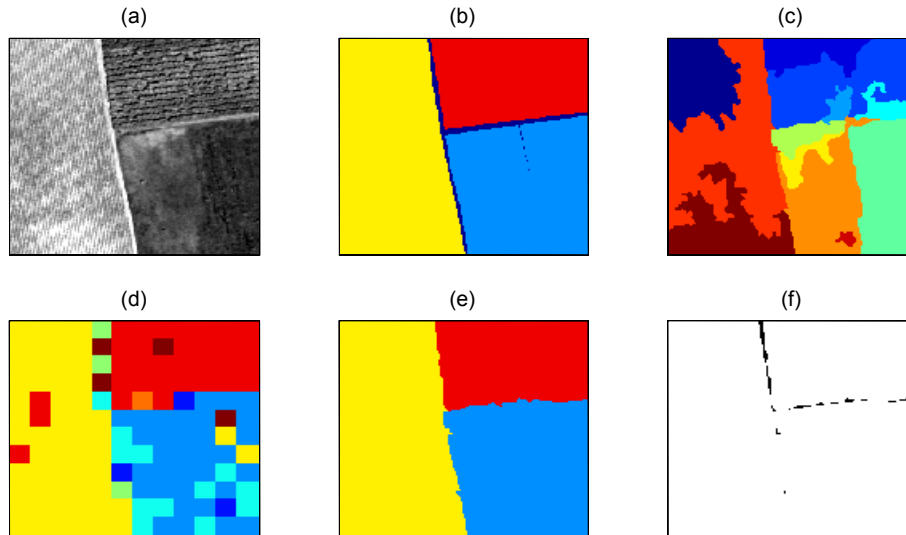


Figure 4.26: (a) the test region 6, (b) its corresponding ground truth, (c) its segmentation after polygon filling, (d) grid-classification map, (e) fused classification map, and (f) the error image

Table 4.12: The Kappa statistics for region 6, after polygon filling

Fused Classification	UA			PA			ECA	OCA	KHAT
	Class 2	Class 5	Class 7	Class 2	Class 5	Class 7			
	Fallows	Vine-yard	Young Orchard	Fallows	Vine-yard	Young Orchard			
Accuracy	99.4%	99.0%	99.7%	99.7%	99.99%	97.6%	35.4%	99.3%	0.9894

### 4.6.3 Feature-level fusion

The final classification map is dependent on the quality of segmentation of the test region. In some cases, like in case of test region 7, segmentation with textural and/or spectral information alone becomes difficult, especially if there is significant presence of manmade structures or other non-class in the test region. Refer figure 4.24. In those cases, the results of segmentation may improve if both, spectral and textural signatures drive the process of segmentation.

## 4.7 Conclusion

The land-cover classification has proved to be a difficult task that becomes quite impossible using only the spectral information. Therefore, this chapter introduced a method of fusing spectral information with the textural information for land-cover classification. The chapter also presented the results of segmenting remote sensing image using spectral as well as textural features and the FOOS clustering method introduced earlier in chapter 3.

Clustering the dataset based on four spectral signatures is considerably less expensive than clustering the dataset based on seven textural features, since the complexity of the clustering algorithms are known to curse of dimensionality [Lu 2008]. This clustering provides the contour information, i.e. delineation of the land-covers or classes. The window size of  $9 \times 9$  pixels is suitable for averaging/smoothing the images carrying spectral signatures before using them for the image segmentation.

The chapter also presented results of classifying the remote sensing images discretised into blocks and showed that the classification of regions fails due to the intra-class variation found in a class and the imperfect delineation of the classes. Several experiments performed over the eight test-regions show that the window/block size of  $18 \times 18$  pixels is workable on all the test images.

Later, the chapter introduces a method of fusing the contour information with the block-wise classification that largely improves the classification accuracy. In the end, the chapter also explores the possibilities of further improvements in the process of land-cover mapping and suggests improving the results using:

- polygon filling
- the two preferred sizes of training-window sizes and the subsequent decision-level fusion
- feature-level fusion for improved segmentation

The first one of these three suggestions has also been tried on a test image that gave improved results, while other options need yet to be explored.



## Conclusion

The remote sensing technologies are now quite mature, and so are the techniques and algorithms of remote sensing. Remote sensing of the environment is a cross road between the fields of image processing, machine learning and the natural sciences. Hence, this thesis is the blend of the three. It introduces the vegetation indices that are the traditional means of discriminating one land-cover from the others. These indices also have practical limitations.

Since a land-cover may vary spectrally quite largely depending on the climate, weather conditions, and many other factors, the vegetation indices prove incapable to recognise the land-covers. Nevertheless, whatever be the conditions, different land-covers however remain spectrally dissimilar. Therefore, although the land-covers may be spectrally unrecognisable, they are always spectrally separable anyways. Keeping this in view, the approach proposed herein suggests using the spectral signatures for segmenting the RS images into regions of homogenous land-covers. This discrimination is realised using clustering techniques. Therefore, this thesis proposes Fusion of Over-Segmentations (FOOS) as an improved version of k-means clustering. The new clustering algorithm outperforms k-means with multiple restarts while segmenting natural digital photos, remote sensing images, and non-image datasets.

Once the regions of homogeneous land-covers become disjoint, it is turn to recognise these regions as belonging to a particular land-cover. The high-resolution RS images have created the possibility of harnessing the image texture features for classification of the land-covers as compared to spectral features alone in the past. However, since a region with homogenous land-cover may contain a small number of pixels, the texture features extraction method must be able to extract the useful signatures from the smallest possible sample of the texture patterns. In the case study presented herein, the available training regions were as small as 28 pixels wide and the texture features based on signal processing approach become impractical in this situation. Therefore, the texture features extracted from immediate pixel-neighbourhood, such as the features based on local Fourier histograms (LFH), remains the only choice available. This thesis proposes two improved versions of the LFH-based texture features. One of those best suites the problem of very small training regions.

The approach of recognising each region with the help of texture features also fails since the regions of homogeneous land-cover comprise assorted regions of different textures. Therefore, the consolidated texture features extracted from a region comparatively bigger than the training regions becomes misleading. The texture features are capable of separately modelling each of the variations within a land-cover. To resolve this problem, the LFH texture features extracted from all the possible variations within a land-cover represent the same class in a lookup table of the k-nearest neighbour classifier. To assure the successful and fast recognition of land-covers, the test image is

divided into blocks as small as the smallest training-region and the block-wise classification of the image follows.

The block-wise or the grid classification alone does not provide high classification accuracies. The scheme of land-cover mapping proposed herein concludes with a new approach to image fusion at the decision level, which is in fact the fusion of spectral information and textural information. The process of image segmentation driven by the spectral signatures provides the regions of homogenous land-covers and the contours between them. The grid classification provides the class or land-cover information on each pixel. Fusion of these two images results in a land-cover map with very high classification accuracies and well-delineated regions.

Naturally, due to the diversity of the problem dealt herein, there remain many possibilities yet unexplored. In this way, this thesis also points out the potential future work to improve the results presented herein. The future work may include:

- Further improvement in the LFH-based features
- Feature-level fusion of spectral and textural characteristics, for better improved segmentation of the test images
- Using the two preferred sizes of training-window sizes and the subsequent decision-level fusion
- Trying other, more involved classifiers such as naïve Bayes' classifier preferably on another RS image with the same land-covers

The methods introduced herein placed the prime importance to the computation simplicity that is vital in determining how many data sources may fuse together to enrich and refine the final product. Despite ever-increasing computational power of the computing machines, the computational simplicity remains the key aspect of any algorithm in the area of image processing and machine intelligence.

## Appendices

### Appendix A: Letter of permission from Defence Image Database for reproducing their images on page 79 of this thesis



MR S LILLEY, IPR-CU1

Mr A Ursani  
20 Avenue des Buttes de Coësmes  
CS 14315  
F ? 45043 Rennes Cedex  
France

By e-mail (Ahsan-Ahmad-Ursani@insa-rennes.fr)

# DCD

## Defence Commercial Directorate

Ministry of Defence  
Directorate of Intellectual Property Rights  
Poplar 2 #2218  
MoD Abbey Wood  
Bristol BS34 8JH

Direct line: 0117-91-32862  
Facsimile: 0117-91-32930  
e-mail: dipr-cu@mod.uk

Our Ref: D/IPR/1X/4/3/3 (5824)  
Your Ref:

Dear Mr Ursani,

13<sup>th</sup> Aug 08

**Reproduction of Crown Copyright Photographs** 45132227, 45132244, 45132272, 45139614, 45142439, 45144975, 45145936, 45146340, 45146406, 45147997

Thank you for your e-mail dated 31<sup>st</sup> July 08 in which you request permission to reproduce the above referenced photographs for the purpose of your PhD thesis on image Segmentation.

I am pleased to inform you that the Ministry of Defence grants permission for the non-commercial use of the images in the manner you describe, on the conditions that the file size of the image when stored in an electronic format shall be no greater than 200Kb and that there is no commercial reproduction of the images on merchandise or any resale of the same. There may be no further reproduction including any by the Institute.

The fee is waived on the condition that your use of the images is not for commercial gain, as we do not allow such gain to be made when the MOD provides free use of its resources.

The acknowledgement should be:

***“Photograph by: .....; © Crown Copyright/MOD, image from www.photos.mod.uk. Reproduced with the permission of the Controller of Her Majesty’s Stationery Office”.***

If alongside the photograph then the first sentence will suffice.

Yours sincerely,

Mr S Lilley, for the  
Crown Copyright Administrator

**Appendix B: Fuller details of the training regions listed in table 4.3 of page 98**

Table B: Details on area and locations of the training regions in the source RS image

Class	Region #	Coordinates				Area	Total Training Area	Percentage of the Training Area
		Up	Down	Left	Right			
1	1	1122	1160	807	900	3666	10803	4.54% (238097)
	2	741	797	1374	1434	3477		
	3	1400	1460	2261	2320	3660		
2	1	1122	1260	1080	1140	8479	27007	4.15% (650099)
	2	1290	1360	1970	2110	10011		
	3	1190	1250	1435	1485	3111		
	4	1740	1790	1496	1550	2805		
	5	1915	1965	2025	2075	2601		
3	1	245	295	1645	1695	2601	39209	3.87% (1013992)
	2	210	260	1350	1400	2601		
	3	95	155	1735	1795	3721		
	4	385	435	1780	1830	2601		
	5	1220	1250	880	930	1581		
	6	1900	1955	1330	1430	5656		
	7	1025	1085	310	370	3721		
	8	615	665	2345	2395	2601		
4	1	660	715	23	73	2856	9121	5.87% (155475)
	2	480	530	485	540	2856		
	3	945	992	310	360	2448		
5	1	280	310	770	800	961	19819	2.21% (896382)
	2	490	520	1740	1770	961		
	3	675	725	1725	1775	2601		
	4	770	820	1880	1930	2601		
	5	630	680	1915	1965	2601		
	6	1000	1050	1955	2002	2448		
	7	1000	1050	2040	2100	3111		
6	1	35	85	95	145	2601	3511	7.94% (44194)
	2	455	504	5	55	2550		
	3	1090	1135	1190	1240	2346		
7	1	450	480	710	740	961	14794	1.41% (1052101)
	2	465	495	590	640	1581		
	3	500	530	660	690	961		
	4	675	705	465	495	961		
	5	780	810	365	395	961		
	6	860	890	480	510	961		
	7	960	990	570	600	961		
	8	1400	1430	565	595	961		
	9	1885	1915	840	870	961		
	10	1960	1990	855	885	961		
	11	2065	2095	810	840	961		
	12	2070	2100	980	1010	961		
	13	1925	1965	700	740	1681		
8	1	1400	1430	700	730	961	9702	5.34% (181810)
	2	1495	1525	730	760	961		
	3	1805	1835	665	695	961		
	4	1815	1845	830	860	961		
	5	650	680	1235	1265	961		
	6	835	875	1094	1121	1148		
	7	835	875	1130	1157	1148		
	8	903	953	1170	1220	2601		
Total	50 training regions					133966	3.1654% (4232150)	

## List of figures

Figure 1.1:	Geometric Distortions in an image captured by airborne sensor due to Pitch, Yaw and Roll motions of the platform	5
Figure 1.2:	Sensor bands of QuickBird2 remote sensing satellite	6
Figure 1.3:	An airborne LIDAR remote sensing system	14
Figure 1.4:	Land-cover types identified on global scale	15
Figure 1.5:	Spectral reflectance of vegetation, soil and water	16
Figure 1.6:	NDVI vs. LAI	20
Figure 1.7:	The fusion at three different levels of abstraction	24
Figure 1.8:	The pixel level fusion	25
Figure 1.9:	The multilevel fusion	27
Figure 1.10:	The region-based fusion	28
Figure 2.1:	The 9-pixel neighbourhood in the spatial domain	35
Figure 2.2:	Histograms of the coefficients $X_0$ through $X_4$ for image D2	36
Figure 2.3:	The circular neighbourhood as suggested in [Arof 1998]	37
Figure 2.4:	Histograms of $\phi_2$ (middle column) and $\phi_3$ (right most column) for image orientations, $\Theta = 0^\circ, 30^\circ, 45^\circ, 60^\circ,$ and $90^\circ$	41
Figure 2.5:	The four 4-pixel sequences	42
Figure 2.6:	The training setup generating three lookup tables, each resulting from a different training window size	45
Figure 2.7:	The testing setup, generating the noised and noiseless test image sets	46
Figure 2.8:	The subset of 32 texture images from Brodatz collection selected for experimentation	47
Figure 2.9:	(a) the noiseless texture image D87, (b) the same image with Gaussian white noise	48
Figure 2.10:	Performance of texture features while retrieving the images from the database comprising 107 classes and 8560 test-images using the classifier trained with window size of $71 \times 71$ pixels	54
Figure 2.11:	Performance of texture features while retrieving the images from the database comprising 107 classes and 8560 test-images using the classifier trained with window size of $45 \times 45$ pixels	54
Figure 2.12:	Performance of texture features while retrieving the images from the database comprising 107 classes and 8560 test-images using the classifier trained with window size of $27 \times 27$ pixels	55
Figure 2.13:	Retrieval results on the subset of 32 test images using the training window of $71 \times 71$ pixels	55
Figure 2.14:	Retrieval results on the subset with 32 test images using the training-window of $45 \times 45$ pixels	56

Figure 2.15:	The retrieval results on the subset of 32 test-images using the training-window of 27×27 pixels	56
Figure 2.16:	The retrieval results on subset comprising 32 test images using the training window size of 27×27 pixels	58
Figure 3.1:	The fusion of over-segmentations	67
Figure 3.2:	(a) The image with four textures from Brodatz collection, Segmentation into (b) 5 clusters (c) 6 clusters (d) The 4 largest classes of common member pixels	69
Figure 3.3:	(a) The image with 4 textures from Brodatz collection, (b) its ideal segmentation, segmentation result with (c) k-means, (d) Gene Cluster, (e) FOOS algorithm	71
Figure 3.4:	(a) A natural photo of leopard from BSD, its segmentation by (b) a human subject, (c) k-means, (d) Gene Cluster, (e) FOOS	73
Figure 3.5:	(a) The zebra image from BSD, its segmentation (b) as provided by BSD, (c) with k-means, (d) with Gene Cluster, (e) with FOOS	74
Figure 3.6:	The results of segmenting (a) remote sensing image with (b) ground truth into 3 classes using (c) k-means (d) Gene Cluster, (e) FOOS, and into 4 classes using (f) k-means (g) Gene Cluster, (h) FOOS	80
Figure 3.7:	(a) an RS image (b) ground truth with 3 land-covers, (c) ground truth with 4 land-covers, segmentation into 3 clusters using (d) k-means (e) Gene Cluster, (f) FOOS, segmentation into 4 clusters using (g) k-means, (h) Gene Cluster, (i) FOOS	81
Figure 4.1:	The ground truth of the image	90
Figure 4.2:	Percentile population of the eight classes and the non-class	91
Figure 4.3:	The regions selected for classifier's training (filled rectangles) and the square test regions (hollow squares)	92
Figure 4.4:	test region 1 and (b) its ground truth	93
Figure 4.5:	(a) The test region 2 and (b) its ground truth	93
Figure 4.6:	(a) The test region 3 and (b) its ground truth	93
Figure 4.7:	(a) The test region 4 and (b) its ground truth	93
Figure 4.8:	(a) The test region 5 and (b) its ground truth	94
Figure 4.9:	(a) The test region 6 and (b) its ground truth	94
Figure 4.10:	(a) The test region 7 and (b) its ground truth	94
Figure 4.11:	(a) The test region 8 and (b) its ground truth	94
Figure 4.12:	(a) The test region 3, its segmentation using (b) templates, (c) texture signatures, and (d) spectral signatures	96
Figure 4.13:	The test region 4, its segmentation using (b) templates, (c) texture signatures, and (d) spectral signatures	96
Figure 4.14:	(a) the test region 3, (b) its ground truth, (c) its grid classification with the block size of 17×17 pixels	100
Figure 4.15:	(a) the test region 6, (b) its ground truth, (c) its grid classification with block size of 17×17 pixels	100
Figure 4.16:	The process of fusing the region/frontier and the land-cover information	101
Figure 4.17:	Scheme of fusing the spectral information and the textural information	102
Figure 4.18:	(a) the test region 1, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image	108
Figure 4.19:	(a) the test region 2, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image	108

Figure 4.20:	(a) the test region 3, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image	109
Figure 4.21:	(a) the test region 4, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image	109
Figure 4.22:	(a) the test region 5, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image	110
Figure 4.23:	(a) the test region 6, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image	110
Figure 4.24:	(a) the test region 7, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) the fused classification map, and (f) the error image	111
Figure 4.25:	(a) the test region 8, (b) its ground truth, (c) its segmentation, (d) its grid classification map, (e) fused classification map, and (f) the error image	111
Figure 4.26:	(a) the test region 6, (b) its corresponding ground truth, (c) its segmentation after polygon filling, (d) grid-classification map, (e) fused classification map, and (f) the error image	113





## List of pictures

Picture 1.1:	An IKONOS Panchromatic image of Washington DC [Jensen 2005]	7
Picture 1.2:	A true colour IKONOS image	7
Picture 1.3:	An image remotely sensed in two visible and one NIR band displayed and printed in one RGB colour space	9
Picture 1.4:	Natural colour composite of a SPOT image	9
Picture 1.5:	A high-resolution airborne SAR image of Washington DC	11
Picture 1.6:	Three-band SAR image of New York City metropolitan area [Rees 2001]	12
Picture 1.7:	LIDAR images (a) shows the canopy heights in green shades, (b) shows the surface terrain	13
Picture 4.1:	The panchromatic remote sensing image captured by QuickBird	89
Picture 4.2:	The MS image in false colour composite	89



## List of tables

Table 1.1:	Sources of satellite-borne optical images from different nations (Note: All the measurements are in $\mu$ meters)	8
Table 1.2:	Characteristics of AVHRR sensor of remote sensing satellites from NOAA series	10
Table 1.3:	Simplified classification of vegetation in Victoria	17
Table 1.4:	Subclasses of wetland in the region of St. Lawrence, Canada	18
Table 1.5:	Typical NDVI values for various land-covers	20
Table 2.1:	The templates to extract the DFT-based texture signatures	36
Table 2.2:	XCC between the histograms of $\phi_2$ and $\phi_3$ corresponding to $0^\circ$ and to $30^\circ$ , $45^\circ$ , $60^\circ$ , $90^\circ$ averaged over all the images from Brodatz album	40
Table 2.3:	The SNR values in decibels for the images with Gaussian white noise	48
Table 2.4:	Recognition results using the training window size of $71 \times 71$ pixels	51
Table 2.5:	Overall and orientation-wise recognition results using the training window size of $45 \times 45$ pixels	51
Table 2.6:	Overall and orientation-wise recognition results using the training window size of $27 \times 27$ pixels	51
Table 2.7:	Overall and orientation-wise recognition results using the training window size of $71 \times 71$ pixels	52
Table 2.8:	Overall and orientation-wise recognition results on the smaller set of test images using the training window size of $45 \times 45$ pixels	52
Table 2.9:	Overall and orientation-wise recognition results using the training window size of $27 \times 27$ pixels	53
Table 2.10:	Recognition results on the smaller set of test images with Gaussian noise	57
Table 3.1:	Statistics of the clusters formed after over-segmentation of the image in 5 clusters (left column) and 6 clusters (top row). The other cells in the table show the number of common pixels ( $T_{mn}$ ) found in the 2 over-segmentation, with the highest 4 numbers in bold	69
Table 3.2:	The list and details of the images with a collage of four Brodatz textures each. The last column shows how many times the Gene Cluster found clusters using k-means.	72
Table 3.3:	Ten more segmentation results on the images from Berkley segmentation database	75
Table 3.4:	The segmentation results on five images from Defence Image database	79
Table 3.5:	FOOS Results with different over-segmentations	82
Table 4.1:	The land-covers in the source RS image	91
Table 4.2:	Description of the test regions	92
Table 4.3:	The training regions	98
Table 4.4:	Details of training the kNN classifiers – number of feature vectors representing each class for a given window-size	99
Table 4.5:	The EM for the grid classification of the test region 3 shown in figure 4.10	103
Table 4.6:	The EM for the grid classification for test region 6 shown in figure 4.11	104
Table 4.7:	The PM corresponding to the EM of table 4.5	105
Table 4.8:	The PM corresponding to the EM of table 4.6	105

Table 4.9:	The statistics derived from EM of table 4.5 and PM of table 4.7	106
Table 4.10:	The statistics derived from EM of table 4.6 and PM of table 4.8	106
Table 4.11:	The classification results in terms of Kappa statistics for eight test regions	107
Table 4.12:	The Kappa statistics for region 6, after polygon filling	113
Table B:	Appendix B: Details on area and locations of the training regions in the source RS image	118

## List of publications

### Journal Papers

- [1] **Texture features based on local Fourier histogram: self-compensation against rotation**, Journal of Electronic Imaging, Vol. 17, 030503 (2008); DOI:10.1117/1.2965439
- [2] **Fusion of over-segmentations for improved clustering**, (to be submitted soon to Data Mining and Knowledge Discovery, Springer)
- [3] **Fusion of classification and segmentation as applied to very high resolution remote sensing images**, (to be submitted soon to International Journal of Remote Sensing, the Remote Sensing and Photogrammetry Society)

### Conference Papers

- [4] **Over-segmentation of Feature Space for Initialization of k-means clustering**, World Congress on Engineering and Computer Engineering, San Francisco, USA, pp. 735-740, 2007
- [5] **Texture features based on Fourier transform and Gabor filters: an empirical comparison**, Int. Conf. on Machine Vision (ICMV), Islamabad, Pakistan, pp. 67-72, 2007
- [6] **Improved Texture Description with Features based on Fourier Transform**, International Multi-Topic Conference, Jamshoro, Pakistan, April 2008



## Bibliography

- [Adegoke 2002] Jimmy O. Adegoke and Andrew M. Carleton, "Relations between Soil Moisture and Satellite Vegetation Indices in the U.S. Corn Belt", *Journal of Hydrometeorology*, American Meteorological Society 3: 395-405, 2002
- [Akono 2000] Alain Akono, André Ndi Nyoungui, Emmanuel Tonyé, Jean-Paul Rudant, "Utilization of radar remote sensing and texture analysis for Urban cartography in cameroon: application to yaounde city", ERS – ENVISAT symposium, Gothenburg, 2000
- [Akbari 2006] M. Akbari, A. R. Mamanpoush, A. Gieske, M. Miranzadeh, M. Torabi, H. R. Salemi, "crop and land cover classification in Iran using Landsat 7 imagery", *int'l Journal of Remote Sensing* 27(19), 4117-4135, 2006
- [Alon 1999] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine, "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays", *PNAS* 8, (96) 12: 6745-6750, 1999
- [Arof 1998] H. Arof, F. Deravi, "Circular neighborhood and 1-D DFT features for texture classification and segmentation", *IEE Proc. Vision, Signal processing* 145: 167-172, 1998
- [Baek 2005] Sangho Baek, Oh-Ig Kwoun, Alexander Braun, Zhong Lu, and C. K. Shum, "Digital Elevation Model of King Edward VII Peninsula, West Antarctica, from SAR Interferometry and ICESat Laser Altimetry", *IEEE Geo-science and Remote Se Sensing Letters* 2 (4): 413 – 417, 2005
- [Bannari 2002] A. H. Bannari, Asalhi, and P. M. Teillet, "Transformed Difference Vegetation Index (TDVI) for Vegetation Cover Mapping", *Proceedings of the IEEE IGARSS and 24<sup>th</sup> Canadian Symposium on Remote Sensing*, 2002
- [Belongie 2002] Serge Belongie, Jitendra Malik, Jan Puzicha, "Shape matching and object recognition using shape context", *IEEE trans. On pattern recognition and machine intelligence* 24(24), 2002
- [Birth 1968] Birth G. S. and G. McVey, "Measuring the color of growing turf with a reflectance spectroradiometer", *Agronomy Journal*, 60:640-643, 1968
- [Borghys 1998] D. Borghys, P. Verlinde, C. Perneel, and M. Acheroy, "Multilevel Data Fusion for Detection of Targets using Multi-spectral Image Sequences", *Optical Fusion 1998*
- [Brekke 2005] Camilla Brekke, Anne H. S. Solberg, Oil spill detection by satellite remote sensing, *Remote Sensing of Environment* 95: 1-13, 2005
- [Breukelen 1998] M. Van Breukelen and R. P. W. Duin, "Neural Network Initialization by combined classifiers", *Proc. 14<sup>th</sup> int. Conf. Pattern Recognition* 1:215, 1998
- [Brodatz 1966] Phil Brodatz, "Textures: A photographic album for artists and designers (Book)", Dover Publications, 1966
- [Bruzzone 2002] Lorenzo Bruzzone, Roberto Cossu, Gianni Vernazza, "Combining parametric and non-parametric algorithms for a partially unsupervised classification of multitemporal remote-sensing images", *Information Fusion* 3(4): 289-297, 2002
- [Cablk 2007] Marry E. Cablk, Christopher Kratt, "A methodology for mapping Shrub Canopy Cover in the Great Basin Desert using High Spatial Resolution Satellite Imagery", Publication No. 41236, Desert Research Institute, Nevada System of Higher Education, 2007
- [Chang 2002] J. H. Chang, K. C. Fan, and Y. L. Chang, "Multimodal gray-level histogram modelling and decomposition", *Image and Vision Computing* 20:203-216, 2002
- [Chen 1998] C. H. Chen, L. F., Pau, P. S. P. Wang (eds.), Chapter 2: Textural Analysis, *The handbook of Pattern Recognition and Computer Vision* (2<sup>nd</sup> Edition), World Scientific Publishing Co., 1998

- [Chen 2005] Bernard Chen, Phang C. Tai, R. Harrison, and Yi Pan, "Novel Hybrid Hierarchical-K-means Clustering Method for Microarray Analysis", Proceedings of IEEE Computation Systems Bioinformatics Conference Workshops, 2005
- [Chen 2005b] P. Chen, S. C. Liew and L. K. Kwok, "Tsunami Damage Assessment Using High Resolution Satellite Imagery: A Case Study of Aceh, Indonesia", Proc. Geoscience and Remote Sensing Symposium, IGARSS '05, IEEE 2: 1405-1408, 2005
- [Chen 2005c] Shui-sen Chen, Liang-fu Chen, Qin-huo Liu, Xia Li, Qiyu Tan, "Remote sensing and GIS-based integrated analysis of coastal changes and their environmental impacts in Lingding Bay, Pearl River Estuary, South China", Ocean & Coastal Management 48: 65-83, 2005
- [Cicala 2004] Luca Cicala, Giovanni Poggi, and Giuseppe Scarpa, "Supervised Segmentation of Remote-Sensing Multitemporal Images based on the Tree-Structured Markov Random Field Model", :1569-1572, 2004
- [Curran 1987] Curran, Paul J., Remote Sensing in Agriculture: An Introductory Review, Journal of Geography, vol: 86 (4): 147-56, 1987
- [Cutter 2003] G. R. Cutter Jr., Y. Rzhannov, L. A. Mayer, "Automated segmentation of seafloor bathymetry from multibeam echosounder data using local Fourier histogram texture features", Journal of Experimental Marine Biology and Ecology, 355-370, 2003
- [Clausi 2001] David A. Clausi, "Comparison and Fusion of Co-occurrence, Gabor and MRF Texture Features for Classification of SAR Sea-Ice Imagery, ATMOSPHERE-OCEAN 39 (3): 183-194, 2001
- [DiGiacomo 2004] Paul M. DiGiacomo, Libe Washburn, Benjamin Holt, Burton H. Jones, "Coastal pollution hazards in southern California observed by SAR imagery: stormwater plumes, wastewater plumes, and natural hydrocarbon seeps", Marine Pollution Bulletin 49:1013-1024, 2004
- [Do 2002] Minh N. Do and Martin Vetterli, "Rotation Invariant Texture Characterization and Retrieval using Steerable Wavelet-Domain Hidden Markov Models", IEEE Transactions on Multimedia, 4(4):517-527, 2002
- [Doraiswamy 2007] Paul C. Doraiswamy, Alan J. Stern, Bakhyt Akhmedov, "Crop Classification in the U.S. Corn Belt Using MODIS Imagery", Presentation at the International Geoscience and Remote Sensing Symposium, 2007
- [Eduardo 2005] Eduardo R. Hruschka, Estevam R. Hruschka, Thiago F. Covoes, Nelson F. F. Ebecken, "Feature Selection for clustering problems: a hybrid algorithm that iterates between k-means and a Bayesian filter", Int'l Conf. on hybrid intelligent systems (5): 405-410, 2005
- [Elber 2005] Gershon Elber, "Gemoetric Texture Modelling", IEEE Computer Graphics and Applications 25(4): 66-76, 2005
- [Fingas 1997] Mervin F. Fingas and Carl E. Brown, "Review of Oil Spill Remote Sensing", Spill Science and Technology Bulletin 4(4): 199-208 (10), Elsevier, 1997
- [Feng 2006] Jianmin Feng, Tao Wang and Changwei Xie, Eco-Environmental Degradation in the Source Region of the Yellow River, Northeast Qinghai-Xizang Plateau, Environmental Monitoring and Assessment, 122 (1-3):125-143, 2006
- [Franke 2006] J Franke, V Heinzl, G Menz, "Assessment of NDVI-Differences Caused by Sensor Specific Relative Spectral Response Functions", Geoscience and Remote Sensing Symposium (IGARSS), 2006
- [Geerling 2007] G. W. Geerling, M. Labrador-Garcia, J. G. P. W. Clevers, A. M. J. Ragas, and A. J. M. SMITS, "Data-fusion of Spectral (CASI) and LiDAR data", International Journal of Remote Sensing 28 (19): 4263 - 4284, 2007
- [Gloaguen 2007] R. Gloaguen, P. R. Marpu, and I. Niemyer, "Automatic extraction of faults and fractal analysis from remote sensing data", Nonlinear. Processes in Geophysics, 14: 131-138, 2007



- [Gnauck 1999] Gary E. Gnauck, Commercial use of remote sensing in agriculture: a case study, Proc. SPIE 3870: 632-640, Sensors, Systems, and Next-Generation Satellites III, Hiroyuki Fujisada; Joan B. Lurie; Eds. SPIE, 1999
- [Gordon 1983] H. R. Gordon, A. Y. Morel, "Remote assessment of ocean color for interpretation of satellite visible imagery: A review", Springer-Verlag (Lecture Notes on Coastal and Estuarine Studies 4: 118, 1983
- [Goshtasby 2005] A. A. Goshtasby, "Fusion of multi-exposure images", Image and Vision Computing, Elsevier, 611-618, 2005
- [Greiwe 2005] A. Greiwe, M. Ehlers, "Combined Analysis of Hyperspectral and High Resolution Image Data in an Object Oriented Classification Approach", 3rd International Symposium Remote Sensing and Data Fusion Over Urban Areas (URBAN), 2005
- [Haboudane 2004] Haboudane Driss, Miller John R., Pattey Elizabeth, Zarco-Tejada Pablo J., Strachan, Ian B., "Hyperspectral vegetation indices and novel algorithms for predicting green LAI of crop canopies: Modelling and validation in the context of agriculture", Remote Sensing of Environment, 90: 337-352, 2004
- [Hall 2003] David L. Hall, Perspectives on the Fusion of Image and Non-image Data, Applied Imagery Pattern Recognition Workshop, pp. 217, 2003
- [Harman 2002] Lawrence J. Harman, Uma Shama, Kirk Dand, Brendan Kidwell, "Remote sensing and spatial information for transportation demand management (TDM) assessment", Pecora 15/Land Satellite Information IV/ISPRS Commission I/FIEOS 2002 Conference Proceedings
- [Heinz 2003] Grete Heinz, Louis J. Peterson, Roger W. Johnson, and Carter J. Kerk, "Exploring relationships in body dimensions", Journal of Statistics Education, 11(2), 2003
- [Herold 2002] Herold Martin, Scepan Joseph, André Müller, "Object-oriented mapping and analysis of urban land use/cover using IKONOS data", EARSEL symposium on Geoinformation on Europe-wide integration, 2002
- [Hirose 2004] Y. Hirose, M. Mori, Y. Akamatsu, Y. Li , Vegetation Cover Mapping Using Hybrid Analysis of IKONOS Data, Isprs 2004
- [Hoon 2004] M. J. L. de Hoon, S. Imoto, J. Nolan, and S. Miyano, "Open Source Clustering Software", *Bioinformatics*, 20 (9): 1453-1454, 2004.
- [Huete 1988] A. R. Huete, "A Soil-Adjusted Vegetation Index", Remote Sensing of Environment 25: 295-309, 1998
- [Hunt 2005] Hunt Earle, Cavigelli Michel Daughtry, Craig McMurtrey Iii James, Walthall Charles, "Evaluation of Aerial Photography from Model Aircraft for Remote Sensing Crop Biomass and Nitrogen Status", Precision Agriculture 6(4): 359-378, 2005
- [Hyer 1999] Heyer, L.J., Kruglyak, S. and Yooseph, S., Exploring Expression Data: Identification and Analysis of Coexpressed Genes, Genome Research 9:1106-1115, 1999.
- [Inaba 1994] M. Inaba, N. Katoh, and H. Imai, "Applications of weighted Voronoi diagrams and randomization to variance-based k-clustering", In Proc. 10<sup>th</sup> Annu. ACM Sympos. Comput. Geom., 332-339, 1994.
- [Jensen 1999] Jensen, J.R. and Cowen, D.C., "Remote Sensing of urban/suburban infrastructure and socio-economic attributes", Photogrammetric Engineering and Remote Sensing 65(5): 611-622, 1999
- [Jensen 2003] Troy Jensen, Armando Apan, Frank Young, Les Zeller and Ken Cleminson, "Assessing grain crop attributes using digital imagery acquired from a low-altitude remote controlled aircraft", Spatial Sciences 2003
- [Jensen 2005] John R. Jensen, "Introductory digital image processing, a remote sensing perspective" (Book), 3<sup>rd</sup> edition, Prentice Hall series in Geographic Information Series, 2005
- [Jones 2007] Carol L. Jones, P. R. Weckler, Niels O. Maness, Marvin L. Stone, Roshani Jayasekara, "Remote sensing to estimate chlorophyll concentration in spinach using multi-spectral plant reflectance", Paper number 043081, 2004 ASABE Annual Meeting, 2007

- [Julesz 1973] B. Julesz, E. N. Gilbert, L. A. Shepp, H. L. Frisch, "Inability of humans to discriminate between visual textures that agree in second-order statistics - revisited", *Perception* 2: 391-405, 1973
- [Kaufman 1992] Y. J. Kaufman, D. Tanre, "Atmospherically resistant vegetation index (ARVI) for EOS-MODIS", *Proc. IEEE Int. Geoscience and Remote Sensing Symposium*, 261-270, 1992
- [Kauth 1976] Kauth, R. J. and Thomas, G.S. "The tasseled cap--A graphic description of the spectral-temporal development of agricultural crops as seen by Landsat", *Proceedings of the Symposium on Machine Processing of Remotely Sensed Data*, Purdue University, West Lafayette, Indiana, 41-51, 1976
- [Koetz 2007] Benjamin Koetz, Guoqing Sun, Felix Morsdorf, K. J. Ranson, Mathias Kneubühler, Klaus Itten, Britta Allgöwer, "Fusion of imaging spectrometer and LIDAR data over combined radiative transfer models for forest canopy characterization", *Remote Sensing of Environment* 106: 449-459, 2007
- [Kotsiantis 2007] S. B. Kotsiantis, "Supervised Machine Learning: A review of classification techniques", *Informatica* 31: 249-268, 2007
- [Kriegler 1969] F. J. Kriegler, W. A. Malila, R. F. Nalepka, and W. Richardson, "Preprocessing transformations and their effects on multispectral recognition", in *proceedings of the sixth international symposium on remote sensing of environment*, University of Michigan, Ann Arbor, MI, 97-131, 1969
- [Laba 2008] Magdeline Laba, Roger Downs, Stephen Smith, Sabrina Welsh, Chuck Neider, Susan White, Milo Richmond, William Philpot, Philippe Baveye, "Mapping invasive wetland plants in the Hudson River National Estuarine Research Reserve using QuickBird satellite imagery", *Remote Sensing of Environment* 112, pp. 286-300, 2008
- [Lee 1996] Tai Sing Lee, "Image representation using 2D Gabor wavelets", *IEEE Transactions on pattern analysis and machine intelligence* 18 (10), 1996
- [Lelong 2003] Camille C. D. Lelong and Audrey Thong-Chane, "Application of textural analysis on very high resolution panchromatic images to map coffee orchards in Uganda", *Geoscience and Remote Sensing Symposium, proc. IGARSS '03, IEEE*, 2003
- [Likas 2003] A. Likas, N. Vlassis, J. J. Verbeek. "The global k-means clustering algorithm", *Pattern Recognition* 36, 451-461, 2003
- [Liu 2006] Liu Xongxue, Li Manchun, Mao Liang, Xu Feifei, Huang Shuo, "Review of Remotely Sensed Imagery Classification Patterns based on Object-oriented Image Analysis", *Chinese Geographical Sciences* 16(3): 282-288, 2006
- [Lozano 1999] J. A. Lozano, J. M. Pena, P. Larrañaga, "An empirical, comparison of four initialization methods for the k-means, algorithm", *Pattern Recognition Letters* 20: 1027-1040, 1999.
- [Lu 1996] Jian Lu, Taiichi Yuasa, "A New Algorithm for 2D Convolution on Mesh-connected SIMD Computers", *Transactions of Information Processing Society of Japan* 37 (12): 2390-2397, 1996
- [Lucchese 2001] L. Lucchese, L. S. K. Mitra, "Colour Image Segmentation: A State-of-the-Art Survey", *proc. Indian National Science Academy, Part A*, 67(2): 207-222, 2001
- [Lu 2008] J. F. Lu, J. B. Tang, Z. M. Tang, J. Y. Yang, "Hierarchical initialization approach for K-Means clustering", *Pattern Recognition Letters* 29 (2008) 787-795
- [Mandelbrot 1982] Benoît Mandelbrot, "The Fractal Geometry of Natures", W. H. Freeman, 1982
- [Manjunath 1996] B. S. Manjunath and W. Y. Ma, "Texture Features for Browsing and Retrieval of Image data", *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18(8): 837-842, 1996
- [Martin 2001] David R. Martin, Charless Fowlkes, Doron Tal and Jitendra Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics", *proceedings of 8<sup>th</sup> International conference on computer vision* (2):416-423, 2001

- [Marshall 2004] Geoffrey Marshall and Xiaobing Zhou, “Drought Detection in Semi-Arid Regions using Remote Sensing of Vegetation Indices and Drought Indices”, Proc. IEEE International Geoscience and Remote Sensing Symposium 3:1555-1558, 2004
- [Mathiew 1998] D. B. Mathiew, “Synthetic Aperture Radar in the Role of Surveillance”, GEC Review 13(3), 1998
- [Matsuoka 2004] M. Matsuoka and F. Yamazaki, “Use of Satellite SAR Intensity Imagery for Detecting Building Areas Damaged due to Earthquakes, Earthquake Spectra”, EERI 20(3): 975-994, 2004
- [McQueen 1967] J. B. McQueen, “Some Methods for Classification and Analysis of Multivariate Observations”, Proc. 5<sup>th</sup> Berkeley Symposium on Mathematics and Probability, 281-297, University of California Press, Berkeley, CA, 1967
- [Miguel 1994] Miguel A. Mayorga and Lonnie C. Ludeman, “Shift and Rotation Invariant Texture Recognition with Neural Nets”, IEEE International Conference on Neural Networks, 1994
- [Mumby 2004] P. J. Mumby, J. D. Hedley, J. R. M. Chisholm, C. D. Clark, H. Ripley, J. Jaubert, “The cover of living and dead corals from airborne remote sensing”, Coral Reefs 23: 171–183, 2004
- [Murphy 1994] P. M. Murphy and D. W. Aha, “UCI Repository of Machine Learning databases”, Irvine, CA: University of California, Dept. of Information & Computer Science, 1994
- [Myneni 1995] Myneni, R. B., F. G. Hall, P.J. Sellers, and A.L. Marshak, “The interpretation of spectral vegetation indexes”, IEEE Transactions on Geoscience and Remote Sensing 33, 481-486, 1995
- [Ogawa 1999] Susumu Ogawa, Takuhiko Murakami, Naoki Ishitsuka, and Genya Saito, “Evapotranspiration Estimates from fine-resolution NDVI”, Asian Conference on Remote Sensing (ACRS), 1999
- [Pantani 2000] Luca Pantani, Gaia Ballerini, Giovanna Cecchi, Hans Edner, David Lognoli, Thomas Johansson, Valentina Raimondi, Sune Svanberg, Piero Tiano, Luisa Tomaselli and Peter Weibring, “Experiments on stony monument monitoring by laser-induced fluorescence”, Journal of Cultural Heritage 1(1): S345-S348, 2000
- [Petja 2004] Brilliant M. Petja, Johan Malherbe, Dawie van Zyl, “Using satellite imagery and rainfall data to track climate variability in South Africa”, IEEE Geoscience and Remote Sensing Symposium, IGARSS'04, 2004
- [Petrovic 2003] Vladimir Petrovic, “Multilevel Image Fusion”, Proc. SPIE Vol. 5099, Multisensor, Multisource Information Fusion: Architectures, algorithms, and applications, 87-96, 2003
- [Pettorelli 2006] Nathalie Pettorelli, Jean-Michel Gaillard, Atle Mysterud, Patrick Duncan, Nils Chr. Stenseth, Daniel Delorme, Guy Van Laere, Carole Toigo, Francois Klein “Using a proxy of plant productivity (NDVI) to find key periods for animal performance: the case of roe deer”, Oikos 112 (3): 565–572, 2006
- [Phol 1998] C. Phol, and J. L. van Genderen, “Multisensor image fusion in remote sensing: concepts, methods and applications”, International Journal of Remote Sensing 19(5): 823-854, 1998
- [Piella 2003] Gemma Piella, “A General Framework for Multi-Resolution Image Fusion: from Pixels to Regions”, Information Fusion 4: 259– 280, 2003
- [Pun 2003] Chi-Man Pun, “Rotation-invariant texture feature for image retrieval”, Computer Vision and Image Understanding 89: 24-43, 2003
- [Qi 1994] J. Qi, A. Chehbouni, A. R. Huete, and Y. H. Kerr, “Modified Soil Adjusted Vegetation Index (MSAVI)”, Remote Sensing of Environment 48: 119-126, 1994
- [Qi 1994b] J. Qi, Y. Kerr, and A. Chehbouni, “External Factor Consideration in Vegetation Index Development”, in Proc. of Physical Measurements and Signatures in Remote Sensing, ISPRS, 723-730, 1994

- [Qu 2001] Guihong Qu, Dali Zhang and Pignfan Yan, "Medical Image Fusion by Wavelet transform modulus maxima", *Optics Express* 9(4): 184-190, 2001
- [Ranchin 2000] T. Ranchin and L. Wald, "Fusion of high spatial and spectral resolution images: the ARSIS concept and its implementation", *Journal of Photometric Engineering and Remote Sensing* 66 (8), 49-61, 2000
- [Raymond 2006] Crystal L. Raymond, Lara-Karena B. Kellogg, and Donald McKenzie "Mapping Fuels on the Okanogan and Wenatchee National Forests", *USDA Forest Service Proceedings RMRS-P-41:293-304*, 2006
- [Raymond 1994] Raymond T. Ng., Jiawei Han, "Efficient and Effective Clustering Methods for Spatial Data Mining", *Intl. conf. on very large databases*: 144-155, 1994
- [Rees 2001] W. G. Rees, *Physical Principles of Remote Sensing (Book)*, 2<sup>nd</sup> edition, Cambridge University Press, pp. 246-249, 2001
- [Rouse 1973] J. W. Rouse, R. H. Hase, J. A. Schell, D. W. Deering, "Monitoring vegetation systems in the great plain with ERTS", in the proceedings of the third ERTS symposium, US Government printing office, NASA, Washington, DC, 309–317, 1973.
- [Sankaragomathi 2007] B. Sankaragomathi, L. Ganesan, and S. Arumugam, "Fractal Image Compression Applied to Remote Sensing", *Proceedings of World Academy of Science, Engineering and Technology* 21:1307-6884, 2007
- [Sande 2003] C. J. van der Sande, S.M. de Jong, A.P.J. de Roo, A segmentation and classification approach of IKONOS-2 imagery for land cover mapping to assist flood risk and flood damage assessment, *International Journal of Applied Earth Observation and Geoinformation* 4: 217-229, 2003
- [Schaale 2000] M. Schaale, I. Keller, and J. Fischer, "Land cover texture information extraction from remote sensing image data", *proc. ASPRS-RTI annual conference*, Washington DC, 2000
- [Schröder 1998] Michael Schröder, Hubert Rehrauer, Klaus Seidel, and Mihai Datcu, "Spatial Information Retrieval from Remote-Sensing Images—Part II: Gibbs–Markov Random Fields", *IEEE Transactions on Geo-science and remote sensing* 36(5):1446-1455, 1998
- [Scott 2003] Scott J. Goetz, Robb K. Wright, Andrew J. Smith, Elizabeth Zinecker, Erika Schaub, "IKONOS imagery for resource management: Tree cover, impervious surfaces, and riparian buffer analyses in the mid-Atlantic region", *Remote Sensing of Environment*, 2003
- [Scott 2008] E. Scott Flynn, Charles T. Dougherty and Ole Wendroth, "Assessment of pasture biomass with the normalized difference vegetation index from active ground-based sensors", published in *Agronomy Journal* 100:114-121, 2008
- [Sellers 1985] Sellers, P. J. "Canopy reflectance, photosynthesis, and transpiration", *International Journal of Remote Sensing* 6: 1335-1372, 1985
- [Shackelford 2003] Aaron K. Shackelford, and Curt H. Davis, "Fully Automated Road Network Extraction from High-Resolution Satellite Multispectral Imagery", 2003
- [Shi 2000] Jianbo Shi, Jitendra Malik, "Normalized Cuts and Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22 (8): 888-905, 2000
- [Simone 2002] G. Simone, A. Farina, F. C. Morabito, S. B. Serpico and L. Bruzzone, "Image fusion techniques for remote sensing applications", *Information Fusion* 3(1): 3-15, 2002
- [Stamos 2002] Ioannis Stamos and Peter K. Allen, "Geometry of Texture Recovery of Scenes of Large Scale", *Computer Vision and Image Understanding* 88: 94-118, 2002
- [Steinwolf 1993] Alexander Steinwolf, "Software for kurtosis and skewness fitting by way of piecewise-Gaussian approximation", *Bulletin of the international statistical institute*, contributed papers of 49<sup>th</sup> ISI session, Florence, Book 2, p. 433-434, 1993.
- [Steinwolf 1996] Alexander Stienwolf, "Approximation and simulation of probability distributions with a variable kurtosis value", *Computational Statistics and Data Analysis* 21:163-180, Elsevier,1996

- [Steinwolf 2006] Alexander Stienwolf and Stephen A. Rizzi, "Non-Gaussian analysis of turbulent boundary layer fluctuating pressure on aircraft skin panels", *Journal of Aircraft*, 43(6):1662-1675, 2006
- [Stubberud 2003] C. S. Stubberud, P. J. Shea, Klamer, M. Dale, "Data Fusion: A Conceptual Approach to Level 2 Fusion (Situational Assessment)", *Proceedings of the SPIE (5096)*, 455-462, 2003
- [Tatiana 2006] Tatiana Plantier, Marta Loureiro, Pedro Marques and Mario Caetano, *Spectral Analysis and Classification of IKONOS images for forest cover characterisation, 2<sup>nd</sup> Workshop of the EARSel SIG on land use and Land Cover*, 2006
- [Thomasson 2001] Thomasson A., Mondelain D., Ménard T., Nominé M., Godet Y., Wolf J., "Air pollution mapping using all solid-state LIDAR systems", *C.R. Academy of Science* 7(IV): 923-929, 2001
- [Tian 2004] J. Tian, Y. Zheng, and L. Shen, "A Comparative Study on Wavelet based Pixel Level Image Fusion," *Signal and Image Processing*, 444-112, 2004
- [Toyoda 2005] T. Toyoda and O. Hasegawa, "Texture Classification using Extended Higher Order Local Autocorrelation Features", *Proceedings of the 4<sup>th</sup> Int. Workshop on Texture Analysis and Synthesis*, 131-136, 2005
- [Tran 2002] Tran Annelise, Gardon Jacques, Weber Sacha, Polidori Laurent, "Mapping disease incidence in suburban areas using remotely sensed data", *American Journal of Epidemiology*, 156(7): 662-668, 2002.
- [Trigg 2005] S. N. Trigg, D. P. Roy, S. P. Flasse, "An in situ study of the effects of surface anisotropy on the remote sensing of burned savannah", *International Journal of Remote Sensing* 26(21): 4869-4876, 2005
- [Tsiligirides 1998] T.A. Tsiligirides, *Remote sensing as a tool for agricultural statistics: a case study of area frame sampling methodology in Hellas*, *Computers and Electronics in Agriculture* 20: 45-77, 1998
- [Ulrich 1998] Ulrich P., "Laser remote sensing", *Trends in Analytical chemistry* 17(8+9), 491-500, 1998
- [Ursani 2007] Ahsan Ahmad Ursani, Kidiyo Kpalma, Joseph Ronsin, "Over-segmenting the feature space for improved initialisation of k-means clustering", *Intl. conf. Machine Learning and Data Analysis, WCECS*, 2007
- [Ursani 2008a] Ahsan Ahmad Ursani, Kidiyo Kpalma, Joseph Ronsin, "Improved texture features based on local Fourier histogram", *IMTIC 2008*
- [Ursani 2008b] Ahsan Ahmad Ursani, Kidiyo Kpalma, Joseph Ronsin, "Texture features based on local Fourier histogram: self-compensation against rotation", *Journal of Electronic Imaging* 17(3) / *JEI Letters*, 2008
- [Varma 2003] Manik Varma, Andrew Zisserman, "Texture Classification: Are Filter Banks Necessary?", *proceedings Conference on Computer Vision and Pattern Recognition* 2: 691-698, 2003
- [Varshney 1997] P. K. Varshney, "Multi-sensor data fusion", *Electronics and Communication Engineering Journal* (9) 6: 245-253, 1997
- [Vasconcelos 2006] Manuela Vasconcelos, Nuno Vasconcelos, Gustavo Carneiro, "Weakly Supervised Top-down Image Segmentation", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (1): 1001-1006, 2006
- [Verbyla 1995] David Verbyla, *Satellite Remote Sensing of Natural Resources*, LEWIS publishers, 1995
- [Wald 1999] Lucien Wald, "Some terms of reference in data fusion", *IEEE Transactions on Geoscience and Remote Sensing* (37) 3: 1190-1193, 1999
- [Wang 2004] L. Wang, W.P. Sousa, P. Gong and G.S. Biging, "Comparison of IKONOS and QuickBird images for mapping mangrove species on the Caribbean coast of Panama", *Remote Sensing of Environment* 91: 432-440, 2004

- [Wang 2005] Zhijun Wang, Djemel Ziou, Costas Armenakis, Daren Li, and Qingquan Li, "A comparative analysis of image fusion methods", IEEE transactions on Geoscience and Remote Sensing 43(6): 1391-1402, 2005
- [Ward 2000] M. H. Ward, J. R. Nuckols, S. J. Weigel, S. K. Maxwell, K. P. Cantor, and R. S. Miller, Identifying populations potentially exposed to agricultural pesticides using remote sensing and a Geographic Information System, Environmental Health Perspectives 108(1): 5-12, 2000.
- [Westra 2004] Toon Westra, Koen C. Mertens and Robert R. De Wulf, "Wavelet-Based Fusion of Spot/Vegetation and Envisat/ ASAR Wide Swath Data for Wetland Mapping", 2004
- [Wein 2005] W. Wein, B. Roper, and Nassir Navab, "Automatic Registration and Fusion of Ultrasound with CT for Radiotherapy", 8th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), Palm Springs, California USA, 2005
- [Winker 2004] Winker D. and Pelon J., "LIDAR remote sensing of aerosols and Clouds from space", 35<sup>th</sup> COSPAR scientific assembly COSPAR04, 2004
- [Wright 2007] Chris Wright, Alisa Gallant, Improved wetland remote sensing in Yellowstone National Park using classification trees to combine TM imagery and ancillary environmental data, Remote Sensing of Environment 107: 582-605, 2007
- [Xia 2005] Yong Xia, (David) Dagan Feng, and Rongchun Zhao, "Semi-supervised Segmentation of Textured Images by Using Coupled MRF Model", proceedings of IEEE Region 10 Conference (TENCON'05), Melbourne, 81-85, 2005
- [Xiao 2003] Xiangming Xiao, Bobby Braswell, Qingyuan Zhang, Stephen Boles, Stephen Froking, Berrien Moore III, "Sensitivity of vegetation indices to atmospheric aerosols: continental-scale observations in Northern Asia", Remote Sensing of Environment 84: 385-392, 2003
- [Xue 2003] Zhiyun Xue, Rick S. Blum, "Concealed Weapon Detection Using Color Image Fusion", Fusion 2003, ISIS, IEEE and AES 2003
- [Zhang 1997] Z. Zhang and R. Blum, A Region-based Image Fusion Scheme for Concealed Weapon Detection, proc. 5<sup>th</sup> annual conf. Information Sciences and Systems, 168-173, 1997
- [Zhang 2000] Dengsheng Zhang, Aylvin Wong, Maria Indrawan, Guojun Lu, "Content-based Image Retrieval Using Gabor Texture Features", IEEE Intern. Symposium on Multimedia Information Processing, 2000
- [Zhang 2002] Yun Zhang, "Problems in the fusion of commercial high-resolution satellite images as well as Landsat 7 images and initial solutions", International Archives of Photogrammetry and Remote Sensing (IAPRS) 34 (4) "GeoSpatial Theory, Processing and Applications", 2002
- [Zhao 2005] Yindi Zhao, Liangpei and Zhang, Pingxiang Li, "Texture Feature Fusion for High Resolution Satellite Image Classification", Proc. Computer Graphics, Imaging and Vision (CGIV): New trends, 2005
- [Zhou 2001] Feng Zhou, Ju-Fu Feng, Qing-yun Shi, "Texture Feature Based on Local Fourier Transform", IEEE Transactions on Image Processing 2: 610-613, 2001
- [Zoran 2006] Maria Zoran, Sabina Stefan, "Climatic changes effects on spectral vegetation indices for forested areas analysis", proceedings of the 2<sup>nd</sup> Environmental Physics Conference, 73-83, 2006

## AVIS DU JURY SUR LA REPRODUCTION DE LA THESE SOUTENUE

Titre de la thèse : ~~Féodétection pour la gestion de l'environnement - une approche de fusion multiniveau~~  
*Fusion multiniveau pour la classification d'images de télédétection à très haute résolution spatiale.*  
Nom Prénom de l'auteur : **URSANI Ahsan Ahmad**

Membres du jury : Monsieur BRUZZONE                      Monsieur KPALMA  
                                 Madame HUBERT-MOY                      Monsieur RONSIN  
                                 Monsieur FERRO-FAMIL  
                                 ~~Monsieur~~ <sup>à la une</sup> LELONG

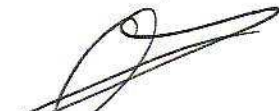
Président du jury : *Laurent Ferro-famil*

Date de la soutenance :                      04/11/2008

### Reproduction de la thèse soutenue :

- Thèse pouvant être reproduite en l'état
- Thèse ne pouvant être reproduite
- Thèse pouvant être reproduite après corrections suggérées

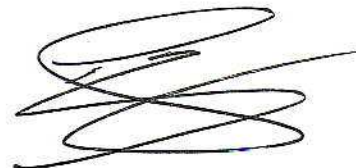
Le Directeur,



A. JIGOREL

Rennes, le                      04/11/2008

Signature du Président du jury



## Résumé

La télédétection s'avère être une discipline aux nombreuses applications telles que, la défense du territoire, la planification urbaine, la santé et la gestion de l'environnement. La collecte d'informations statistiques sur le rendement des cultures dans un pays, est une tâche importante de la télédétection. L'acquisition et le traitement d'images satellitaires à très haute résolution (THR) fournissent les moyens d'accomplir de telles tâches. Le traitement de ces images satellitaires exige non seulement de la puissance de calcul mais aussi les algorithmes efficaces en segmentation et classification d'images.

Cette thèse présente un travail de mise en œuvre de traitements efficaces en analyse, dans le domaine spectral et celui de la texture, sur des images à très haute résolution (THR). Ce travail combine les résultats de ces deux analyses pour une classification améliorée du couvert végétal. L'analyse spectrale présentée ici s'appuie sur une classification non supervisée, tandis que l'analyse de texture adopte une procédure de classification supervisée. La fusion des informations de type contour, issues de l'analyse spectrale non supervisée, et des informations de type bloc, issues de l'analyse texturale supervisée, conduit à des résultats de classification intéressants et encourageants.

En guise d'application, la thèse étudie le cas d'un site comportant vergers, cultures maraîchères, vignes, forêts, jachères de la région de Nîmes en France.

La contribution apportée ici concerne, d'une part, une amélioration de la méthode des « k-means », d'autre part, une solution à l'invariance en rotation des caractéristiques texturales issues d'une transformée de Fourier discrète, et enfin une méthode de fusion d'une classification supervisée avec une classification non supervisée.

Mots clefs : Télédétection, Gestion de l'environnement, Fusion d'images, Analyse de textures

## Abstract

Remote sensing is a promising technology that finds as diverse applications as defence, urban planning, healthcare, and environmental management. Collecting countrywide statistics of crop yield is one of the main tasks of remote sensing. Acquiring and processing very high-resolution (VHR) satellite images are means accomplishing this task. Processing these remotely sensed (RS) images requires not only great computational power but also efficient algorithms for image segmentation and classification.

This thesis aims at presenting the work carried out for applying computationally efficient spectral and textural analysis on very high-resolution RS images, and combining the results from the two analyses for improved classification of vegetation covers. The spectral analysis presented here adopts the unsupervised approach of classification, whereas the textural analysis adopts the supervised approach of classification. The fusion of the contour information from the unsupervised spectral analysis with the pixel class information from the supervised textural analysis yields successful classification results.

The thesis takes as a test case, a site covered with orchards, truck crops, crop fields, vineyards, forest, and fallows from Nîmes region, France. The real contribution includes improved version of the unsupervised classification method based on k-means clustering, a method of introducing rotation invariance into the texture features based on discrete Fourier transform, and a method of fusing a supervised classification with an unsupervised classification. This thesis is all about developing these algorithms.

Key words: Remote Sensing, Environmental management, Image fusion, Texture analysis