Remotely Sensed Data Segmentation under a Spatial Statistics Framework

by

Yu Li

A thesis

presented to the University of Waterloo

in fulfilment of the

thesis requirement for the degree of

Doctor of Philosophy

in

Geography

Waterloo, Ontario, Canada, 2009

AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Yu Li

ABSTRACT

In remote sensing, segmentation is a procedure of partitioning the domain of a remotely sensed dataset into meaningful regions which correspond to different land use and land cover (LULC) classes or part of them. So far, the remotely sensed data segmentation is still one of the most challenging problems addressed by the remote sensing community, partly because of the availability of remotely sensed data from diverse sensors of various platforms with very high spatial resolution (VHSR). Thus, there is a strong motivation to propose a sophisticated data representation that can capture the significant amount of details presented in a VHSR dataset and to search for a more powerful scheme suitable for multiple remotely sensed data segmentations.

This thesis focuses on the development of a segmentation framework for multiple VHSR remotely sensed data. The emphases are on VHSR data model and segmentation strategy. Starting with the domain partition of a given remotely sensed dataset, a hierarchical data model characterizing the structures hidden in the dataset locally, regionally and globally is built by three random fields: Markova random field (MRF), strict stationary random field (RF) and label field. After defining prior probability distributions which should capture and characterize general and scene-specific knowledge about model parameters and the contextual structure of accurate segmentations, the Bayesian based segmentation framework, which can lead

to algorithmic implementation for multiple remotely sensed data, is developed by integrating both the data model and the prior knowledge.

To verify the applicability and effectiveness of the proposed segmentation framework, the segmentation algorithms for different types of remotely sensed data are designed within the proposed segmentation framework. The first application relates to SAR intensity image processing, including segmentation and dark spot detection by marked point process. In the second application, the algorithms for LiDAR point cloud segmentation and building detection are developed. Finally, texture and colour texture segmentation problems are tackled within the segmentation framework.

All applications demonstrate that the proposed data model provides efficient representations for hierarchical structures hidden in remotely sensed data and the developed segmentation framework leads to successful data processing algorithms for multiple data and task such as segmentation and object detection.

ACKNOWLEDGEMENTS

It is a pleasure to thank all people who made this thesis possible.

I would like to express my sincere gratitude to my Ph.D. supervisors, professors Dr. Jonathan Li and Dr. Michael A. Chapman, for their patience, motivation, enthusiasm, and immense knowledge.

I wish to express my warm and sincere thanks to the members of my Ph.D. Thesis Committee at the University of Waterloo, professors Dr. David Clausi, Dr. Claude Duguay, and The PhD Comprehensive Examination Committee members at Ryerson University, professors Dr. Kunquan Lan and Dr. Songnian Li for their encouragement, insightful and constructive comments, and hard questions. Thanks also go to Dr. Jing Ming Chen, Professor at the Department of Geography and Program in Planning, Toronto University for his critical and helpful comments on the thesis.

I am especially grateful to Mrs. Lynn Finch, Graduate Program Administrator at the Department of Geography and Environmental Management for her administrative assistance, Ms. Anne Grant, Environmental Remote Sensing Specialist in Mapping,

Analysis and Design (MAD) at the Faculty of Environment for her timely technical support, and also to Ms. Lori McConnell, Assistant to Associate Dean Graduate Studies for all her helps.

Thanks to the Public Safety Canada, Association of Universities and Colleges of Canada, Ontario Ministry of Training, Colleges and Universities, University of Waterloo, and Ryerson University for their financial supports to my PhD study.

Many thanks also to my classmates at the University of Waterloo, Xiangyu Si, Yipeng Yuan, Wenxia Tan, Yao Lu, Xian Guan, Rui Zheng, Yuanming Shu, and Qingxu Huang for providing a warn and fun environment where to learn and to do my research happily.

I cannot end without thanking my family, on whose constant encouragement and love I have relied throughout my time on my Ph.D. study. Lastly and most importantly, I wish to thanks my parents and my beloved son for their immense love. To them I dedicate this thesis.

To my parents and my beloved son

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

∈	is a member of set
∉	is not a member of set
∀	for all
∝	varies directly as
⊕	Kronecker produce
⊂	is a sunset of
Π	the product of the terms indicated
Σ	the sum of the terms indicated
~	neighbour relationship between two sub-regions or distribute to
|	conditional on
|.|	the absolution value of or distance
\	difference of sets
∫	integral
*	proposal
∂	partial derivative
[ ]	a matrix

$\{\ \}$	a set
$\cap$	intersection
$\cup$	union
$\hat{}$	hat for estimation
!	factorial
$\lVert \cdot \rVert$	norm
/	divided by
%	per cent
a	direction of a window
c	coefficient for interactions of neighbour polygons in improves Potts model
D	data domain
D_j	j'th polygon or j'th window
E	expectation operator
$\boldsymbol{G}$	set of generating points
i	index of data points in a remote sensed dataset
j	index of Voronoi polygons or windows
k	the number of homogeneous regions or objects
l	index of homogeneous regions or objects
$\boldsymbol{L}$	label field
L_j	label for j'th sub-regions or objects
m	number of sub-regions or windows

r	acceptance rate
$\boldsymbol{s}$	vector for dimension match in RJMCMC scheme
t	indicator function
T_m	number of maximum iteration
u_i	horizontal coordination of data point *i*
v_i	vertical coordination of data point *i*
Z	random function or remotely sensed dataset
Z_i	measure of *i*’th data point
$\boldsymbol{Z}_j$	Set of measures of data points in *j*’th polygon or window
α	shape parameter of Gamma distribution
β	Scale parameter of Gamma distribution
δ	indicator of direction for clique
ε	Standard deviation of the Gaussian distribution for proposals
$\boldsymbol{\Theta}$	set of model parameters
μ	mean of Gaussian distribution for prior distributions
$\boldsymbol{\Xi}$	set of constants in posterior distribution
ρ	correlation coefficient
$\boldsymbol{\Sigma}$	covariance matrix
τ	Standard deviation of multivariate Gaussian distribution
$\boldsymbol{\Psi}$	set of parameters in posterior distribution

LIST OF ABBREVIATIONS

BGB	Bivariate Gamma Distribution
BGMRF	Bivariate Gaussian Markov Random Field
EM	Electromagnet
GMRF	Gaussian Markov Random Field
LiDAR	Light Detection And Ranging
LULC	Land Use and Land Cover
MAP	Maximum A Posteriori
MCMC	Markov Chain Monte Carlo
MGD	Multivariate Gamma Distribution
MGMRF	Multivariate Gaussian Markov Random Field
MRF	Markov Random Field
MS	Multi-spectral
PDF	probability density function
RF	Random Field
RJMCMC	Reversible Jump Markov Chain Monte Carlo
SAR	Synthetic Aperture Radar

Chapter 1

INTRODUCTION

This chapter describes motivations and problems, the goal and objectives of the study followed by the thesis structure. The motivations behind this thesis are discussed in Section 1.1. Section 1.2 describes the segmentation problems. Section 1.3 summarizes the objectives and scope of the study. Section 1.4 outlines the organization of this thesis.

1.1 Motivations and Goal

In remotely sensed data processing, segmentation is the most essential and critical task, not only because it is a prelude for further high-level data processing tasks such as feature extraction, object recognition and classification, but also because its quality has a prominent impact on the global qualities in all data analysis and interpretation stages (Jensen, 2006; Richards and Jia, 2006).

By far, remotely sensed data segmentation is still a difficult task. The difficulties stem from the intractability of the segmentation problem itself and the increasing complexity of remotely sensed data. According to Hadamard (1952), a mathematical problem is well-posed when its solution exists, is unique and robust to noise. While the acquisition of remotely sensed data is shown to be a direct well-posed problem, their segmentation is considered to be an inverse ill-posed problem since the solution is usually not unique (Poggio et al., 1985). The complexity of remotely sensed data is usually determined by the following factors: (1) the diversity of land use and land cover (LULC) in the real world and the manifold geometries of objects in scenes, (2) the variety of remotely sensed data ranging from optical, infra-red, and microwave and laser; and (3) the increasingly improved characteristics of remote sensors, for example, multiple imaging modes by new generation synthetic aperture radar (SAR) sensors for Radarsat-2 and TerraSAR sensors and very high spatial resolution (VHSR) sensors for IKONOS and GeoEye-1 (Curran, 2001).

In order to implement significantly accurate segmentation for different types of remotely sensed data, the development of sophisticated algorithms is one of the most challenging tasks addressed by the remote sensing community. Motivated by the above, the goals of this thesis will be directed toward devising a robust methodology suitable for modeling remotely sensed data and solving the problem of remotely sensed data segmentation.

1.2 Problems

Remotely sensed data segmentation is a procedure of partitioning the scene presented in a given remotely sensed dataset into meaningful regions. These regions correspond to land cover, and use all or part of (LULC) classes. To attain this aim, it is necessary to investigate scattering properties of LULC classes and infer constraints to characterize them. The challenges with the above include:

(1) Translating these constraints into criteria applicable at a data level to characterize complex scenes presented in a given remotely sensed dataset;

A remotely sensed dataset is the collection of measurements acquired from discrete georeferenced locations on the Earth's surface by an airborne or spaceborne sensor (Jensen, 2006). These sensors record the scatters of electromagnetic (EM) waves at different locations. A remotely sensed dataset can also be viewed as the realization of a random field (RF), that is, the incomplete observation of a single realization of a random experiment on a spatial domain. As a consequence, representing remotely sensed data is equivalent to modeling a RF related to the dataset. In general, the RF many reveal data specific local, regional and global structures. The local structure of remotely sensed data is caused by the similarity of measurements from neighbouring locations. The regional structure is determined by the fact of the matter coherency (Marr, 1982), that is, the homogeneity of measurements acquired from the region corresponding to a certain LULC class or part of it. The global structures reveal either the coherency or the diversity of backscatter properties from different LULC classes, even different parts of a certain LULC. Consequently, a hierarchical data model is a proper solution for characterizing those structures on different scales.

(2) Modeling the segmentation problem by using these criteria and a suitable scheme to obtain an optimal segmentation.

More precisely, it is intended to introduce a framework for remotely sensed data segmentation, which is able to integrate the hierarchical data model with prior knowledge on the segmentation, and to be applicable for multiple remotely sensed data segmentation, such as SAR, Light Detection and Ranging (LiDAR), and multispectral (MS) data.

1.3 Objectives

From the previous discussion, two broad issues need to be addressed to develop a framework for remotely sensed data segmentation, which will be the objectives focused on in this thesis.

(1) Figuring out how to model the remotely sensed data representation in order to characterize the structures at distinct scales, that is, local (or neighbour), regional and global structures. Using a mathematical framework, how are criteria formulated for expressing these structures?

Accordingly, in this thesis a hierarchical representation for remotely sensed data is proposed. The methodology is grounded in three RFs, namely, local, regional and global RFs, which together provide a uniform model for remotely sensed data representation.

(2) Modeling the segmentation problem under an operative framework which obtaining the optimal segmentation. What are the computational procedures necessary for this purpose?

In recent years, many researchers have used Bayesian estimation techniques for solving segmentation problem (Chan and Shen, 2005). They form a general framework to combine statistical models of the contextual structure and prior knowledge about data behaviour of accurate segmentations. The data is then segmented by finding an approximate maximum a posteriori (MAP) estimate of the unknown scene since this is the most likely segmentation given the remotely sensed data (Koch, 2007). Under this framework, the remotely sensed data segmentation considered in this thesis consists of the following four successive stages (Besag, 1993): (1) defining prior probability distributions for model parameters, (2) modeling the joint probability density (likelihood) for a given remotely sensed dataset based on the three previously mentioned RFs, (3) integrating the prior densities and the likelihood by Bayes' theorem to induce the posterior density of all model parameters conditional on the given dataset, and (4) building any inference about the model parameters.

In conclusion, the objectives will focus on two issues: (1) developing a hierarchical model for the representation of remotely sensed data and (2) developing a framework for remotely sensed data segmentation. To determine the applicability and effectiveness of these frameworks, this thesis also attempts to deal with segmentation problems for three kinds of remotely sensed data. These include: satellite SAR intensity imagery, airborne LiDAR point cloud data, and texture imagery.

1.4 Thesis Organization

The current chapter will attempt to introduce the problem requiring a solution, as well as the main objectives addressed in this thesis. Chapter 2 provides a review of related work. Fig. 1.1 gives the organization of Chapter 3 to Chapter 6. Chapter 3 presents the framework for remotely sensed data segmentation. In such a framework, remotely sensed data are characterized by RF models and then a Bayesian paradigm is used for modeling segmentation problem. Chapter 4 addresses the issues of SAR intensity image processing including SAR intensity image segmentation (Section 4.1) and oil spill detection by marked point processing (Section 4. 2). Charter 5 focuses on the issues related to LiDAR data processing. Firstly, the proposed segmentation framework in Chapter 3 is used to design the algorithm for LiDAR point cloud data segmentation, with application to object recognition (Section 5.1). A new algorithm based on the marked point process is developed for automated extraction of buildings from airborne LiDAR point cloud data (Section 5.2). Charter 6 then presents algorithms for texture and colour texture segmentations under the proposed framework. Finally, Chapter 7 concludes the thesis work with a summary of new contributions and addresses some recommendations for future research.

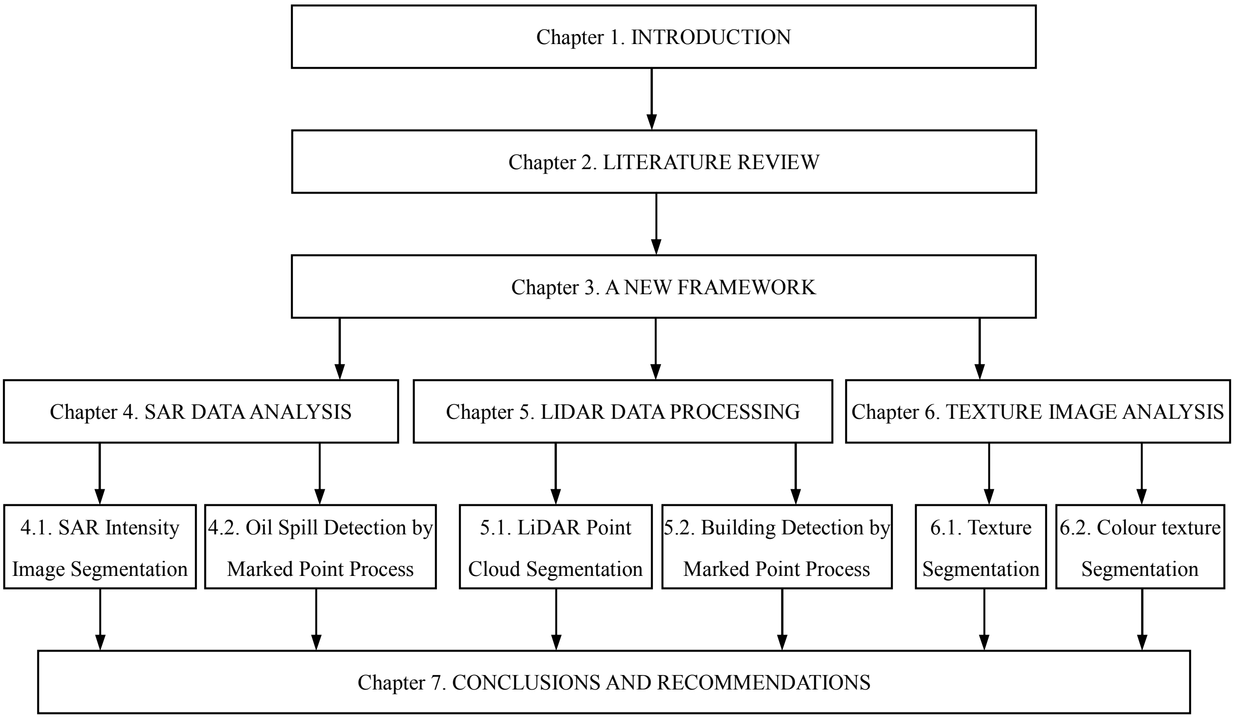

Chapter 1. INTRODUCTION
Chapter 2. LITERATURE REVIEW
Chapter 3. A NEW FRAMEWORK
Chapter 4. SAR DATA ANALYSIS
Chapter 5. LIDAR DATA PROCESSING
Chapter 6. TEXTURE IMAGE ANALYSIS
4.1. SAR Intensity Image Segmentation
4.2. Oil Spill Detection by Marked Point Process
5.1. LiDAR Point Cloud Segmentation
5.2. Building Detection by Marked Point Process
6.1. Texture Segmentation
6.2. Colour texture Segmentation
Chapter 7. CONCLUSIONS AND RECOMMENDATIONS

Fig. 1.1 Organization of the thesis.

Chapter 2

LITERATURE REVIEW

This chapter summarizes the algorithms commonly used for remotely sensed data segmentation and reviews the work closely related to the segmentation algorithms proposed in this thesis. The following sections briefly describe these schemes.

2.1 Introduction

Ever since remotely sensed data became available from civilian airborne and space-borne sensors in the early 1970s, a significant proportion of the research carried out over more than four decades in the remote sensing community concerns segmentation in one way or another (Jensen, 2006). Some researchers have focused on the improvement of segmentation algorithms specified for remotely sensed data while others on the use of well-known segmentation algorithms in particular types of remote sensing applications. Segmentation is regarded as a fundamental process in remote sensing for most high level remotely sensed data analysis tasks such as feature extraction, object identification and classification (Richards and Jia, 2006). The general segmentation problem involves the partitioning of a given dataset into a number of homogeneous regions, such that any two neighbouring regions are heterogeneous. In practice, there are several qualities that can be used to define the contextual homogeneity in terms of the particular objective of the segmentation process, for example, intensity, range, texture and so on (Russ, 2008). Broadly speaking, the segmentation task can be accomplished by edge-based, pixel-based and region-based schemes, respectively (Jähne, 2005).

2.2 Edge-based Segmentation

Edge-based segmentation relies on the information provided by boundaries of regions or objects, i.e., the dissimilarity between heterogeneous regions. In practice, they attempt to find the position of an edge by an extreme of the first-order derivative or a zero crossing in the second-order derivative (Qiu, 2005), which can be carried by either deriving evolving contours and topologically adaptable surfaces with level set (Cao and Yang, 2007), or directly detecting edge points with differential operators such as Sobel (Rekik et al., 2007), Canny (Rignot et al., 1991) and Laplacian (Greenfeld, 1991) operators. Usually, edge-based segmentation takes into account that an object is characterized by adjacent data points.

There are limitations on edge-based algorithms for VHSR remotely sensed data segmentation: (1) they produce a lot of slices for each connectively homogeneous region, (2) edge points are usually not closed and are often very sensitive to local measurement variations (Kermad and Chehdi, 2002), (3) they poorly perform in the detection of textured regions (Yu et al., 2006), (4) edge detection from a multi-spectral image is complicated by the inconsistent location of edges in the multiple bands (Yu et al., 2006).

2.3 Pixel-based Segmentation

Pixel-based segmentation is conceptually the simplest algorithm which will only use the measure values of the individual data points. As popular pixel-based algorithms, *K*-means algorithms (Bezdek et al., 1984) are implemented in some commercial software such as PCI Geomatica, ERDAS Imagine, ARC Info and ENVI, since they are suited to deal with the imprecise and uncertain nature of remotely sensed data. Most of *K*-means algorithms are based on an objective function (Pal and Bezdek, 1995). They make an optimal segmentation by minimizing the objective function. In a *K*-means algorithm, each cluster is usually represented by a cluster model. This model consists of a cluster center. The degrees of membership of each data point in different clusters are computed from the distances of the data point to the cluster centers. So the division of a dataset into *c* clusters can be stated as the minimization of the distances (Cannon et al., 1986). Thresholding algorithms are also popular due to their simplicity and efficiency (Mardia and Hainsworth, 1988), which partition data points in the dataset into region interested and background classes based upon the relationship between the measure value of a data point and a parameter called the threshold. Usually, thresholding algorithms are classified into two main groups: global and local. In global methods, a fixed threshold is used for the whole dataset, whereas in local methods the threshold changes dynamically. However, thresholding algorithms cannot separate those areas which have the same measurements but do not belong to the same part. In addition, they cannot process images whose histograms are nearly unimodal, especially when the target region is much smaller than the background area (Ding et al., 2008). For some remote sensing applications, both *K*-mean and thresholding algorithms are not successful. The principal reason is that they are very sensitive to local measurement variations and ignore useful spatial information among data points.

2.4 Region-based Segmentation

Currently, a large number of algorithms for remotely sensed data segmentation are region-based, which exploit information provided by the entire region, i.e., the similarity of measurements within a homogeneous region. They always provide closed boundary of regions or objects and makes use of relatively larger neighbourhoods for decision making. Let D be the domain of a remotely sensed dataset, and C be a logical criterion for the homogeneity of a region. The region-based segmentation can be defined as a partition of D into disjoint non-empty regions, that is, $D = \{D_l\,; l = 1, \ldots, k\}$, so that the following conditions hold (Zucker, 1976): (1) D_l is digitally connected, i.e., the region must be composed of contiguous data points (Rosenfeld, 1970); (2) $\bigcup_{l=1}^{k} D_l = D$; (3) $C(D_l) = TRUE$, for all $l = 1, \ldots, k$; (4) $C(D_l \cup D_{l'}) = FALSE$, for $l \neq l'$, where D_l and $D_{l'}$ are adjacent. Region-based segmentation algorithms can be implemented by either structural or model-based techniques. The former analyzes the data structure in terms of organization and relationship of data points and regions (Udupa and Samarasekera, 1996). The representative and robust algorithms include region-growing (Ballard and Brown, 1982; Ryan, 1985), region merging-and-splitting, and sophisticated hierarchical methods. The latter tries to use probability distribution functions of data points and regions to characterize the data (Dubes and Jain, 1989), such as random field (RF) model.

The region-growing technique is an iterative process by which regions increase from individual data points, or another initial segmentation and grow iteratively until every data point is processed (Tremeau and Borel, 1997; Mehnert and Jackway, 1997). To this end, a similarity measurement between neighbouring data points is defined such that it produces a high value if the data points are similar and a low one otherwise. First, consider a data point adjacent to another. Then the latter can be added to the former's region if and only if their similarity is larger than some threshold. Second, similarly consider the neighbours of the latter and add them likewise if they are similar enough (Chang and Li, 1994). However, it has drawbacks: (1) Decisions about similarity measure are often very difficult; (2) It is hard to find objects that span multiple disconnected regions; (3) Region-growing techniques are computationally expensive because they start from individual data points or such small initial regions. Region splitting-and-merging is a top-down method that begins with the entire image (Russ, 2008). Some data property is selected as a criterion to decide the homogeneity of LULC class (de Jong and ven der Meer, 2004). This criterion is often based on the statistics from the measure histogram. If the histogram is multimodal, then the measurements of the data points in the dataset are assumed to be non-uniform and its domain is divided into sub-regions. Each sub-region is examined in the same way and subdivided again if necessary. The procedure continues until the individual data point level is reached. There is another way to split the data domain or sub-regions and encode the resulting data structure. Thresholding can be used to govern the partition of each sub-region into arbitrary smaller sub-regions iteratively (Laprade, 1988). This can produce final results having less blocky boundaries, but the data processing is much more complex, since all of the regions must be defined, and the time required for the process is much greater. Region-splitting alone does not create useful segmentation. After each splitting, the divided region is compared to adjacent ones. If they are similar, they are merged together. The definition of similarity may be the same as the one applied to the region-splitting, or comparisons may be made only for data points along the common edge. The latter has the advantage of tolerating gradual changes across the data (Russ, 2008). The disadvantages of this algorithm are that: (1) the splitting-and-merging algorithm depends on the definition of the dissimilarity for detecting heterogeneity in each divided region. Small sub-regions within large uniform regions can easily be missed, (2) in most cases, the merging of two sub-regions will change the value of the similarity measure and, the resulting regions will depend on the search strategy employed among the neighbours; (3) It is difficult to combine statistical models for remotely sensed data into the splitting-and-merging techniques.

2.5 Related Work

Statistical model-based algorithms based on Markov RF (MRF) models (Besag, 1974; Geman and Geman, 1984; Besag, 1986; Winkler, 1995; Li, 2009) have been used with a remarkable success for remotely sensed data segmentation because they model the local statistical dependence among neighbouring data points. The MRF model is a relatively simple, yet effective, tool to encompass prior knowledge in the segmentation process, and in fact the interest in MRFs has been steadily growing in recent years (D'Elia et al., 2003). Since MRF models express global statistics in terms of the local neighbourhood potentials, MRF based segmentation algorithms search all data points in a specific order and actually characterize the local properties only.

However, not all of the segmentation algorithms are feasible for VHSR and multiple remotely sensed data sources due to the following facts: (1) VHSR remotely sensed data increases the internal spectral variability (intra-region variability) of each homogeneous region and decreases the spectral variability between different heterogeneous regions (inter-region variability) (Bruzzone and Carlin, 2006). Due to the high complexity of the data, segmentation algorithms considering each data point independently or interaction between neighbouring data points have shown their limits (Mueller et al., 2004). In fact, the VHSR remotely sensed data allows for a more precise detection of boundaries and hence a finer definition of the homogeneous regions, possibly at multiple scales of observation. They call for new solutions to cope with the increased complexity and new peculiarities of these data (Gaetano et al., 2009). (2) The currently available multiple remote sensed data such SAR, LiDAR and multispectral provides detailed information from spectral, to polarization, to range. However, a segmentation algorithm cannot provide the solution for all kinds of the data. It is necessary to develop a uniform framework under which segmentation algorithms for different remotely sensed data can be designed. (3) The common limitation of all the above mentioned techniques is that they are based only on local information and even use this information only partly.

To address these issues, a number of problems remain open, the more important being (1) how to define a data model that is able to capture prior information on hierarchical structures hidden in VHSR remotely sensed data while remaining mathematically and numerically applicable; (2) how to simulate the numerical parameters of such a data model; (3) how to solve the optimal estimation problem with reasonable computational complexity. The first problem is certainly the most significant and intriguing, as it amounts to defining abstract structures that fit the observed remotely sensed data well. A hierarchical model is proposed, in order to capture the local, regional and global dependencies by data domain partitioning. The segmentation procedure is formulated as a posterior distribution under a Bayesian framework. In practice, the posterior distribution can be simulated by Markov chain Monte Carlo (MCMC) sampling methods and the optimal segmentation can be considered as a maximum a MAP estimation problem. Table 2.1 shows the comparison of segmentation schemes mentioned before, several criteria are applied for the comparison here, including, (1) typical algorithms used in each segmentation scheme (termed typical algorithm in the table), (2) the degree on which the spatial correlation is considered (termed spatial correlation), (3) whether or not suitable VHSR remotely sensed data (termed VHSR data), (4) whether or not to be used for multiple remotely sensed data sources (termed multiple sources), (5) whether or not to extend to other remote sensed data processing tasks (termed multiple tasks), (6) can prior knowledge of data context (for example, object geometry, correlation of LULC classes, and so on) be considered during designing algorithms (termed data texture)? From processing VHRS remote sensed data point of view, the segmentation framework introduced in this thesis is better than others mentioned above.

Table 2.1 Comparison of above mentioned segmentation schemes and one developed in the thesis

Criteria	Edge-based schemes	Pixel-based schemes	Region-based schemes	Scheme proposed in the thesis
Typical algorithm(s)	differential operators	K-means Histogram	Region-growing Splitting-merging	Segmentation by Voronoi tessellation Object detection by marked point process
Spatial correlation	Local (neighbour)	no	Local (neighbour)	Local, regional and global
VHSR data	fairly	hardly	fairly	suitable
Multiple sources	Raster data	Raster data	Raster data	Raster data Irregular point cloud data
Multiple tasks	Segmentation	Segmentation	Segmentation	Segmentation Object detection Edge detection*
Data context	no	no	Partly considered	Easy to model
* If a fibre process is used for domain partition, edge detection can be carried out under the segmentation framework proposed in the next chapter.				

The regional partition based algorithm to image segmentation has been introduced by Green (1995) in his pioneering work on developing a reversible jump MCMC (RJMCMC) algorithm. In that work, the image segmentation problem is viewed as an example of two-dimensional (2D) analogues of change-point analysis. The image domain is partitioned into sub-regions by a Voronoi tessellation and a synthesized noisy image is modeled by step functions defined on those polygons. Following Green (1995)'s idea, Dryden et al. (2006) investigated a Bayesian method for the segmentation of muscle fibre images containing two types of fibres. In their work, a fibre image is reasonably approximated by Voronoi polygons and a deformable template model based on Voronoi polygons is used to represent the segmented image.

Similar work was carried out independently by Mϕller and Skare (2001). They considered 2D image segmentation and also further extensions to three-dimensional (3D) reservoir modeling. In their work, a flexible prior is defined in terms of interacting colouring Voronoi polygons. Other related work includes Blackwell and Mϕller (2003), in which they defined a class of tessellation models based on perturbing or deforming the Voronoi tessellation, to model images and represent the animal territories using Voronoi polygon edges. Instead of Voronoi polygons, in the work from Dryden et al. (2003), an image is subdivided into regular blocks and each block is modeled as a single texture. This algorithm is developed for the segmentation of weed and crop images. Further, a hierarchical Bayesian procedure is used, in which the texture label field is characterized by a Potts model and the pixels within a block are distributed according to Gaussian MRF (GMRF), with the parameters dependent on texture types.

Chapter 3

A NEW FRAMEWORK

This chapter presents a new framework for remotely sensed data segmentation, with emphasis on two basic issues in segmentation problems: data model and segmentation strategy.

3.1 Description of the Proposed Framework

From a spatial statistics point of view, a remotely sensed dataset can be considered as a collection of random variables or vectors

$$\boldsymbol{Z} = \{Z_i = Z(x_i, y_i);\ i = 1, ..., n,\ \ (x_i, y_i) \in D\} \tag{3.1}$$

where Z is the spatially continuous random scale or vector function, (x_i, y_i) are the georeferenced ground points which are distributed on D regularly or irregularly, D is the domain of the dataset, Z_i is the data point that is the spatial sample of Z at (x_i, y_i), i

is the index of sampled data points and n is the number of data points in $\boldsymbol{Z}$.

For a given dataset $\boldsymbol{Z}$ with k homogeneous regions, it means that its domain D can be partitioned into k disjoint regions, that is,

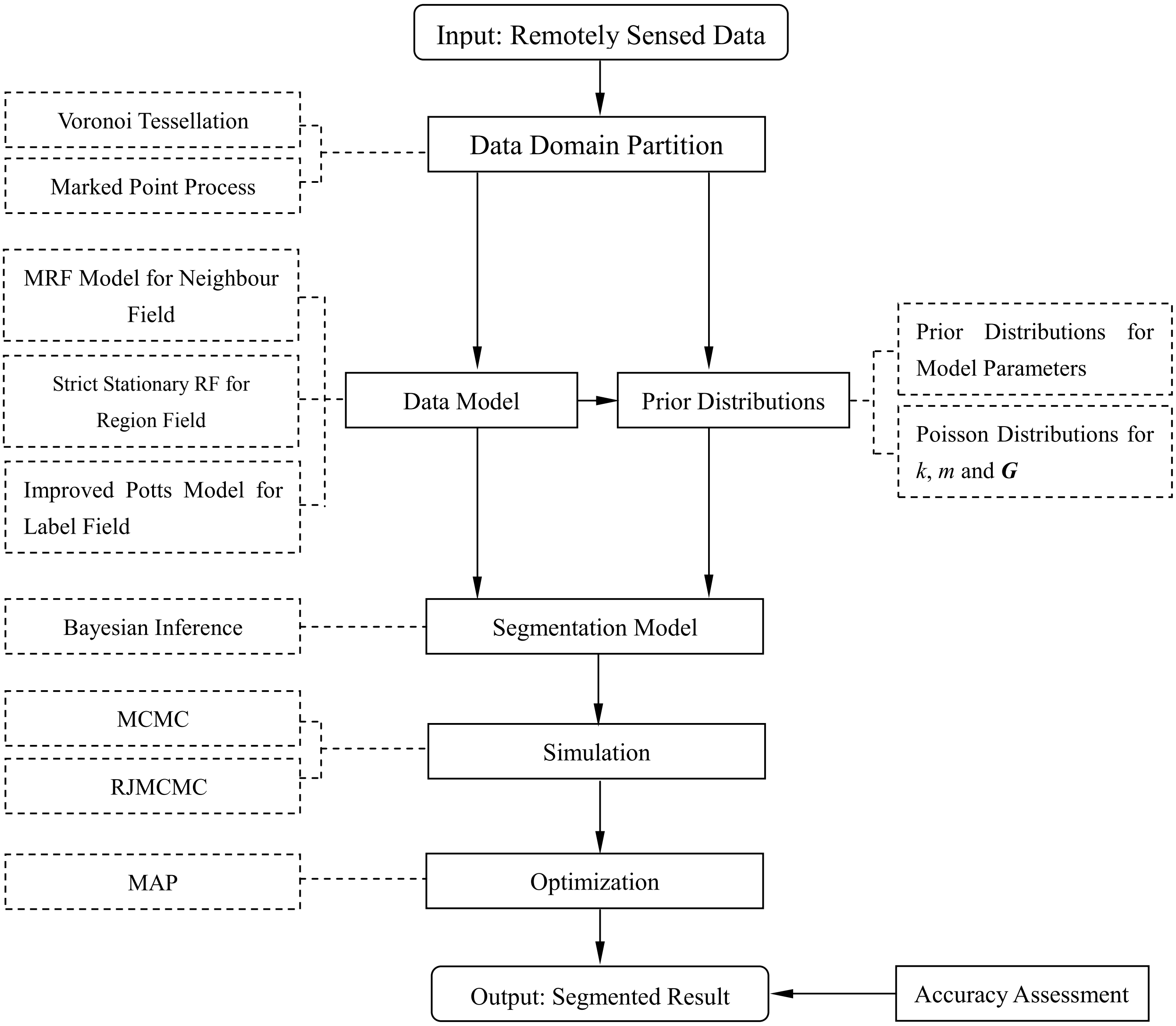
$$D = \{S_l;\ l = 1,\ \ldots,\ k\} \tag{3.2}$$

where S_l is l'th homogeneous region and l is the index of homogeneous regions. The data points in each homogeneous region $\boldsymbol{Z}_l$ = $\{Z_i$; (x_i, y_i) ∊ $S_l\}$ demonstrate some homogeneity or similarity. The purpose of a segmentation algorithm is to characterize the homogeneity and find an optimal solution of Eq. (3.2), called the optimal segmentation

$$D = \{\hat{S}_l;\ l = 1,\ \ldots,\ k\} \tag{3.3}$$

Based on the data structure in Eq. (3.1), a statistical region-based framework for the remotely sensed data segmentation is developed. Fig. 3.1 shows the flowchart of the proposed segmentation framework.

To hierarchically characterize remotely sensed data, the domain of a given remotely sensed dataset is partitioned into sub-regions corresponding to the components of homogeneous regions. Both the Voronoi tessellation and the marked



Input: Remotely Sensed Data
Voronoi Tessellation
Marked Point Process
Data Domain Partition
MRF Model for Neighbour Field
Strict Stationary RF for Region Field
Improved Potts Model for Label Field
Data Model
Prior Distributions
Prior Distributions for Model Parameters
Poisson Distributions for k, m and $\boldsymbol{G}$
Bayesian Inference
Segmentation Model
MCMC
RJMCMC
Simulation
MAP
Optimization
Output: Segmented Result
Accuracy Assessment

Fig. 3.1 Flowchart of the proposed segmentation framework.

point process are employed for this purpose. Each sub-region is assigned a label to indicate the homogenous region to which the sub-region belongs. The data model characterizing the homogeneous structures hidden in remotely sensed data are grounded in three RFs: label field for encoding global structures, neighbour field built among neighbouring pixels for modeling local structures, and region field on each sub-region for characterizing regional structures. They together provide a uniform framework which can flexibly model different types of remotely sensed data. The label, neighbour, and region fields are characterized by an improved Potts model, MRF model and strict stationary RF model, respectively. Based on the three RFs, the data model is expressed as the joint probability density (likelihood) for a given dataset.

The Bayesian paradigm for data analysis takes advantage of explicit probability models and general and scene-specific prior knowledge and provides a uniform framework under which many different data processing tasks can be tackled (Besag, 1993). After constructing prior probability distributions for model parameters such as the numbers of homogeneous regions and sub-regions partitioned and distribution parameters used in data models, the posterior probability distribution function (PDF) on label filed and model parameters conditional on the given dataset can be obtained by combining the prior densities and the likelihood.

To sample the posterior PDF, MCMC and RJMCMC based simulation schemes are designed since they can be applied effectively to Bayesian problems (German and German, 1984; Green, 1995). The optimal segmentation is obtained in terms of MAP which searches for the maximum of the posterior PDF of the label field and model parameters given the dataset.

The details of the components of the proposed framework shown in Fig. 3.1 are explained in the following sections.

3.2 Data Domain Partition

Conside a remotely sensed dataset $\boldsymbol{Z} = \{Z_i\,; i = 1, \ldots, n, (x_i, y_i) \in D\}$. Its domain D can be randomly partitioned into m disjoint sub-regions either corresponding to the components of homogeneous regions or being modeled m objects. In this thesis, the domain parathion can be carried out by Voronoi tessellation and marked point process.

A. Voronoi Tessellation for Domain Partition

A tessellation is a partition of space into small units or cells, which are usually polytopes, for example, polygons in R^2 and polyhedra in R^3 (Okabe et al., 1992). Depending on the applications, a tessellation can be considered as a partition of space or as a random function by assigning each cell a value, or even as a population of cells (Lantuejoul, 2002).

A flexible and convenient example of a randomly generating tessellation is the Voronoi tessellation, also called Dirichlet or Thiessen tessellation (Stoyan et al., 1995). Consider a set of points called generating points $\boldsymbol{G} = \{(u_j, v_j) \in D; j = 1, \ldots, m\}$. Associate to the generating point (u_j, v_j) the set D_j of the points $(x, y) \in D$ that are closest to the generating point (u_j, v_j) than to other generating points $(u_{j'}, v_{j'})$ where j'

$\in \{1, \ldots, m\}$ but $j' \neq j$, that is,

$$D_j = \left\{ (x, y) \in D;\ \left\| (x, y) - (u_j, v_j) \right\| < \min_{(u_{j'}, v_{j'}) \in \boldsymbol{G} \setminus (u_j, v_j)} \left\| (x, y) - (u_{j'}, v_{j'}) \right\| \right\} \qquad (3.4)$$

where || **.** || is the norm operator, $\boldsymbol{X} \setminus \boldsymbol{Y}$ denotes the difference of sets $\boldsymbol{X}$ and $\boldsymbol{Y}$, and D_j is called Voronoi polygon which is a convex polygon in D delimited by the perpendicular bisectors of segments joining neighbouring generating points (Okabe et al., 1992). Fig. 3.2 shows an example of a planar Voronoi tessellation with six Voronoi polygons D_1 - D_6 induced by generating points (u_1, v_1) - (u_6, v_6), in which the dashed lines are the segments linking neighbouring generating points and solid segments are the boundaries of Voronoi polygons.

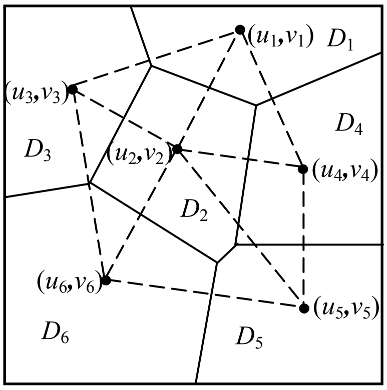


Fig. 3.2 Example of a planar Voronoi tessellation with 6 polygons D_1-D_6 induced by generating points (u_1, v_1) - (u_6, v_6).

As a result, the domain can be partitioned as

$$D = \left\{ D_j \subset D;\ j = 1,\ \ldots,\ m \right\} \qquad (3.5)$$

where D_j's satisfy

$$\begin{aligned} &D_j \cap D_{j'} = \Phi \quad \text{for } j \neq j' \\ &\bigcup_{j=1}^{m} D_j = D \end{aligned} \tag{3.6}$$

where *Φ* denotes the empty set, *m* can be considered to be unknown *a priori* but with a prior distribution $p(m)$.

B. Marked Point Process for Domain Partitions

Generally speaking, a random point process in R^d is a random set in R^d, each realization of which consists of a finite or countable number of points (Stoyan et al., 1995). In this thesis, the random point process will be limited in 2D case, that is, d=2.

A uniformly distributed random point is a trivial random pattern. Correspondingly, a binomial point process can be formed by *m* independent and uniformly distributed random points. Such a process is formed by *m* independent points $\boldsymbol{G} = \{(u_j, v_j); j = 1, \ldots, m\}$ uniformly distributed on *D*. For the purpose of domain partition, a marked point process can be constructed from the random point process ***G*** by attaching a rectangular (or a window) to each point of the process. Thus, the marked point process on *D* is a random sequence $\{(u_j, v_j, l_j, w_j, a_j); \quad j = 1, \ldots, m\}$ where the points

(u_j, v_j) together constitute a point process (not mark) in D while the mark (l_j, w_j, a_j) corresponds the length, width and direction of the window centered at (u_j, v_j). Accordingly, the domain is partitioned as

$$D = \{D_0, D_j; j = 1, \ldots, m\} \tag{3.7}$$

where $D_0 = D \setminus \bigcup_{j=1}^{m} D_j$.

Fig. 3.3 shows an example of the domain partition by marked point process.

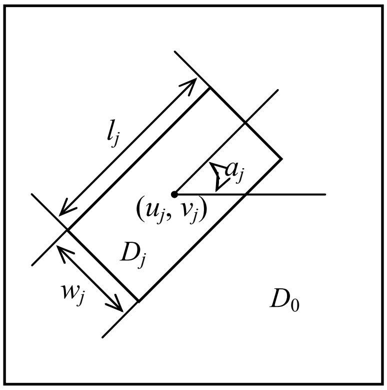


Fig. 3.3 Domain partition by marked point process where D_j is a window with length l_j, width w_j and direction a_j and D_0 is non window part of the domain.

3.3 Data Model

In remote sensing, the recoded measures in remotely sensed data are also determined by the properties of materials, the roughness of an object’s surface, the topological feature of landscapes, the geometries of LULC, and the scattering behaviors of scatter cells, which usually make their impacts on visual features presented by remotely sensed data at different scales (Richards and Jia, 2006). From a remotely sensed data segmentation point of view, proper data models should be capable of capturing the visual features and play a critical role in developing successful segmentation algorithms.

There usually are three visual features or structures hidden in remotely sensed data, which represent local, regional and global variations of measures on different scales. They determine the overall visual smoothness and coarseness of the measures and provide important information about the distribution of physical objects and their spatial relationship within the remotely sensed data. *Local structure* is considered to be caused by the similarity of measures from neighbour scattering cells on the Earth’s surface, which can be described by the correlation of the neighbouring measures. The homogeneity of measures from scattering cells in an object or part of the object, that is, the “Matter is cohesive” property as stated by Marr (1982), gives rise to *regional structure*, which can be characterized by identical distributions for these measures. In practice, what should be regarded as homogeneity depends on data context, for example, texture, colour, intensity, range, and other measures recorded by remote sensing sensors. *Global structure* is derived from the continuity or periodicity of measures for a certain homogeneous structure, or the correlation of measures from heterogeneous structures.

Fig. 3.4 illustrates an example of those structures revealed in remotely sensed data at different scales. Fig. 3.4 (a) shows a pan-sharpened IKONOS image with 1 *m* spatial resolution and the size of 128×128 pixels. This scene is composed of forest patches with varying species, sizes and shapes corresponding to regions with circumferences in blue, pink and green, respectively. In this image, the patch for each tree species corresponds to a global structure, which represents the tonal variation in the whole spatial domain and determines the overall smoothness and coarseness of intensities in the patch. As shown in Fig. 3.4, (b1)-(d1) present the parts of three tree species with the size of 16×16 pixels corresponding to pink, blue and green windows in (a), which presents statistical homogeneity corresponding to regional structures, while Fig. 3.4 (b2)-(d2) show second order neighbourhoods of the pink, blue and green windows (3×3 pixels) in (b1)-(d1), which presents the similarity of the neighbouring structures.

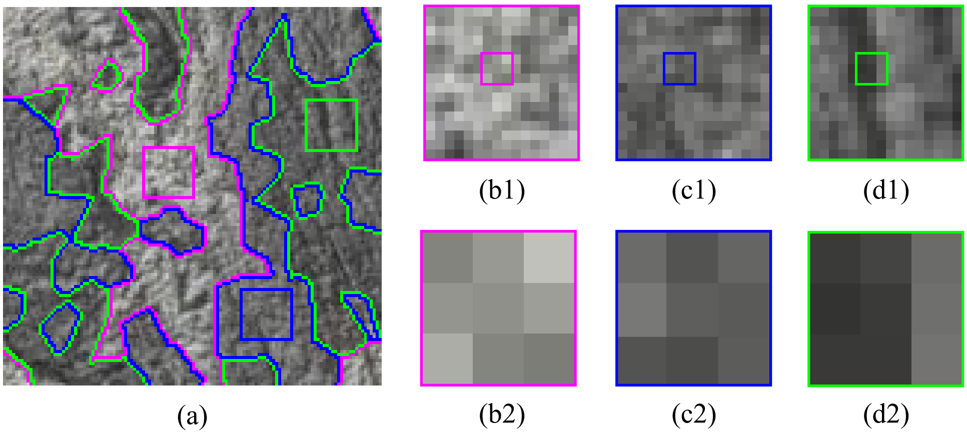


Fig. 3.4 Structures hidden in the IKONOS imagery (a): global (a), regional (b1)-(d1) and neighbour (b2)-(d2) structures.

According to the above discussion, developing a hierarchical data model for remotely sensed data seems to be necessary to characterize the structures revealed at different scales.

Formally, consider a remotely sensed dataset $\boldsymbol{Z} = \{Z_i\,;\, i = 1, \ldots, n, (x_i, y_i) \in D\}$. Assume that $\boldsymbol{Z}$ contains k homogenous regions representing different LULC or objects, each of which is considered to be statistically homogenous where k can be considered to be either unknown but with a prior distribution $p(k)$ or known *a priori*. Further, the domain D is randomly partitioned into m disjoint sub-regions by domain partition defined in Eq. (3.5).

Each sub-region D_j is assigned a label L_j to indicate the homogeneous region to which D_j belongs. Therefore, the labels for all sub-regions form a collection of random labels

$$\boldsymbol{L} = \left\{ L_j;\ j = 1, \ldots,\ m, L_j \in \{1,\ \ldots,\ k\} \right\} \tag{3.8}$$

The realization of a random partition in Eq. (3.5) and the random collection of labels in Eq. (3.8) completely characterize a segmentation of D as defined in Eq. (3.3), that is,

$$S_l = \{D_j; L_j = l\} \tag{3.9}$$

Fig 3.5 shows an example of the domain partition with labels $\boldsymbol{L} = \{1, 1, 2, 1, 2, 3, 3, 2\}$, in which the red lines divide the domain into three homogenous regions S_1 - S_3 and the green lines partition the domain into eight sub-regions D_1 - D_8 corresponding to the components of the regions, that is $S_1 = \{D_1, D_2, D_4\}$, $S_2 = \{D_3, D_5, D_8\}$ and $S_3 = \{D_6, D_7\}$. In this thesis, Voronoi tessellation and marked point process are used for the purpose.

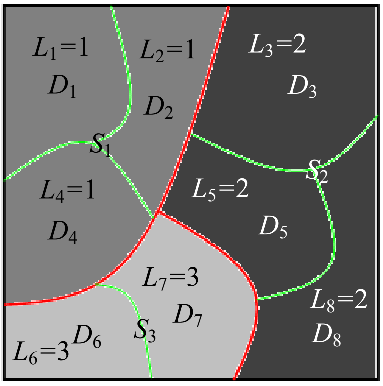


Fig. 3.5 Domain partition with 3 homogenous regions S_1-S_3 into 8 sub-regions D_1-D_8.

A. Label Field for Global Structure

To hierarchically model the remotely sensed dataset $\boldsymbol{Z}$, the random collection of labels in Eq. (3.8) is modeled with a RF called label field, which characterizes the global structure existing in $\boldsymbol{Z}$ and is represented by conditional probability distribution function $p(\boldsymbol{L} \mid k, m)$.

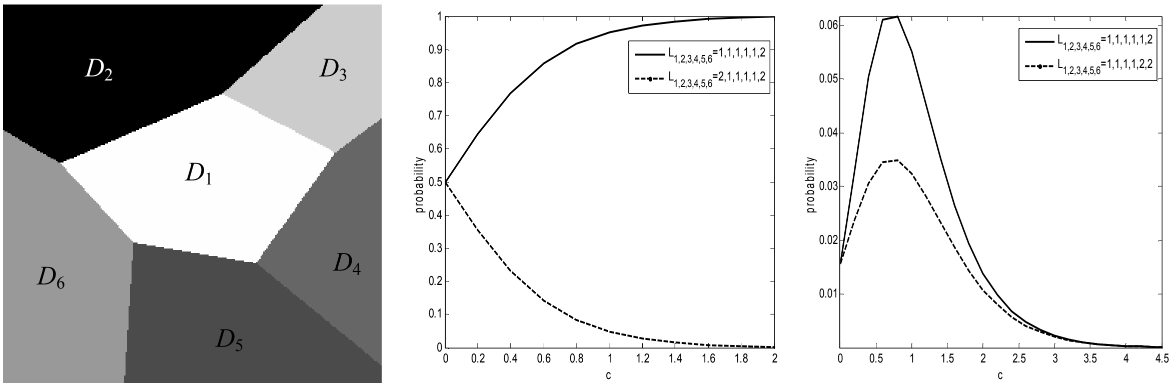
In this thesis, the improved Potts model (Strauss, 1977; Besag, 1986) is used to model the label field, which characterizes the interaction between neighbouring polygons in a Voronoi tessellation. Consider a Voronoi tessellation which partitions the data domain D into disjoint polygons, that is, $D = \{D_j\,;\, j = 1, \ldots, m\}$. Any two distinct Voronoi polygons D_j and $D_{j'}$ are neighbours, denoted by $D_j \sim D_{j'}$, if and only if D_j and $D_{j'}$ have mutual boundary, where the operator ' $\sim$ ' donates a neighbourhood relationship. For a polygon D_j, let $ND_j = \{D_{j'}\,;\, D_{j'} \sim D_j, j' \neq j\}$ be the set of its neighbouring polygons. The conditional distribution of the label L_j for the polygon D_j given the labels of its neighbouring polygons can be expressed as

$$p(L_j \mid L_{j'}, D_{j'} \in ND_j) = \frac{\exp\left(c \sum_{D_{j'} \in ND_j} t(L_j, L_{j'}) \right)}{\sum_{l=1}^{k} \exp\left(c \sum_{D_{j'} \in ND_j} t(l, L_{j'}) \right)} \tag{3.10}$$

where t function is defined as follows

$$t(x, y) = \begin{cases} 1, & \text{if } x = y \\ 0, & \text{otherwise} \end{cases} \tag{3.11}$$

and $c \geq 0$ is the coefficient which controls the neighbourhood dependences between a pair of neighbouring polygons. Depending on the labels for a polygon and its neighbouring polygons, Eq. (3.10) is monotone increasing or decreasing with c. Fig. 3.6 (a) shows a domain partition and Fig. 3.6 (b) gives the monotonic function of c under different segmentation. On the other hand, updating the label of a polygon causes the changes of the conditional probabilities of the labels for the polygon and its neighbouring polygons as defined in Eq. (3.10). From a numerical computing point of view, a large *c* makes some neighbouring relationships unidentifiable. From the Voronoi tessellation shown in Fig. 3.6 (a), Fig. 3.6 (c) gives the joint probabilities for the label field under different observations. It can be seen that when $c > 2.5$ the observations of the label field $\boldsymbol{L}$ = (2, 1, 1, 1, 1, 2) and $\boldsymbol{L}$ = (1, 1, 1, 1, 1, 2) are hard to be recognized. From a number of experiments, the interval [0.5, 1.5] for the constant *c* is recommended.



(a) (b) (c)

Fig. 3.6 (a) Domain partition into 6 regions D_1-D_6. (b) Curves of conditional probabilities to *c* under the labels $\boldsymbol{L}$ = {1, 1, 1, 1, 1, 2} and $\boldsymbol{L}$ = {2, 1, 1, 1, 1, 2}. (c) Curve of joint probabilities to *c* under the labels $\boldsymbol{L}$ = {1, 1, 1, 1, 1, 2} and $\boldsymbol{L}$ = {2, 1, 1, 1, 1, 2}.

The joint probability function of the label field $\boldsymbol{L}$ can be expressed as

$$p(\boldsymbol{L}) = \prod_{j=1}^{m} p(L_j \mid L_{j'}, D_{j'} \in N_j) = \prod_{j=1}^{m} \frac{\exp\left(c \sum_{D_{j'} \in ND_j} t(L_j, L_{j'}) \right)}{\sum_{l=1}^{k} \exp\left(c \sum_{D_{j'} \in ND_j} t(l, L_{j'}) \right)} \tag{3.12}$$

B. MRF Model for Local Structure

In order to characterize the neighbour structure, the measure Z_i at the location $(x_i, y_i) \in D_j$ is represented by a probability distribution function conditional on a set of model parameters $\boldsymbol{\Theta}_{L_j}$ and the measurements from its neighbour locations $\boldsymbol{Z}_{N_i} = \{Z_{i'};\ (x_{i'}, y_{i'}) \in ND_i \cap D_j\}$, that is, $p(Z_i \mid \boldsymbol{Z}_{N_i}, \boldsymbol{\Theta}_{L_j})$. In this thesis, Bivariate GMRF (BGMRF) and Multivariate GMRF (MGMRF) are used for characterizing the neighbouring structures in texture and colour texture images, respectively. They will be described in Chapter 6.

C. Strict Stationary RF for Regional Structure

Regional structure can be characterized by a PDF for all measures in a sub-region D_j, that is, $p(\boldsymbol{Z}_j \mid \boldsymbol{\Theta}_{L_j})$ where $\boldsymbol{Z}_j = \{Z_i;\ (x_i, y_i) \in D_j\}$. Assume that the regional structure is modeled by a strict stationary RF, in which Z_i's are considered to be independent each other. Thus, $p(\boldsymbol{Z}_j \mid \boldsymbol{\Theta}_{L_j})$ can be written as

$$p(\boldsymbol{Z}_j \mid \boldsymbol{\Theta}_{L_j}) = \prod_{(x_i, y_i) \in D_j} p(Z_i \mid Z_{N_i}, \boldsymbol{\Theta}_{L_j}) \qquad (3.13)$$

D. Data Representation

On the assumption that PDFs for all sub-regions are independent, then the joint distribution (likelihood) of $\boldsymbol{Z}$ given $\{\boldsymbol{\Theta}, k, m, D, \boldsymbol{L}\}$ can be expressed

$$p(\boldsymbol{Z} \mid \boldsymbol{\Theta}, k, m, D, \boldsymbol{L}) = \prod_{j=1}^{m} p(\boldsymbol{Z}_j \mid \boldsymbol{\Theta}_{L_j}) \tag{3.14}$$

where $\boldsymbol{\Theta} = \{\boldsymbol{\Theta}_l \,;\, l = 1, \ldots, k\}$.

Consequently, the hierarchical model of remotely sensed data can be represented by modeling the joint distribution of ($\boldsymbol{Z}$, $\boldsymbol{L}$) as

$$p(\boldsymbol{Z}, \boldsymbol{L} \mid \boldsymbol{\Theta}, k, m, D) = p(\boldsymbol{Z} \mid \boldsymbol{\Theta}, k, m, D, \boldsymbol{L})\, p(\boldsymbol{L} \mid m, D) \tag{3.15}$$

The hierarchical model can be demonstrated by a directed acyclic graph (see Fig. 3.7) where circles represent unknown quantities, and squares represent known quantities and let $\boldsymbol{\Xi}$ denote the collection of constants used in prior distributions of random variables $\boldsymbol{\Theta}$, k, m, $\boldsymbol{G}$, and $\boldsymbol{L}$.

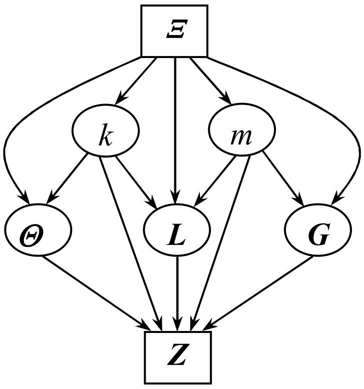


Fig. 3.7 Directed acyclic graph for the data model.

The data model used for representing regional structures is a particular type of mixture models (Richardson and Green, 1997) with m components by considering the independent labels case, that is,

$$p(\boldsymbol{L} \mid k, m) = \prod_{j=1}^{m} p(L_j \mid k) \tag{3.16}$$

$\boldsymbol{Z}_j$*'s* given $\boldsymbol{\Theta}$ and k are independent and with PDF

$$p(\boldsymbol{Z}_j \mid \boldsymbol{\Theta}) = \sum_{l=1}^{k} p(L_j = l \mid k) p(Z_j \mid L_j = l, \boldsymbol{\Theta}_l) = \sum_{l=1}^{k} w_l f(\boldsymbol{Z}_j \mid \boldsymbol{\Theta}_l) \tag{3.17}$$

where labels L_j are independently drawn from the probability distributions $p(L_j = l \mid k) = w_l$, $l = 1, \ldots, k$, and the observations $\boldsymbol{Z}_j$ are independently drawn from the corresponding individual populations given the values L_j, with PDF $p(\mathbf{Z}_j \mid L_j = l, \boldsymbol{\Theta}_l) = f(\boldsymbol{Z}_j \mid \boldsymbol{\Theta}_l)$.

3.4 Prior Distributions

This section defines some prior distributions for model parameters, which will be used throughout this thesis.

A. Prior Distribution for the Number of Homogenous Regions k

The number of homogeneous regions in a particular dataset has a theoretical maximum equal to the total number of sub-regions, which occurs in the case when sub-regions are equal to homogenous regions. In practice, the maximum $k_{\max}$ is set dependent on a given dataset. The number of homogenous regions can be considered to have a prior truncated Poisson distribution with mean λ_k (truncated with $2 \leq k \leq k_{\max}$) (Green, 1995), that is,

$$p(k) = e^{-\lambda_k} \frac{\lambda_k^{\ k}}{k!} \tag{3.18}$$

B. Prior Distribution for the Number of Sub-regions m

The number of sub-regions is modelled to have a Poisson distribution with parameter λ_m (Green, 1995), truncated to $m = 1, \ldots, m_{\max}$. In theory, $m_{\max}$ is equal to n if and only if when each sub-region exactly contains one data point. The prior distribution can be written as

$$p(m) = e^{-\lambda_m} \frac{\lambda_m^{\ m}}{m!} \tag{3.19}$$

C. Prior Distribution for the Partition of Domain D

According to the statement in Section 3.2, the domain partition is specified by the generating points, $\boldsymbol{G} = \{(u_j, v_j) \in D; j = 1, \ldots, m\}$, that is, $p(D \mid m) = p(\boldsymbol{G} \mid m)$. If the position of generating point (u_j, v_j) is assumed to be independently and uniformly distributed on D, then the joint prior distribution $p(\boldsymbol{G} \mid m)$ is

$$p(\boldsymbol{G} \mid m) = \prod_{j=1}^{m} p(u_j, v_j) = \prod_{j=1}^{m} \frac{1}{|D|} = \left(\frac{1}{|D|}\right)^m \tag{3.20}$$

where $p(u_j, v_j)$ is the prior distribution for the position of generating point (u_j, v_j) and $|D|$ denotes the area of domain D.

However, instead of assuming the independency among generating point, if it is necessary to consider interaction between the generating points in some applications, there are point processes that can be used to model the interaction, including Gaussian perturbed points (Dryden et al., 1997), the nearest neighbour Markov process (Baddeley and Mϕller, 1989) and the Strauss process (Strauss, 1977; Ripley, 1987). Strauss process is an example of this: Given a set of generating points $\boldsymbol{G} = \{(u_j, v_j), j = 1, \ldots, m\}$, which distribute on the data domain D, the probability density of Strauss model for point positions conditional on m can be expressed as

$$p(\boldsymbol{G} \mid m) = \frac{1}{c(b)} e^{-bt_r(\boldsymbol{G})} \tag{3.21}$$

where $b \geq 0$ is the intensity parameter of the model, $c(b)$ is the normalizing constant dependent on b, and $t_r(\boldsymbol{G})$ is the number of generating point pairs in $\boldsymbol{G}$ that have the distance less than r, that is,

$$t_r(\boldsymbol{G}) = \sum_{j=1}^{m-1} \sum_{j'=j+1}^{m} I_{[0,r]}\left(\| (u_j, v_j) - (u_{j'}, v_{j'}) \|\right) \tag{3.22}$$

where $\mathrm{I}_{[0,\, r]}\ (s)$ is a indicator defined as

$$I_{[0,r]}(s) = \begin{cases} 1, & \text{if } 0 \leq s \leq r \\ 0, & \text{otherwise} \end{cases} \tag{3.23}$$

Note that if $b = 0$, Eq. (3.21) degrades to a uniform distribution. In more complicated situations, when $\boldsymbol{G}$ is a Poisson point process with variable m, the Strauss process model can be written as

$$p(\boldsymbol{G} \mid m) = \frac{\lambda_m{}^m}{c(b, \lambda_m)} e^{-bt_r(G)} \tag{3.24}$$

where λ_m is the rate of the underlying process.

3.5 Segmentation Model

Bayesian inference-based approaches to image analysis have now become a generally accepted framework under which a wide variety of image processing tasks such as image retrieval (Geman and Geman, 1984; de Ves et al., 2006) and segmentation (D'Elia et al., 2003) can be performed. Bayesian image analysis takes advantage of explicit probability models to incorporate general and scene-specific prior knowledge into the image processing procedure (Besag, 1993). In this thesis, the Bayesian paradigm (Gelman et al., 2004) is used for modeling remotely sensed data segmentation. Combing the prior distributions for $\{\boldsymbol{\Theta}, k, m, \boldsymbol{G}, \boldsymbol{L}\}$ and the likelihood by Bayes' theorem, the posterior PDL $p(\boldsymbol{\Theta}, k, m, \boldsymbol{G}, \boldsymbol{L} \mid \boldsymbol{Z})$ can be induced

$$\begin{aligned} & p(\boldsymbol{\Theta},k,m,\boldsymbol{G},\boldsymbol{L} \mid \boldsymbol{Z}) \propto \\ & \qquad p(\boldsymbol{Z} \mid \boldsymbol{\Theta},k,m,\boldsymbol{G},\boldsymbol{L})p(k)p(m)p(D \mid m)p(\boldsymbol{L} \mid \boldsymbol{G},k,m)p(\boldsymbol{\Theta} \mid k) \end{aligned} \tag{3.25}$$

where the normalizing constant is a function of data and is not generally required.

Since the flexibility of the Bayesian paradigm allows for the introduction of extra variables to the data model (Richardson and Green, 1997), the prior distribution for the model parameter vector $\boldsymbol{\Theta}$ can be considered to depend on a hyper-parameter $\boldsymbol{\Psi}$, which is independent of the number of homogenous regions and label filed. Therefore, the posterior distribution in Eq. (3.25) for all the variables becomes

$$p(\boldsymbol{\Psi},\boldsymbol{\Theta},k,m,\boldsymbol{G},\boldsymbol{L}\mid\boldsymbol{Z})\propto$$

$$p(\boldsymbol{Z}\mid\boldsymbol{\Psi},\boldsymbol{\Theta},k,m,\boldsymbol{G},\boldsymbol{L})p(k)p(m)p(\boldsymbol{\Psi})p(\boldsymbol{G}\mid m)p(\boldsymbol{L}\mid\boldsymbol{G},k,m)p(\boldsymbol{\Theta}\mid\boldsymbol{\Psi},k) \tag{3.26}$$

3.6 Simulation and Optimization

To base any inference about the prior parameters $\{\boldsymbol{\Theta}, k, m, \boldsymbol{G}, \boldsymbol{L}\}$ on the posterior distribution in Eq. (3.26), the simulation scheme in which the inference should be done is context dependent. In this thesis, the MCMC and RJMCMC schemes are developed to simulate from the posterior distribution in Eq. (3.24).

A. Metropolis-Hastings Algorithm

The Metropolis-Hastings algorithm, first introduced by Metropolis et al. (1953) and developed by Hastings (1970), is a MCMC algorithm. It allows for sampling from a distribution when traditional sampling methods such as transformation or inverse fail and even there is no need to know the normalization constant in Eq. (3.26).

Let $\boldsymbol{\Lambda} = \{\boldsymbol{\Theta}, k, m, \boldsymbol{G}, \boldsymbol{L}\}$. In the Metropolis-Hastings algorithm, at iteration t, the next state $\boldsymbol{\Lambda}^{(t+1)}$ is chosen by first sampling a candidate $\boldsymbol{\Lambda}^*$ from a proposal distribution (an arbitrary transition kernel) $q\ (.\ |\boldsymbol{\Lambda}^{(t)})$. The candidate $\boldsymbol{\Lambda}^*$ is then accepted with probability $r\,(\boldsymbol{\Lambda}^{(t)}, \boldsymbol{\Lambda}^{*})$,

$$r(\boldsymbol{\Lambda}^{(t)}, \boldsymbol{\Lambda}^{*}) = \min\left(1,\ \frac{\pi(\boldsymbol{\Lambda}^{*})q(\boldsymbol{\Lambda}^{(t)} \mid \boldsymbol{\Lambda}^{*})}{\pi(\boldsymbol{\Lambda}^{(t)})q(\boldsymbol{\Lambda}^{*} \mid \boldsymbol{\Lambda}^{(t)})}\right) \tag{3.27}$$

In practice, terms drawn from π by using the Markov chain defined by the transition kernel q can be set up by the following steps: (1) Initializing the iteration counter $\upsilon = 1$ and setting an arbitrary initial value $\boldsymbol{\Lambda}^{(0)}$; (2) Moving the chain to a new value $\boldsymbol{\Lambda}^{*}$ generating from the density q (**.** $|\boldsymbol{\Lambda}^{(t)}$); (3) Evaluating the acceptance probability of the move $r(\boldsymbol{\Lambda}^{(t)}, \boldsymbol{\Lambda}^{*})$. If the move is accepted, $\boldsymbol{\Lambda}^{(t+1)} = \boldsymbol{\Lambda}^{*}$. If it is not accepted, $\boldsymbol{\Lambda}^{(t+1)} = \boldsymbol{\Lambda}^{(t)}$, and the chain does not move; (4) Changing the counter from υ to $\upsilon + 1$ and return to Step (2) until convergence is reached.

B. RJMCMC

The RJMCMC algorithm is an effective approach to simulate dependent samples from π(**.**) while the parameter space is variable during sampling. According to Green (1995), at each iteration a new candidate $\boldsymbol{\Lambda}^{*}$ for $\boldsymbol{\Lambda}^{(t)}$ is proposed by an invertible deterministic function $\boldsymbol{\Lambda}^{*} = \boldsymbol{\Lambda}^{*}(\boldsymbol{\Lambda}^{(t)}, \boldsymbol{s})$ (assuming that the dimension of $\boldsymbol{\Lambda}^{*}$ is higher than that of $\boldsymbol{\Lambda}^{(t)}$) where a vector $\boldsymbol{s}$ of continuous random variables is defined for dimension matching and is sampled independently $\boldsymbol{\Lambda}^{(t)}$. The appropriate acceptance probability for the proposed transition from $\boldsymbol{\Lambda}^{(t)}$ to $\boldsymbol{\Lambda}^{*}$ is given by

$$r(\boldsymbol{\Lambda}^{(t)},\boldsymbol{\Lambda}^{*}) = \min\left\{1, \frac{\pi(\boldsymbol{\Lambda}^{*})q(\boldsymbol{\Lambda}^{*})}{\pi(\boldsymbol{\Lambda}^{(t)})q(\boldsymbol{\Lambda}^{(t)})p(\boldsymbol{s})}\left|\frac{\partial(\boldsymbol{\Lambda}^{*})}{\partial(\boldsymbol{\Lambda}^{(t)},\boldsymbol{s})}\right|\right\} \tag{3.28}$$

where $p(\boldsymbol{s})$ is the density function of $\boldsymbol{s}$ and $q(\boldsymbol{\Lambda}^{*})$ and $q(\boldsymbol{\Lambda}^{(t)})$ are the probabilities of a given move type in the states $\boldsymbol{\Lambda}^{*}$ and $\boldsymbol{\Lambda}^{(t)}$, respectively. The Jacobian is due to the change of variables from $(\boldsymbol{\Lambda}^{(t)}, \boldsymbol{s})$ to $\boldsymbol{\Lambda}^{*}$.

C. Optimization

The MAP scheme is used to find the point estimation $\{\hat{\boldsymbol{\Psi}},\hat{\boldsymbol{\Theta}},\hat{k},\hat{m},\hat{\boldsymbol{G}},\hat{\boldsymbol{L}}\}$ of the prior parameters $\{\boldsymbol{\Psi}, \boldsymbol{\Theta}, k, m, \boldsymbol{G}, \boldsymbol{L}\}$, that is,

$$\{\hat{\boldsymbol{\Psi}},\hat{\boldsymbol{\Theta}},\hat{k},\hat{m},\hat{\boldsymbol{G}},\hat{\boldsymbol{L}}\} = \arg\{\max p(\boldsymbol{\Psi},\boldsymbol{\Theta},k,m,\boldsymbol{G},\boldsymbol{L} \mid \boldsymbol{Z})\} \tag{3.29}$$

And then by estimating from $\{\hat{\boldsymbol{\Psi}},\hat{\boldsymbol{\Theta}},\hat{k},\hat{m},\hat{\boldsymbol{G}},\hat{\boldsymbol{L}}\}$ the feature of primary interest $\{\hat{\boldsymbol{G}},\hat{\boldsymbol{L}}\}$ can be obtained, which characterize the optimal segmentation of the given remotely sensed data, that is, $D = \{\hat{S}_l; l = 1,...,k\}$ where $\hat{S}_l = \{\hat{D}_j; L_j = l\}$.

3.7 Accuracy Assessment

In order to qualitatively assess the accuracy of the segmented results obtained, two assessment schemes are carried out, the statistical measure based scheme (Congalton and Green, 1999) and the buffer zone based scheme (Li et al., 2008).

A. Statistical Measure Technique

The statistical measures are used for evaluating the accuracy of the segmented results obtained from proposed algorithms in this thesis, which include error matrix, producer's accuracy, user's accuracy, overall accuracy, and Kappa coefficient (Congalton and Green, 1999).

B. Buffer Zone Technique

In order to assess the accuracy of the extracted linear objects such as road, coastline and object edge, the buffer zone based assessment technique is introduced by Li et al. (2008). Consider a linear object in reference image called reference line (BZ_0), where BZ_0 is a collection of pixels corresponding to the line. A buffer zone around BZ_0 can be constructed as follows

$$bl_i = \bigcup_{(x,y)\in BZ_{i-1}} N_8(x,y) - BZ_{i-1} \tag{3.30}$$

where bl_i is ith buffer layer, $N_8(x, y)$ is the 8-neighbours of the pixel located at (x, y) and BZ_i is the buffer zone contains i buffer layers and reference line BZ_0, that is,

$$BZ_i = \bigcup_{j=0}^{i} BZ_j \tag{3.31}$$

Fig. 3.8 (a) shows the structure of a buffer zone with 3 buffer layers, where the reference line is presented in red and its 3 buffer layers in yellow, blue and green, respectively. Fig. 3.8 (b) shows an extracted line in brown over the buffer zone.

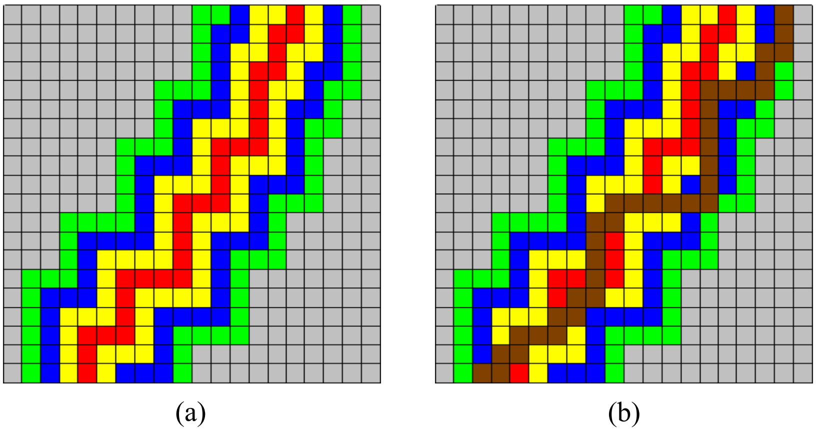


Fig. 3.8 (a) Buffer zone around the line in red with 3 buffer layers in yellow, blue and green, respectively. (b) Extracted line in brown over a buffer zone.

Based on above buffer zone structure, statistical measures for assessing the accuracy of the extracted line *EL* can be defined by counting the number of pixels of the extracted line located in different buffer layers.

$$b_i = \frac{\#\{(x,y);\ \ (x,y) \in bl_i \cup EL\}}{\#\{(x,y);\ \ (x,y) \in EL\}} \times 100, \text{ for } i = 0,\ 1,\ \ldots \tag{3.32}$$

where b_i is the percentage of extracted line over i'th buffer layer, EL is the extracted line and let $bl_0 = BE_0$. An accumulated percentage of the extracted line on buffer zone can be written as,

$$B_i = \sum_{j=0}^{i} b_i \quad \text{for } i = 0, 1, \ldots \tag{3.33}$$

3.8 Chapter Summary

This chapter focuses on the development of a novel framework for remotely sensed data segmentation. Following the Bayesian paradigm, the proposed segmentation framework consists of four successive stages. (1) The joint probability density (likelihood) for a given remotely sensed dataset is first formed. To this end, the domain of the dataset is partitioned into sub-regions corresponding to the components of homogeneous regions. Each sub-region is assigned a label to indicate the homogenous region to which the sub-region belongs. Modelling homogeneous structures is grounded in three RFs: label, neighbour and region fields. (2) Prior probability distributions for model parameters are constructed, which should capture general and scene-specific knowledge about homogeneous structures. (3) Combining the prior distributions and the likelihood, the posterior density of all model parameters conditional on the dataset is induced by Bayes' theorem. (4) The inference about the model parameters is carried out by MCMC, RJMCMC and MAP schemes.

The advantages of the proposed segmentation framework are as follows. First of all, the hierarchical RF-based model specifies a class of data models which can be used to represent a wide range of remotely sensed data such as SAR intensity imagery, LiDAR point cloud data and multispectral imagery. Secondly, the segmentation framework presented in this chapter can be extended to provide solutions to many other image processing problems such as feature extraction and object identification. Third, by using spatial statistics techniques for data domain partition, such as Voronoi tessellation and marked point process, geometrical properties can be easily introduced into segmentation algorithms by the prior distributions since random parameters define the object geometry. Fourth, since the data are modelled at different scales, the robustness of the designed segmentation algorithm is improved when noise cannot be assumed to be independent at the pixel level.

To demonstrate the applicability of the proposed framework, some segmentation algorithms for different sorts of data, such as SAR intensity imagery (in Chapter 4), LiDAR point cloud data (in Chapter 5) and texture imagery (in Chapter 6), are designed under this framework.

Chapter 4

SAR DATA ANALYSIS

This chapter addresses the issues on SAR data analysis. Section 4.1 describes a statistical region-based algorithm for SAR intensity image segmentation under the segmentation described in Chapter 3. In Section 4.2, the segmentation algorithm is used to design a segmentation-based scheme for the extraction of oil spill features. The results reveal that the designed algorithm can extract the oil spill features such as distribution parameters, area and shape. Section 4.3 presents a new algorithm for the detection of oil spills from SAR intensity images. The presented algorithm combines the marked point process, the Bayesian inference, and the MCMC techniques.

4.1 SAR Intensity Image Segmentation

4.1.1 Description of Algorithm for SAR Intensity Image Segmentation

Consider a SAR intensity image $\boldsymbol{Z} = \{Z_i\ ;\ i = 1, \ldots, n\}$, where n is the number of sampling points (pixels), i is the index of sampling points, Z_i is the intensity of pixel at lattice (x_i, y_i) regularly arranged on the image domain D. Given a set of generating points $\boldsymbol{G} = \{(u_j, v_j) \in D; j = 1, \ldots, m\}$ where m is the unknown number of generating points and possesses Poisson distribution defined by Eq. (3.19), D is partitioned into a set of Voronoi polygons $D = \{D_j; j = 1, \ldots, m\}$ by Eq. (3.5). Assume that the image consists of k homogeneous regions known *a priori*. Associated with each polygon, there is a random label variable which indicates the homogenous region to which the polygon belongs, label variables for all polygons form a label field, $\boldsymbol{L} = \{L_j; j = 1, \ldots, m\}$. A realization of $\boldsymbol{L}$, $\boldsymbol{l} = \{l_j \in \{1, \ldots, k\}; j = 1, \ldots, m\}$, corresponds to a segmentation of the image. In a given polygon P_j, the intensity values of pixels, $\boldsymbol{Z}_j = \{Z_i; (x_i, y_i) \in P_j\}$, are characterized by identical and independent Gamma distributions (Lee et al., 1994) conditionally on the label $L_j = l_j$ with the PDF as follows

$$p(\boldsymbol{Z}_j \mid L_j, \boldsymbol{\theta}_{L_j}) = \prod_{(\boldsymbol{x}_i, y_i) \in D_j} \frac{1}{\Gamma(\alpha_{L_j})} \beta_{L_j}{}^{-\alpha_{L_j}} Z_i^{\alpha_{L_j} - 1} \exp\left(-\frac{Z_i}{\beta_{L_j}} \right) \tag{4.1}$$

where $\boldsymbol{\theta}_{L_j} = (\alpha_{L_j}, \beta_{L_j})$ is the parameter vector, α_{L_j} and β_{L_j} are the shape and scale parameters of the Gamma distribution, respectively. The joint PDF of $\boldsymbol{Z}$, given m, $\boldsymbol{L}$, $\boldsymbol{G}$ and the parameters of Gamma distributions for all homogeneous regions, becomes

$$p(\boldsymbol{Z} \mid m, \boldsymbol{L}, \boldsymbol{G}, \boldsymbol{\theta}) = \prod_{j=1}^{m} p(\boldsymbol{Z}_j \mid L_j, \boldsymbol{\theta}_{l_j}) = \prod_{l=1}^{k} \prod_{(\boldsymbol{x}_i, y_i) \in \Delta_l} \frac{1}{\Gamma(\alpha_l)} \beta_l^{-\alpha_l} Z_i^{\alpha_l - 1} \exp\left(-\frac{Z_i}{\beta_l} \right) \tag{4.2}$$

where $\boldsymbol{\theta}$ is the Gamma distribution parameter vector, $\boldsymbol{\theta} = \{\boldsymbol{\theta}_l = (\alpha_l, \beta_l);\ l = 1, \ldots, k\}$, and Δ_l is the set of polygons with the same label l, $\Delta_l = \{D_j;\ L_j = l, j = 1, \ldots, m\}$.

The shape and scale parameters of the Gamma distribution are assumed to be identical independent Gaussian distributions, that is, $\alpha \sim N(\mu_\alpha, \sigma_\alpha)$ and truncated $\alpha > 0$, $\beta \sim N(\mu_\beta, \sigma_\beta)$, where μ_α, μ_β, σ_β and σ_α are the means and standard deviations of the Gaussian distributions. The joint PDF's of $\boldsymbol{\alpha} = \{\alpha_l;\ l = 1, \ldots, k\}$ and $\boldsymbol{\beta} = \{\beta_l;\ l = 1, \ldots, k\}$ can be written, respectively

$$p(\boldsymbol{\alpha}) = \prod_{l=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_\alpha} \exp\left[-\frac{(\alpha_l - \mu_\alpha)^2}{2\sigma_\alpha^{\ 2}}\right] \tag{4.3}$$

$$p(\boldsymbol{\beta}) = \prod_{l=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_\beta} \exp\left[-\frac{(\beta_l - \mu_\beta)^2}{2\mu_\beta^{\ 2}}\right] \tag{4.4}$$

According to Eq. (3.26), the posterior distribution of m, $\boldsymbol{L}$ $\boldsymbol{G}$, and $\boldsymbol{\theta}$ given $\boldsymbol{Z}$ can be rewritten as

$$p(m,\boldsymbol{L},\boldsymbol{G},\boldsymbol{\theta} \mid \boldsymbol{Z}) \propto p(\boldsymbol{Z} \mid m,\boldsymbol{L},\boldsymbol{G},\boldsymbol{\theta})p(\boldsymbol{G} \mid m)p(\boldsymbol{\theta})p(\boldsymbol{L} \mid m)p(m)$$

$$= \prod_{l=1}^{k} \prod_{(\boldsymbol{x}_i,y_i)\in\Delta_l} \frac{1}{\Gamma(\alpha_l)} \beta_l^{-\alpha_l} Z_i^{\alpha_l - 1} \exp\left(-\frac{Z_i}{\beta_l}\right) \times \prod_{j=1}^{m} \frac{\exp\left(c\sum_{j'\in N_j} t(L_j, L_{j'})\right)}{\sum_{l=1}^{k} \exp\left(c\sum_{j'\in N_j} t(l, L_{j'})\right)} \times \frac{1}{|D|^m} \qquad (4.5)$$

$$\prod_{l=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_\alpha} \exp\left[-\frac{(\alpha_l - \mu_\alpha)^2}{2\sigma_\alpha^{\ 2}}\right] \times \prod_{l=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_\beta} \exp\left[-\frac{(\beta_l - \mu_\beta)^2}{2\sigma_\beta^{\ 2}}\right] \times \frac{\lambda^m}{m!} \exp(-\lambda)$$

In order to segment a SAR image, it is necessary to simulate the posterior distribution defined in Eq. (4.5) and estimate its parameters. The RJMCMC algorithm described in Section 3.6 is used.

The move types designed in this paper include: (1) updating Gamma distribution parameters, (2) updating labels, (3) updating positions of generating points, and (4) birth or death of generating points.

Move 1: *updating Gamma distribution parameters*. The parameter vector for Gamma distributions can be written as $\boldsymbol{\theta} = \{\boldsymbol{\theta}_l;\ l = 1, \ldots, k\}$ where $\boldsymbol{\theta}_l = (\alpha_l, \beta_l)$. Assuming that the probability distributions for the proposals α_l^* and β_l^* are Gaussian distributions with means α_l and β_l, and standard differences ε_α and ε_β, respectively, i.e., $\alpha_l^* \sim N(\alpha_l, \varepsilon_\alpha)$ and $\beta_l^* \sim N(\beta_l, \varepsilon_\beta)$, the acceptance probability for the proposals α_l^* and β_l^* can be obtained

$$r_{\alpha,\beta}(\boldsymbol{\theta}_l, \boldsymbol{\theta}_l^*) = \min\left\{1, \prod_{j \in J_l} \frac{p(\boldsymbol{Z}_j \mid \boldsymbol{\theta}_l^*) \times p(\boldsymbol{\theta}_l^*)}{p(\boldsymbol{Z}_j \mid \boldsymbol{\theta}_l) \times p(\boldsymbol{\theta}_l)}\right\} \tag{4.6}$$

where $J_l = \{j'; L_{j'} = l\}$.

Move 2: updating labels. A polygon D_j with the label l_j is randomly drawn. In order to update its label, a new label l_j^* is then uniformly drawn from $\{1, \ldots, k\}$. The acceptance probability for l_j^* can be written as

$$\begin{aligned} r_l(l_j, l_j^*) &= \min\left\{1, \frac{p(\boldsymbol{Z} \mid \boldsymbol{G}^*)}{p(\boldsymbol{Z} \mid \boldsymbol{G})}\right\} \\ &\min\left\{1, \frac{\prod_{(\boldsymbol{x}_i, y_i) \in D_j} \frac{1}{\Gamma(\alpha_{L_j^*})} \beta_{L_j^*}^{-\alpha_{L_j^*}} Z_i^{\alpha_{L_j^*}-1} \exp\left(-\frac{Z_i}{\beta_{L_j^*}}\right) \times \exp\left(c \sum_{j' \in N_j} t(l_j^*, l_{j'})\right)}{\prod_{(\boldsymbol{x}_i, y_i) \in D_j} \frac{1}{\Gamma(\alpha_{L_j})} \beta_{L_j}^{-\alpha_{L_j}} Z_i^{\alpha_{L_j}-1} \exp\left(-\frac{Z_i}{\beta_{L_j}}\right) \times \exp\left(c \sum_{j' \in N_j} t(l_j, l_{j'})\right)}\right\} \end{aligned} \tag{4.7}$$

Move 3: *moving position of generating points*. One of generating points in $\boldsymbol{G} = \{(u_j, v_j); j = 1, \ldots, m\}$ is drawn at random, say (u_j, v_j). A proposed generating point (u_j^*, v_j^*) is drawn uniformly from its corresponding polygon D_j. The new generating point gives rise to the local changes of D_j and its neighbour polygons $ND_j = \{D_{j'} ; j' \in N_j\}$ to D_j^* and $ND_j^* = \{D_{j'}^*, j' \in N_j\}$. Fig. 4.1 shows an example of the changes of Voronoi polygons when the generating point moves from generating point (u_2, v_2) in red to (u_2^*, v_2^*) in blue, where the blue and red lines are the boundaries of new and old Voronoi polygons, respectively, while the green lines are invariant boundaries.

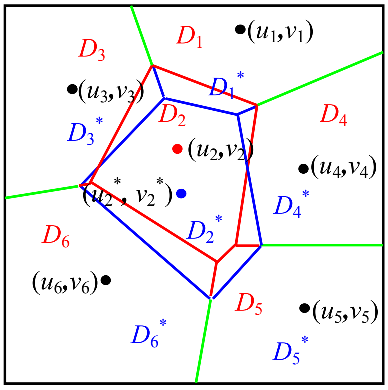


Fig. 4.1 Changes of Voronoi polygons when changing the generating point (u_2, v_2) to (u_2^*, v_2^*); New polygons are delineated by blue and green segments and old polygons by red and green lines.

The acceptance probability for the move turns out to be

$$r_{gp}\left((u_j,v_j)_j,(u_j^*,v_j^*)\right)=\min\left\{1,\frac{\prod_{D_{j'}*\in DN_j^*}\prod_{(x_i,y_i)\in D_{j'}*}\frac{1}{\Gamma(\alpha_{L_j})}\beta_{L_j}^{-\alpha_{l_j}}Z_i^{\alpha_{L_j}-1}\exp\left(-\frac{Z_i}{\beta_{L_j}}\right)}{\prod_{D_{j'}\in DN_j}\prod_{(x_i,y_i)\in D_{j'}}\frac{1}{\Gamma(\alpha_{L_j})}\beta_{L_j}^{-\alpha_{L_j}}Z_i^{\alpha_{L_j}-1}\exp\left(-\frac{Z_i}{\beta_{L_j}}\right)}\right\} \quad (4.8)$$

Move 4: *birth or death of generating points*. Suppose that the current number of generating points is m and let the probabilities of proposing a birth or death operation be b_m or d_m, respectively. Consider a birth operation which increases the number of generating points from m to m +1 and assume that the new generating point is identified with m +1 and its location (u_{m+1}, v_{m+1}) is drawn uniformly from D. Let the polygon induced by (u_{m+1}, v_{m+1}) be D_{m+1} and its label l_{m+1} is drawn from {1, …, k} uniformly. The set of labels of D_{m+1}'s neighbour polygons is N_{m+1} = {j'; $D_{j'}$ ~ D_{m+1}}. The Voronoi tessellation is modified by adding the proposed generating point from D

$= \{D_1, \ldots, D_{j'}, \ldots, D_m\}$ to $D^* = \{D_1, \ldots, D_{j'}^*, \ldots, D_m, D_{m+1}\}$ where $j' \in N_{m+1}$. Fig. 4.2 shows the modified Voronoi tessellation after adding the new generating point (u_7, v_7), in which the original tessellation have six generating points (u_1, v_1)- (u_6, v_6) and they induce six polygons D_1-D_6, see Fig. 4.2 (a). By proposing the new generating point (u_7, v_7), the new polygon D_7 is formed and its neighbours include D_2, D_3 and D_6, that is, $N_7 = \{2, 3, 6\}$.

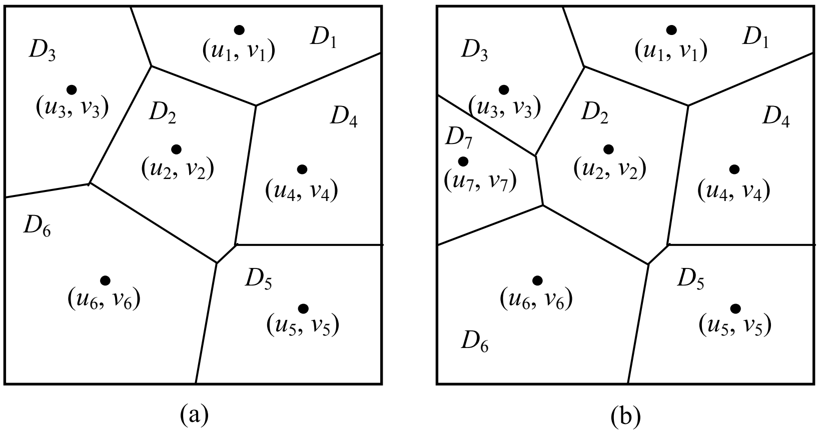


Fig. 4.2 (a) Voronoi tessellation with 6 polygons D_1-D_6 corresponding to generating points (u_1, v_1)- (u_6, v_6); (b) Voronoi tessellation with 7 polygons D_1-D_7 formed by adding the generating point (u_7, v_7).

Birth or death of a generating point does not affect Gamma distribution parameters in $\boldsymbol{\theta}$. As a result, the parameter vector for the birth operation becomes $\boldsymbol{\Theta}^* = (k, m+1, \boldsymbol{L}^*, \boldsymbol{G}^*, \boldsymbol{\theta})$, where $\boldsymbol{G}^* = ((u_1, v_1), \ldots, (u_m, v_m), (u_{m+1}, v_{m+1}))$ and $\boldsymbol{L}^* = (L_1, \ldots, L_m, L_{m+1})$. The acceptance probability for the birth operation can be written as

$$r_b(\boldsymbol{\Theta}, \boldsymbol{\Theta}^*) = \min\{1, R_b\} \tag{4.9}$$

where

$$R_b = \frac{p(\boldsymbol{Z} \mid m+1, \boldsymbol{L}^*, \boldsymbol{G}^*, \boldsymbol{\theta})\, p(m+1)\, p(\boldsymbol{G}^* \mid m+1)\, p(\boldsymbol{L}^* \mid m+1)\, r_{b_m}(\boldsymbol{\Theta}^*)}{p(\boldsymbol{Z} \mid m, \boldsymbol{L}, \boldsymbol{G}, \boldsymbol{\theta})\, p(m)\, p(\boldsymbol{G}^* \mid m+1)\, p(\boldsymbol{L} \mid m)\, r_{d_{m+1}}(\boldsymbol{\Theta})\, q(\boldsymbol{s})} \left| \frac{\partial(\boldsymbol{\Theta}^*)}{\partial(\boldsymbol{\Theta}, \boldsymbol{s})} \right| \tag{4.10}$$

where $r_{bm} = b_m$, $r_{dm+1} = d_{m+1}/(m+1)$, $\boldsymbol{s} = l_{m+1}$ and other terms in Eq. (4.10) can be expressed as follows

$$\frac{p(\boldsymbol{Z} \mid m+1, \boldsymbol{L}^*, \boldsymbol{G}^*, \boldsymbol{\theta})}{p(\boldsymbol{Z} \mid m, \boldsymbol{L}, \boldsymbol{G}, \boldsymbol{\theta})} = \frac{\displaystyle\prod_{j \in \{m+1, N_{m+1}\}} \prod_{(x_i, y_i) \in D_j} \frac{1}{\Gamma(\alpha_{L_j})} \beta_{L_j}^{-\alpha_{L_j}} Z_i^{\alpha_{L_j}-1} \exp\left(-\frac{Z_i}{\beta_{L_j}}\right)}{\displaystyle\prod_{j \in N_{m+1}} \prod_{(x_i, y_i) \in D_j} \frac{1}{\Gamma(\alpha_{L_j})} \beta_{L_j}^{-\alpha_{L_j}} Z_i^{\alpha_{L_j}-1} \exp\left(-\frac{Z_i}{\beta_{L_j}}\right)} \tag{4.11}$$

$$\frac{p(\boldsymbol{L}^* \mid m+1)}{p(\boldsymbol{L} \mid m)} = \frac{\displaystyle\prod_{j \in \{m+1, N_{m+1}\}} \frac{\exp\left(c \sum_{j' \in N_j^*} t(l_j, l_{j'})\right)}{\sum_{l=1}^{k} \exp\left(c \sum_{j' \in N_j^*} t(l, l_{j'})\right)}}{\displaystyle\prod_{j \in N_{m+1}} \frac{\exp\left(c \sum_{j' \in N_j} t(l_j, l_{j'})\right)}{\sum_{l=1}^{k} \exp\left(c \sum_{j' \in N_j} t(l, l_{j'})\right)}} \tag{4.12}$$

$$\frac{p(m+1)}{p(m)} = \frac{\lambda}{m+1} \tag{4.13}$$

$$\frac{p(\boldsymbol{G}^* \mid m+1)}{p(\boldsymbol{G} \mid m)} = \frac{1/|D|^{(m+1)}}{1/|D|^{m}} = \frac{1}{|D|} \tag{4.14}$$

$$\left|\frac{\partial(\boldsymbol{\Theta}^{*})}{\partial(\boldsymbol{\Theta},\boldsymbol{s})}\right| = 1 \tag{4.16}$$

$$q(\boldsymbol{s}) = \frac{1}{k} \tag{4.17}$$

The acceptance probability for the death of generating point is given by

$$r_d(\boldsymbol{\Theta},\boldsymbol{\Theta}^{*}) = \min\{1, R_d\}, \text{ and } R_d = R_b^{-1} \tag{4.18}$$

For any given proposal with acceptance probability *r*, it is accepted if and only if $r \geq \xi$, where ξ is drawn from [0, 1] uniformly, that is, ξ ~ *U*(0, 1). The MAP estimation defined in Eq. (3.29) is used to obtain optimal parameters.

4.1.2 Experimental Results on SAR Intensity Image Segmentation

The proposed algorithm is tested with four real C-band RADARSAT-1/2 images. In addition, a simulated SAR image is also used to evaluate the proposed algorithm quantitatively.

A. RADARSAT-1/2 Images

Fig. 4.3 shows three real Radarsat-1/2 ScanSAR images with dimensions of

256×256 pixels. Among them, (a) presents a RADARSAT-2 standard mode image with horizontal-vertical (HV) polarization and spatial resolution of 25 *m*, which covers part of Stanley Park, Vancouver, British Columbia, Canada, including urban area (white), forest (grey) and water (black); (b) shows a RADARSAT-1 image of a coastal scene with vertical-vertical (VV) polarization and spatial resolution of 30 *m*. Visually, both of them include three homogeneous regions; (c) also presents a Radarsat-1 4-look image with VV polarization and spatial resolution of 50 *m*, which reveals four types of sea ice structures in Ungava Bay, Quebec, Canada. In the remainder of this paper, we use a, b and c to indicate the test images shown in Figs. 4.3 (a)-(c) and 1, 2, 3 and 4 denote the homogenous regions in the decreasing order of their means in each test image.

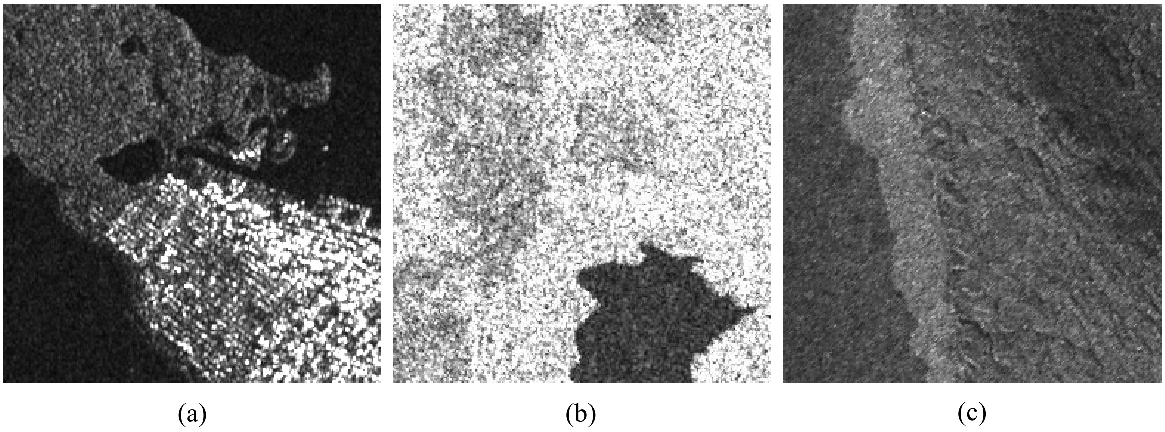


Fig. 4.3 Radarsat-1/2 ScanSAR images with 256×256 pixels. (a) RADARSAT-2 HV standard mode image; (b) RADRASTA-1 VV image; (c) RADARSAT-1 VV image.

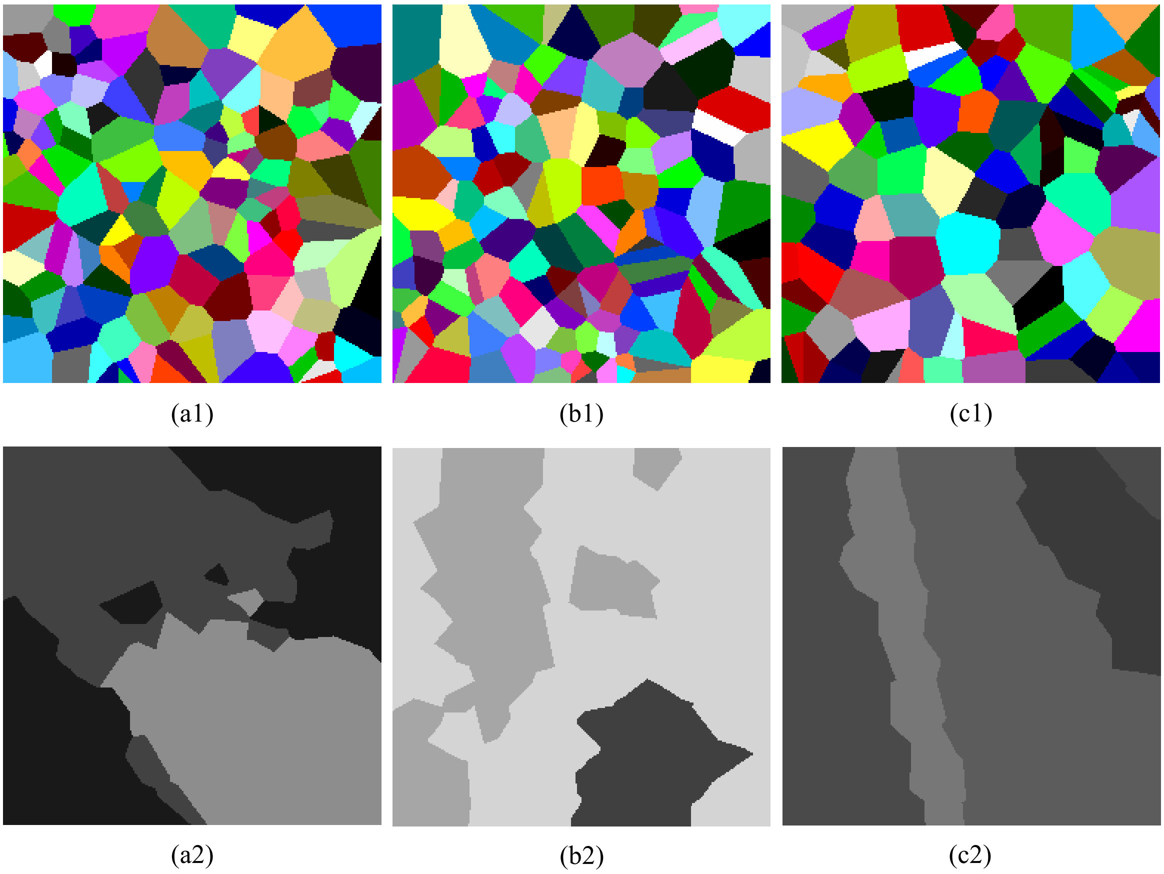
The constants used for testing the proposed segmentation algorithm are listed in Table 4.1.

Table 4.1 Constants used in Eq. (4.5)

Image	c	μ_α	σ_α	μ_β	σ_β	λ	ε_α	ε_β	k	T_m
a	1	4	0.5	32	4	96	0.5	1	3	8000
b									3	4000
c									4	4000

The coefficient defined in Eq. (3.6), c is set to be 1. The constants μ_α and μ_β are the means of shape parameter α and scale parameter β of the Gamma distributions in Eq. (4.3) and Eq. (4.4), respectively, i.e., $\mu_\alpha = E(\alpha)$ and $\mu_\beta = E(\beta)$. Given a multi-look SAR image in which the intensities of pixels are characterized by Gamma distribution, the shape parameter α is equal to the number of its looks (Lee et al., 1994). In this thesis, since α is considered as a random variable the value μ_α is set as the number of looks. For a Gamma distribution with shape parameter α and scale parameter β, the product of the two parameters, $\mu_\alpha \times \mu_\beta$, is equal to its mean. Then the value $\mu_\alpha \times \mu_\beta = E(\alpha) \times E(\beta) = E(\alpha \times \beta)$ (the last equation is true, since α and β are independent) is taken 128 = 256/2 (i.e. the midpoint of 256 grey levels) since the pixel intensities in a grey-scale image vary in the range of 0 and 255. The constant λ_m is the mean of a Poisson distribution in Eq. (3.18). The constants ε_α and ε_β are the proposal variances for α and β, respectively, which affect the sampling and convergence of the algorithm under the MCMC scheme (Dryden et al., 2003). Besag and Moran (1975) suggested choosing the proposal variances so that the acceptance probability lies in the interval (0.3-0.7). However, it has been found from this experiment that the proposal variances causing the acceptance probability around 0.1 still make the algorithm work well. For simplicity, the number of homogeneous regions k for the scene presented in a SAR image is determined by manual inspection *a priori*. In practice, selecting the number is not very reliable as it depends on the experience of human operators and is sometimes impossible since the ground truth is always unknown in advance. Therefore, an algorithm for automatically identifying the number of homogenous regions is necessary. The constant T_m is the maximum iterations of the algorithm. Usually, it depends on the complexity of the scene revealed in a SAR image and requirement of segmentation accuracy. The T_m's selected in this experiment are uninformative and larger than practically used ones.

The initial partitions of image domain D is carried out by the Voronoi tessellation, in which the number of generating points m_0 is drawn from the Poisson distribution with the mean 96 and the locations of m_0 generating points are drawn from D uniformly. The initial segmentation is performed by randomly assigning a label to each polygon in the initial partition of D from the Bernoulli distribution with probabilities $p_j = 1/k$, where $j = \{1, \ldots, m\}$ and k is the number of homogeneous regions in each test image. It is found that there is no notable impact of the initial segmentation on the final segmentation. Fig. 4.4 (a1)-(c1) show the results of the optimal partitions of D with 146, 140 and 104 polygons, respectively. Fig. 4.3 (a2)-(c2) show the results of the optimal segmentation in terms of the MAP estimation after all iterations obtained at the 7,997th out of 8,000, 3,978th out of 4,000 and 3,984th out of 4,000 iterations, respectively, where the tone of each region is represented by its estimated mean.



(a1) (b1) (c1)

(a2) (b2) (c2)

Fig. 4.4 Results of optimal partitions (a1)-(c1) and segmentations (a2)-(c2).

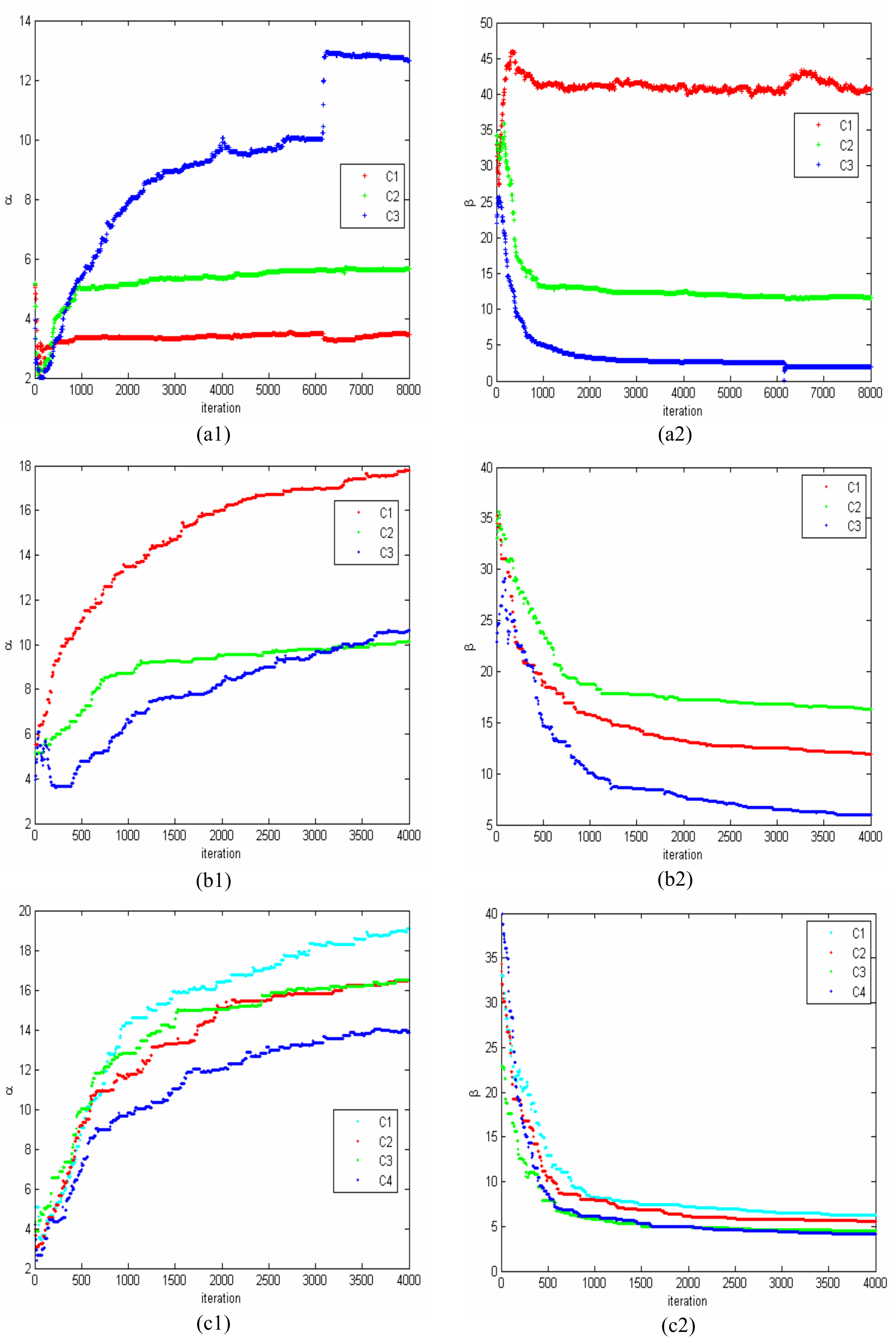
Table 4.2 summarizes estimated shape parameters $\alpha_{1,\,..,\,k}$ and scale parameters $\beta_{1,\,...\,,\,k}$ for the Gamma distributions corresponding to the segmented homogenous regions, the final number of polygons *m* and the number of the iteration *ite* at which the maximum posterior is obtained.

Table 4.2. Estimated parameters, the number of polygons and iterations for optimal segmentation

image	α_1	α_2	α_3	α_4	β_1	β_2	β_3	β_4	*m*	*ite*
a	3.47	5.65	12.66	-	40.70	11.69	1.97	-	146	7997
b	10.65	10.19	17.80	-	5.98	16.34	11.94	-	140	3978
c	4.20	4.47	5.57	6.23	13.88	16.51	16.47	19.06	104	3984

Fig. 4.5 (a1)-(c1) give the changes of the estimated shape parameters while Fig.

4.5 (a2)-(c2) show the changes of the estimated scale parameters for the Gamma



(a1) (a2)

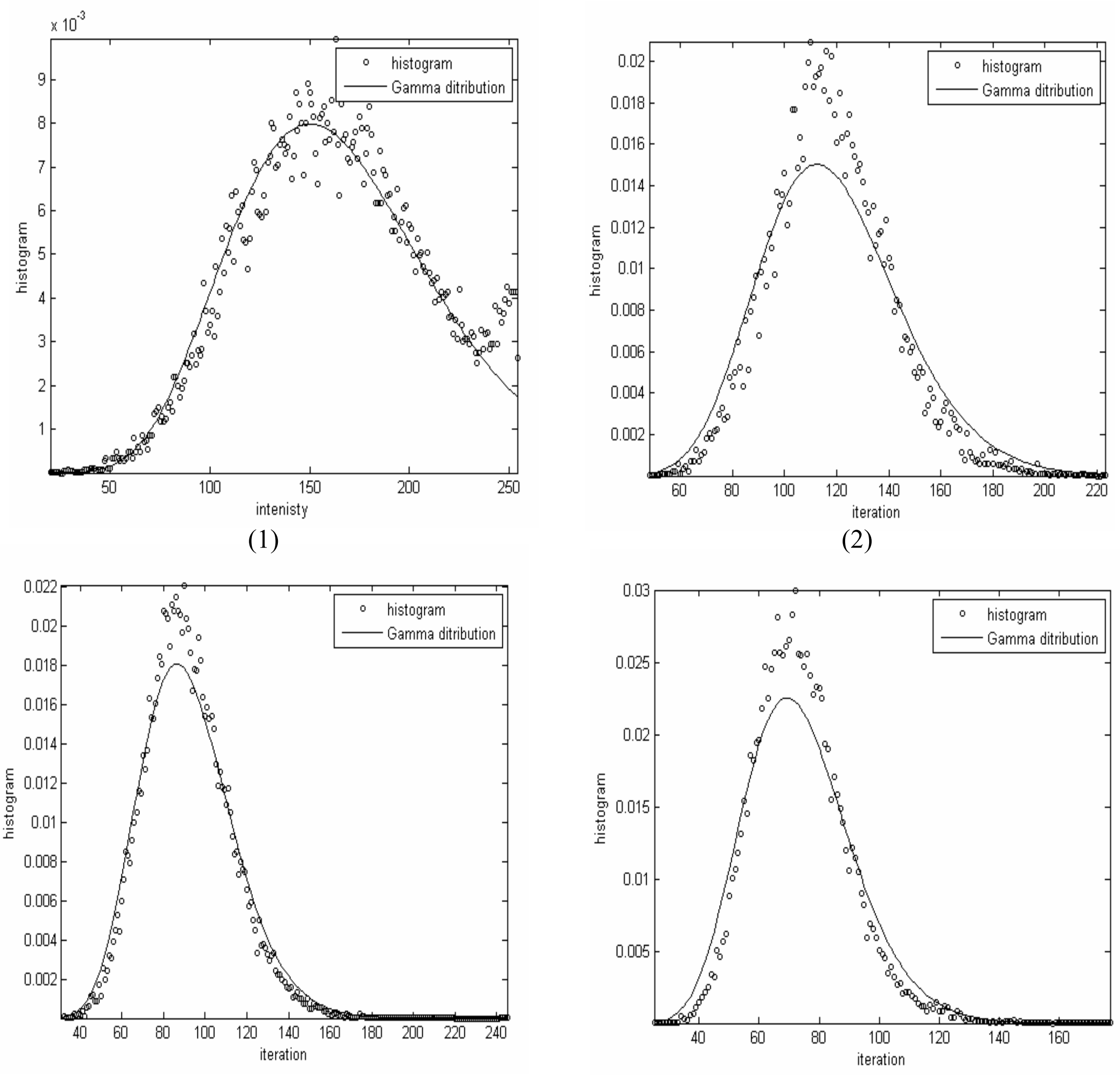
(b1) (b2)

(c1) (c2)

Fig. 4.5 Changes of Gamma distribution parameters during iterations, (a1)-(b1) for α and (a2)-(b2) for β.

distributions of the segmented homogeneous regions during iterations. Here C1, C2 and C3 indicate the homogenous regions. It can be seen that the shape and scale parameters converge to their stable values finally.

Fig. 4.6 shows the histogram of intensities and Gamma distributions with the estimated shape and scale parameters of the segmented homogeneous regions for the images shown in Fig. 4.3 (c). Fig. 4.7 shows the results for the image in Fig. 4.3 (a)



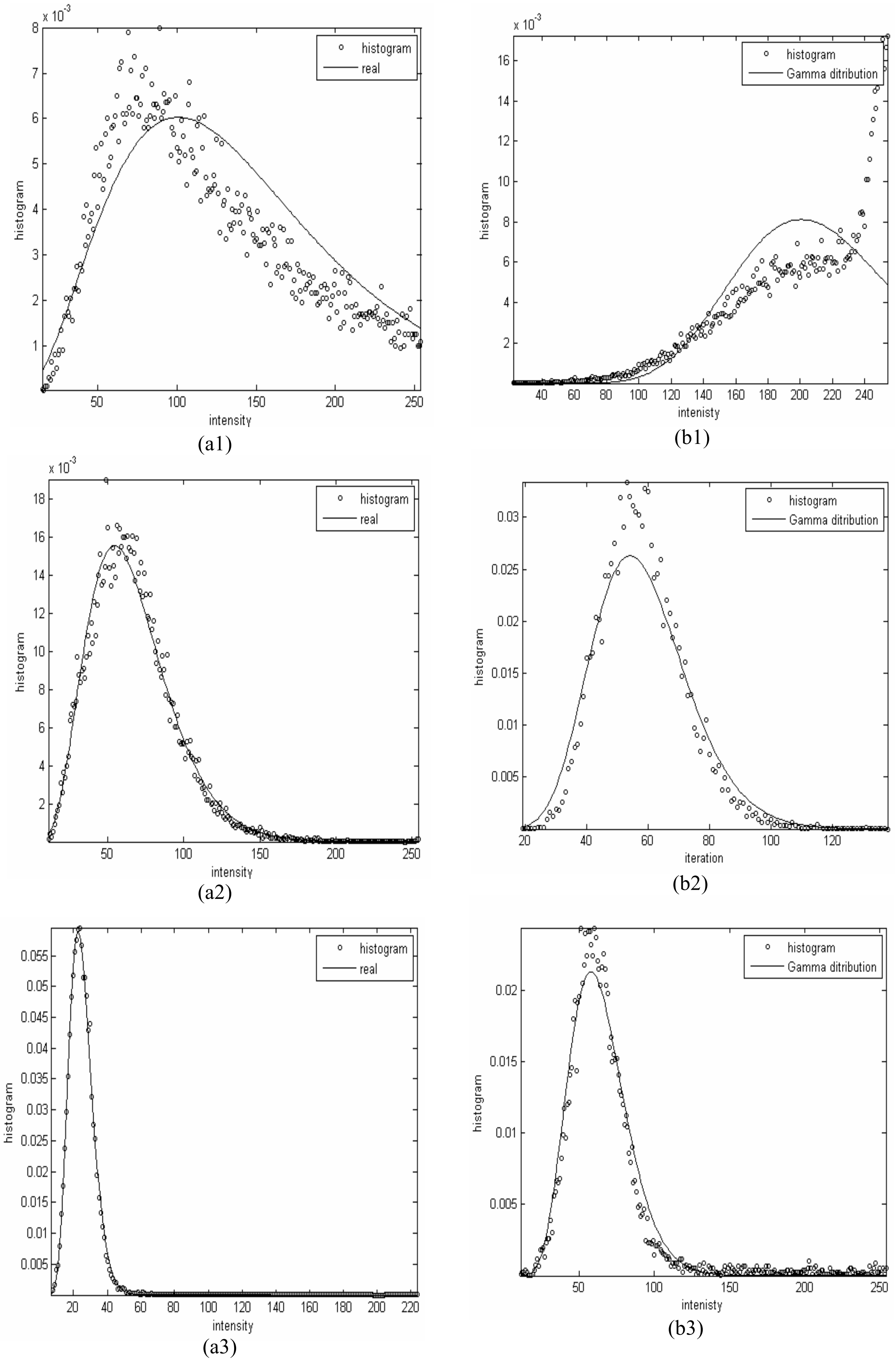
(1)

(2)

(3)

(4)

Fig. 4.6 Histograms and Gamma distributions with estimated parameters of segmented regions (1, 2, 3 and 4) for SAR intensity image shown in Fig 4.3 (c).



(a1) (b1)

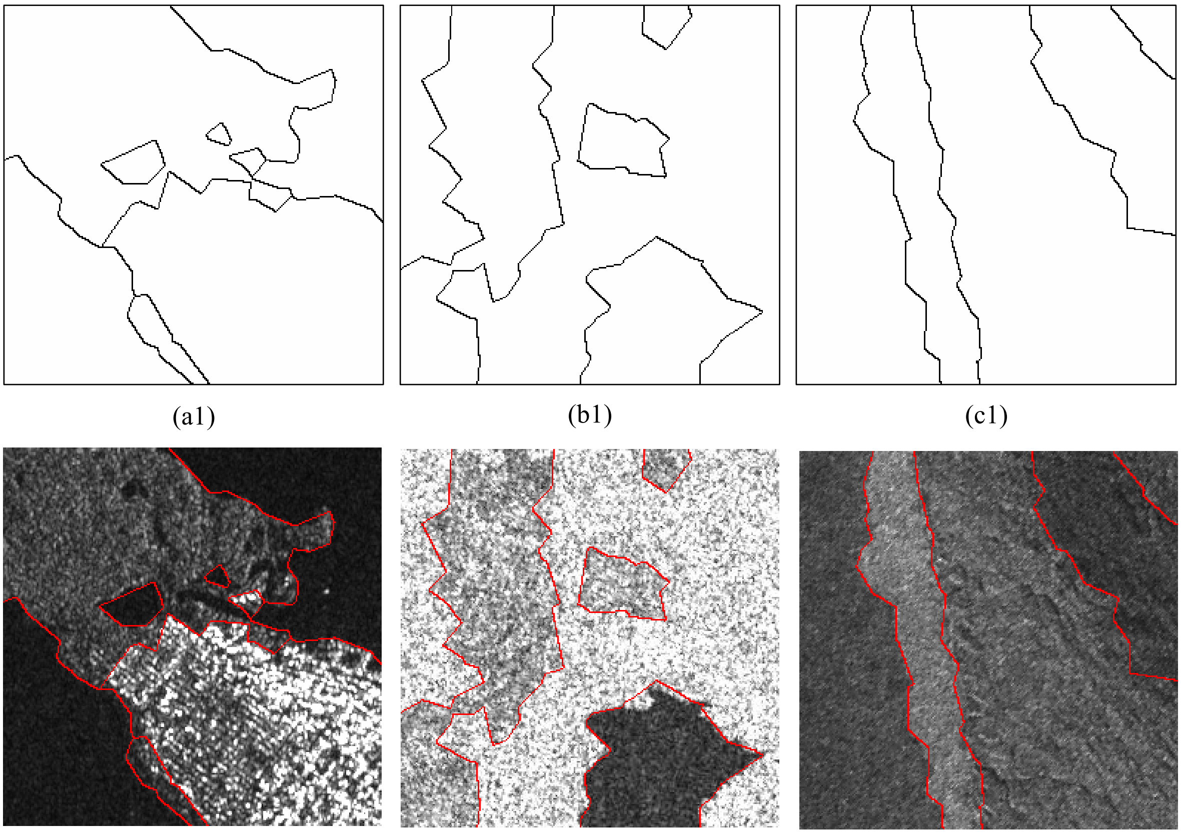
(a2) (b2)

(a3) (b3)

Fig. 4.7 Histograms and Gamma distributions with estimated parameters of segmented regions (1, 2 and 3) for SAR intensity image shown in Fig 4.3 (a) and (b).

and (b). As shown in Figs. 4. 6 (1) and Fig. 4.7 (b1) the curves of Gamma distributions for the regions with the maximum estimated means do not fit their histograms well. As shown in Fig. 4.3 (b) and (c), there are many light pixels in these regions and they make the distributions of intensities in the regions out of Gamma distributions. Nevertheless, the proposed algorithm still accurately identifies these regions. In other cases, the histograms match the Gamma distributions well.

For a visual assessment of whether the result is accurate, the outlines of the segmented homogeneous regions in Fig. 4.5 are delineated, see Fig. 4.8 (a1)-(c1), and then overlaid on the original images in red, see Fig. 4.8 (a2)-(c2). By visual inspection, the delineated outlines of the segmented homogenous regions match the edges of the real homogenous regions quite well.



(a1) (b1) (c1)

(a2) (b2) (c2)

Fig. 4.8. Delineated outlines (red) (a1)-(c1) and overlaid on test images (a2)-(c2).

B. Simulated SAR Image

Fig. 4.9 shows a simulated SAR image, which is generated based on the partition of a domain as shown in Fig. 4.9 (a). In the simulated image shown in Fig. 4.9 (b), the intensity values of pixels in each homogeneous region are drawn from Gamma distributions with shape parameters equal to 3, 4, and 5, and the scale parameters equal to 24, 32 and 40, respectively.

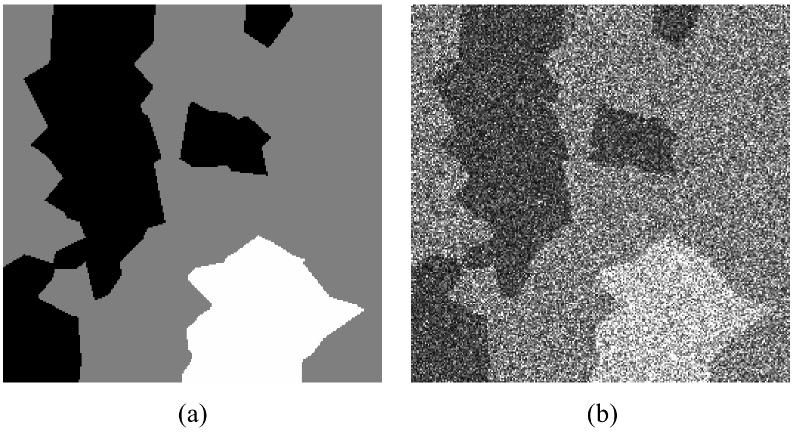
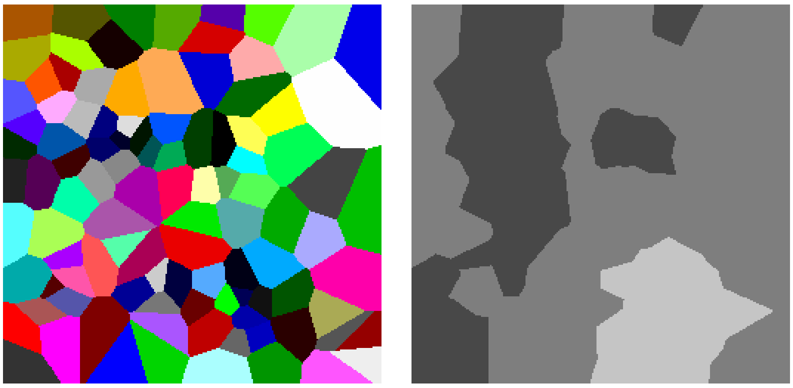

(a) (b)

Fig. 4.9 (a) Domain partition and (b) the simulated SAR image.

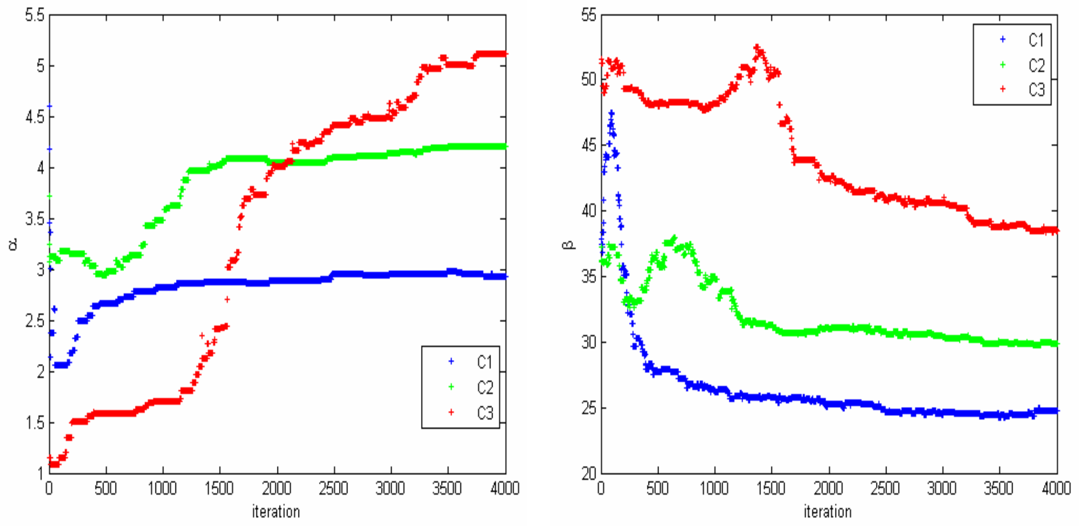
In the experiment using the simulated image, the constants used in our algorithm are the same as those listed in Table 4.1. The optimal results are obtained by MAP estimation at the 3,887th iteration out of 4,000. Fig. 4.10 (a) presents the results of the optimal partition of *D* with 140 polygons. Fig. 4.10 (b) shows the optimal segmentation of the simulated image, in which the homogeneous regions are presented by the estimated means.



(a) (b)

Fig. 4.10 (a) Optimal partition of image domain and (b) Optimal segmentation obtained at the 3,887th iteration out of 4,000 iterations.

Fig. 4.11 shows the changes of the shape and scale parameters during 4,000 iterations and the estimated parameters converge to their stable values finally.



(a) (b)

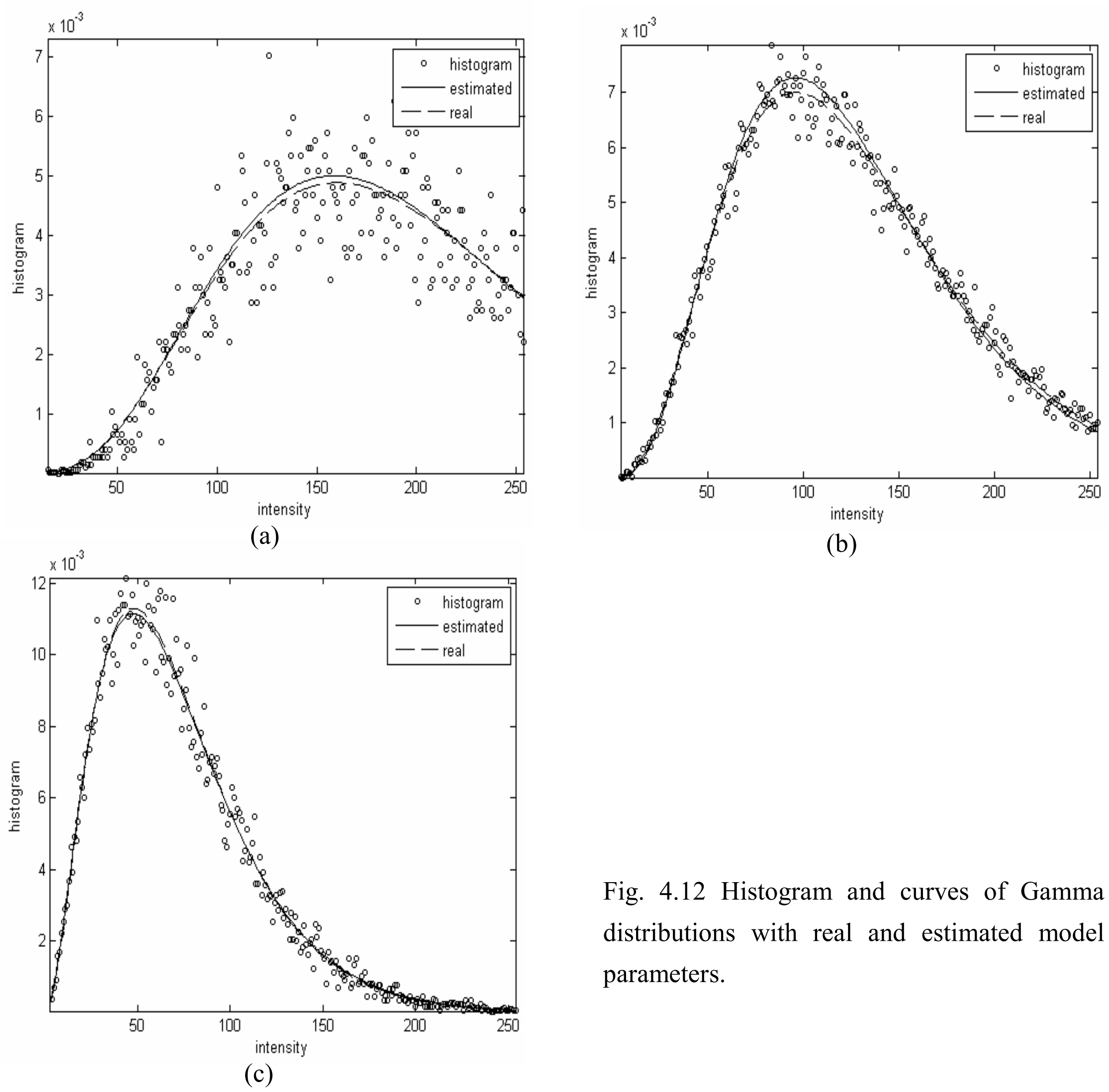
Fig. 4.11 Changes of estimated shape parameters (a) and scale parameters (b) during 4,000 iterations.

Table 4.3 reports the estimation values of the shape parameters (α's), scale parameters (β's) and their percentage error (e_α and e_β), respectively. The minimum accuracy of those estimated parameters is larger than 93 % (≈ 100-6.55). It can be concluded that the estimated values of shape and scale parameters are close to their real values listed in Table 4.3.

Table 4.3 Estimated model parameters α and β, errors e_α and e_β.

	C1	C2	C3
α	5.12	4.20	2.93
e_α (%)	2.35	5.00	2.32
β	24.71	29.90	38.53
e_β (%)	3.66	6.55	2.95

Fig. 4.12 shows the histograms, Gamma distributions with the real parameters and



(a)

(b)

(c)

Fig. 4.12 Histogram and curves of Gamma distributions with real and estimated model parameters.

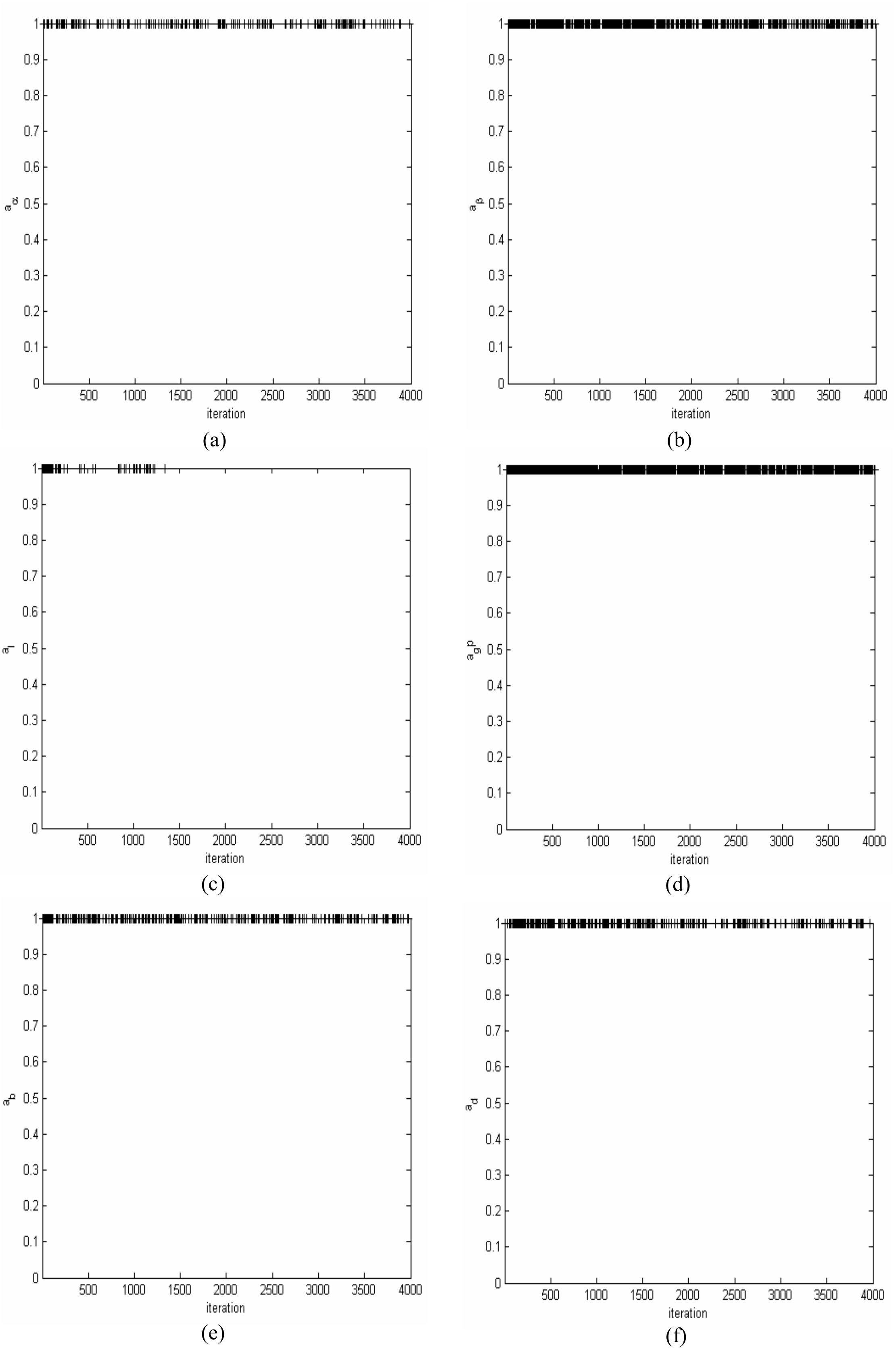
the estimated parameters. It can be seen that the histograms and distributions of pixel intensities for each homogeneous regions match very well.

Table 4.4 shows the acceptance rate for each type of moves, where r_α, r_β, r_l, r_{gp}, r_b and r_d are the acceptance rates of moves for updating shape parameter, updating scale parameter, updating label, moving generating point, birth and death of generating point, respectively.

Table 4.4 Acceptance rates of moves for shape parameter, scale parameter, label, generating point, birth and death (%)

r_α	r_β	r_l	r_{gp}	r_b	r_d
5.10	16.65	1.78	31.4	8.63	8.48

Fig. 4.13 presents the acceptances of the proposals in each move. As shown in Fig. 4.13 and Table 4.4, the operation of updating labels has the minimum acceptance rate and the accepting of the proposed labels is concentrated in the initial phase of iterations, within first 1400 iterations, see Fig. 4.13(c). The phenomenon implies that the proposed algorithm can rapidly segment the image into the homogenous regions. The operation of moving generating points has the maximum acceptance rate. It can be explained by the fact that the algorithm drives many more changes on the shape of polygons to fit the details of homogenous regions. The birth and death of generating points operate throughout the period of the iterations, though with lower acceptance rates. Since the computation burden for both moves is very heavy, it is necessary to find a better way to control them effectively.



(a) (b) (c) (d) (e) (f)

Fig. 4.13 Acceptances of proposals in different move types indicated by 1: (a) shape parameter, (b) scale parameter, (c) label, (d) moving generating point, (e) birth, (f) dead.

Fig. 4.14 demonstrates the procedure of fitting regions during the iterations, in which the outlines of the real regions (blue) and the segmented regions within 2,000, 3,000 and 4,000 iterations (cyan, red, and green, respectively) are overlaid on the simulated image. As shown in Fig. 4.14 the delineated outlines of the segmented regions move asymptotically toward the outlines of the real regions when the algorithm is iterating.

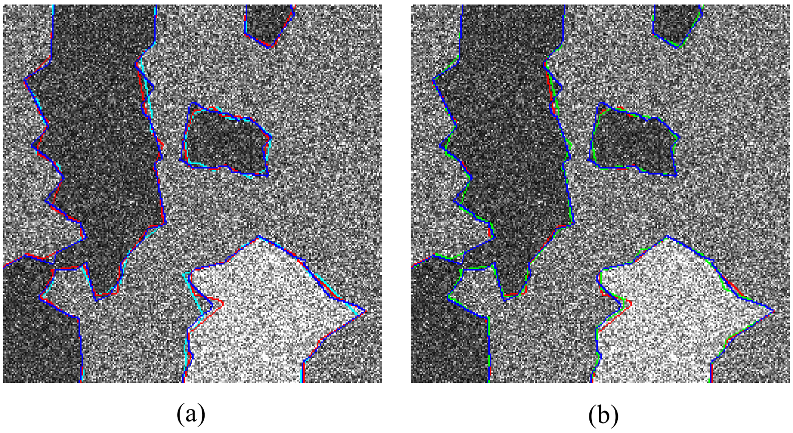


Fig. 4.14 Outlines of the real regions (blue) overlaid on the delineated outlines of the segmented regions within (a) 2,000 iterations (cyan) and 3,000 iterations (red), and (b) 3,000 iterations (red) and 4,000 iterations (green).

In order to assess the accuracy of the segmented results obtained by the proposed algorithm qualitatively, both the outlines of the real and the segmented homogenous regions are overlaid on the simulated image. By visual inspection of Fig. 4.13 (b), the delineated outlines of the segmented homogenous regions (green) are very close to the outline of the real homogenous regions (blue).

In this experiment, two assessment schemes are carried out for quantitative evaluation, the statistical measure based scheme (Congalton and Green, 1999) and the buffer zone based scheme (Li et al., 2008). In the statistical measure based scheme, some common measurements are used for accuracy assessment, including producer's accuracy, user's accuracy, overall accuracy and Kappa coefficient (Congalton and Green, 1999). Table 4.5 presents an error matrix, where C_1, C_2 and C_3 indicate the homogenous regions and $\sum C_{r.}$ and $\sum C_{.s}$ the row and column totals.

Table 4.5 Error matrix for segmented result shown in Fig. 4.10.

	C_1	C_2	C_3	$\sum C_{r.}$
C_1	7478	401	0	7560
C_2	203	38759	477	39439
C_3	0	82	18136	18537
$\sum C_{.s}$	7681	39242	18613	65536

Table 4.6 gives the producer's accuracy, user's accuracy, overall accuracy and Kappa coefficient. As shown in Table 4.6, in the worst case 97.36 % of real pixels (7,478 out of 7,681) in the lightest block on the bottom of the simulated image are correctly segmented. The algorithm incorrectly omitted 2.64 % of pixels (203 out of 7,681) in the worst case. Correspondingly, 97.84 % of pixels (7,478 out of 7,560) segmented in the same region are correctly identified and only 2.16 % of pixels (401 out of 7,560) are incorrectly segmented to other homogenous regions. In the similar way, the segmented results for other homogeneous regions in the simulated image can be evaluated. As a conclusion, the high segmentation accuracy is anticipated when the proposed algorithm is applied. As shown in Table 4.6, the Kappa coefficient for the

segmented result is up to 0.968. According to the general interpretation rules for thematic accuracy assessment, the Kappa coefficients 0.81-1.00 can be interpreted as almost perfect (Fleiss, 1975).

Table 4.6 Producer's accuracy, user's accuracy, overall accuracy and Kappa coefficient

Measure	C_1	C_2	C_3	Overall accuracy (%)	Kappa
Producer's accuracy (%)	97.36	98.77	97.44	98.28	0.968
User's accuracy (%)	97.84	98.28	98.92		

The buffer zone based accuracy assessment described in Section 3.7 is also used for evaluating the proposed segmentation algorithm. Fig. 4.15 shows the extracted outlines (block) of the segmented homogenous regions lying in the buffer zone (grey) with 4 pixels width around the real outlines at each side. It can be clearly seen that almost all the extracted outlines of the segmented regions lie within the buffer zone.

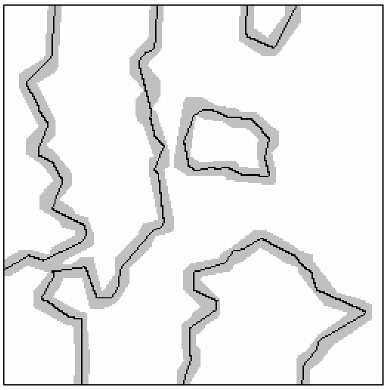


Fig. 4. 15 Extracted outlines overlaid on the buffer zones around the outlines of real regions.

Table 4.7 presents the percentage of the extracted outlines on each buffer layer.

Over 80% of the extracted outlines of the segmented homogenous regions are within the buffer zone with one pixel width around the real outlines and almost all the extracted outlines (around 99%) are on the buffer zone with 4 pixel widths around the real outlines.

Table 4.7. Percents of the extracted outlines on each buffer layer

B_0 (%)	B_1 /Σ_1 (%)	B_2/ Σ_2 (%)	B_3/ Σ_3 (%)	B_4/Σ_4 (%)
43.97	44.08/87.05	9.64/96.68	2.76/99.45	0.55/100.00

The proposed algorithm is developed using MATLAB running on a DELL Optiple GX 745 computer. The average computing time for an iteration, in which all four moves are accepted, is around 3.6sec. As a result, the average computation time for 4000 iteration is about 240 min. The computation burden for updating the model parameters, updating label, moving the generating point and birth/death of the polygon account for 5%, 20%, 35% and 40%, respectively, since the operations in the experiment for the Voronoi tessellation and finding neighbouring polygons are time-consuming.

4.2 Oil Spill Detection by Marked Point Process

Dark spot detection is a critical and fundamental step as a prelude for further feature extraction and classification of oil spills. To detect dark spots in SAR intensity imagery, it is reasonable to approximate the dark spots by a series of rectangular windows modeled with a marked point process, which is formed by a group of points and attaching to each point a window centred at the point. The number of points is assumed unknown *a priori* but satisfies a Poisson distribution with fixed mean given by a human operator. The intensities of pixels inside and outside the windows are modelled by two strict stationary random fields, respectively. That is, both of them satisfy independent and identical Gamma distributions, but the means for the former are less than that for the latter. Following the Bayesian paradigm, the mathematical form for the posterior distribution is obtained up to an integrating constant. A RJMCMC algorithm is introduced for simulation from the posterior. And the optimal locations of dark spots can be obtained by a MAP scheme.

4.2.1 Description of Algorithm for Dark Spot Detection

A. Bayesian Model for Dark Spots Diction

Consider a SAR intensity image with n pixels, $\boldsymbol{Z} = \{Z_i\,;\, i = 1, \ldots, n\}$, containing an unknown but a bounded number k for dark spots corresponding to the candidate of oil spills and k follows Poisson distribution defined in Eq. (3.14). The dark spots are represented by k windows with length l_j, width w_j, direction a_j, and centred at pixel (u_j, v_j) called central points randomly distributed on D with a prior density $p(u_j, v_j)$ as defined in Eq. (3. 18). Let $\boldsymbol{G} = \{(u_j, v_j); j =1, \ldots, k\}$ and $\boldsymbol{\Phi} = \{(l_j, w_j, a_j); j = 1, \ldots, k\}$

be the sets of central points and the geometric parameters of windows, respectively. Then D can be partitioned by the marked point process described in Section 3.2, that is, $D = \{D_0, D_1, \ldots, D_k\}$.

Associated with each pixel i is a class label $L_i \in \{0, 1, \ldots, k\}$. $L_i = 0$ means the pixel (x_i, y_i) is out of any windows $(x_i, y_i) \in D_0$, while $L_i = j \neq 0$ indicates the pixel (x_i, y_i) locates at the j'th window D_j. As a result, the class labels of all pixels form the label field, $\boldsymbol{L} = \{L_i\,; i = 1, \ldots, n\}$. For a give pixel with class label $L_i = j \in \{0, 1, \ldots, k\}$, the pixel intensity is modeled by Gamma distribution (Lee et al., 1994)

$$p(Z_i \mid L_i = j) = p(Z_i \mid \boldsymbol{\theta}_j) = \frac{Z_i^{\alpha_j - 1}}{\beta_j^{\alpha_j} \Gamma(\alpha_j)} \exp\left(-\frac{Z_i}{\beta_j} \right) \tag{4.19}$$

with $\boldsymbol{\theta}_j = (\alpha_j, \beta_j)$ the parameter vector for the class j. Accordingly, the likelihood of the image data given the parameter vector $\boldsymbol{\theta} = \{\boldsymbol{\theta}_j\,; j = 0, 1, \ldots, k\}$ can be written as

$$p(\boldsymbol{Z} \mid \boldsymbol{\theta}) = \prod_{i=1}^{n} p(Z_i \mid \boldsymbol{\theta}_{L_i = j}) = \prod_{i=1}^{n} \frac{Z_i^{\alpha_j - 1}}{\beta_j^{\alpha_j} \Gamma(\alpha_j)} \exp\left(-\frac{Z_i}{\beta_j} \right) \tag{4.20}$$

The dark spots can be identified by the constrain on the distribution parameters, that is,

$$\alpha_0 \beta_0 > \alpha_j \beta_j \ \text{ for all } j = 1, \ldots, k \tag{4.21}$$

The prior distributions of shape and scale parameters α_j and parameter β_j are assumed to be a normal distribution, but with different means and standard deviations for dark spot class and no dark spot class, that is, $\alpha_0 \sim N(\mu_{\alpha 0}, \sigma_{\alpha 0})$, $\beta_0 \sim N(\mu_{\beta 0}, \sigma_{\beta 0})$, $\alpha_j \sim N(\mu_\alpha, \sigma_\alpha)$, $\beta_j \sim N(\mu_\beta, \sigma_\beta)$ where $j \neq 0$. Their probability density function can be expressed as

$$p(\alpha_j) = \frac{1}{\sqrt{2\pi}\sigma_\alpha} \exp\left[-\frac{(\alpha_j - \mu_\alpha)^2}{2\sigma_\alpha{}^2} \right] \tag{4.22}$$

$$p(\beta_j) = \frac{1}{\sqrt{2\pi}\sigma_\beta} \exp\left[-\frac{(\beta_j - \mu_\beta)^2}{2\sigma_\beta{}^2} \right] \tag{4.23}$$

$$p(\alpha_0) = \frac{1}{\sqrt{2\pi}\sigma_{\alpha 0}} \exp\left[-\frac{(\alpha_0 - \mu_{\alpha 0})^2}{2\sigma_{\alpha 0}{}^2} \right] \tag{4.24}$$

$$p(\beta_0) = \frac{1}{\sqrt{2\pi}\sigma_{\beta 0}} \exp\left[-\frac{(\beta_0 - \mu_{\beta 0})^2}{2\sigma_{\beta 0}{}^2} \right] \tag{4.25}$$

Assume that all distribution parameters are independent of each other, the prior distribution of $\boldsymbol{\theta} = \{\alpha_0, \alpha_1, \ldots, \alpha_k, \beta_0, \beta_1, \ldots, \beta_k\}$ can be written as follows.

$$p(\boldsymbol{\theta} \mid k) = \frac{1}{2\pi\sigma_{\alpha 0}\sigma_{\beta 0}} \times \exp\left[-\frac{(\alpha_0 - \mu_{\alpha 0})^2}{2\sigma_{\alpha 0}{}^2}\right] \times \exp\left[-\frac{(\beta_0 - \mu_{\beta 0})^2}{2\sigma_{\beta 0}{}^2}\right] \times$$

$$\prod_{j=1}^{k} \frac{1}{2\pi\sigma_{\alpha}\sigma_{\beta}} \exp\left[-\frac{(\alpha_j - \mu_{\alpha})^2}{2\sigma_{\alpha}{}^2}\right] \exp\left[-\frac{(\beta_j - \mu_{\beta})^2}{2\sigma_{\beta}{}^2}\right] \tag{4.26}$$

The lengths and widths of windows are assumed to be normal distributions, that is, $l_j \sim N(\mu_l,\ \sigma_l^2)$, $w_j \sim N(\mu_w,\ \sigma_w^2)$, while the directions of windows are the uniform distribution on [-π/2, π/2], that is, $a_j \sim U(-\pi/2,\ \pi/2)$. Assume that the geometric parameters are independent. As a result, the prior distribution of $\boldsymbol{\Phi}$ can be written as

$$p(\boldsymbol{\Phi} \mid k) = \prod_{j=1}^{k} \frac{1}{2\pi^2\sigma_l\sigma_w} \exp\left[-\frac{(l_j - \mu_l)^2}{2\sigma_l{}^2}\right] \exp\left[-\frac{(w_j - \mu_w)^2}{2\sigma_w{}^2}\right] \tag{4.27}$$

The posterior distribution in Eq. (3.21) can be rewritten as

$$p(\boldsymbol{\theta}, \boldsymbol{G}, k \mid \boldsymbol{Z}) \propto p(\boldsymbol{Z} \mid \boldsymbol{\theta}) p(\boldsymbol{G} \mid k) p(\boldsymbol{\Phi} \mid k) p(\boldsymbol{\theta} \mid k) p(k)$$

$$= \prod_{i=1}^{n} Z_i^{\alpha_{L_i} - 1} \frac{1}{\beta_{L_i}{}^{\alpha_{L_i}} \Gamma(\alpha_{L_i})} \exp\left(-\frac{Z_i}{\beta_{L_i}}\right) \times \left(\frac{1}{|D|}\right)^k \times \frac{\lambda^k}{k!} \exp(-\lambda) \times$$

$$\frac{1}{2\pi\sigma_{\alpha 0}\sigma_{\beta 0}} \times \exp\left[-\frac{(\alpha_0 - \mu_{\alpha 0})^2}{2\sigma_{\alpha 0}{}^2}\right] \times \exp\left[-\frac{(\beta_0 - \mu_{\beta 0})^2}{2\sigma_{\beta 0}{}^2}\right] \times$$

$$\prod_{j=1}^{k} \frac{1}{2\pi\sigma_{\alpha}\sigma_{\beta}} \exp\left[-\frac{(\alpha_j - \mu_{\alpha})^2}{2\sigma_{\alpha}{}^2}\right] \exp\left[-\frac{(\beta_j - \mu_{\beta})^2}{2\sigma_{\beta}{}^2}\right] \times$$

$$\prod_{j=1}^{k} \frac{1}{2\pi^2\sigma_l\sigma_w} \exp\left[-\frac{(l_j - \mu_l)^2}{2\sigma_l{}^2}\right] \exp\left[-\frac{(w_j - \mu_w)^2}{2\sigma_w{}^2}\right] \tag{4.28}$$

C. Simulation and Optimization

In order to detect dark spots from a SAR image, it is necessary to simulate from the posterior distribution defined in Eq. (4.28) and estimate its parameters. In this thesis, a RJMCMC scheme is employed for the purpose.

The move types designed in this paper include: (1) updating the geometric parameters of windows: the length, width and direction; (2) updating Gamma distribution parameters; (3) updating the positions of central points; and (4) birth or death of central points.

Move 1: *updating the geometric parameters of windows*. The geometric parameters can be rearranged as $\boldsymbol{\Phi} = \{(l_j, w_j, a_j); j = 1, \ldots, k\} = \{\Phi_{j'}; j' = 1, 2, \ldots, 3k\}$. During the t'th iteration the proposal $\Phi_{j'}^{*}$ is drawn from a Gaussian distribution with mean $\Phi_{j'}^{(t-1)}$ and variance ε which is equal to ε_l, ε_w, and ε_a as $\Phi_{j'} = l_j, w_j, a_j$, respectively. That is, $\Phi_{j'}^{*} \sim N(\Phi_{j'}^{(t-1)}, \varepsilon)$. Fig. 4.16 shows the changes of window structures in terms of their length l, width w and direction a, respectively.

The acceptance probability for the proposal $\Phi_{j'}^{*}$ is given by

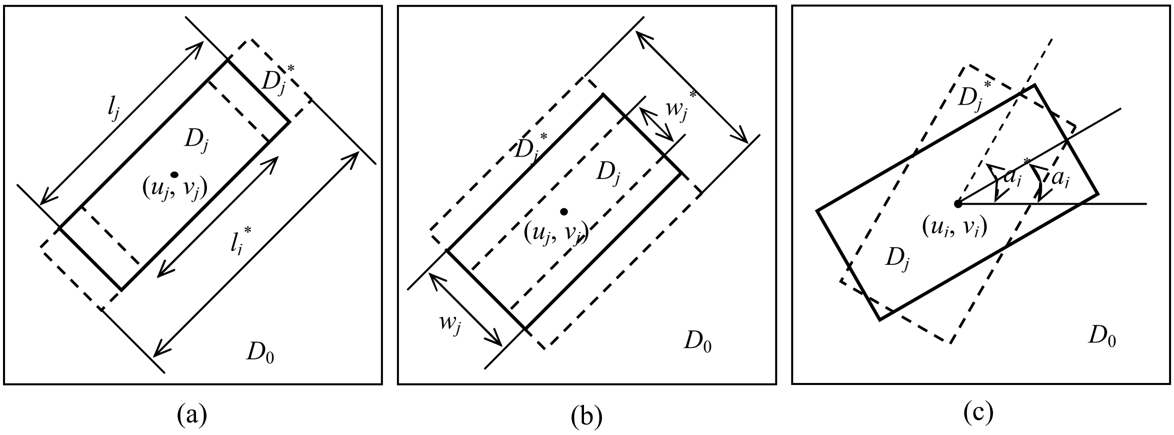


Fig. 4.16 Changes of windows in terms of length l_j, width w_j and direction a_j .

$$r_l =$$

$$\begin{cases} \min\left\{1,\ \dfrac{\prod\limits_{(x_i,y_i)\in D_j^*\setminus D_j} \dfrac{Z_i^{\alpha_j-1}}{\beta_j^{\alpha_j}\Gamma(\alpha_j)}\exp\left(-\dfrac{Z_i}{\beta_j}\right)}{\prod\limits_{(x_i,y_i)\in D_j^*\setminus D_j} \dfrac{Z_i^{\alpha_0-1}}{\beta_0^{\alpha_0}\Gamma(\alpha_0)}\exp\left(-\dfrac{Z_i}{\beta_0}\right)} \times \dfrac{\exp\left[-\dfrac{(l_j^*-\mu_l)^2}{2\sigma_l^2}\right]}{\exp\left[-\dfrac{(l_j-\mu_l)^2}{2\sigma_l^2}\right]}\right\}, & \text{if } \ l_j^* > l_j \\ \min\left\{1,\ \dfrac{\prod\limits_{(x_i,y_i)\in D_j\setminus D_j^*} \dfrac{Z_i^{\alpha_0-1}}{\beta_0^{\alpha_0}\Gamma(\alpha_0)}\exp\left(-\dfrac{Z_i}{\beta_0}\right)}{\prod\limits_{(x_i,y_i)\in D_j^*\setminus D_j} \dfrac{Z_i^{\alpha_j-1}}{\beta_j^{\alpha_j}\Gamma(\alpha_j)}\exp\left(-\dfrac{Z_i}{\beta_j}\right)} \times \dfrac{\exp\left[-\dfrac{(l_j^*-\mu_l)^2}{2\sigma_l^2}\right]}{\exp\left[-\dfrac{(l_j-\mu_l)^2}{2\sigma_l^2}\right]}\right\}, & \text{if } \ l_j^* < l_j \end{cases} \qquad (4.29)$$

$$r_w =$$

$$\begin{cases} \min\left\{1,\ \dfrac{\prod\limits_{(x_i,y_i)\in D_j^*\setminus D_j} \dfrac{Z_i^{\alpha_j-1}}{\beta_j^{\alpha_j}\Gamma(\alpha_j)}\exp\left(-\dfrac{Z_i}{\beta_j}\right)}{\prod\limits_{(x_i,y_i)\in D_j^*\setminus D_j} \dfrac{Z_i^{\alpha_0-1}}{\beta_0^{\alpha_0}\Gamma(\alpha_0)}\exp\left(-\dfrac{Z_i}{\beta_0}\right)} \times \dfrac{\exp\left[-\dfrac{(w_j^*-\mu_w)}{2\sigma_w^2}\right]}{\exp\left[-\dfrac{(w_j-\mu_w)}{2\sigma_w^2}\right]}\right\}, & \text{if } \ w_j^* > w_j \\ \min\left\{1,\ \dfrac{\prod\limits_{(x_i,y_i)\in D_j\setminus D_j^*} \dfrac{Z_i^{\alpha_0-1}}{\beta_0^{\alpha_0}\Gamma(\alpha_0)}\exp\left(-\dfrac{Z_i}{\beta_0}\right)}{\prod\limits_{(x_i,y_i)\in D_j^*\setminus D_j} \dfrac{Z_i^{\alpha_j-1}}{\beta_j^{\alpha_j}\Gamma(\alpha_j)}\exp\left(-\dfrac{Z_i}{\beta_j}\right)} \times \dfrac{\exp\left[-\dfrac{(w_j^*-\mu_w)}{2\sigma_w^2}\right]}{\exp\left[-\dfrac{(w_j-\mu_w)}{2\sigma_w^2}\right]}\right\}, & \text{if } \ w_j^* < w_j \end{cases} \qquad (4.30)$$

$r_a =$

$$\min\left\{1,\ \frac{\prod_{(x_i,y_i)\in D_j{*}\backslash D_j} \frac{Z_i^{\alpha_j-1}}{\beta_j^{\alpha_j}\Gamma(\alpha_j)}\exp\left(-\frac{Z_i}{\beta_j}\right)}{\prod_{(x_i,y_i)\in D_j{*}\backslash D_j} \frac{Z_i^{\alpha_0-1}}{\beta_0^{\alpha_0}\Gamma(\alpha_0)}\exp\left(-\frac{Z_i}{\beta_0}\right)} \times \frac{\prod_{(x_i,y_i)\in D_j\backslash D_j^*} \frac{Z_i^{\alpha_0-1}}{\beta_0^{\alpha_0}\Gamma(\alpha_0)}\exp\left(-\frac{Z_i}{\beta_0}\right)}{\prod_{(x_i,y_i)\in D_j{*}\backslash D_j} \frac{Z_i^{\alpha_j-1}}{\beta_j^{\alpha_j}\Gamma(\alpha_j)}\exp\left(-\frac{Z_i}{\beta_j}\right)}\right\} \tag{4.31}$$

where D_j^* and D_j are the windows constructed by (l_j^*, w_j^*, a_j^*) and (l_j, w_j, a_j), respectively.

Move 2: updating Gamma distribution parameters. The state $\boldsymbol{\theta} = \{\boldsymbol{\theta}_j, j = 0, 1, \dots, k\}$ can be written where $\boldsymbol{\theta}_j = (\alpha_j, \beta_j)$. Assume that the probability distributions for the proposal α_j^* and β_j^* are Gaussian distributions with means α_j and β_j, and standard difference ε_α and ε_β, that is, $\alpha_j^* \sim N(\alpha_j, \varepsilon_\alpha)$ and $\beta_j^* \sim N(\beta_j, \varepsilon_\beta)$. The acceptance probability for the proposal α_j^* and β_j^* can be obtained as

$$r_{\alpha,\beta} = \min\left\{1,\ \frac{\prod_{(x_i,y_i)\in D_j} p(Z_i \mid \boldsymbol{\theta}_j^*) \times p(\boldsymbol{\theta}_j^*)}{\prod_{(x_i,y_j)\in D_j} p(Z_i \mid \boldsymbol{\theta}_j) \times p(\boldsymbol{\theta}_j)}\right\} \tag{4.32}$$

Move 3: moving the position of generating points. One of central points in $\boldsymbol{G} = \{(u_j, v_j), j = 1, \dots, k\}$ is drawn at random, say (u_j, v_j). A proposed position is (u_j^*, v_j^*) by drawing uniformly in the window D_j. The new position of the central point gives rise to the local changes of D_j to D_j^*.

The acceptance probability for the move turns out to be

$$r_c = \min\left\{1,\ \frac{\prod\limits_{(x_i,y_i)\in D_j{}^*\backslash D_j} \frac{\beta_j{}^{-\alpha_j}}{\Gamma(\alpha_j)}\exp\left(-\frac{Z_i}{\beta_j}\right)\times \prod\limits_{(x_i,y_i)\in D_j\backslash D_j{}^*} \frac{\beta_0{}^{-\alpha_0}}{\Gamma(\alpha_0)}\exp\left(-\frac{Z_i}{\beta_0}\right)}{\prod\limits_{(x_i,y_i)\in D_j{}^*\backslash D_j} \frac{\beta_0{}^{-\alpha_0}}{\Gamma(\alpha_0)}\exp\left(-\frac{Z_i}{\beta_0}\right)\times \prod\limits_{(x_i,y_i)\in D_j\backslash D_j{}^*} \frac{\beta_j{}^{-\alpha_j}}{\Gamma(\alpha_j)}\exp\left(-\frac{Z_i}{\beta_j}\right)}\right\} \tag{4.33}$$

Fig. 4.17 shows the example for the change for move 3.

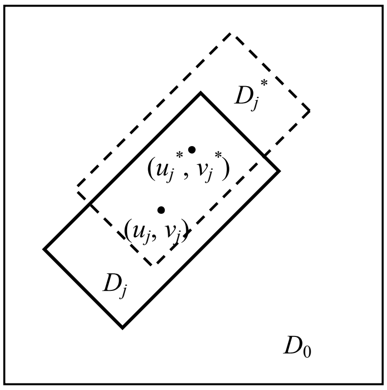


Fig. 4.17 Change of window D_j to D_j^* by moving central point (u_j, v_j) to (u_j^*, v_j^*).

Move 4: *birth or death of generating points*. Suppose that the current number of central points is k and let the probabilities of proposing a birth or death operation be b_k or d_k, respectively. Consider a birth operation which increases the number of central points from k to k+1 and assume that the new central point is labelled with k+1 and its location (u_{k+1}, v_{k+1}) is drawn uniformly from $D\backslash\bigcup_{j=1}^{k} D_j$. Let the window induced by (u_{k+1}, v_{k+1}) be D_{k+1}. The proposal position vector becomes $\boldsymbol{G}^* = \{(u_1, v_1), \ldots, (u_k, v_k),$

(u_{k+1}, v_{k+1})}. As a result, the parameter vector for the birth operation becomes $\boldsymbol{\Theta}^*$ = ($\boldsymbol{\theta}$, $\boldsymbol{G}^*$, k+1). The acceptance probability for the birth can be written as

$$r_b(\boldsymbol{\Theta},\boldsymbol{\Theta}^*) = \min\{1, R_b\} \tag{4.34}$$

where

$$R_b = \frac{p(\boldsymbol{Z} \mid \boldsymbol{\theta},\boldsymbol{G}^*,k+1)p(k+1)p(\boldsymbol{G}^* \mid k+1)r_{b_k}(\boldsymbol{\Theta}^*)}{p(\boldsymbol{Z} \mid \boldsymbol{\theta},\boldsymbol{G},k)p(k)p(\boldsymbol{G} \mid k)r_{d_{k+1}}(\boldsymbol{\Theta})q(\boldsymbol{s})}\left|\frac{\partial(\boldsymbol{\Theta}^*)}{\partial(\boldsymbol{\Theta},\boldsymbol{s})}\right| \tag{4.35}$$

where $r_{b_k} = b_k$, $r_{d_{k+1}} = d_{k+1}/(k+1)$, $\boldsymbol{s}$ = (u_{k+1}, v_{k+1}) and other terms in Eq. (4.35) can be expressed as follows

$$\frac{p(\boldsymbol{Z} \mid \boldsymbol{\theta},\boldsymbol{G}^*,k+1)}{p(\boldsymbol{Z} \mid \boldsymbol{\theta},\boldsymbol{G},k)} = \frac{\displaystyle\prod_{(x_i,y_i)\in D_{k+1}} \frac{\beta_{k+1}{}^{-\alpha_{k+1}}}{\Gamma(\alpha_{k+1})}\exp\left(-\frac{Z_i}{\beta_{k+1}}\right)}{\displaystyle\prod_{(x_i,y_i)\in D_{k+1}} \frac{\beta_0{}^{-\alpha_0}}{\Gamma(\alpha_0)}\exp\left(-\frac{Z_i}{\beta_0}\right)} \tag{4.36}$$

where α_{k+1} and β_{k+1} are drawn from their prior distribution functions, respectively.

$$\frac{p(k+1)}{p(k)} = \frac{\lambda}{k+1} \tag{4.37}$$

$$\left|\frac{\partial(\boldsymbol{\Theta}^*)}{\partial(\boldsymbol{\Theta},\boldsymbol{s})}\right| = 1 \tag{4.38}$$

$$q(\boldsymbol{s}) = \frac{1}{|D_{k+1}|} \tag{4.39}$$

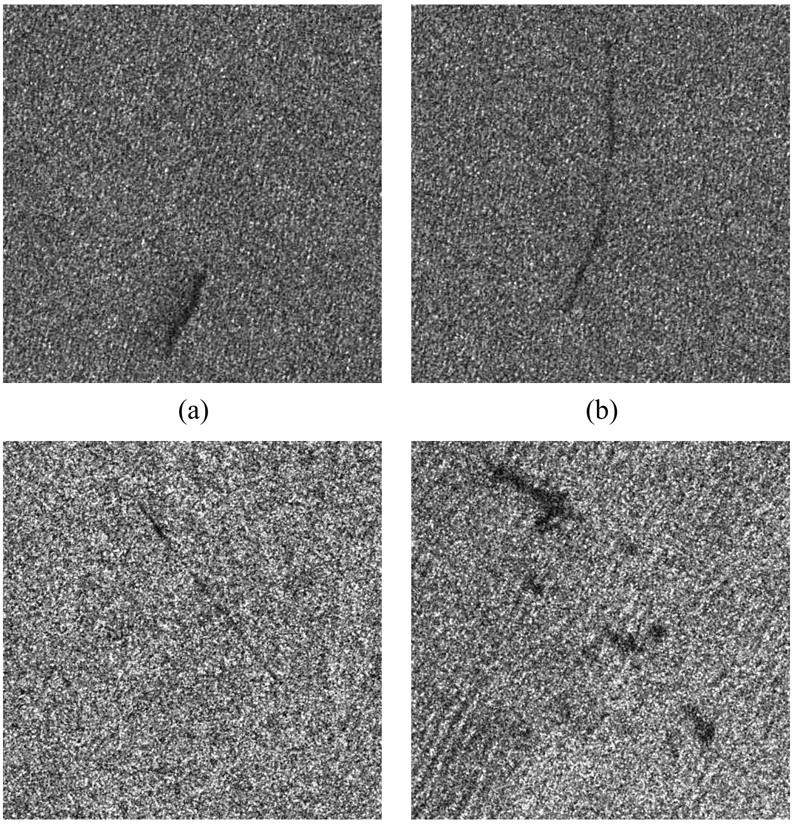
where $|D_{k+1}|$ is the area of the window D_{k+1}. The acceptance probability for a death of central point is

$$r_d(\boldsymbol{\Theta}, \boldsymbol{\Theta}^*) = \min\{1, R_d\}, \text{ and } R_d = {R_b}^{-1} \tag{4.40}$$

For any given proposal with acceptance probability r, it is accepted if and only if $r \geq \xi$, where ξ is drawn from [0, 1] uniformly, that is, $\xi \sim U(0, 1)$. The optimal detection can be obtained by the MAP estimation defined in Eq. (3.29).

4. 2.2 Experimental Results on Dark Spot Detection

The proposed algorithm is applied to SAR intensity images containing oil spills indicated by human analysts from Canada Ice Service (CIS). Fig. 4.18 shows four-look Radarsat-1 ScanSAR intensity images with size 512×512 pixels.



(a) (b)

(c) (d)

Fig. 4.18 Radarsat-1 ScanSAR intensity images.

Table 4.8 lists the constants used in the proposed algorithm for the experiment where λ is the mean of the Poisson distribution for the number of dark spots, which is set as 2 to encourage a low number of central points since dark spots as the candidates of oil spills are fewer. μ_l (μ_w) and σ_l (σ_w) are the mean and standard deviation of length (width) of window. μ_α ($\mu_{\alpha 0}$) and σ_α ($\sigma_{\alpha 0}$) are the mean and standard deviation of the shape parameter of the Gamma distributions defined for the dark spots and their background. The constants $\varepsilon_{l,w,\ a,\alpha,\beta}$ are the proposal variances for l, w, a, α, α_0, β and β_0, respectively. The T_m used in this experiment is uninformative and larger than practically used ones.

Table 4.8 Constants used in Eq. (4.28)

λ	μ_l, μ_w	σ_l, σ_w	μ_α, α_0	σ_α	μ_β	σ_β
3	50	10	4	0.5	16	2
β_0	$\varepsilon_{l,w}$	ε_a	ε_α	ε_β	T_m	
32	2	π/36	0.25	1	10000	

The numbers of initial window are drawn from the Poisson distribution with the mean 3. From our experience, the initial window numbers have no significant impact on the final results. The geometric parameters of the initial window are drawn from their distributions. The initial Gamma distribution parameters are also drawn from their distributions.

Table 4.9 gives the estimated parameters for windows such as the length, width, direction, location of central point and the Gamma distribution parameters.

Table 4.9 Estimated geometric and distribution parameters

image	window	l	w	$a(^\circ)$	u	v	α	β
a	D_1	64.1	27.6	-59.1	235.0	428.4	3.47	12.32
	D_2	59.3	16.5	-67.4	265.7	374.5	3.85	11.90
	D_3	66.3	18.0	-52.2	203.0	430.6	3.79	11.82
	D_4	49.3	6.5	65.7	388.4	132.5	3.86	12.07
b	D_1	157.2	44.9	-74.7	247.6	333.8	3.87	15.33
	D_2	162.4	27.1	-87.7	273.0	124.6	3.43	13.94
c	D_1	90.7	12.0	59.1	202.7	107.1	4.07	11.28
	D_2	129.1	8.0	51.1	302.2	237.6	4.10	13.08
d	D_1	26.9	30.7	10.2	396.4	391.5	4.21	13.62
	D_2	57.9	38.6	-83.4	186.6	96.3	4.42	13.35
	D_3	50.8	22.4	11.7	290.5	273.7	4.16	16.95
	D_4	28.1	24.9	13.9	165.9	193.6	4.00	15.34
	D_5	65.1	20.0	24.5	140.3	54.5	4.14	14.00
	D_6	19.4	55.4	-33.6	339.7	270.9	3.41	16.83

Fig. 4.19 gives the distribution of final windows after 10000 iterations, which correspond to the locations of detected dark spots. The numbers of the detected dark spots are 4, 2, 2 and 6 for testing images shown in Fig. 4.18 (a)-(d), respectively.

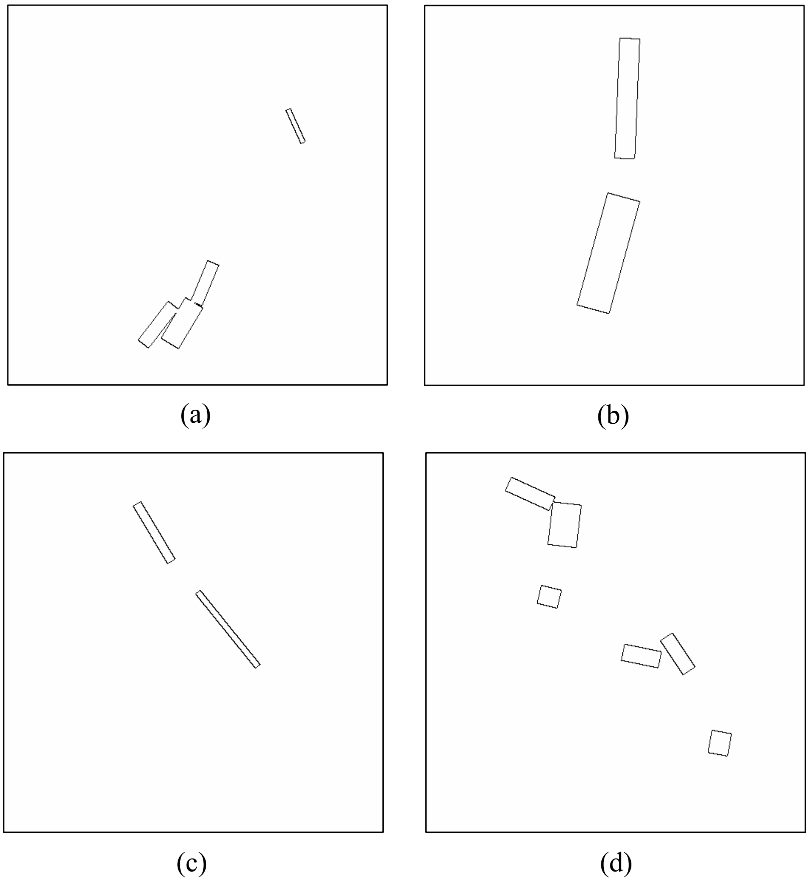

(a) (b) (c) (d)

Fig. 4.19 Outlines of detected windows for the candidates of oil spills.

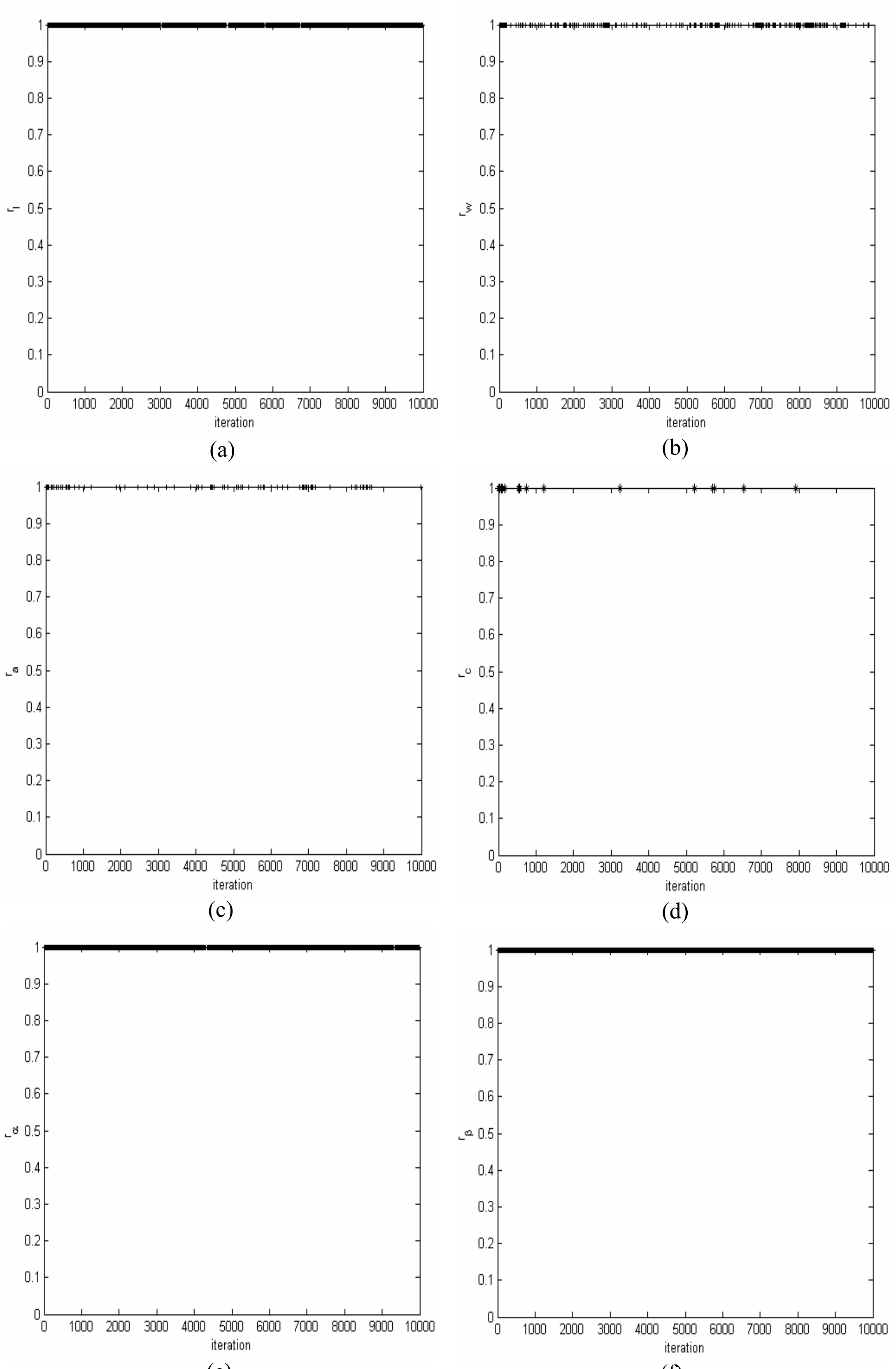
Table 4.10 lists the acceptance rate for each type of moves, where r_l, r_w, r_a, r_c, r_α, r_β, r_b and r_d are the acceptance rates of moves for updating length, width, direction, central point of window, shape parameter, scale parameter, moving central point, birth and death of windows, respectively.

Table 4.10. Acceptance rates of moves 1-4 (%)

image	r_l	r_w	r_a	r_c	r_α	r_β	r_b	r_d
a	9.13	14.87	2.25	29.75	70.47	0.49	0.1	0.02
b	23.43	22.49	4.21	29.81	71.25	1.36	1.91	1.95
c	20.97	2.95	1.13	29.04	84.5	0.23	0.01	0.05
d	16.44	17.68	5.13	28.79	91.84	1.05	0.54	0.61

Fig. 4.20 and Fig. 4.21 give the acceptances of proposals for the SAR image shown in Fig. 4.18 (c). From Fig.4.20-21 and Table 4.10, the operation of birth and death of central points have the minimum acceptance rate and the accepting of the proposed labels is concentrated in the initial phase of the iterations, within the first 6000 iterations, see Fig. 4.20-21, where Fig. 4.21 (c) indicates the change of the number of windows. The phenomenon implies that the proposed algorithm can quickly decide the number of windows which represent the candidates for oil spills. In contrast, the operation of updating the shape parameter α has the maximum acceptance rate, which is caused by the small ε_α. The large acceptance rates for moving central points can be explained by the fact that the algorithm drives many more changes on the location of windows to fit the details of dark spots. The updating length of the windows operates throughout the period of the iterations, though with middle acceptance rates. Since the computational burden for both of the moves is very heavy, it is necessary to find an innovative way to control them, though they have the lowest acceptance rates.

To evaluate the accuracy of the proposed algorithm for the detection of dark spots visually, the detected windows (in red) are overlaid on the SAR images, see Fig. 4.22.



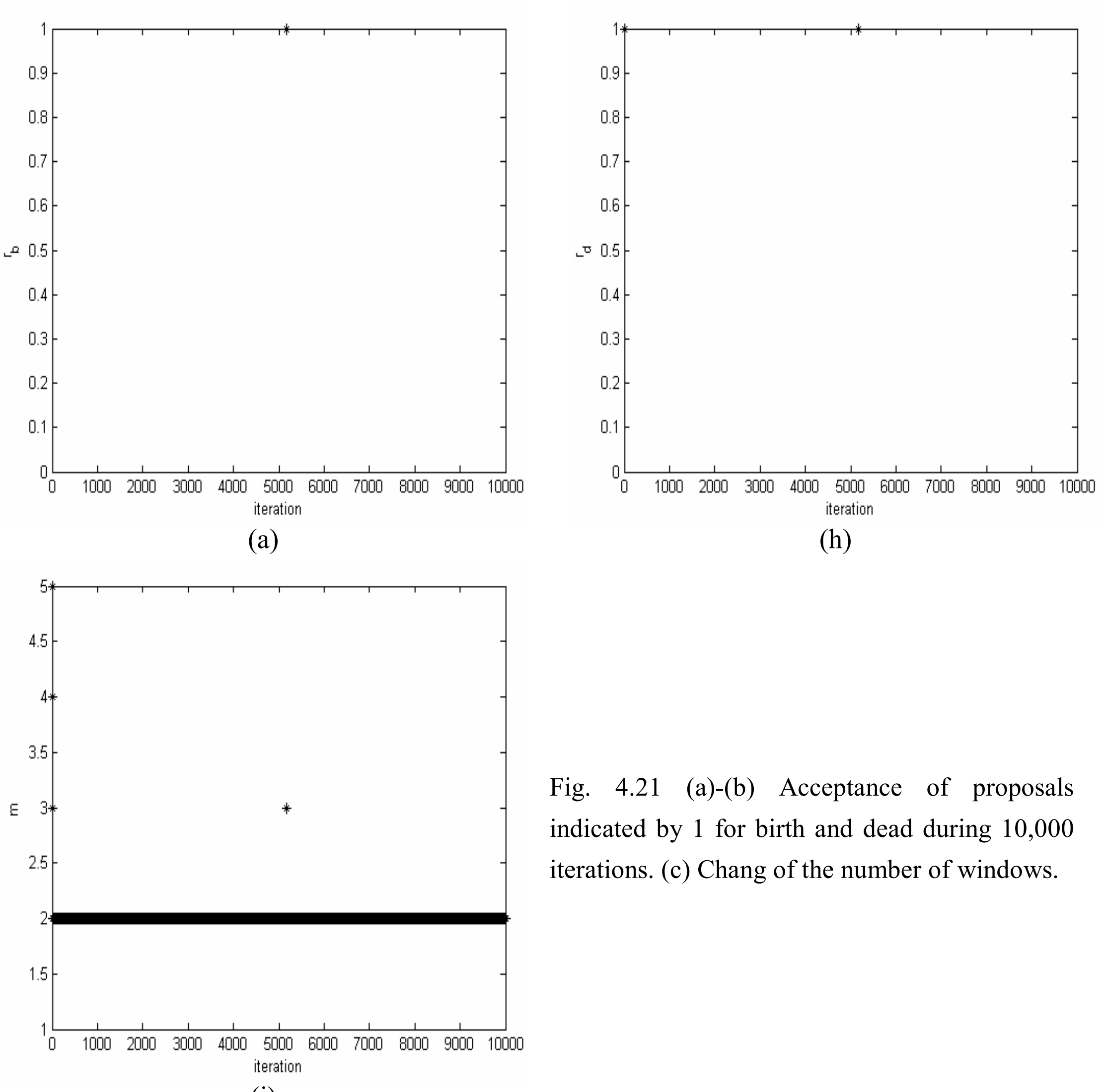
(a) (b)

(c) (d)

(e) (f)

Fig. 4.20 (a)-(g) Acceptance of proposals indicated by 1 for *l*, *w*, *a*, central point, α and β during 10,000 iterations.

From Fig. 4.22, it can be observed that the algorithm can detect the dark spots well.



(a)

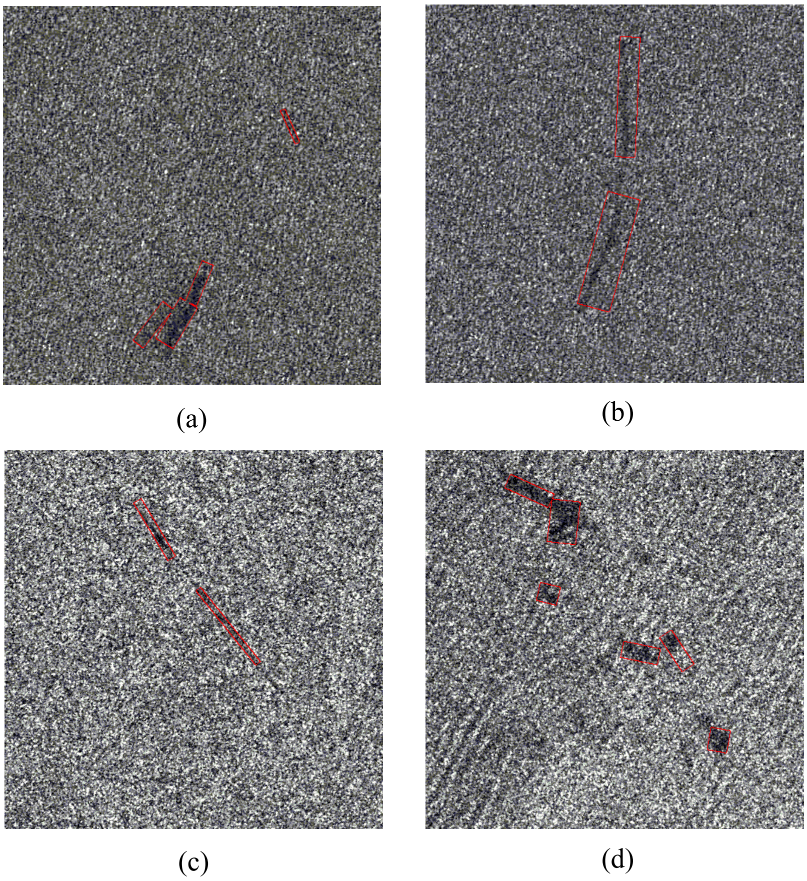
(h)

(i)

Fig. 4.21 (a)-(b) Acceptance of proposals indicated by 1 for birth and dead during 10,000 iterations. (c) Chang of the number of windows.

4.3 Chapter Summary

In this chapter, a new segmentation algorithm developed from the segmentation framework has been presented first. The algorithm has been evaluated based on extensive experiments using both real Radarsat-1/2 images and a simulated SAR image. The experimental results show the efficiency of the proposed segmentation algorithm.



(a) (b)

(c) (d)

Fig. 4.22 Overlaying detected outline for dark spots on the test SAR images.

A new algorithm for the detection of dark spots as the candidates of oil spills based on marked point process is also presented. In the proposed algorithm, the dark spots are modeled as windows. Each window is attached with length, width and direction, and a Gamma distribution with scale and shape parameters. Results from Radarsat-1 ScanSAR intensity imagery show that the proposed algorithm can detect the dark spots very well. Instead of processing the image pixel by pixel for the purpose of dark spot detection, the proposed algorithm processes the pixels in and out of windows. Therefore, it is suitable for searching for oil spills in a large sea area. The proposed algorithm is a statistical region-based algorithm that can reduce the effect of speckle noise on the detection of dark spots.

Chapter 5

LIDAR DATA PROCESSING

This chapter presents two LiDAR point cloud processing techniques: region-based segmentation and building detection using a marked point process. Section 5.1 describes the developed segmentation algorithm based on the segmentation framework proposed in Chapter 3. Section 5.1 shows the building detection from LiDAR cloud point data by taking advantage of a marked point process.

5.1 LiDAR Point Cloud Segmentation

5.1.1 Description of Algorithm for LiDAR Point Cloud Segmentation

A 3D LiDAR point cloud can be expressed as a collection of spatial points, $\boldsymbol{z} = \{(x_i, y_i, z_i); i = 1, \ldots, n\}$, where i is the index of data point (x_i, y_i, z_i); $(x_i, y_i) \in D \subset R^2$ called the ground point is the geo-referenced coordination of the data point i on the earth surface, which is usually distributed on D irregularly; z_i is one of the elevations (returns) acquired at the ground point (x_i, y_i); D is the data domain in which the elevations are acquired by LiDAR sensor, and n is the total number of data points in the point cloud. Fig. 5.1 shows the diagram for the distribution of ground points on a domain D.

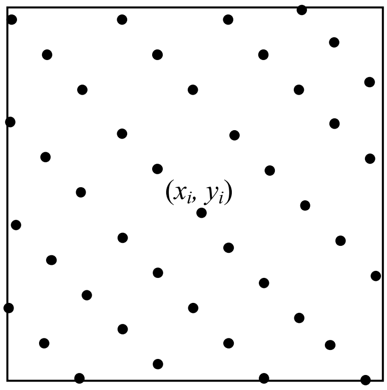


Fig. 5.1 Diagram of ground points distributed on the domain D.

On the other hand, the 3D point cloud can also be rewritten as $\mathbf{z} = \{z_i = Z(x_i, y_i); i = 1, \ldots, n\}$ where the elevation z_i is considered as an observation of the random variable $Z(x_i, y_i)$ defined on the ground point (x_i, y_i). As a consequence, the collection of observations from n ground points that makes up the 3D point cloud represents an incomplete single sampling from the random field, $\mathbf{Z} = \{Z_i = Z(x_i, y_i); i = 1, \ldots, n\}$.

Given a point cloud $\mathbf{Z} = \{Z_i = Z(x_i, y_i); i = 1, \ldots, n\}$ with k known homogeneous regions, it's domain D is partitioned into m sub-regions by a Voronoi tessellation defined in Eq. (3.5).

Under the partition, the elevation Z_i for the ground point $(x_i, y_i) \in D_j$ follows the Gaussian distribution conditional on the label L_j

$$p(Z_i \mid L_j = l) = \frac{1}{\sqrt{2\pi}\sigma_l} \exp\left\{ -\frac{(Z_i - \mu_l)^2}{2\sigma_l^{\ 2}} \right\} \tag{5.1}$$

where μ_l and σ_l are the mean and standard deviation of the Gaussian distribution for the l'th homogenous region. The elevations for the ground points in D_j, $\boldsymbol{Z}_j = \{Z_i; (x_i, y_i) \in P_j\}$, can be modeled by a joint probability distribution of identical independent Gaussian distributions conditional on the label of the polygon, that is,

$$p(\boldsymbol{Z}_j \mid L_j = l) = \prod_{(x_i, y_i) \in D_j} p(Z_i \mid L_j = l) \tag{5.2}$$

On the assumption that the joint distributions for all polygons are independent, then the likelihood can be written as

$$P(\boldsymbol{Z} \mid \boldsymbol{L}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{G}) = \prod_{j=1}^{m} P(\boldsymbol{Z}_j \mid L_j = l) \tag{5.3}$$

where $\boldsymbol{\mu} = (\mu_l\,;\ l = 1, \ldots, k)$ and $\boldsymbol{\sigma} = (\sigma_l\,;\ l = 1, \ldots, k)$ are the mean and variance parameter vectors.

Assume the prior distributions of μ's are an independent Gaussian distributions μ

~ $N(\mu_\mu, \sigma_\mu)$, its PDF is

$$p(\mu_l) = \frac{1}{\sqrt{2\pi}\sigma_\mu} \exp\left\{ -\frac{(\mu_l - \mu_\mu)^2}{2\sigma_\mu{}^2} \right\} \tag{5.4}$$

while the prior distribution of σ's are independent Gamma distributions $\sigma \sim \Gamma(\alpha_\sigma, \beta_\sigma)$, that is,

$$p(\sigma_l) = \frac{\sigma_l{}^{\alpha_\sigma - 1}}{\Gamma(\alpha_\sigma)\beta_\sigma{}^{\alpha_\sigma}} \exp\left\{ -\frac{\sigma_l}{\beta_\sigma} \right\} \tag{5.5}$$

where $\Gamma(\cdot)$ is the Gamma function. Thus, the join prior distributions for $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ can be written as

$$p(\boldsymbol{\mu}) = \prod_{l=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_\mu} \exp\left\{ -\frac{(\mu_l - \mu_\mu)^2}{2\sigma_\mu{}^2} \right\} \tag{5.6}$$

$$p(\boldsymbol{\sigma}) = \prod_{l=1}^{k} \frac{\sigma_l{}^{\alpha_\sigma - 1}}{\Gamma(\alpha_\sigma)\beta_\sigma{}^{\alpha_\sigma}} \exp\left\{ -\frac{\sigma_l}{\beta_\sigma} \right\} \tag{5.7}$$

The posterior distribution of $\boldsymbol{L}$, $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$ and $\boldsymbol{G}$ given $\boldsymbol{Z}$ can be written as

$$p(\boldsymbol{L},\boldsymbol{\mu},\boldsymbol{\sigma},\boldsymbol{G}\mid \boldsymbol{Z}) \propto p(\boldsymbol{Z}\mid \boldsymbol{L},\boldsymbol{\mu},\boldsymbol{\sigma},\boldsymbol{G})p(\boldsymbol{\mu})p(\boldsymbol{\sigma})p(\boldsymbol{L})p(\boldsymbol{G}) =$$

$$\prod_{j=1}^{m}\prod_{(x_i,y_i)\in D_j}\frac{1}{\sqrt{2\pi}\sigma_{L_j}}\exp\left\{-\frac{(Z_i-\mu_{L_j})^2}{2}\right\}\times\prod_{l=1}^{k}\frac{1}{\sqrt{2\pi}\sigma_{\mu}}\exp\left\{-\frac{(\mu_l-\mu_{\mu})^2}{2\sigma_{\mu}^{2}}\right\}\times \tag{5.8}$$

$$\prod_{l=1}^{k}\frac{\sigma_l^{\alpha_\sigma-1}}{\Gamma(\alpha_\sigma)\beta_\sigma^{\alpha_\sigma}}\exp\left\{-\frac{\sigma_l}{\beta_\sigma}\right\}\times\prod_{j=1}^{m}\frac{\exp\left(c\sum_{D_{j'}\in NP_j}t(L_j,L_{j'})\right)}{\sum_{l=1}^{k}\exp\left(c\sum_{D_{j'}\in NP_j}t(l,L_{j'})\right)}\times\left(\frac{1}{|D|}\right)^{m}$$

The Metropolis-Hastings algorithm described in Section 3.6 is used to simulate the posterior distribution in Eq. (5.8). The moves designed in the algorithm include (1) updating distribution parameters in $\boldsymbol{\psi} = \{\boldsymbol{u}, \boldsymbol{\sigma}\} = \{\mu_1, \ldots, \mu_k, \sigma_1, \ldots, \sigma_k\} = \{\Psi_1, \ldots, \Psi_{2k}\}$; (2) updating labels in $\boldsymbol{L} = \{L_j;\ j = 1, \ldots, m\}$; (3) updating the location of generating points in $\boldsymbol{G} = \{(u_j, v_j);\ j = 1, \ldots, m\ \}$. The procedure of the scheme is described as follows.

Move 1: updating the distribution parameters. Let the proposal $\Psi_l^* \sim N\ (\Psi_{l'}, \varepsilon_\Psi)$ where $\varepsilon_\Psi = \varepsilon_\mu$ or ε_σ dependent on the type of Ψ. The acceptance probability for the proposal Ψ_l^* (μ_l^* or σ_l^*) can be calculated by its marginal distribution

$$r_\Psi(\psi_l,\psi_l^*) = \min\left\{1, \frac{p(\boldsymbol{Z}\mid\psi_l^*)p(\psi_l^*)}{p(\boldsymbol{Z}\mid\psi_l)p(\psi_l)}\right\} \tag{5.9}$$

Move 2: updating the label. Uniformly draw a proposal L_j^* from $\{1, \ldots, k\}$ conditional on $L_j^* \neq L_j$. The acceptance probability for the proposal is determined by its marginal distribution

$$r_L(L_j, L_j^*) = \min\left\{1, \frac{p(\boldsymbol{Z} \mid L_j^*)p(L_j^*)}{p(\boldsymbol{Z} \mid L_j)p(L_j^{(t-1)})}\right\} \tag{5.10}$$

Move 3: updating the location of the generating point. Draw the proposal for the generating point $G_j = (u_j, v_j) \in D_j$, that is, $G_j^* = (u_j^*, v_j^*) \sim U(D_j)$, where U denotes a uniform distribution. The acceptance probability for the proposal is as follows

$$r_G(G_j, G_j^*) = \min\left\{1, \frac{p(\boldsymbol{Z} \mid G_j^*)p(G_j^*)}{p(\boldsymbol{Z} \mid G_j)p(G_j)}\right\} \tag{5.11}$$

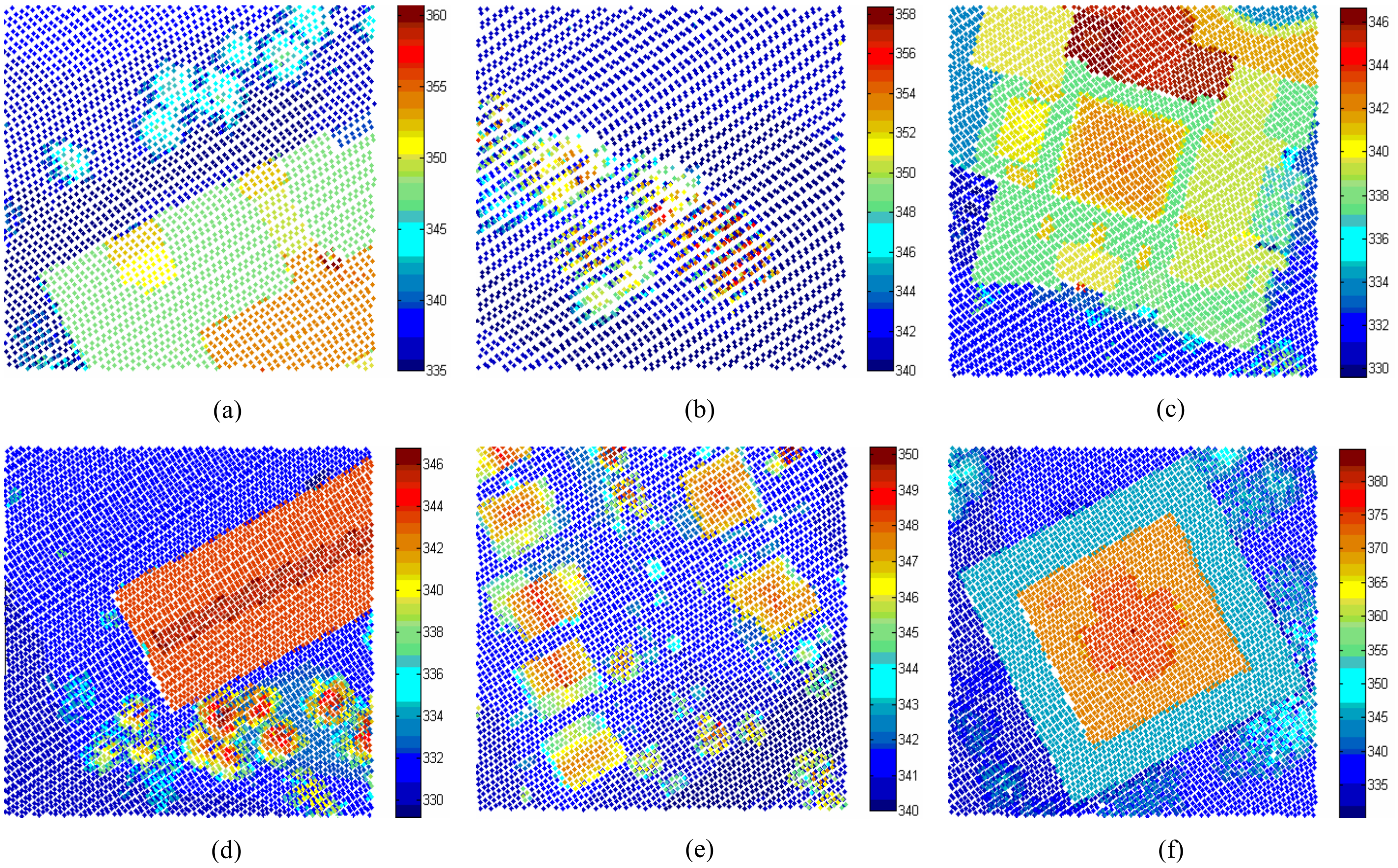
The optimal segmentation can be obtained by MAP defined in Eq. (3.29).

5.1.2 Experimental Results on LiDAR Point Cloud Segmentation

The proposed segmentation algorithm is tested with the LiDAR point cloud data acquired by an Optech ALTM GEMINI system with an Applanix-POS/AV subsystem on March 11, 2006 over the main campus of the University of Waterloo (UW). The dataset consists of 7.9 million points. The flying height was 1,200 *m* above the mean sea level. The speed of the airplane was 66.9 *m*/*sec*. The scan frequency and swath were 35 Hz and 873.53 *m* with the scan resolution and point density of 0.908 *m* and 1.1 point per *sq. m*, respectively. In addition, the simulated point cloud data are also used to evaluate the proposed algorithm quantitatively.

A. ALTM Point Cloud Data

Fig. 5.2 presents the six LiDAR point cloud dataset with 3,382, 3,517, 6,246, 7,094, 5,037 and 7,201 data points, respectively, in which the colour bars indicate the elevations acquired at corresponding ground points.



(a) (b) (c)

(d) (e) (f)

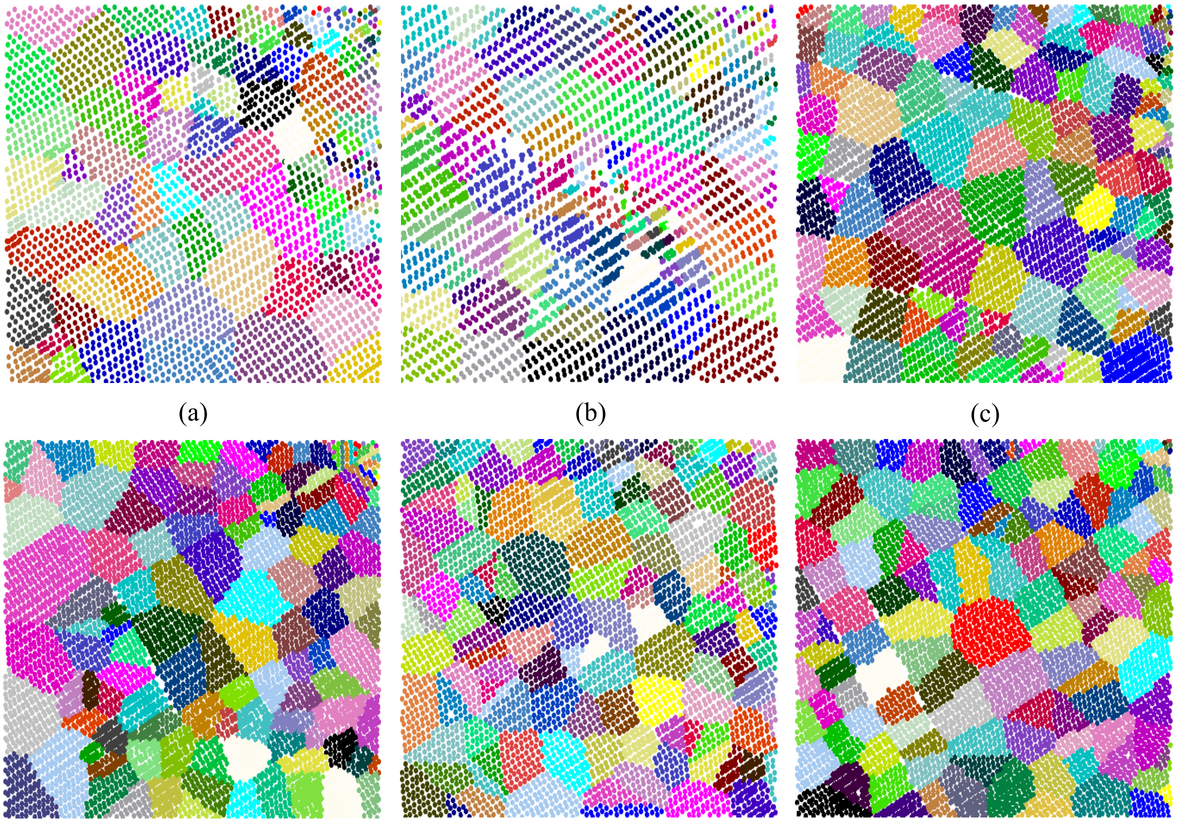
Fig. 5.2 LiDAR point cloud data, elevations are demonstrated by colours.

Table 5.1 lists the constants used in this experiment, where μ_μ takes the mean of the elevations in each point cloud dataset. σ_μ takes one quarter of the maximum difference of the elevations. α_σ and β_σ are selected from prior knowledge on the scenes covered by the test datasets. T_{max} is the maximum iterations. k is the number of homogenous regions needed to segment determined by a human operator. ε_μ and ε_σ are the proposal variances for μ and σ, respectively.

Table 5.1 Constants defined in Eq. (5.8)

Dataset	μ_μ	σ_μ	k	α_σ	β_σ	ε_μ	ε_σ	c	m	T_{max}
a	347	6.0	2	2	1	1	0.25	1	128	20000
b	349	4.5	2							
c	337	7.0	2							
d	337	7.0	3							
e	345	2.5	3							
f	357	12.5	4							

Fig. 5.3 shows the partitions corresponding to the optimal segmentation after

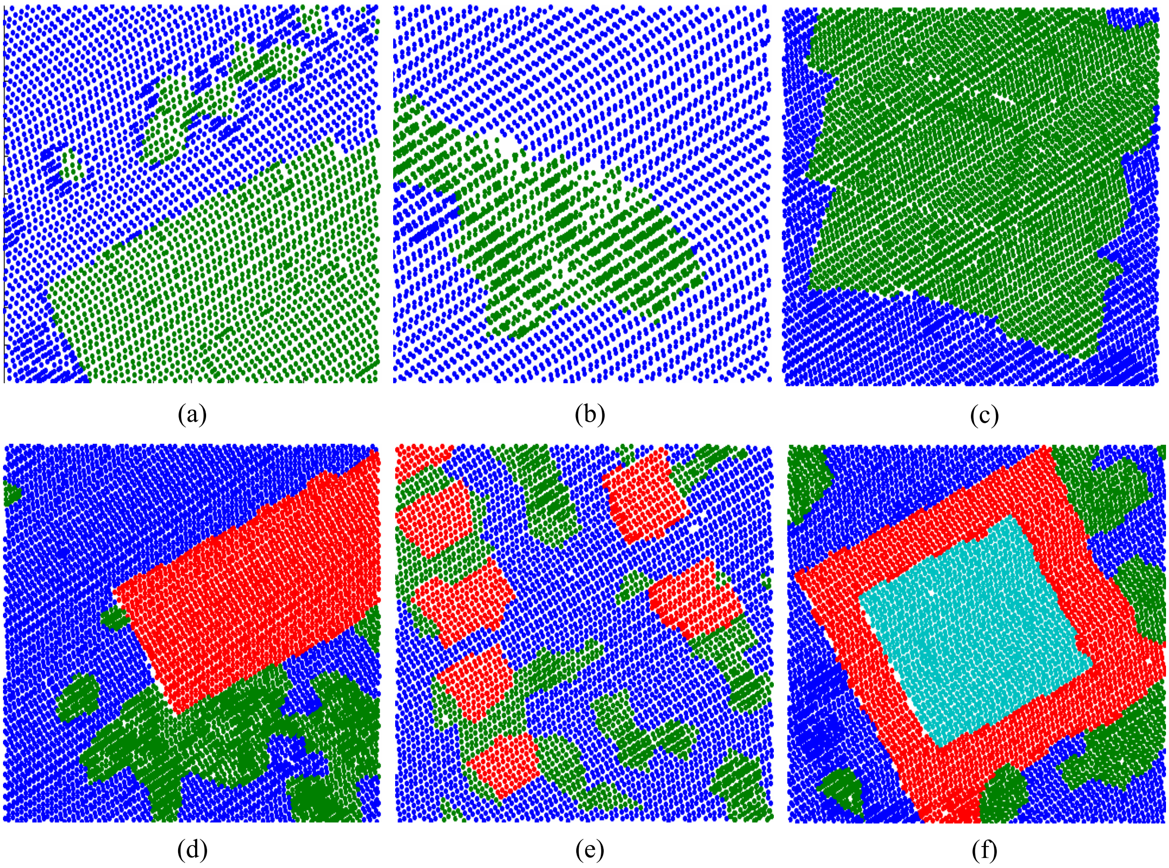

(a) (b) (c)

(d) (e) (f)

Fig. 5.3 Optimal partitions after 20,000 iterations.

20,000 iterations.

Fig.5.4 shows the optimal segmentations obtained at the 19,989th, 19,994th, 19,984th, 19,978th, 19,991th and 19,978th iteration out of 20,000 iterations, in which the segmented regions are presented in blue, green, red, and cyan with respect to the estimated means in an increasing order.


(a) (b) (c)

(d) (e) (f)

Fig. 5.4 Optimal segmentations after 20,000 iterations.

Figs. 5.5 and 5.6 show the accepted means and standard deviations for each homogenous region during 20,000 iterations, in which the colours of curves are coincided with ones presenting the segmented regions in Fig. 5.4 and C1-C4 indicate

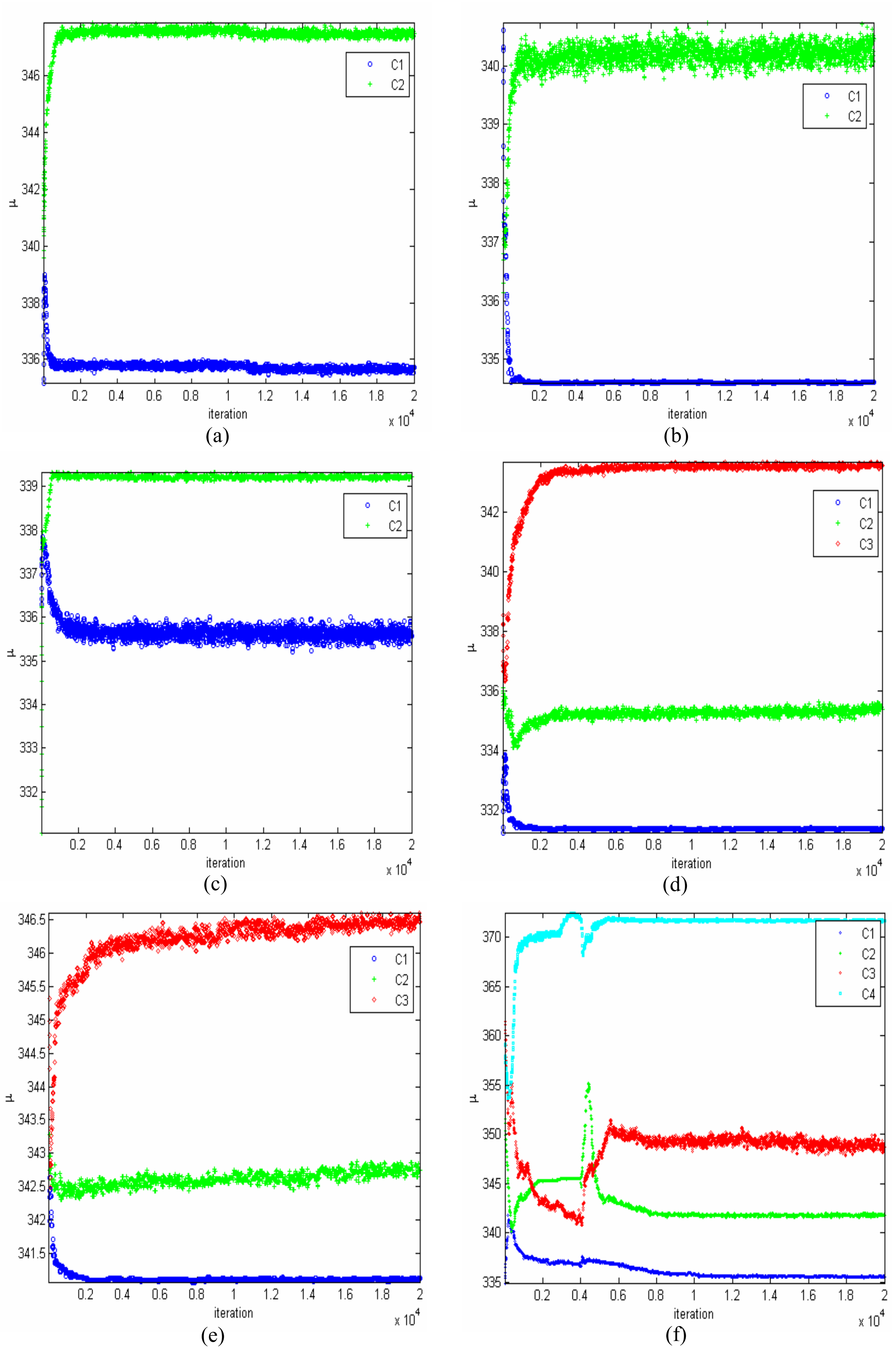


Fig. 5.5 Curves of estimated means during 20,000 iterations.

the segmented regions in terms of the estimated means from low to high. It can be

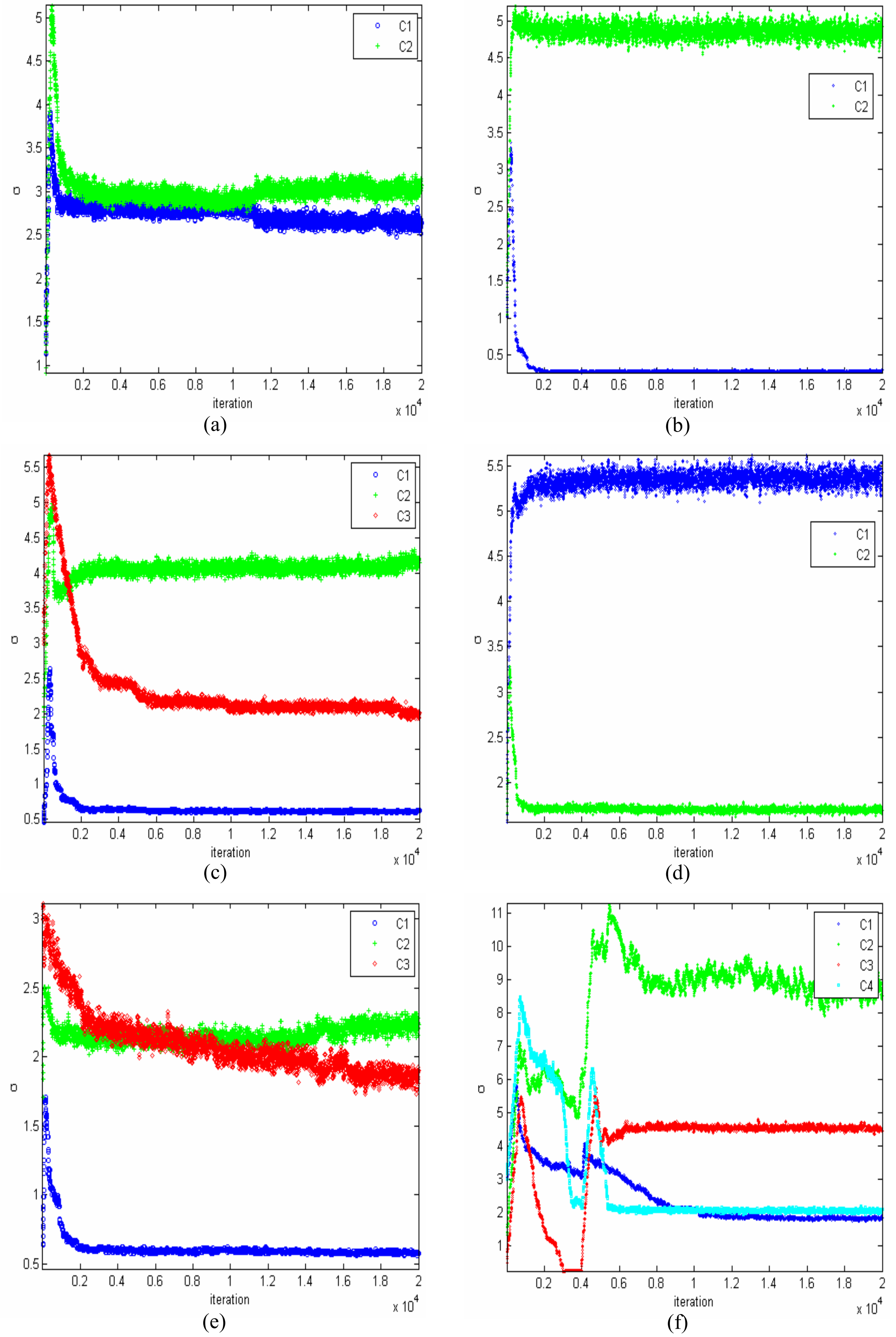


Fig. 5.6 Curves of estimated variances during 20,000 iterations.

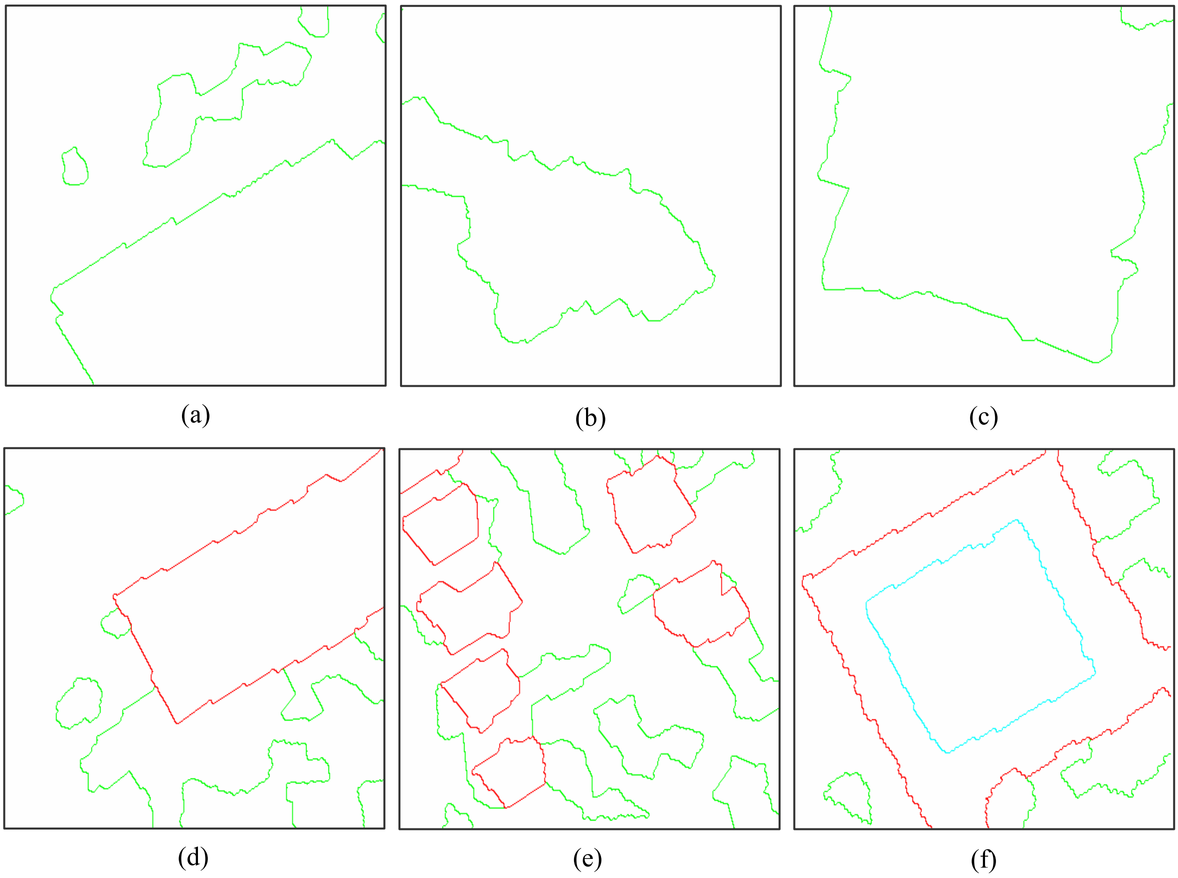
seen from these curves that both mean and variance can quickly converge their stable values.

Table 5.2 presents the estimated means and variances corresponding to the optimal segmentation while indicating the iteration at which the estimated values are obtained.

Table 5.2 Estimated means and standard deviations, iterations for optimal segmentation

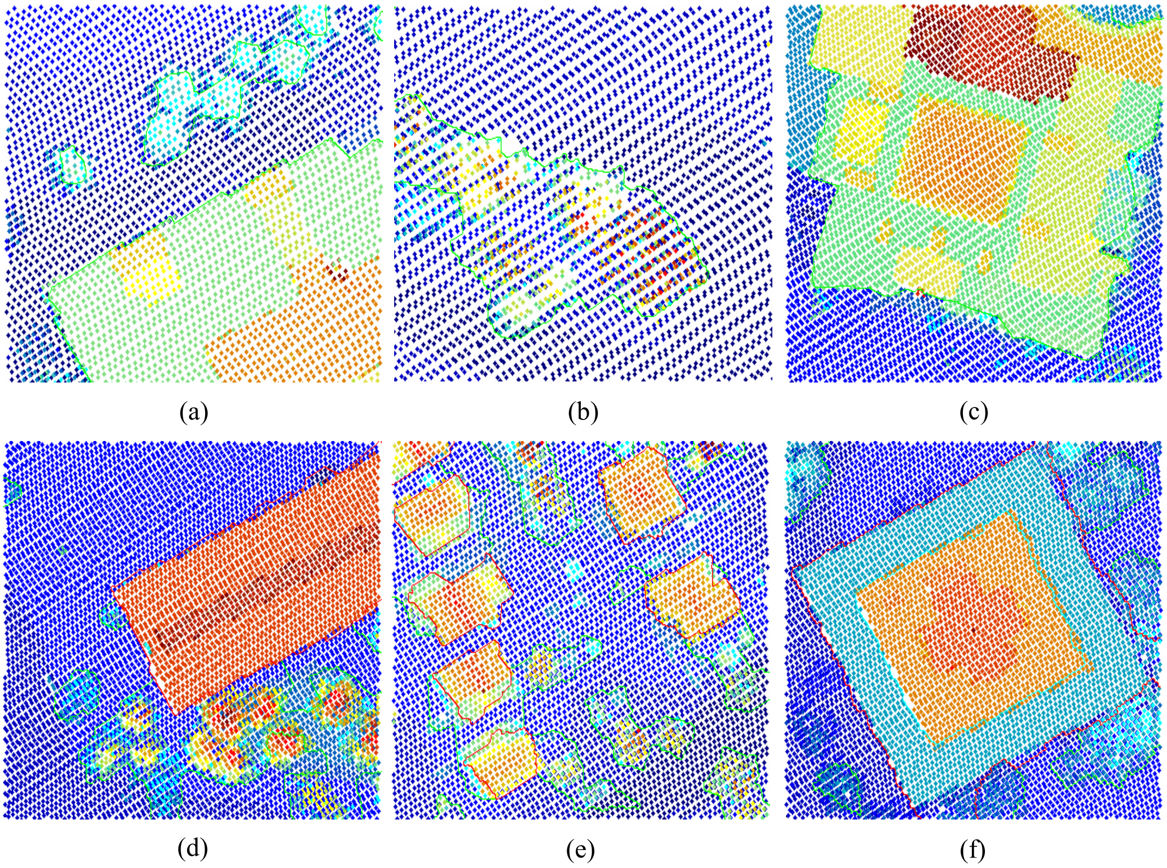
Dataset	Region	μ	σ	*ite*
a	C1	335.61	2.67	19,989
	C2	347.43	3.03	
b	C1	334.61	0.30	19,994
	C2	340.47	4.87	
c	C1	335.61	5.43	19,984
	C2	339.23	1.66	
d	C1	331.38	0.63	19,978
	C2	335.47	4.19	
	C3	343.54	1.94	
e	C1	341.12	0.58	19,991
	C2	342.75	2.23	
	C3	346.46	1.57	
f	C1	335.53	1.81	19,978
	C2	341.79	8.48	
	C3	348.48	4.50	
	C4	371.70	2.07	

In order to visually evaluate the accuracy of segmented regions, Fig. 5.7 gives the outlines of the segmented regions and Fig. 5.8 presents the outlines overlaid on the diagrams of LiDAR point cloud datasets. By visual inspection, the segmented regions match the objects very well.


(a) (b) (c)

(d) (e) (f)

Fig. 5.7 Outlines of the segmented regions.

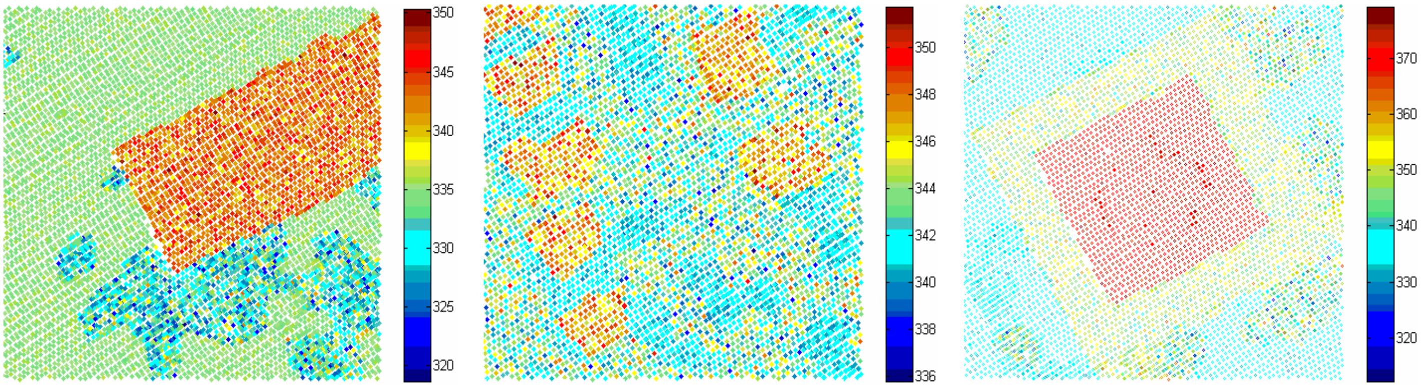

(a) (b) (c)

(d) (e) (f)

Fig.5.8 Outlines of segmented regions overlaid on the diagrams of datasets.

B. Simulated Point Cloud Data

1) Generating Simulated Data. To assess the accuracy of the segmentation algorithm qualitatively and quantitatively, three datasets, which simulate 3D LiDAR point clouds, are synthesized. Fig. 5.9 shows the datasets. Based on the results from the segmentation of LiDAR point cloud data shown in Fig. 5.4 (d)-(e), the simulated datasets are generated by drawing the elevation values for the ground points in each segmented homogeneous regions from the Gaussian distribution with fixed means and variances as listed in Table 5.3.



(a) (b) (c)

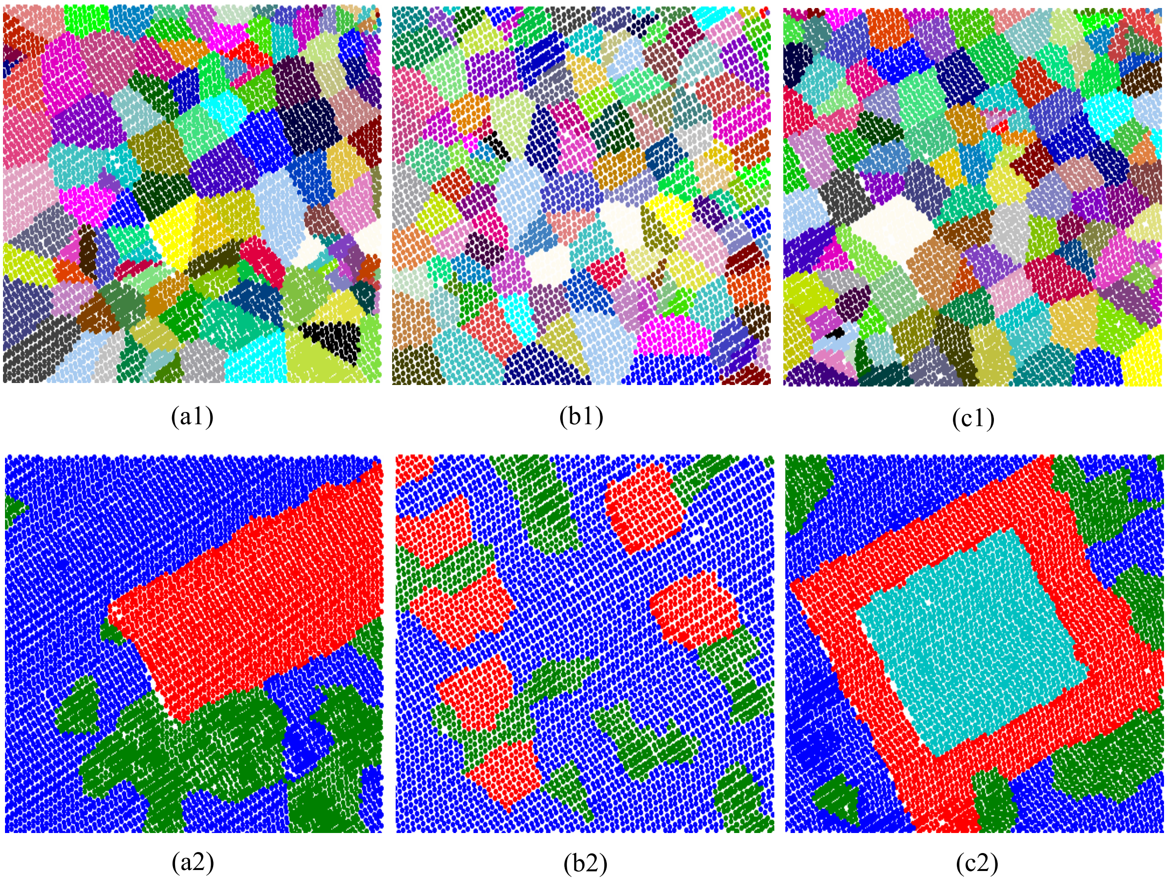
Fig. 5.9 Simulated LiDAR point clouds, elevations demonstrated by colours.

Table 5.3 Parameters for synthetic data.

Dataset	μ_1	μ_2	μ_3	μ_4	σ_1	σ_2	σ_3	σ_4
a	331	335	343	-	4	0.5	2	-
b	341	343	347	-	0.6	2.2	1.6	-
c	336	342	348	372	1.8	8.5	4.5	2.1

2) Testing Simulated Data. The proposed algorithm is tested on the synthetic datasets. The constants for the experiment are the same as these listed in Table 5.1, except for the number of iterations. In the experiment, the numbers are 8,000 for all simulated datasets (a)-(c).

Fig. 5.10 shows the final results after running 8,000 iterations for the simulated datasets (a)-(c). Fig. 5.10 (a1)-(c1) give the final partitions and Fig. 5.10 (a2)-(c2) show the optimal segmentations in which the segmented homogeneous regions are indicated in cyan, red, green and blue in decreased order of their means.

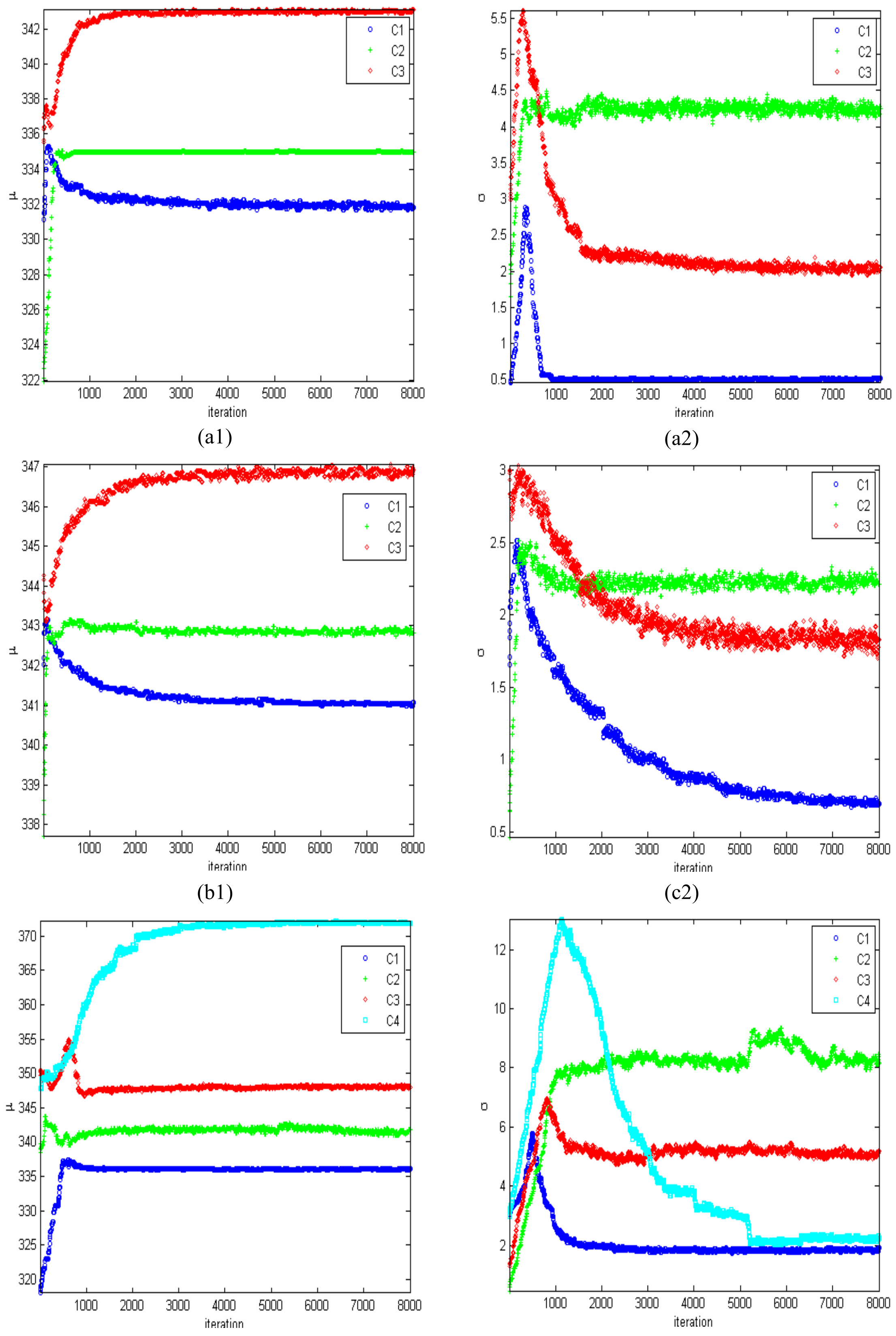


(a1) (b1) (c1)

(a2) (b2) (c2)

Fig.5.10 Optimal partition (a1) - (c1) and optimal segmentation (a2) - (c2) of simulated datasets.

Fig. 5.11 shows the changes of estimated parameters during 8,000 iterations. The Fig. 5.11 also demonstrates that the estimated parameters converge their stable values.



(a1)

(a2)

(b1)

(c2)

(b1)

(c2)

Fig.5.11 Curves of estimated means (a1)-(c1) and standard deviations (a2)-(c2) during 8,000 iterations.

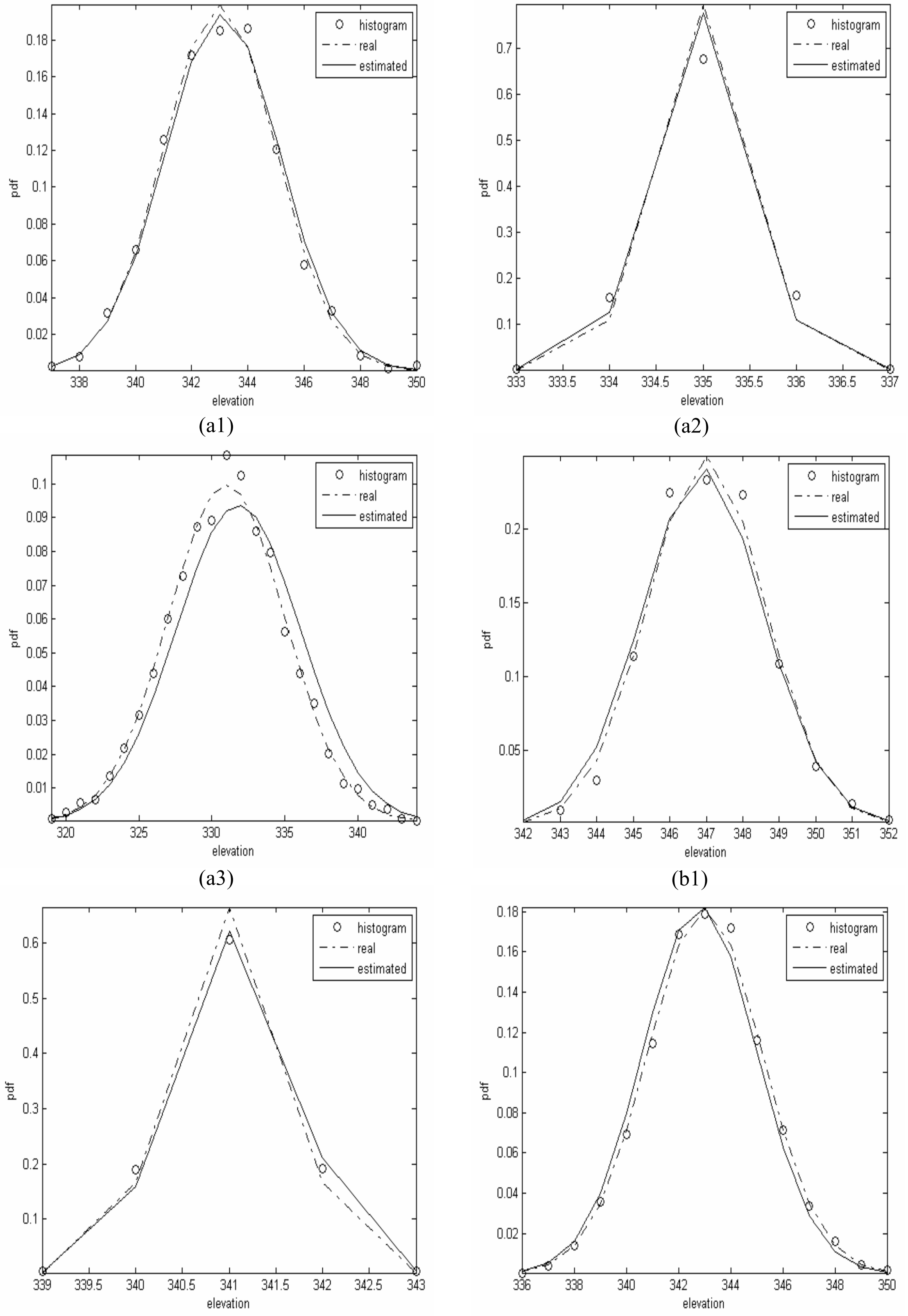
Table 5.4 gives the estimated values of the model parameters μ and σ, the percentage error of the estimated scale parameters (*e*, %), and the number of iterations (*ite*) by which the optimal segmentations are obtained. It is obvious that the estimated values of the model parameters are close to their real values by which the datasets in Fig. 5.9 (a)-(c) are synthesized. The values listed in Table 5.4 demonstrate that the proposed algorithm can estimate the model parameters with high accuracy.

Table 5.4 Estimates model parameters, their errors and iterations for the estimated parameters.

Dataset	μ_1/ e(%)	μ_2/ e(%)	μ_3/ e(%)	μ_4/ e(%)	σ_1/ e(%)	σ_2/ e(%)	σ_3/ e(%)	σ_4/ e(%)	*ite*
a	331.80/ 0.24	334.98/ 0.06	343.09/ 0.03	-	0.51/ 6.43	4.26/ 2.52	2.05/ 2.69	-	7898
b	341.06/ 0.17	242.81/ 0.06	346.91/ 0.03	-	0.64/ 6.67	2.18/ 0.91	1.66/ 7.78	-	7945
c	336.04/ 0.04	341.87/ 0.01	347.89/ 0.03	371.89/ 0.03	1.92/ 6.59	8.17/ 3.90	5.19/ 15.30	2.25/ 7.18	7964

Fig. 5.12 shows the histograms, Gaussian distributions with real and estimated parameters for the synthetic images shown in Fig. 5.9 (a)-(b). Fig. 5.13 shows the histograms, Gaussian distributions with real and estimated parameters for the synthetic images shown in Fig. 5.9 (c). It can be seen that the distributions of the elevations in extracted homogeneous regions match those in the real homogeneous regions.

3) Evaluation of Algorithm. In this experiment, the statistical measurement based assessment scheme is carried out for the purpose of quantitative evaluation. To this end, some of the common measurements are used for accuracy assessment, including



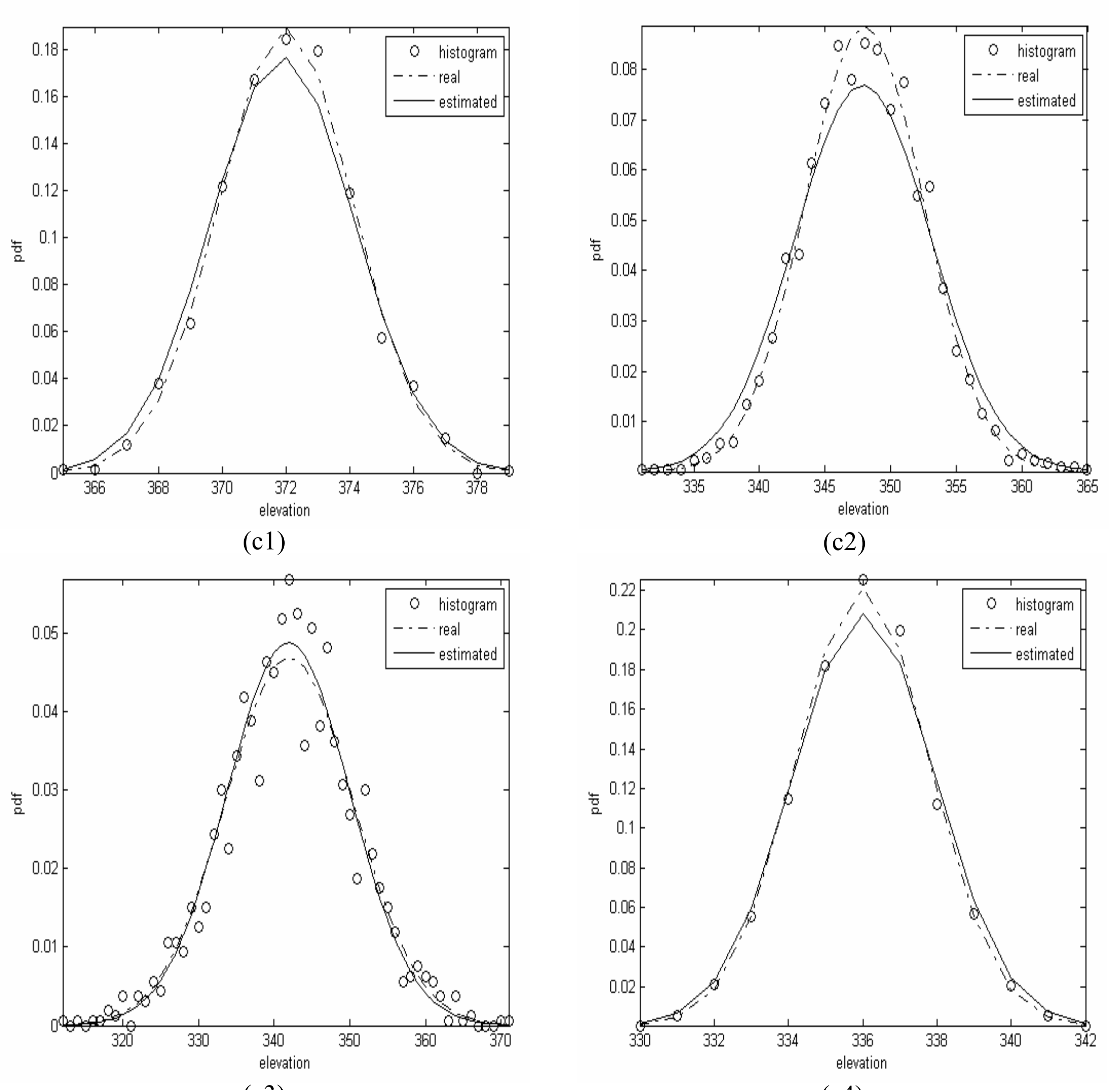
(a1) (a2)

(a3) (b1)

(b2) (b3)

Fig.5.12 Histograms, Gaussian distributions with real parameter and estimated parameters for synthetic data (a) and (b).

error matrix, producer’s, consumer’s, overall accuracies and Kappa coefficient.



(c1) (c2)

(c3) (c4)

Fig.5.13 Histograms, Gaussian distributions with real parameter and estimated parameters for synthetic dataset (c).

Table 5.5 gives the error matrixes

Table 5.5 Error matrix for the segmented results in Fig. 5.10.

Data	Label	C1	C2	C3	C4	$\sum C_{s.}$
a	C1	3058	168	22	-	3248
	C2	172	1933	13		2118
	C3	2	16	1710		1728
	$\sum C_{.r}$	3232	2117	1745		7094
b	C1	2439	66	21	-	2521
	C2	353	1194	16		1568

	C3	75	1	872		948
	$\sum C_{,r}$	2867	1261	909		5037
c	C1	1991	90	15	0	2096
	C2	80	1463	55	0	1598
	C3	13	31	2125	2	2171
	C4	0	0	12	1324	1336
	$\sum C_{,r}$	2084	1584	2207	1326	7201

Based on the error matrix in Table 5.5, the producer's accuracy, user's accuracy, overall accuracy and Kappa coefficient can be calculated. Table 5.6 gives the values of those measurements by taking the synthetic dataset (a) shown in Fig. 5.9 (a) as an example.

Table 5.6 Producer's accuracy, consumer's accuracy, overall accuracy and Kappa coefficient.

Data	Measure	C1	C2	C3	C4	Overall accuracy (%)	Kappa coefficient
a	Producer's accuracy (%)	94.62	91.31	97.99	-	94.46	0.9374
	User's accuracy (%)	94.15	91.27	98.96			
b	Producer's accuracy (%)	85.07	94.67	95.93	-	89.44	0.8802
	User's accuracy (%)	96.74	76.14	91.98			
c	Producer's accuracy (%)	95.53	92.36	96.28	96.63	95.86	0.9558
	User's accuracy (%)	94.99	91.55	97.88	99.10		

In the worst case, 91.31 % of real ground points (1,933 out of 2,117), which correspond to the area covered by trees, are correctly segmented. The algorithm incorrectly omitted 8.69 % of pixels (184 out of 2,117) in the worst case. Correspondingly, 94.46% of the ground points (6,701 out of 7,094) are correctly identified and only 393 out of 7,094 or 5.54% of ground points are incorrectly segmented to other homogenous regions. In a similar way, all components in the matrix can be evaluated. As a conclusion, one would anticipate a high degree of accuracy for the segmented results from the proposed algorithm. As shown in Table 6, the Kappa coefficients for the synthetic datasets shown in Fig. 5.9 (a)-(c) are up to 0.9374, 0.8802 and 0.9558, respectively. According to the general interpretation rules for thematic accuracy, a Kappa coefficient of between 0.81-1.00 for the assessed data can be interpreted as being almost perfect (Fleiss, 1975).

C. Application to Object Identification

The results from the segmentation of testing data, including the mean, variance and shape for each segmented region, can be used to identify the objects existing in LiDAR data. From experiments on datasets (a)-(c), the ground region can be easily separated from objects in the ground by segmenting the dataset into two regions. The region with smaller the mean corresponds to the ground no matter its variance. In this case, the variance characterizes the undulation of the ground. For example, in dataset (c) the ground region consists of two terraces with a difference in height of around $2m$.

For the datasets (d) and (e), which include 3 regions: ground, trees and building, the ground can always be identified by its minimum mean. The trees can be recognized by their variances. They usually have the maximum variances since the regions corresponding trees consist of the points at which the laser beam can touch ground, trunk, branches, and crown.

The dataset (f) is segmented into 4 homogenous regions. Except for the ground, the trees can be identified by the variance. For example, as shown in Table 5.2 that for dataset (f) the variance of the C2 region, in green in Fig. 5.4 (f), is up to 8.48. The region can therefore be deduced to be trees. Though the C3 region, in red in Fig. 5.4 (f), has high variance (4.50) it is possessed of a regular shape, see its outline in red in Fig. 5.8 (f), the region can be judged to be a building.

5.2 Building Detection by Marked Point Process

5.2.1 Description of Algorithm for Building Detection

A. Data Model for Build Detection

Consider a LiDAR point cloud $\boldsymbol{Z} = \{Z_i = Z(x_i, y_i);\ i = 1, \ldots, n, (x_i, y_i) \in D\}$covering k buildings, where k is unknown random variable with prior PDF $p(k)$ defined in Eq. (3.14). A building hidden in $\boldsymbol{Z}$ is characterized by a cuboid with attribute vectors $(u_j, v_j, \mu_{bj}, l_j, w_j, a_j)$, where (u_j, v_j) is the central position of the orthographic projection of the cuboid (called window D_j), $(\mu_{bj}, l_j, w_j, a_j)$ are the height, length, width and direction of the cuboid, respectively. Fig. 5.14 shows the geometric structure of the building. In practice, the distribution of buildings is obtained by the marked point process based domain partition defined in Eq. (3.7). The marks associated with the central points of buildings are the geometric attributes corresponding to the buildings $\boldsymbol{M} = \{(\mu_{bj}, l_j, w_j, a_j); j = 1, \ldots, k\}$.

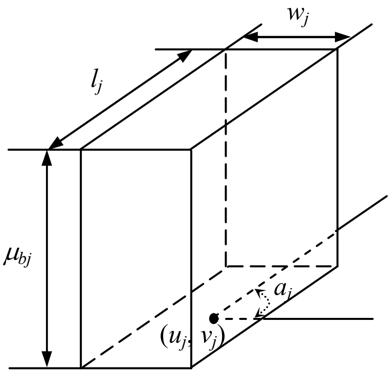


Fig. 5.14 Cuboid structure of the building centered at (u_j, v_j) with height μ_{bj}, length l_j, width w_j and direction a_j.

To distinguish different buildings in D, D is considered to have three homogeneous regions, that is, $\boldsymbol{D} = \{S_g, S_t, S_b\}$ where S_g, S_t and S_b correspond to ground, tree and building regions, respectively, and $S_b = \{D_j; j = 1, \ldots, k\}$. In this thesis, assume that the elevations in these regions are characterized by Gaussian distributions, that is,

$$p(Z_i) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma_g} \exp\left(-\frac{(Z_i - \mu_g)^2}{2\sigma_g{}^2} \right) & \text{if } (x_i, y_i) \in S_g \\ \frac{1}{\sqrt{2\pi}\sigma_t} \exp\left(-\frac{(Z_i - \mu_t)^2}{2\sigma_t{}^2} \right) & \text{if } (x_i, y_i) \in S_t \\ \frac{1}{\sqrt{2\pi}\sigma_{bj}} \exp\left(-\frac{(Z_i - \mu_{bj})^2}{2\sigma_{bj}{}^2} \right) & \text{if } (x_i, y_i) \in D_j \end{cases} \tag{5.12}$$

where μ_g, μ_t, $\mu_{bj,}$ σ_g, σ_t and σ_{bj} are the means and standard deviations of Gaussian distributions for the elevations of ground, tree and the j'th building, respectively, μ_g,

μ_t, σ_g, and σ_t are defined as constants, and μ_{bj} and σ_{bj} are random variables drawn from prior distributions $p(\mu_{bj})$ and $p(\sigma_{bj})$. As a result, these objects can be identified by the constraints on these distribution parameters, that is,

$$\begin{cases} \mu_g << \mu_t, \mu_{bj} \\ \sigma_t >> \sigma_g, \sigma_{bj} \end{cases} \tag{5.13}$$

Assume that all elevations are independent. Then the joint distributions for the elevations in object regions can be expressed as follows

$$p(\boldsymbol{Z}_g \mid S_g) = \prod_{(x_i, y_i) \in S_g} \frac{1}{\sqrt{2\pi}\sigma_g} \exp\left(-\frac{(Z_i - \mu_g)^2}{2\sigma_g{}^2} \right) \tag{5.14}$$

$$p(\boldsymbol{Z}_t \mid S_t) = \prod_{(x_i, y_i) \in S_t} \frac{1}{\sqrt{2\pi}\sigma_t} \exp\left(-\frac{(Z_i - \mu_t)^2}{2\sigma_t{}^2} \right) \tag{5.15}$$

$$\begin{aligned} p(\boldsymbol{Z}_b \mid \boldsymbol{G}, \boldsymbol{M}, k, \boldsymbol{\mu}_b, \boldsymbol{\sigma}_b) &= \prod_{W_j \in S_b} p(\boldsymbol{Z}_{D_j}) \\ &= \prod_{D_j \in S_b} \prod_{(x_i, y_i) \in D_j} \frac{1}{\sqrt{2\pi}\sigma_{bj}} \exp\left(-\frac{(Z_i - \mu_{bj})^2}{2\sigma_{bj}{}^2} \right) \end{aligned} \tag{5.16}$$

$$p(\boldsymbol{Z} \mid \boldsymbol{D}, \boldsymbol{G}, \boldsymbol{M}, k, \boldsymbol{\mu}_b, \boldsymbol{\sigma}_b) = p(\boldsymbol{Z}_g \mid D_g) p(\boldsymbol{Z}_t \mid D_t) p(\boldsymbol{Z}_b \mid \boldsymbol{G}, \boldsymbol{M}, k, \boldsymbol{\mu}_b, \boldsymbol{\sigma}_b) \tag{5.17}$$

where $\boldsymbol{Z}_g = \{Z_i; (x_i, y_i) \in S_g\}$, $\boldsymbol{Z}_t = \{Z_i; (x_i, y_i) \in S_t\}$, $\boldsymbol{Z}_{D_j} = \{Z_i; (x_i, y_i) \in D_j\}$, $\boldsymbol{\mu}_b = \{\mu_{bj}; j = 1, \ldots, k\}$ and $\boldsymbol{\sigma}_b = \{\sigma_{bj}; j = 1, \ldots, k\}$.

Assume that the directions of the windows are uniformly distributed on D and $(-\pi/2, \pi/2]$, respectively, then their PDFs can be written

$$p(\boldsymbol{\alpha} \mid k) = \prod_{j=1}^{k} p(a_j) = \prod_{j=1}^{k} \frac{1}{\pi} = \left(\frac{1}{\pi}\right)^k \tag{5.18}$$

Other geometric parameters of the buildings are assumed to be Gaussian distributions, that is,

$$p(\boldsymbol{l} \mid k) = \prod_{j=1}^{k} p(l_j) = \prod_{j=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_l} \exp\left(-\frac{(l_j - \mu_l)^2}{2\sigma_l^{\;2}}\right) \tag{5.19}$$

$$p(\boldsymbol{w} \mid k) = \prod_{j=1}^{k} p(w_j) = \prod_{j=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_w} \exp\left(-\frac{(w_j - \mu_w)^2}{2\sigma_w^{\;2}}\right) \tag{5.20}$$

$$p(\boldsymbol{\mu}_b \mid k) = \prod_{j=1}^{k} p(\mu_{bj}) = \prod_{j=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_{\mu_b}} \exp\left(-\frac{(\mu_{bj} - \mu_{\mu_b})^2}{2\sigma_{\mu_b}^{\;2}}\right) \tag{5.21}$$

$$p(\boldsymbol{\sigma}_b \mid k) = \prod_{j=1}^{k} p(\sigma_{bj}) = \prod_{j=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_{\sigma_b}} \exp\left(-\frac{(\sigma_{bj} - \mu_{\sigma_b})^2}{2\sigma_{\sigma_b}^{\;2}}\right) \tag{5.22}$$

The prior distribution of $\boldsymbol{G}$ is the same as in Eq. (3.19)

Accordingly, the parameter set of the data model can be written as $\boldsymbol{\Theta} = \{\boldsymbol{G}, \boldsymbol{l}, \boldsymbol{w}, \boldsymbol{a}, \boldsymbol{\mu}_b, \boldsymbol{\sigma}_b\}$, where $\boldsymbol{l} = \{l_j\,; j = 1, \ldots, k\}$, $\boldsymbol{w} = \{w_j\,; j = 1, \ldots, k\}$, $\boldsymbol{a} = \{a_j\,; j = 1, \ldots, k\}$, which completely represents the detected buildings. By the Bayesian paradigm, the posterior distribution of the parameter set conditional on a given dataset can be expressed as

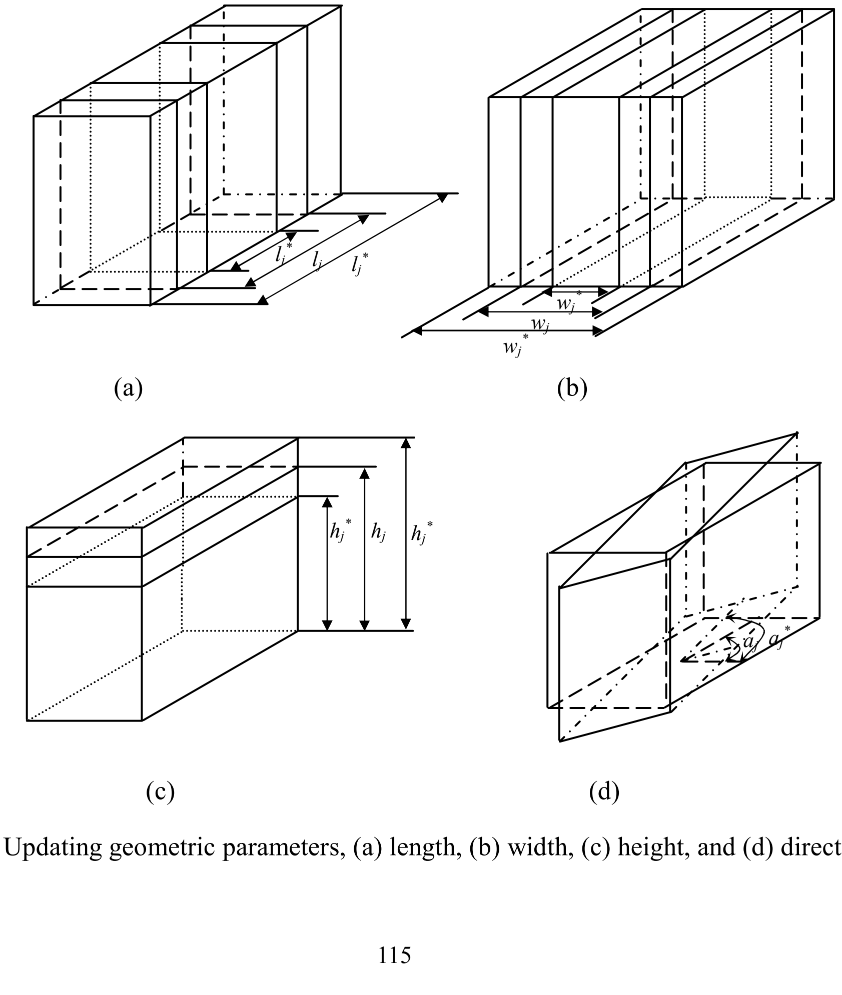
$$\begin{aligned} &p(\boldsymbol{\Theta} \mid \boldsymbol{Z}) \propto \\ &\quad p(\boldsymbol{Z} \mid \boldsymbol{\Theta}) p(\boldsymbol{D}) p(\boldsymbol{G} \mid k) p(\boldsymbol{l} \mid k) p(\boldsymbol{w} \mid k) p(\boldsymbol{a} \mid k) p(\boldsymbol{\mu}_b \mid k) p(\boldsymbol{\sigma}_b \mid k) p(k) \end{aligned} \tag{5.23}$$

The RJMCMC scheme described in Section 3.6 is used to simulate the posterior distribution in Eq. (5.28). The move types include (a) updating building model parameters in $\boldsymbol{\Phi} = \{\boldsymbol{l}, \boldsymbol{w}, \boldsymbol{a}, \boldsymbol{\mu}_b, \boldsymbol{\sigma}_b\}$; (b) moving the locations of detected buildings $\boldsymbol{G} = \{(u_j, v_j); j = 1, \ldots, k\}$; (c) birth or death of a building.

Move 1: Updating the distribution parameters. Draw a proposal for the updating parameter, $\Phi_j^* \sim N(\Phi_j, \varepsilon)$ where $\varepsilon = \varepsilon_l, \varepsilon_w, \varepsilon_a, \varepsilon_\mu, \varepsilon_\sigma$ in terms of the type of Φ. The acceptance probability for the proposal can be calculated by its marginal distribution

$$r_\Phi(\Phi_j, \Phi_j^*) = \min\left\{1, \frac{p(\boldsymbol{Z}_{D_j} \mid \Phi_j^*) p(\Phi_j^*)}{p(\boldsymbol{Z}_{D_j} \mid \Phi_j) p(\Phi_j)}\right\} \tag{5.24}$$

Fig. 5.15 illustrates the operations of updating geometrical parameters.



(a) (b)

(c) (d)

Fig.5.15 Updating geometric parameters, (a) length, (b) width, (c) height, and (d) direction.

Move 2: Updating the location of a central point. Propose a new central point for the building *j* by uniformly dawning a point from D_j, that is, $G_j^* = (u_j^*, v_j^*) \sim U(D_j)$. Fig. 5.16 shows the operation of updating a central point. Since there are two non-building classes there are four combinations for the operation to calculate the acceptance probability

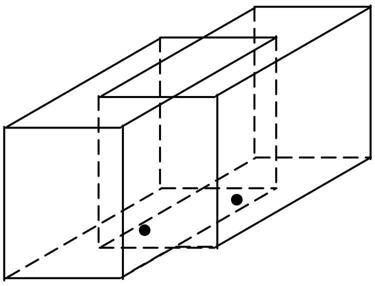


Fig. 5.16 Moving central point.

$$r_{g1}(G_j, G_j^*) = \min\left\{1, \frac{p(\boldsymbol{Z}_{D_j^*\backslash D_j} \mid \mu_{bj}, \sigma_{bj})\, p(\boldsymbol{Z}_{D_j\backslash D_j^*} \mid \mu_g, \sigma_g)\, p(G_j^*)}{p(\boldsymbol{Z}_{D_j^*\backslash D_j} \mid \mu_g, \sigma_g)\, p(\boldsymbol{Z}_{D_j\backslash D_j^*} \mid \mu_{bj}, \sigma_{bj})\, p(G_j)}\right\} \tag{5.25}$$

$$r_{g2}(G_j, G_j^*) = \min\left\{1, \frac{p(\boldsymbol{Z}_{D_j^*\backslash D_j} \mid \mu_{bj}, \sigma_{bj})\, p(\boldsymbol{Z}_{D_j\backslash D_j^*} \mid \mu_t, \sigma_t)\, p(G_j^*)}{p(\boldsymbol{Z}_{D_j^*\backslash D_j} \mid \mu_g, \sigma_g)\, p(\boldsymbol{Z}_{D_j\backslash D_j^*} \mid \mu_{bj}, \sigma_{bj})\, p(G_j)}\right\} \tag{5.26}$$

$$r_{g3}(G_j, G_j^*) = \min\left\{1, \frac{p(\boldsymbol{Z}_{D_j^*\backslash D_j} \mid \mu_{bj}, \sigma_{bj})\, p(\boldsymbol{Z}_{D_j\backslash D_j^*} \mid \mu_g, \sigma_g)\, p(G_j^*)}{p(\boldsymbol{Z}_{D_j^*\backslash D_j} \mid \mu_t, \sigma_t)\, p(\boldsymbol{Z}_{D_j\backslash D_j^*} \mid \mu_{bj}, \sigma_{bj})\, p(G_j)}\right\} \tag{5.27}$$

$$r_{g4}(G_j, G_j^*) = \min\left\{1, \frac{p(\boldsymbol{Z}_{D_j^*\backslash D_j} \mid \mu_{bj}, \sigma_{bj})\, p(\boldsymbol{Z}_{D_j\backslash D_j^*} \mid \mu_t, \sigma_t)\, p(G_j^*)}{p(\boldsymbol{Z}_{D_j^*\backslash D_j} \mid \mu_t, \sigma_t)\, p(\boldsymbol{Z}_{D_j\backslash D_j^*} \mid \mu_{bj}, \sigma_{bj})\, p(G_j)}\right\} \tag{5.28}$$

G_j^* is accepted if and only if all combinations are accepted.

Move 3: Birth or death of a building. Let the probability of proposing a birth or death operation be b_k or d_k, respectively. Consider a birth operation which increases the number of buildings from k to k+1 and assume that the new building is labelled k+1. Draw the central point ($u_{k+1}{}^*$, $v_{k+1}{}^*$) from $D\backslash S_b$ uniformly. Draw the geometric parameter for the new building from their distribution, that is, ($l_{k+1}{}^*$, $w_{k+1}{}^*$, $a_{k+1}{}^*$, $\mu_{bk+1}{}^*$, $\sigma_{bk+1}{}^*$). Let the window induced by { $u_{k+1}{}^*$, $v_{k+1}{}^*$, $l_{k+1}{}^*$, $w_{k+1}{}^*$, $a_{k+1}{}^*$} be $D_{k+1}{}^*$. As a result, the parameter space after the birth operation becomes from $\boldsymbol{\Theta}$= {$\boldsymbol{G}$, $\boldsymbol{\Phi}$, k} to $\boldsymbol{\Theta}^*$ = { $\boldsymbol{G}$, $\boldsymbol{\Phi}$, k+1, $u_{k+1}{}^*$, $v_{k+1}{}^*$, $l_{k+1}{}^*$, $w_{k+1}{}^*$, $a_{k+1}{}^*$, $\mu_{bk+1}{}^*$, $\sigma_{bk+1}{}^*$}. The acceptance probability for the birth operation can be written as

$$r_b(\boldsymbol{\Theta},\boldsymbol{\Theta}^*) = \min\{1, R_b\} \tag{5.29}$$

and

$$R_b = \frac{p(\boldsymbol{Z}\mid\boldsymbol{\Theta}^*)p(\boldsymbol{\Theta}^*)r_{b_k}(\boldsymbol{\Theta}^*)}{p(\boldsymbol{Z}\mid\boldsymbol{\Theta})p(\boldsymbol{\Theta})r_{d_{k+1}}(\boldsymbol{\Theta})p(\boldsymbol{s})}\left|\frac{\partial(\boldsymbol{\Theta}^*)}{\partial(\boldsymbol{\Theta},\boldsymbol{s})}\right| \tag{5.30}$$

where $\boldsymbol{s}$ is a vector for $\boldsymbol{\Theta}^*$ =$\boldsymbol{\Theta}^*$($\boldsymbol{\Theta}$, $\boldsymbol{s}$). For simplicity, let $\boldsymbol{s}$ = {u_{k+1}, v_{k+1}, l_{k+1}, w_{k+1}, a_{k+1}, μ_{bk+1}, σ_{bk+1}}. Thus

$$r_{b_k} = b_k \tag{5.31}$$

$$r_{d_{k+1}} = \frac{d_{k+1}}{k+1} \tag{5.32}$$

$$\frac{p(\boldsymbol{\Theta}^*)}{p(\boldsymbol{\Theta})p(\boldsymbol{s})} = \frac{p(k+1)}{p(k)} = \frac{\lambda}{k+1} \tag{5.33}$$

$$\left|\frac{\partial(\boldsymbol{\Theta}^*)}{\partial(\boldsymbol{\Theta}, \boldsymbol{s})}\right| = 1 \tag{5.34}$$

Since there are two non-building classes: ground and tree, to create a new building, it is necessary to calculate the acceptance probability of birth of a building by comparing the two object classes, that is

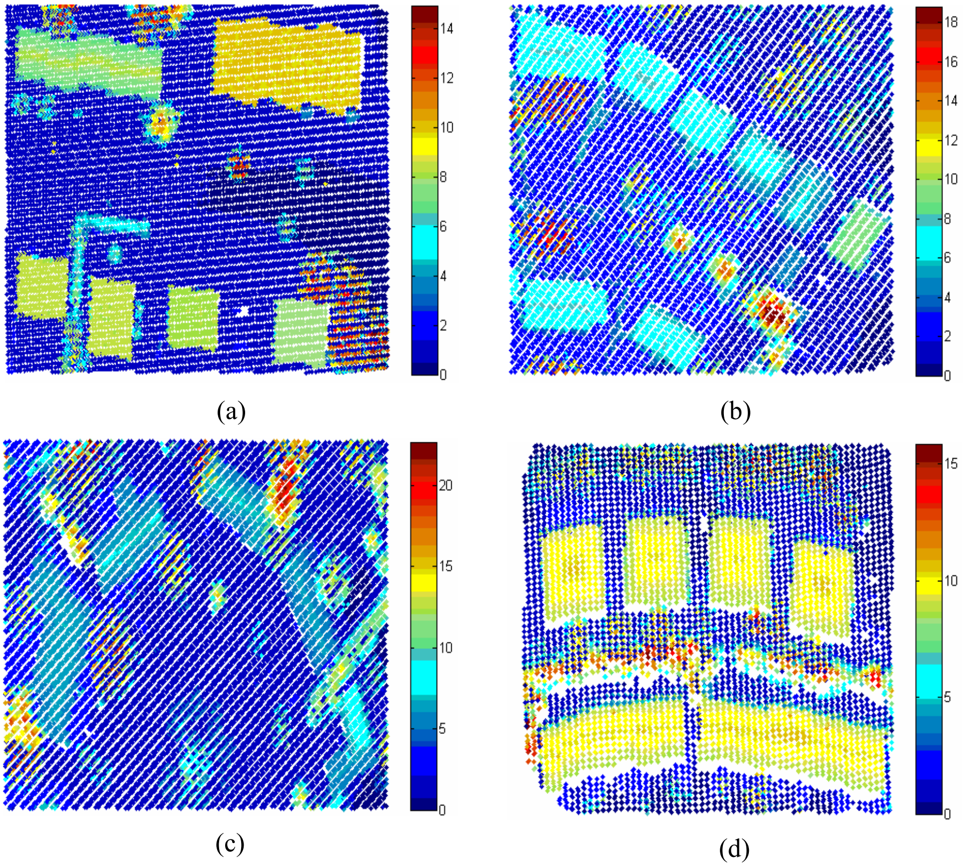
$$R_{bg} = \frac{b_k \lambda}{d_{k+1}} \frac{p(\boldsymbol{Z} \mid \boldsymbol{\Theta}^*)}{p(\boldsymbol{Z} \mid \boldsymbol{\Theta})} = \frac{\prod_{(x_i, y_i) \in D_{k+1}^*} \frac{1}{\sqrt{2\pi}\sigma_{bk+1}^*} \exp\left(-\frac{(Z_i - \mu_{bk+1}^*)^2}{2(\sigma_{bk+1}^*)^2} \right)}{\prod_{(x_i, y_i) \in D_{k+1}^*} \frac{1}{\sqrt{2\pi}\sigma_g} \exp\left(-\frac{(Z_i - \sigma_g)^2}{2\sigma_g^2} \right)} \tag{5.35}$$

$$R_{bt} = \frac{b_k \lambda}{d_{k+1}} \frac{p(\boldsymbol{Z} \mid \boldsymbol{\Phi}^*)}{p(\boldsymbol{Z} \mid \boldsymbol{\Phi})} = \frac{\prod_{(x_i, y_i) \in D_{k+1}^*} \frac{1}{\sqrt{2\pi}\sigma_{bk+1}^*} \exp\left(-\frac{(Z_i - \mu_{bk+1}^*)^2}{2(\sigma_{bk+1}^*)^2} \right)}{\prod_{(x_i, y_i) \in D_{k+1}^*} \frac{1}{\sqrt{2\pi}\sigma_t} \exp\left(-\frac{(Z_i - \mu_t)^2}{2\sigma_t^2} \right)} \tag{5.36}$$

For any move, its proposal is accepted if and only if the acceptance probability $r \geq \xi$ where $\xi \sim U(0, 1)$. The MAP scheme defined in Section 3.6 is used to obtain the optimal detection of buildings.

5.2.2 Experimental Results on Building Detection

The presented algorithm for building detection is applied to LiDAR point cloud data covering residential areas. For simplicity and without loss of generality, the elevations in those scenes are normalized to [0, E], where $E = \max(\boldsymbol{Z}) - \min(\boldsymbol{Z})$. Fig. 5.17 shows the testing data where the colours indicate the elevations.



(a) (b)

(c) (d)

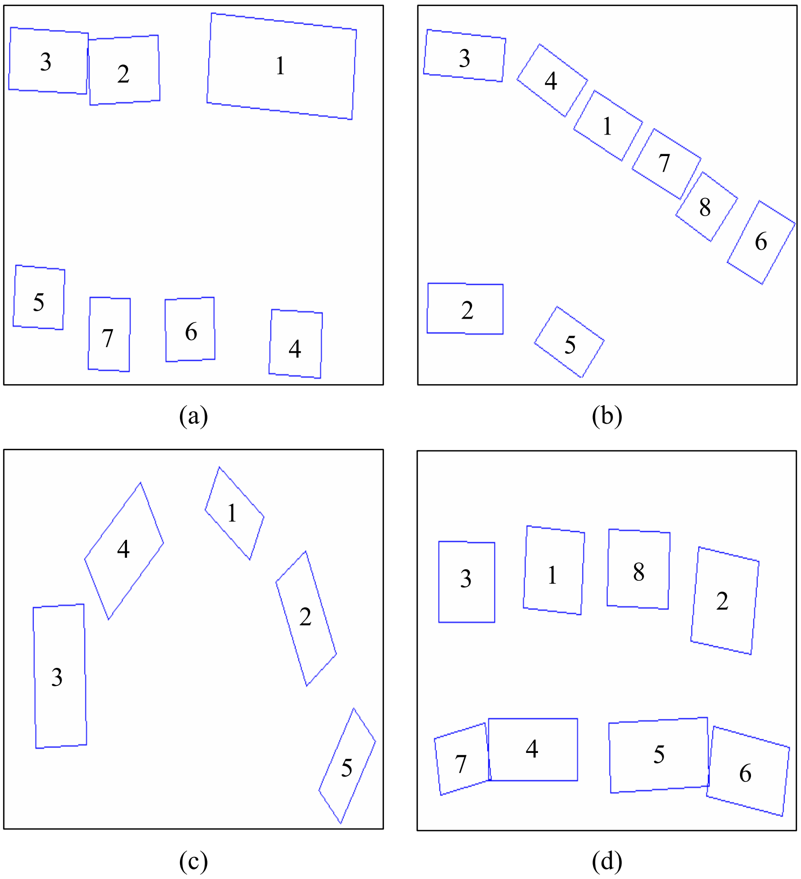
Fig. 5.17 Testing LiDAR point cloud data.

Table 5.7 lists the constants used in the posterior distribution for the experiment purpose. μ_g = 0 corresponds to the minimum elevation in a scene while σ_g = 2 indicates the vertical displacement of the ground around 2 m. The mean height of trees is set as the median of elevation difference $E/2$, and $\sigma_t = E/2$ means the laser beams can touch the ground or the top of the trees on a tree area. The mean and standard deviation of the Gaussian distribution for building heights is set as $E/2$ and $E/4$. The relief of building heights is around 2 m, that is, set $\sigma_{\sigma_b} = 2$. λ is the mean of Poisson distribution for the number of buildings, which is set as 6. The constants $\varepsilon_{l,\ w,\ a,\ \mu,\ \sigma}$ are the proposal variances for l, w, a, μ_b and σ_b, respectively. The T_m used in this experiment is uninformative and larger than practically used ones.

Table 5.7 Constants used in the posterior distribution

μ_g	σ_g	μ_{μ_b}	σ_{μ_b}	μ_{σ_b}	σ_{σ_b}	μ_t	σ_t
0	2	$E/2$	$E/4$	2	1	$E/2$	$E/2$
$\mu_{l,w}$	15	λ	$\varepsilon_{l,w}$	ε_a	ε_h	ε_τ	T_m
$\sigma_{l,w}$	4	6	0.5	$\pi/36$	0.5	0.25	20000

Fig. 5.18 shows the distributions of detected windows in blue corresponding to the



(a) (b)

(c) (d)

Fig. 5.18 Detected windows corresponding to the outlines of building roofs.

outlines of the building roofs. The numbers in the windows indicate their labels.

Table 5.8 gives the estimated geometric parameters of the detected buildings, including length, width, direction, height, and standard deviation of height.

Table 5.8 Estimated geometric parameters

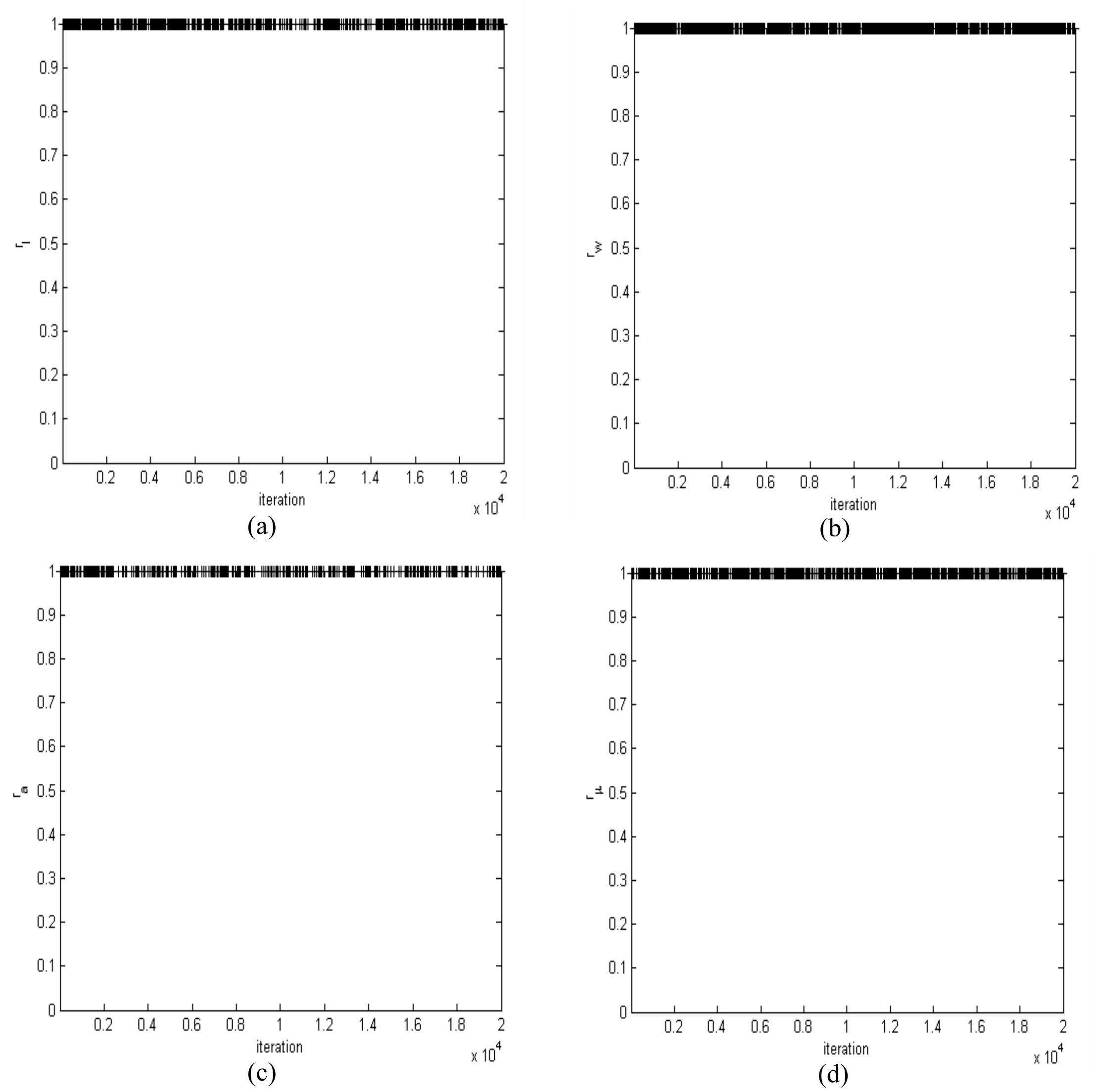
Dataset	Windows	l	w	a	μ_b	σ_b
a	1	17.78	44.20	85.7	8.81	2.48
	2	21.17	12.93	2.4	8.18	3.48
	3	12.17	23.44	87.6	8.19	2.98
	4	12.80	15.44	87.3	8.18	1.03
	5	12.02	14.99	87.1	8.21	1.18
	6	14.83	12.19	1.6	8.22	1.18
	7	14.32	12.29	88.0	8.19	0.07
b	1	9.26	12.66	58.7	7.07	0.82
	2	10.12	16.98	89.2	7.06	1.90
	3	8.97	17.78	84.7	7.06	0.87
	4	9.02	13.16	55.1	7.05	0.51
	5	9.12	12.67	56.0	7.05	0.88
	6	14.44	9.12	59.6	7.70	2.06
	7	9.61	12.40	59.7	7.05	1.36
	8	10.73	9.49	55.8	7.05	1.88
c	1	8.91	17.30	59.6	6.26	0.61
	2	11.38	21.04	29.39	5.41	1.40
	3	17.08	24.93	2.26	5.59	1.39
	4	25.00	19.32	58.7	4.88	1.40
	5	18.59	9.38	51.4	5.56	1.40
d	1	14.05	16.48	85.60	9.32	0.40
	2	16.24	17.35	82.14	9.44	0.65
	3	13.83	15.97	89.91	9.67	0.39
	4	10.77	25.27	90.00	9.50	0.53
	5	28.09	11.88	.2.12	9.60	0.86
	6	12.08	21.94	80.64	9.41	0.53
	7	13.15	17.39	87.88	9.30	0.38
	8	14.84	9.79	10.18	9.08	1.39

Table 5.9 lists the acceptance rate for each operation, where r_l, r_w, r_a, r_μ, r_σ, r_g, r_b, and r_d are the acceptance rates of updating length, width, direction, height, height error, moving central point, birth of window and dead of window, respectively.

Table 5.9 Acceptance rates of all operations (%)

Dataset	r_l	r_w	r_a	r_μ	r_σ	r_g	r_b	r_d
a	8.11	7.72	7.03	2.66	25.91	1.00	0.43	0.4
b	3.77	4.59	3.28	1.69	20.30	0.12	0.04	0.00
c	4.34	6.87	2.39	4.51	9.31	0.09	0.025	0.005
d	11.47	7.00	2.39	1.86	9.47	0.12	0.035	0.00

Fig. 5.19 shows the accepted geometric parameters of the building in the testing dataset shown in Fig. 5.17 (d), including length *l*, width *w*, direction *a*, height μ_b.



(a)

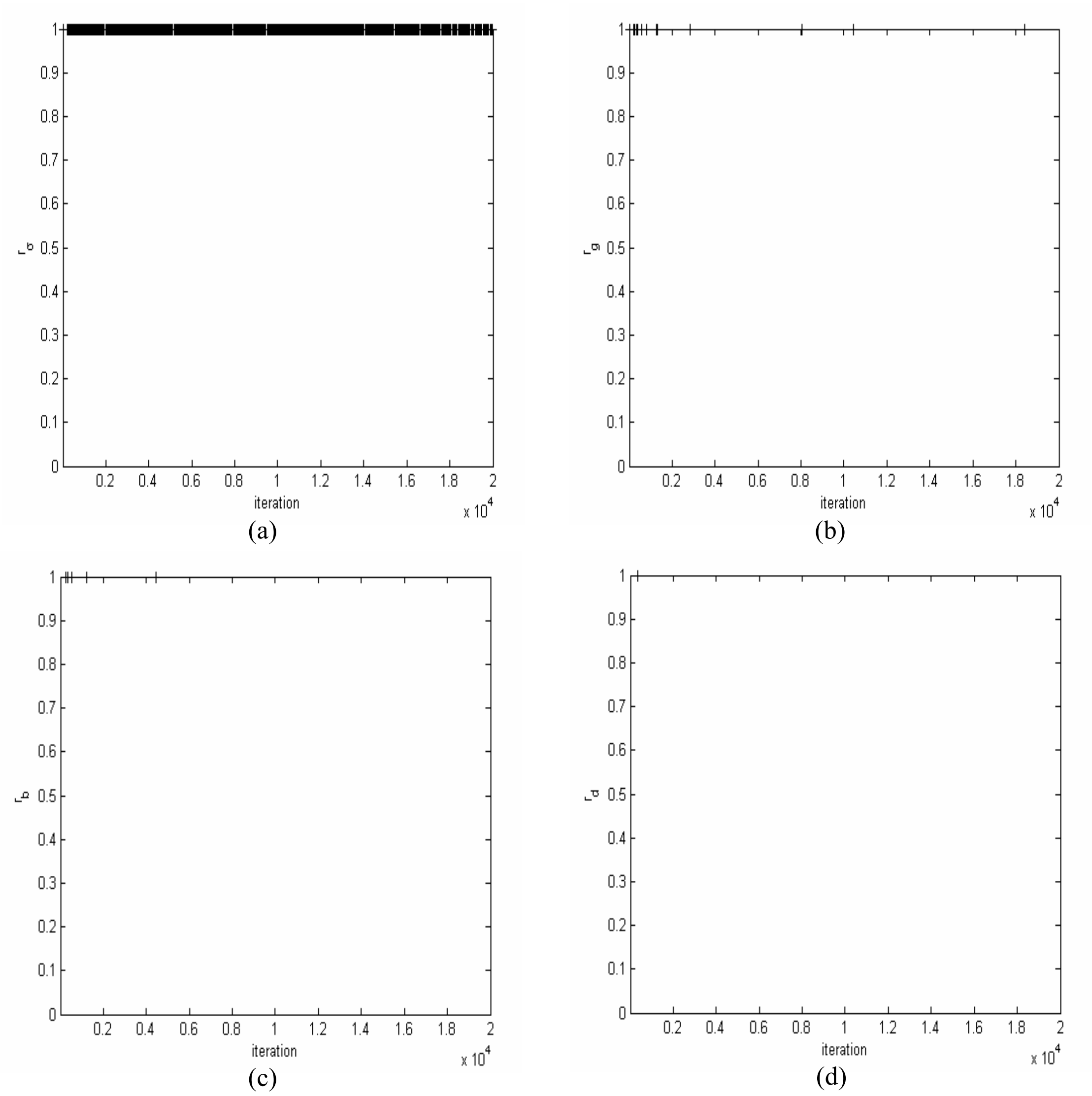
(b)

(c)

(d)

Fig. 5.19 Acceptance of length *l*, width *w*, direction *a* and height μ_b.

Fig. 5.20 shows the accepted standard deviation of the building, moving central point, birth and death for the testing dataset shown in Fig. 5.17 (d).



(a)

(b)

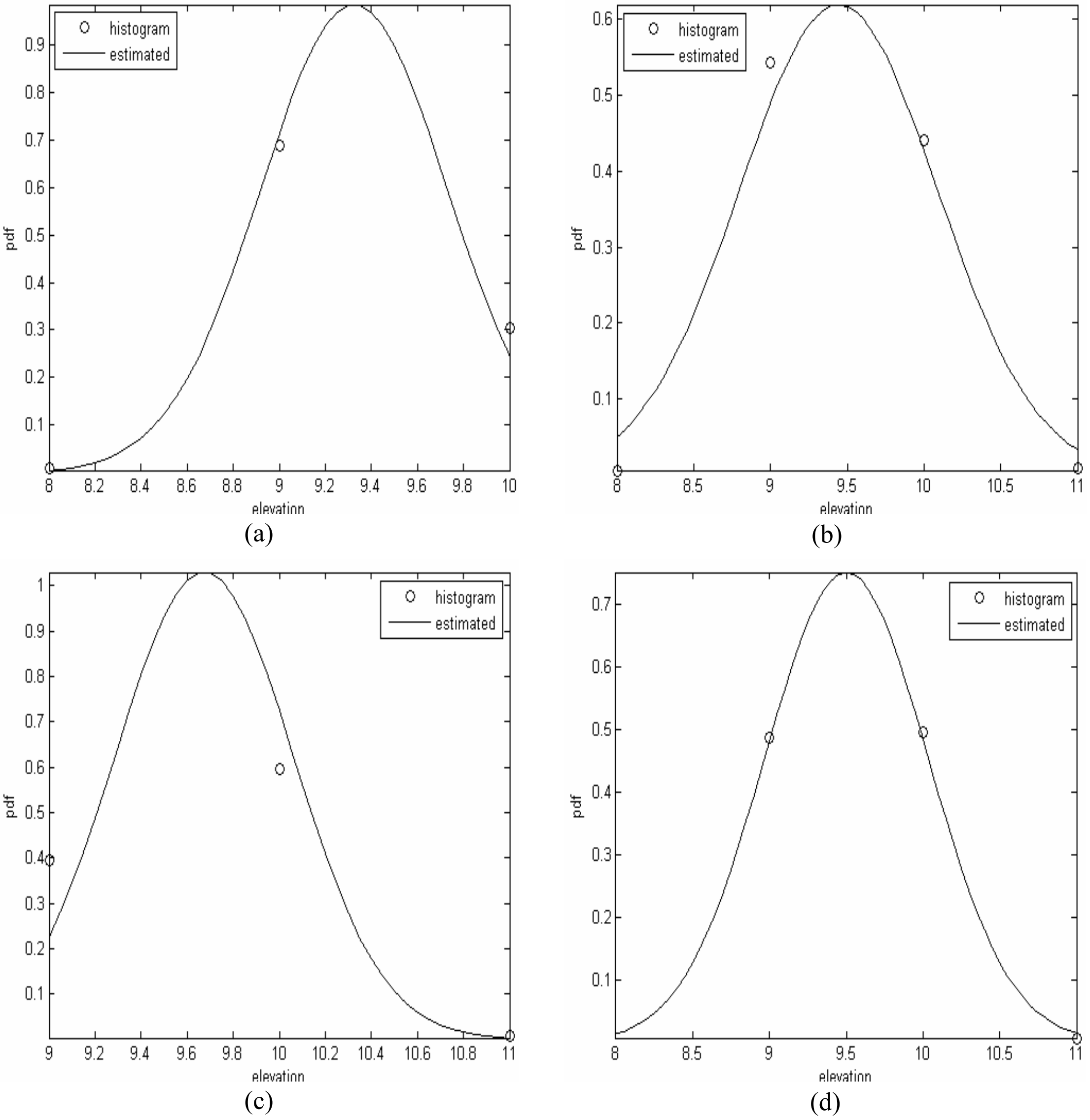
(c)

(d)

Fig. 5.20 Acceptance of standard deviation σ_b, moving central point, birth and dead.

From Fig. 5.19, Fig. 5.20 and Table 5.9, it can be observed that the operation of birth and death of central points have the minimum acceptance rate and the accepting of the new buildings is concentrated in the initial phase of the iterations, within the first 5,000 iterations, see Fig. 5.20 (c) and (d). The phenomenon implies that the presented algorithm can quickly decide the number of windows which represent the detected buildings. The operation of updating the standard deviation σ_b has the maximum acceptance rate, which is partly caused by the small ε_τ and it also means that the parameter is sensitive to the data. Comparing the operations of updating length, width, direction, and height, the smaller acceptance rate for moving central points implies the former plays a more important role in the detection than the latter.

Figs. 5.21-22 shows the Gaussian distributions with the estimated means and standard deviation (μ_b's and σ_b's listed in Table 5.9), and the histogram of elevations within window (1)-(4) and (5)-(8), respectively, for test data shown in Fig. 5.17 (d). As shown in Figs. 5.21-22 the histograms match the Gaussian distributions well.



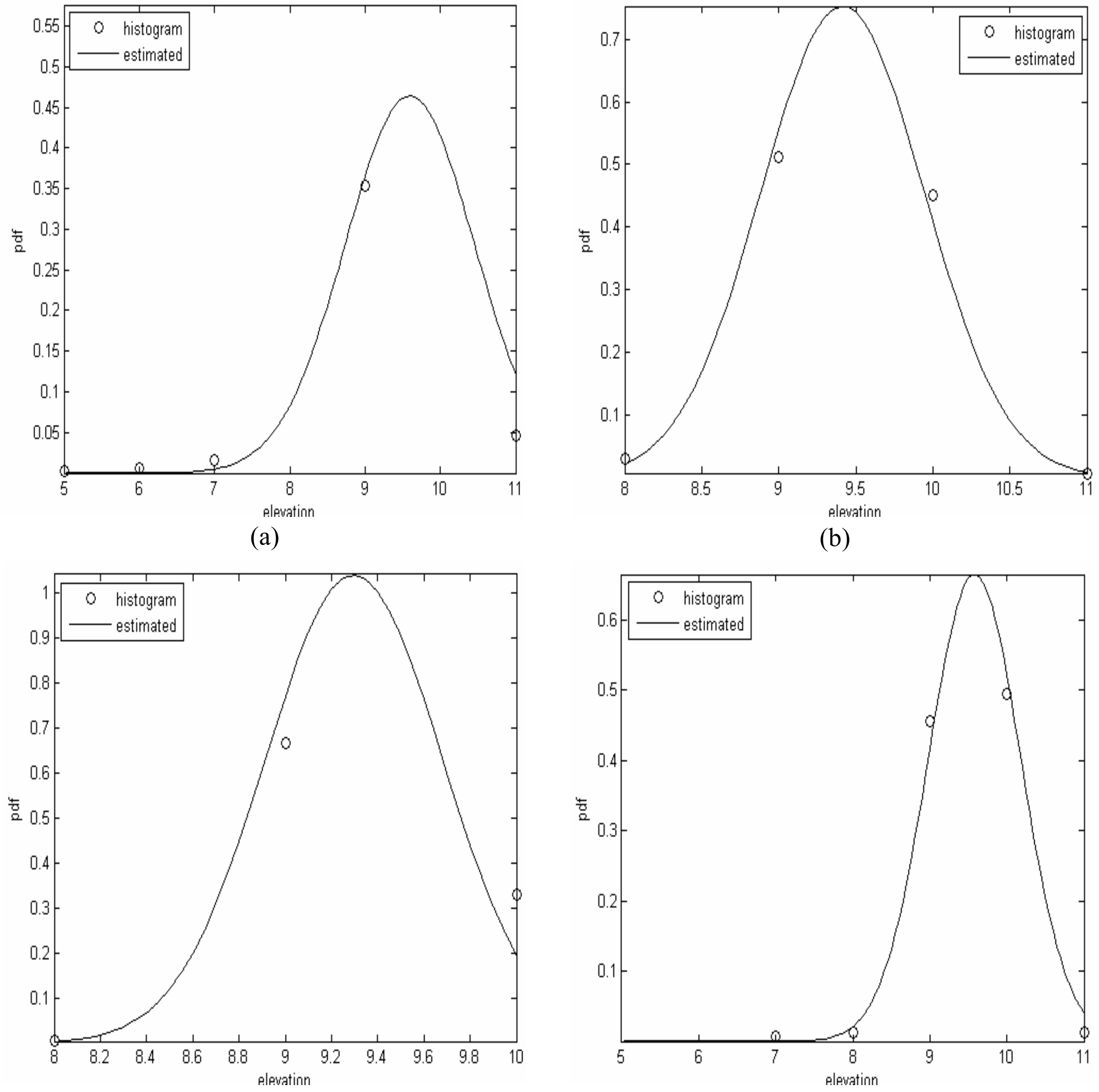
(a)

(b)

(c)

(d)

Fig. 5.21 Gaussian distributions with the estimated means and standard deviations and histogram of elevations within window (1)-(4) for test dataset shown in Fig. 5.17 (d).



(a)

(b)

(c)

(d)

Fig. 5.22 Gaussian distributions with the estimated means and standard deviations and histogram of elevations within window (5)-(8) for test dataset shown in Fig. 5.17 (d).

Fig. 5.23 shows the Gaussian distributions with parameter μ_g, μ_t, σ_g, and σ_t listed in Table 5.8 and histogram of ground and tree classes for the testing dataset shown in Fig 5.17 (d). In Fig. 5.23, the Gaussian distribution of elevations for tree class does not match their histogram. The problem can be solved in future work by assuming that the distribution parameters are variables instead of constants.

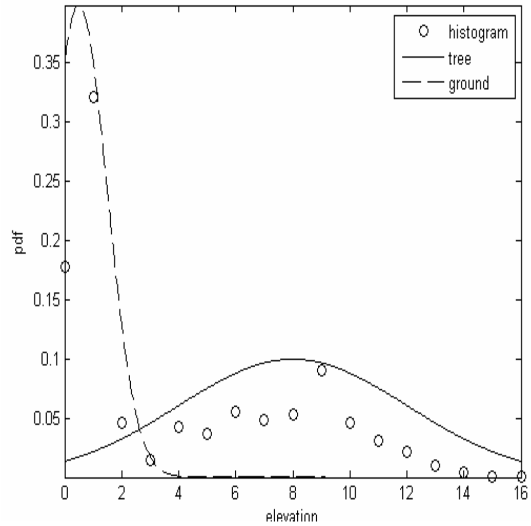
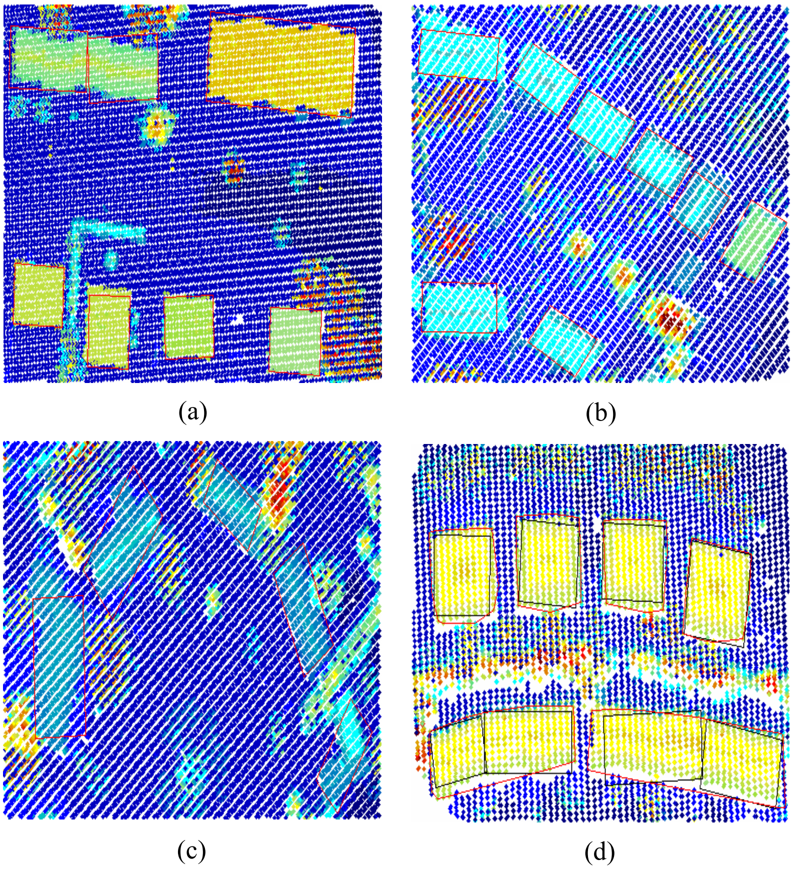


Fig. 5.23 Gaussian distributions with the fix means and standard deviations and histogram of elevations of tree and ground classes for test dataset shown in Fig. 5.17 (d).

In order to visually evaluate the accuracy of the presented algorithm for the building extraction, the detected windows (in black) are overlaid on the test data, see Fig. 5.24. From Fig. 5.24 (a), (b) and (d), it can be observed that the presented algorithm can detect the building shape well, if it is rectangular. It can also be observed that some long buildings are detected by two segments, for example, 2 and 3 in Fig. 5.18 (a), 4 and 7, 5 and 6 in Fig. 5.18 (d). For buildings with non-rectangular shape, the algorithm doesn't work well, for example, the L shape buildings (3) - (5) in Fig. 5.17 (c). In practice, polygon model for build shape might solve the problem.

The proposed algorithm is developed using MATLAB running on a DELL Optiple GX 745 computer. Take the dataset shown in Fig. 5.17 (d) as an example, which includes 9,731 data points and covers 108×65 m^2 area. The average computing time of each iteration, in which all three moves are accepted, is around 0.20 sec. As a result,



(a)

(b)

(c)

(d)

Fig. 5.24 Overlaying the detected windows on testing data.

the maximum computation time for 20,000 iterations is about 65 minutes. The computational burden for updating distribution parameters, moving location of central points, and birth or death of windows account for 5%, 35%, and 60%, of the time respectively.

In this experiment, some statistical measurements are calculated for quantitative evaluation to show the accuracy of detected building dimensions. To this end, the ground truth of building-roof outlines is manually generated, as shown with red lines in Fig. 5.24 (d). Table 5.10 gives the numbers of manually generated building (*GT*), detected building (*DP*), correctly detected building (DP_0) and statistical measurements (DP_0/GT (%) and DP_0/DP (%)). As shown in Table 5.10, in the worst case 74.3% of building points (251 out of 338) in building 1 is detected. From algorithmic point of view, in the worst case 95.2% of detected points (572 out of 601) are building points.

Table 5.10 Statistical measurements for accuracy assessment of testing dataset shown in Fig. 5. 17 (d).

measurement	1	2	3	4+8	5+6	7
GT	338	317	324	503	742	317
DP	251	298	243	446	601	257
DP_0	251	298	243	436	572	257
DP_0/GT (%)	74.3	94.0	75.0	86.7	77.1	81.1
DP_0/DP(%)	100.0	100.0	97.8	100.0	95.2	100.0

5.3 Chapter Summary

In this chapter, a statistical region-based segmentation algorithm for raw LiDAR point cloud data is first developed, which is based on the segmentation framework proposed in Chapter 3. The experimental results indicate that the proposed segmentation algorithm provides an accurate estimate of not only geometric parameters such as height and shape but also a statistical parameter such as variance. All of these parameters can be used for object identification.

Future research will focus on (1) the investigation of the impact of the number of polygons by setting it as variable; (2) the improvement of the current algorithm to automatically search the effective number of homogenous regions rather than indicate it *a priori*; (3) the development of a more efficient MRF model to characterize the spatial interactions among pixels in a given sub-region; (4) the development of a novel MCMC strategy for posterior simulation.

A new algorithm for the building detection from LiDAR point cloud data is presented which is based on a marked point process and Bayesian inference. More specifically, in the presented algorithm, the buildings hidden in the point cloud data are modeled as a marked point process which is created by a group of points uniformly distributed on the domain of a given dataset and attached as a cuboid with length, width, direction and height, while the elevations of buildings follows a Gaussian distribution. Results from the LiDAR point clouds show that the presented algorithm can detect buildings very well. Instead of processing data point by point for the purpose of building detection, the presented algorithm processes the data points in and out of windows simultaneously. On the other hand, non-building areas are modeled by two Gaussian distributions, which characterize the ground and tree, respectively. As a result, the presented algorithm can clearly distinguish buildings from trees.

The further work on improving the presented algorithm focuses on (1) considering a melting process to joint the windows corresponding to one building; (2) improving the proposed algorithm suitable for the scenes with uneven ground.

Chapter 6

TEXTURE IMAGE ANALYSIS

This chapter focuses on the problem of texture segmentation. Section 6.1 details a region-based algorithm for colour texture segmentation, which is developed from the segmentation framework described in Chapter 3. In the proposed algorithm, the MGMRF model is used to characterize colour texture structures of inter- and intra-polygons. The proposed algorithm for colour texture segmentation is tested on three types of images, including synthetic and real colour texture images as well as IKONOS satellite images. The accuracy assessments are preformed qualitatively on all of these images and quantitatively on the synthetic images. The experimental results demonstrate that the proposed algorithm is both efficient and effective. Section 6.2 describes a new algorithm for texture segmentation, which is based on the segmentation framework described in Chapter 3. In the proposed algorithm the BGMRF model is employed to characterize the spectral interactions between neighbouring pixels, respectively. The proposed algorithm is tested with both of synthesized and real texture images.

6.1 Colour Texture Segmentation

6.1.1 Description of Algorithm for Colour Texture Segmentation

Consider a colour texture image $\boldsymbol{Z} = \{\boldsymbol{Z}_i;\ i = 1, \ldots, n\}$, $\boldsymbol{Z}_i$ is the colour vector of *i'*th pixel at (x_i, y_i) made of three additive primaries: red, green and blue, that is, $\boldsymbol{Z}_i = (Z_{ri}, Z_{gi}, Z_{bi})$ where Z_{ri}, Z_{gi}, and Z_{bi} represent its red, green, and blue primaries, respectively.

Assume the colour image contains a known number of homogenous regions k and its domain D is partitioned into m sub-regions (for simplicity, m is also known *a priori*) by a Voronoi tessellation defined in Eq. (3.5). The label field is formed by $\boldsymbol{L} = \{L_j\,;\, j = 1, \ldots, k\}$.

The colour vectors for the pixels in the polygon D_j, $\boldsymbol{Z}_j = \{\boldsymbol{Z}_i,\ (x_i, y_i) \in D_j\}$, are modeled by the MGMRF model which characterizes the local structure of colour texture by specifying a multivariate Gaussian distribution for colour vectors of neighbouring pixels. The cliques used in this model include all pairs of horizontally, diagonally or vertically adjacent pixels in a second-order neighbourhood system (Li, 2009). Fig. 6.1 shows neighbouring systems, cliques on the first-order neighbourhood system, and cliques on the second-order neighbourhood system, whereδ is used for indicating the directions of two pixel cliques.

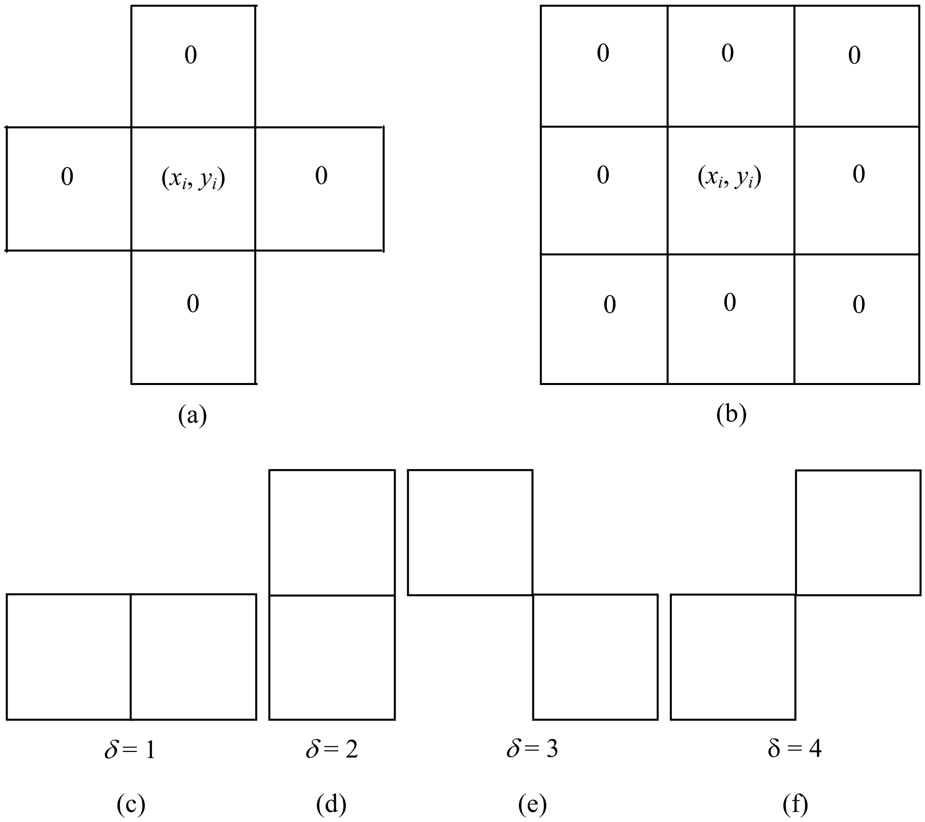


Fig. 6.1 (a) and (b) First and second-order neighbourhood system. (c)-(f) Cliques of two pixels on where δ is indicator of direction.

The second-order MGMRF (Anderson, 2003) for modeling neighbour pixels spatially and spectrally can be written as

$$p(\boldsymbol{Z}_i, \boldsymbol{Z}_{i'}) \sim N(\boldsymbol{\mu}_l \otimes \boldsymbol{1}_2, \boldsymbol{\Sigma}_{l\delta}\tau_l^{\,2}) \qquad (6.1)$$

where $(\boldsymbol{Z}_i, \boldsymbol{Z}_{i'}) = (Z_{ri}, Z_{gi}, Z_{bi}, Z_{ri'}, Z_{gi'}, Z_{bi'})$ is a colour vector pair of neighbour pixels; $L_j = l \in \{1, 2, \ldots k\}$ is the label for the polygon D_j including (x_i, y_i) and $(x_{i'}, y_{i'})$; $\boldsymbol{\mu}_l = (\mu_{rl}, \mu_{gl}, \mu_{bl})^T$ is mean vector for l'th homogenous region; $\boldsymbol{1}_2 = (1, 1)^T$ is 2-vector of 1s; $\otimes$ is kronecker product (Horn and Johnson, 1991). For two matrices $A_{N\times N'}$ and $B_{M\times M'}$,

$A \otimes B$ is defined as a $NM \times N'M'$ matrix formed by placing the element at the location (i, i') in A the copy of B scaled by the (i, i') element of A; τ_l is the standard deviation for the l'th homogenous region; and $\boldsymbol{\Sigma}_{l\delta}$ is the correlation coefficient for the l'th homogenous region at the direction indicated by δ, which characterizes the correlations between colours for the neighbour pixels spatially and spectrally. $\boldsymbol{\Sigma}_{l\delta}$ can be expressed as

$$\boldsymbol{\Sigma}_{l\delta} = \boldsymbol{\Theta}_{l\delta} \otimes \boldsymbol{B}_l \tag{6.2}$$

where δ indicates the direction of neighbour pixels i and i'; $\boldsymbol{\Theta}_{l\delta}$ is a 2×2 matrix and $\boldsymbol{B}_l$ is a 3×3 matrix, which model spatial and spectral correlations, respectively. They can be written as

$$\boldsymbol{\Theta}_{l\delta} = \begin{bmatrix} 1 & \theta_{l\delta} \\ \theta_{l\delta} & 1 \end{bmatrix} = I_2 + (\boldsymbol{I}_2 - I_2)\theta_{l\delta} \tag{6.3}$$

$$\boldsymbol{B}_l = \begin{bmatrix} 1 & \beta_l & \beta_l \\ \beta_l & 1 & \beta_l \\ \beta_l & \beta_l & 1 \end{bmatrix} = I_3 + (\boldsymbol{I}_3 - I_3)\beta_l \tag{6.4}$$

where I is identical matrix, $\theta_{l\delta}$ and β_l are spatial and spectral correlation coefficients, respectively. Thus, Eq. (6.1) can be rewritten as

$$p(\boldsymbol{Z}_i,\boldsymbol{Z}_{i'}) =$$

$$\frac{1}{(2\pi)^{3/2}\,|\boldsymbol{\Sigma}_{l\delta}\,|\,\tau_l}\exp\left\{-\frac{1}{2}\left[(\boldsymbol{Z}_i,\boldsymbol{Z}_{i'})-(\boldsymbol{\mu}_l,\boldsymbol{\mu}_l)\right]^T(\boldsymbol{\Sigma}_{l\delta}{\tau_l}^2)^{-1}\left[(\boldsymbol{Z}_i,\boldsymbol{Z}_{i'})-(\boldsymbol{\mu}_l,\boldsymbol{\mu}_l)\right]\right\} \tag{6.5}$$

On the assumption that $p(\boldsymbol{Z}_i, \boldsymbol{Z}_{i'})$'s for all cliques in P_j are independent, the probability $p(\boldsymbol{Z}_j \mid L_j)$ distribution conditional on the label $L_i = l$ can be obtained

$$p(\boldsymbol{Z}_j \mid L_j = l) = \prod_{\substack{\delta_{ii'}\in\{1,2,3,4\}\\(u_i,v_i),(u_i,v_i)\in D_j}} p(\boldsymbol{Z}_i,\boldsymbol{Z}_{i'}) \tag{6.6}$$

where $\delta_{ii'}$ indicates the direction of pixel i and i', if pixel i and i' are not a clique shown in Fig. 6.1 (c)-(f), then $\delta_{ii'} = 0$. Also assume that the joint distributions for all polygons are independent, the MGMRF can be expressed

$$P(\boldsymbol{Z} \mid \boldsymbol{L},\boldsymbol{\theta},\boldsymbol{\beta},\boldsymbol{\mu},\boldsymbol{\tau}) = \prod_{j=1}^{m} P(\boldsymbol{Z}_j \mid L_j = l,\boldsymbol{\theta}_l,\boldsymbol{\beta}_l,\boldsymbol{\mu}_l,\boldsymbol{\tau}_l) \tag{6.7}$$

where $\boldsymbol{\theta}_l = (\theta_{l1}, \theta_{l2}, \theta_{l3}, \theta_{l4})$, $\boldsymbol{\theta} = (\boldsymbol{\theta}_l;\ l = 1, \ldots, k)$, $\boldsymbol{\beta} = (\beta_l;\ l = 1, \ldots, k)$, and $\boldsymbol{\mu} = (\boldsymbol{\mu}_l;\ l = 1, \ldots, k)$, and $\boldsymbol{\tau} = (\tau_l;\ l = 1, \ldots, k)$.

For simplicity, assume the prior distributions for μ's, β's, and θ's to be independently truncated Gaussian distributions and the prior distribution for τ a Gamma distribution, that is, $\mu_{rl,\ gl,\ bl}$ ~ truncated $N(\mu_\mu, \sigma_\mu)$, β_l ~ truncated $N(\mu_\beta, \sigma_\beta)$, $\theta_{l1,2,3,4}$ ~ truncated $N(\mu_\theta, \sigma_\theta)$, and $\tau \sim \Gamma(\alpha_\tau, \beta_\tau)$. Thus, the join prior distributions for $\boldsymbol{\mu}$,

$\boldsymbol{\tau}$, $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ can be written

$$p(\boldsymbol{\mu}) = \prod_{l=1}^{k} \prod_{\mu_{\gamma l} \in \{\mu_{rl}, \mu_{gl}, \mu_{bl}\}} \frac{1}{\sqrt{2\pi}\sigma_{\mu}} \exp\left\{ -\frac{(\mu_{\gamma l} - \mu_{\mu})^2}{2\sigma_{\mu}^{2}} \right\} \tag{6.8}$$

$$p(\boldsymbol{\tau}) = \prod_{l=1}^{k} \frac{1}{\Gamma(\alpha_{\tau})\beta_{\tau}^{\alpha_{\tau}}} (\tau_l)^{\alpha_{\tau}-1} \exp\left\{ -\frac{\tau_l}{\beta_{\tau}} \right\} \tag{6.9}$$

$$p(\boldsymbol{\beta}) = \prod_{l=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_{\beta}} \exp\left\{ -\frac{(\beta_l - \mu_{\beta})^2}{2\sigma_{\beta}^{2}} \right\} \tag{6.10}$$

$$p(\boldsymbol{\theta}) = \prod_{l=1}^{k} \prod_{\theta_{l\delta} \in \{\theta_{l1}, \theta_{l2}, \theta_{l3}, \theta_{l4}\}} \frac{1}{\sqrt{2\pi}\sigma_{\theta}} \exp\left\{ -\frac{(\theta_{l\delta} - \mu_{\theta})^2}{2\sigma_{\theta}^{2}} \right\} \tag{6.11}$$

The truncated distributions for $\boldsymbol{\mu}_l$, $\boldsymbol{\beta}_l$ and $\boldsymbol{\theta}_l$ are used because the intensities of colour primaries vary in the range [0, 255] and the matrix $\boldsymbol{\Sigma}_{l\delta}$ should be a positive definite matrix. From Eq. (6.2), it can be proved that $\boldsymbol{\Sigma}_{l\delta}$ is positive definite if and only if $\boldsymbol{\Theta}_{l\delta}$ and $\boldsymbol{B}_l$ are positive definite simultaneously. From Eqs. (6.3) and (6.4), if $-1 < \theta_{l\delta} < 1$ and $-0.5 < \beta_l < 1$, $\boldsymbol{\Theta}_{l\delta}$ and $\boldsymbol{B}_l$ are positive definite, respectively. In this thesis, the $\boldsymbol{\Theta}_{l\delta}$ and $\boldsymbol{B}_l$ are truncated (0, 1).

Using the Bayesian segmentation model described in Section 3.5, the posterior distribution of $\boldsymbol{L}$, $\boldsymbol{\mu}$, $\boldsymbol{\tau}$, $\boldsymbol{\theta}$, $\boldsymbol{\beta}$ and $\boldsymbol{G}$ given $\boldsymbol{Z}$ can be written as,

$$p(\boldsymbol{L},\boldsymbol{\mu},\boldsymbol{\tau},\boldsymbol{\beta},\boldsymbol{\theta},\boldsymbol{G}\mid\boldsymbol{Z})\propto p(\boldsymbol{Z}\mid\boldsymbol{L},\boldsymbol{\mu},\boldsymbol{\tau},\boldsymbol{\beta},\boldsymbol{\theta},G)p(\boldsymbol{\mu})p(\boldsymbol{\tau})p(\boldsymbol{\beta})p(\boldsymbol{\theta})p(\boldsymbol{L})p(\boldsymbol{G})=$$

$$\prod_{j=1}^{m}\prod_{\substack{\delta_{ii'}\in\{1,2,3,4\}\\(u_i,v_i),(u_{i'},v_{i'})\in ND_j}}\frac{1}{(2\pi)^{3/2}\,|\boldsymbol{\Sigma}_{L_j\delta_{ii'}}\,|\,\tau_{L_j}}\exp\left\{-\frac{1}{2}\Big[(\boldsymbol{Z}_i,\boldsymbol{Z}_{i'})-(\boldsymbol{\mu}_{L_j},\boldsymbol{\mu}_{L_j})\Big]^T(\boldsymbol{\Sigma}_{L_j\delta_{ii'}}\tau_{L_j}{}^2)^{-1}\Big[(\boldsymbol{Z}_i,\boldsymbol{Z}_{i'})-(\boldsymbol{\mu}_{L_j},\boldsymbol{\mu}_{L_j})\Big]\right\}\times$$

$$\prod_{l=1}^{k}\prod_{\mu_{\gamma l}\in\{\mu_{rl},\mu_{gl},\mu_{bl}\}}\frac{1}{\sqrt{2\pi}\sigma_\mu}\exp\left\{-\frac{(\mu_{\gamma l}-\mu_\mu)^2}{2\sigma_\mu{}^2}\right\}\times\prod_{l=1}^{k}\frac{1}{\Gamma(\alpha_\tau)\beta_\tau{}^{\alpha_\tau}}(\tau_l)^{\alpha_\tau-1}\exp\left\{-\frac{\tau_l}{\beta_\tau}\right\}\times$$

$$\prod_{l=1}^{k}\frac{1}{\sqrt{2\pi}\sigma_\beta}\exp\left\{-\frac{(\beta_l-\mu_\beta)^2}{2\sigma_\beta{}^2}\right\}\times\prod_{l=1}^{k}\prod_{\theta_{l\delta}\in\{\theta_{l1},\theta_{l2},\theta_{l3},\theta_{l4}\}}\frac{1}{\sqrt{2\pi}\sigma_\theta}\exp\left\{-\frac{(\theta_{l\delta}-\mu_\theta)^2}{2\sigma_\theta{}^2}\right\}\times$$

$$\prod_{j=1}^{m}\frac{\exp\left(c\sum_{D_{j'}\in ND_j}t(L_j,L_{j'})\right)}{\sum_{l=1}^{k}\exp\left(c\sum_{D_{j'}\in ND_j}t(l,L_{j'})\right)}\times\left(\frac{1}{|D|}\right)^m$$

(6.12)

where ND_j is the set of D_j's neighbour polygons.

To simulate the posterior distribution in Eq. (6.12), the Metropolis-Hastings scheme described in Section 3.6 is used. The moves proposed in the scheme include (1) updating distribution parameters in $\boldsymbol{\psi} = \{\boldsymbol{u}, \boldsymbol{\tau}, \boldsymbol{\beta}, \boldsymbol{\theta}\}$; (2) updating labels in $\boldsymbol{L} = \{L_j;\ j = 1, \ldots, m\}$; (3) moving the locations of generating points in $\boldsymbol{G} = \{(u_j, v_j); j = 1, \ldots, m\}$.

Move 1: Updating distribution parameters. Draw a proposal for the parameter, $\psi^* \sim N(\psi, \varepsilon)$ where $\varepsilon = \varepsilon_\mu, \varepsilon_\tau, \varepsilon_\beta, \varepsilon_\theta$ in terms of the type of ψ. The acceptance probability for the proposal can be calculated by its marginal distribution

$$r_\Psi(\psi,\psi^*)=\min\left\{1,\frac{p(\boldsymbol{Z}\mid\psi^*)p(\psi^*)}{p(\boldsymbol{Z}\mid\psi)p(\psi)}\right\} \tag{6.13}$$

Move 2: Updating texture class labels. Draw a proposal L_j^* from {1, …, k} conditional on $L_j^* \neq L_j$. The acceptance probability for the proposal can be calculated by its marginal distribution

$$r_L(L_j,L_j^*)=\min\left\{1,\frac{p(\boldsymbol{Z}\mid L_j^*)p(L_j^*)}{p(\boldsymbol{Z}\mid L_j)p(L_j)}\right\} \tag{6.14}$$

Move 3: Update the location of generating point. Draw the proposal for the generating point by uniformly dawning a point from D_j, that is, $G_j^* = (u_j^*, v_j^*) \sim U(D_j)$. The acceptance probability for the proposal can be calculated by its marginal distribution

$$r_G(G_j,G_j^*)=\min\left\{1,\frac{p(\boldsymbol{Z}\mid G_j^*)p(G_j^*)}{p(\boldsymbol{Z}\mid G_j)p(G_j)}\right\} \tag{6.15}$$

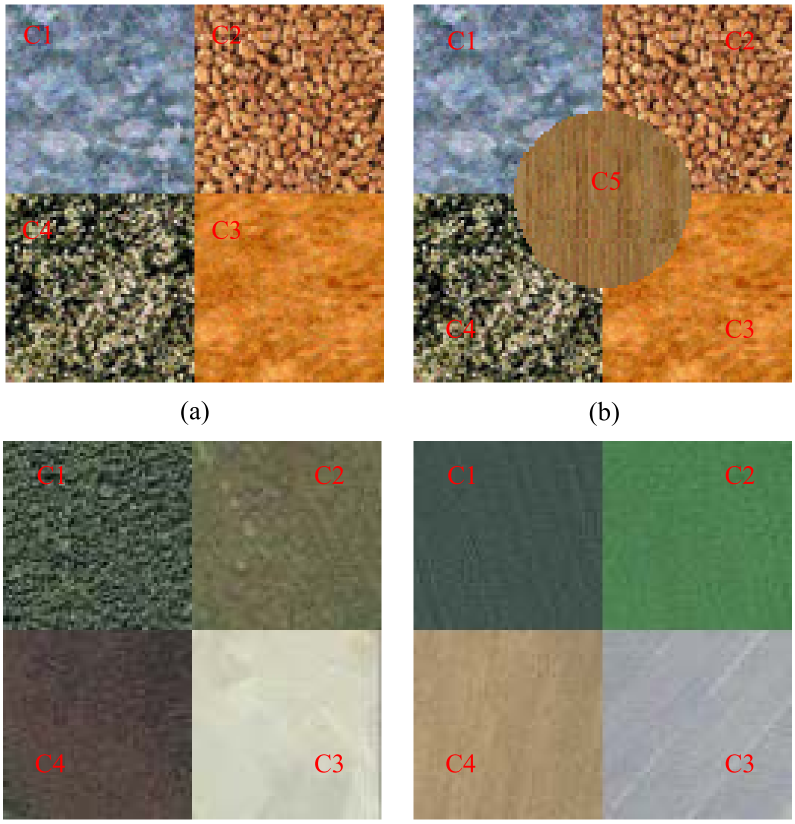
The MAP criterion defined in Eq. (3.29) is used to obtain the optimal segmentation.

6.1.2 Experimental Results on Colour Texture Segmentation

The proposed algorithm for colour texture segmentation is evaluated by testing several images, including synthetic colour images, colour texture image and colour remote sensing images.

A. Synthetic Colour Images

Fig. 6.2 shows several synthetic colour texture images. Fig. 6.2 (a) is comprised of four colour textures, donated by C1-C4. Fig. 6.2 (b) is generated by overlaying a disc colour texture C5 on (b). Fig. 6.2 (c) and (d) are synthesized by pitches with



(a)

(b)

(c)

Fig. 6.2 Synthetic colour texture images.

different LULC classes from IKONOS images.

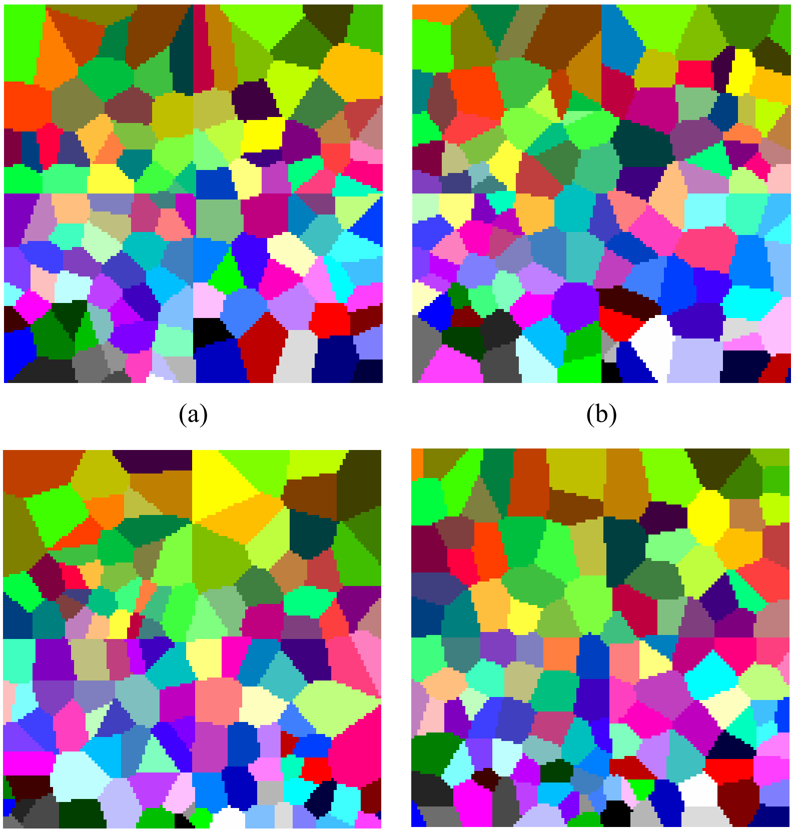
Table 6.1 lists the constants used in the experiments. The constant c is taken to be 1. The constants μ_μ and σ_μ are the mean and standard deviation of the Gaussian distribution for the components of a colour vector defined in Eq. (6.8). Since the means for the components of colour vectors is truncated in a range from 0 to 255, μ_μ is taken as the midpoint, i.e. 256/2 (= 128) and σ_μ is set as 32. In this experiment, the spectral and spatial correlation coefficients θ and β is truncated in the interval (0, 1). μ_β and μ_θ are set to be 1/2, and σ_β and σ_θ are 1/4. α_τ and β_τ are the shape and scale parameters of Gamma distribution for τ. They are taken to be 1 and 2, respectively. The constants ε_μ, ε_τ, ε_β and ε_θ are the standard deviations of proposals for μ, τ, β and θ, respectively. The constant m is the number of polygons. These constants are set according to the experiment. The constant k is the number of homogenous regions in each colour image. For a synthetic image, the number is easy to indicate. For a real colour image, selecting the number is not very reliable as it depends on the experience of human operators. The constant T_m is the maximum iteration of the algorithm. T_m normally depends on the complexity of the scene revealed in an image and requirement of segmentation accuracy. The T_m's selected in this experiment are uninformative and larger than practically used ones.

Table 6.1 Constants used in the posterior distribution in Eq. (6.12).

c	μ_μ	σ_μ	α_τ	β_τ	μ_β	σ_β	μ_θ
1	128	32	1	2	1/2	1/4	1/2
σ_θ	ε_μ	ε_τ	ε_β	ε_θ	m_a	m_b	$m_{c,d}$

1/4	2	1/2	1/16	1/16	128	96	128
k_a	k_b	k_c	k_d	T_{maxa}	T_{maxb}	T_{maxc}	T_{maxd}
4	5	4	4	8000	12000	8000	8000

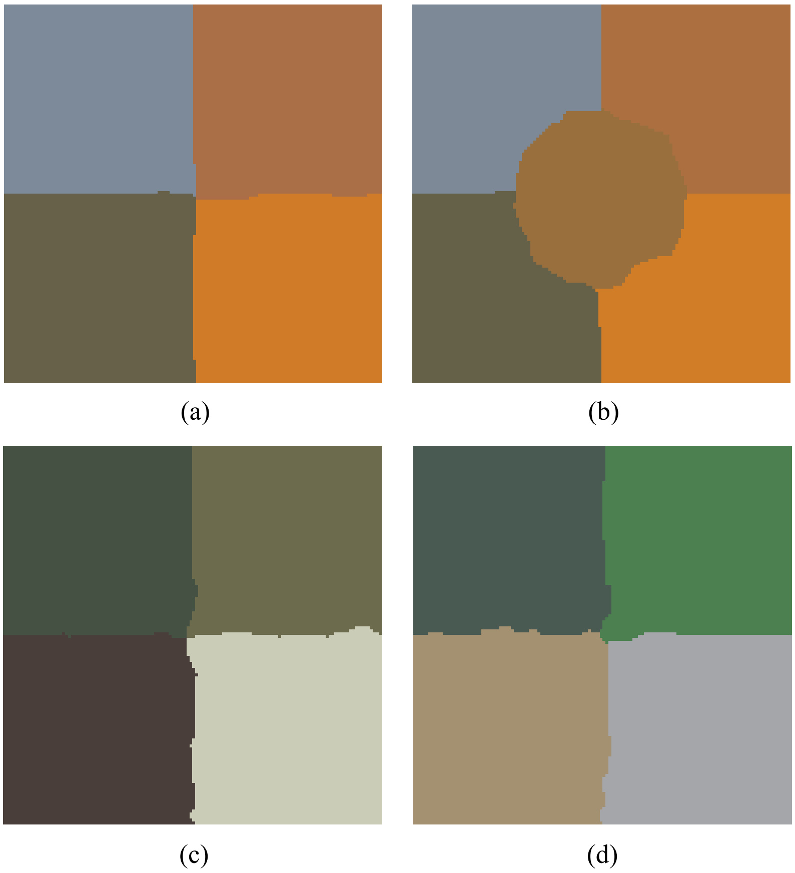
Fig. 6.3 show the optimal partitions obtained at the 7,984th 11,988th, 7,993rd and 7,988th iteration out of T_{maxa}, T_{maxb}, T_{maxc}, and T_{maxd} iterations, respectively. Fig. 6.4 gives the optimal segmentations corresponding to the optimal partitions, in which the segmented homogenous regions are represented by the estimated mean colours.



(a) (b)

(c) (d)

Fig. 6.3 Optimal partitions of texture images (a)-(d).



(a) (b)

(c) (d)

Fig. 6.4 Optimal segmentations of texture images (a)-(d).

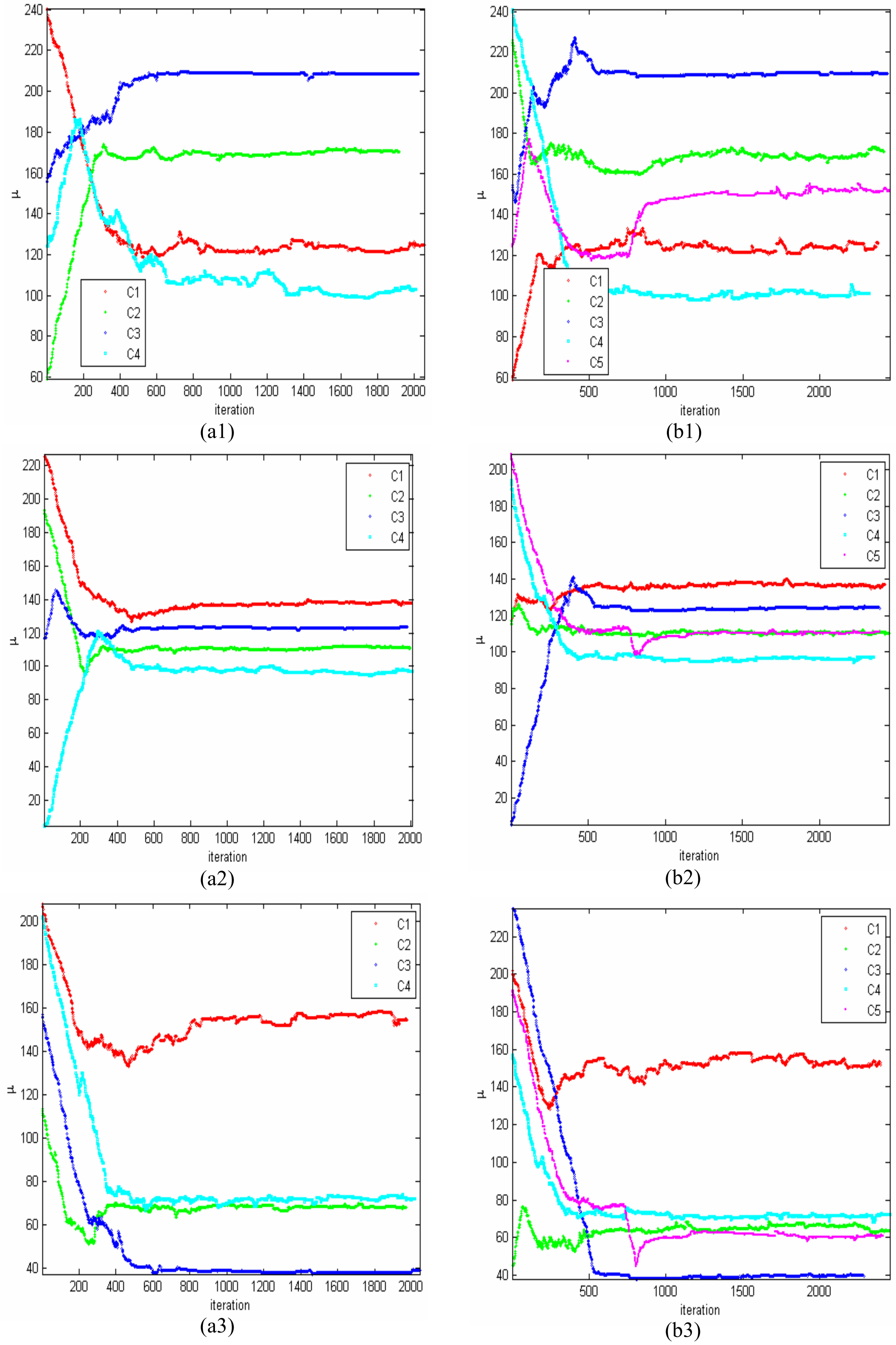
Table 6.2 gives the estimated parameters of the MGMRF model corresponding to the optimal segmentations for synthetic images shown in Fig. 6.2 where *ite* is the iteration at which the optimal segmentations are obtained under the MAP criterion, C1-C5 indicate the homogeneous regions labelled in Fig. 6.2. It is obvious that the estimated colour means (μ_r, μ_g, μ_b), β, τ and θ for the homogenous regions C1-C4 in Fig. 6.2 (a) and (b) are very close. Fig. 6.4 reveals the visual similarity of the estimated mean vectors to the colours for each homogeneous region in testing images.

Table 6.2 Estimated parameters used in Eq. (6.12)

Image	Region	μ_r	μ_g	μ_b	β	τ	θ_1	θ_2	θ_3	θ_4	*ite*
a	C1	124.74	137.82	154.78	0.81	4.18	0.78	0.74	0.89	0.81	7984
	C2	170.45	111.21	69.92	0.89	6.48	0.40	0.41	0.65	0.57	
	C3	208.41	123.36	39.14	0.80	1.00	0.49	0.47	0.71	0.67	
	C4	102.89	97.31	71.99	0.90	7.36	0.53	0.53	0.69	0.65	
d	C1	126.01	137.18	151.76	0.74	4.16	0.78	0.78	0.88	0.84	11977
	C2	171.39	110.51	63.74	0.85	7.07	0.57	0.56	0.72	0.68	
	C3	209.67	124.11	39.28	0.88	0.96	0.48	0.48	0.58	0.73	
	C4	101.36	96.98	72.12	0.95	9.48	0.46	0.47	0.67	0.60	
	C5	151.96	110.88	60.75	0.58	0.91	0.63	0.55	0.82	0.67	
c	C1	69.06	81.65	66.66	0.99	3.97	0.68	0.62	0.76	0.76	7993
	C2	109.02	106.99	78.28	0.97	1.25	0.79	0.80	0.87	0.86	
	C3	202.84	203.75	184.11	0.96	1.34	0.72	0.75	0.91	0.82	
	C4	73.13	62.31	57.39	0.99	2.11	0.78	0.79	0.91	0.82	
d	C1	72.46	89.88	82.47	0.96	2.95	0.84	0.85	0.97	0.85	7988
	C2	75.58	127.73	79.63	0.92	0.23	0.60	0.57	0.71	0.65	
	C3	163.55	165.53	170.40	0.90	0.21	0.65	0.57	0.71	0.66	
	C$	164.13	145.09	112.68	0.75	1.46	0.88	0.88	0.91	0.93	

Fig. 6.5 shows the changes of the estimated colour means where (a1)-(a3) and (b1)-(b3) are for the means of the red, green and blue components for testing images shown in Fig. 6.2 (a) and (b), respectively. Fig. 6.6 (a1)-(a3) and (b1)-(b3) do the same thing but for the testing images shown in Fig. 6.2 (c) and (d), respectively. It can

be seen that the means for all colour components converge to their stable values.



(a1)

(b1)

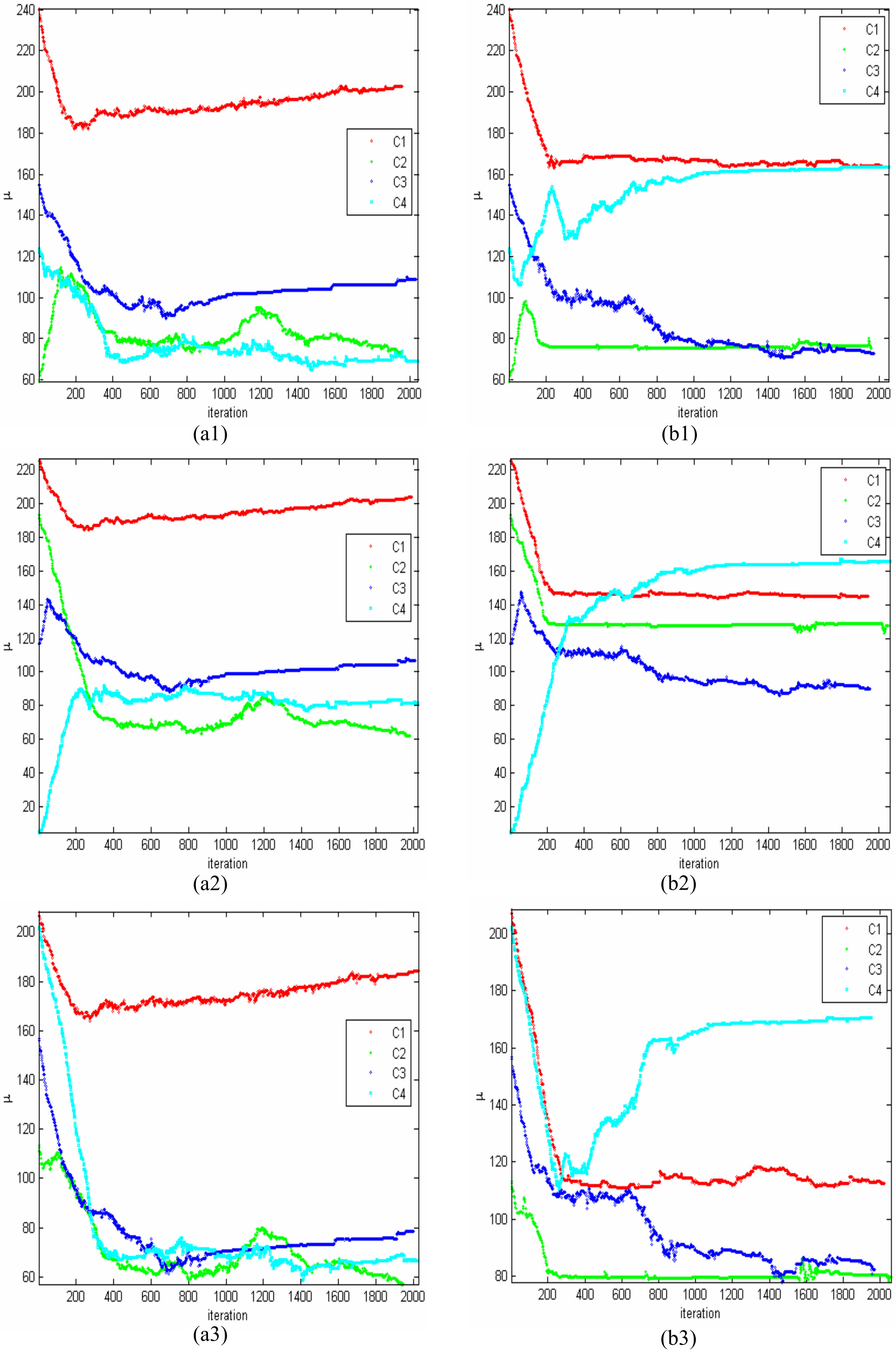
(a2)

(b2)

(a3)

(b3)

Fig. 6.5 Changes of estimated means for red, blue and green components for texture images Fig.6.2 (a)-(b).



(a1) (b1)

(a2) (b2)

(a3) (b3)

Fig. 6.6 Changes of estimated means for red, blue and green components for texture images Fig.6.2 (c)-(d).

Fig. 6.7 shows the outlines of segmented homogenous regions.

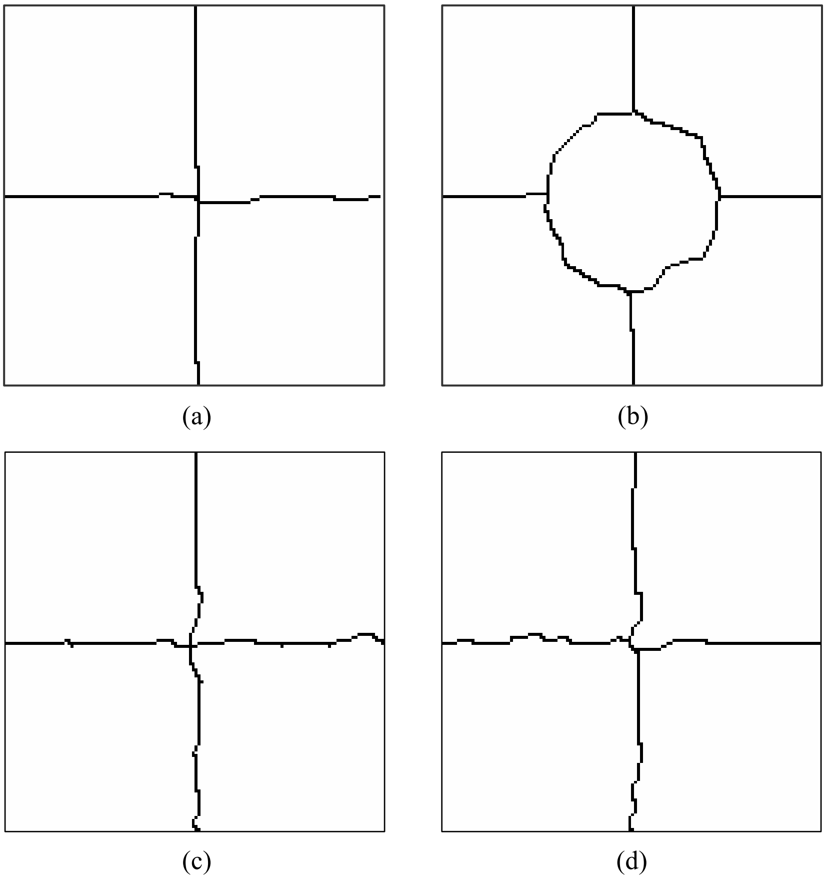
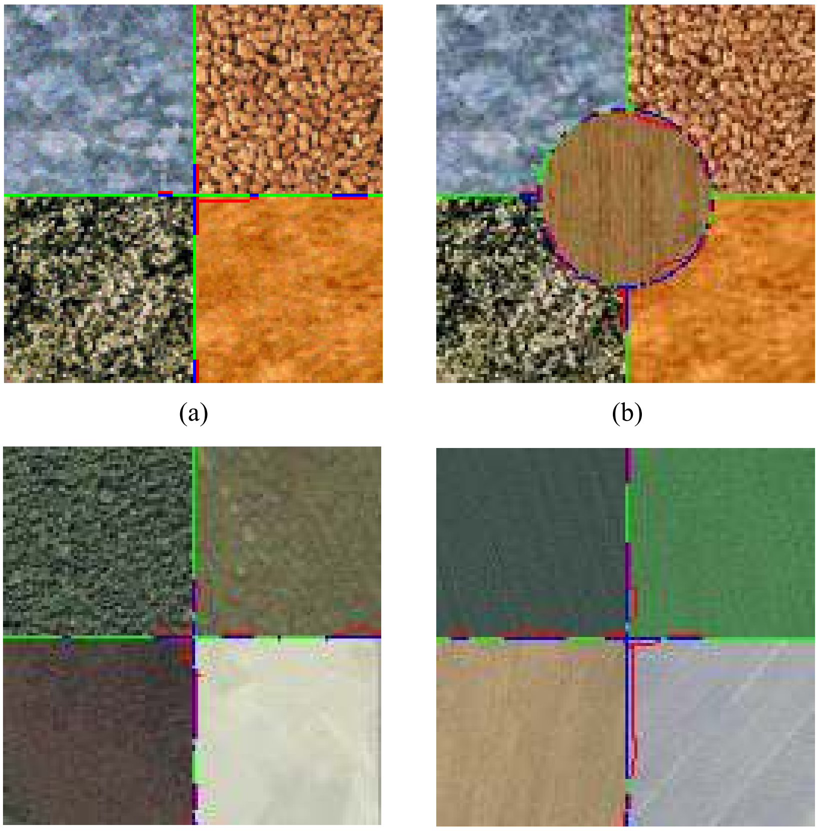


Fig. 6.7 Outlines of segmented regions.

To assess the accuracy of segmented homogenous regions, the outlines of segmented regions shown in Fig. 6.8 and the outlines of real homogenous regions are overlaid on the testing images, see Fig. 6.8. In Fig. 6.8, the blue and red lines are the outlines of real and segmented regions, respectively, while the green lines correspond to the overlapping part of both outlines. The Fig. 6.8 reveals that both outlines match very well visually.



(a) (b)

(c) (d)

Fig. 6.8 Outlines of segmented regions in red and real homogenous regions in blue overlaid on the original synthetic colour texture images, overlaying parts of both lines in green.

To perform accuracy estimates quantitatively, two different techniques are used to assess the accuracy of segmented regions. First of all, statistical measurements including error matrix, producer's accuracy, consumer's accuracy, overall and Kappa coefficient, are used for this purpose. Table 6.3 gives the error matrix

Table 6.3 Error matrix for the segmented results in Fig. 6.4

Image	Measure	C1	C2	C3	C4	C5	$\Sigma C_{..r}$
a	C1	4092	10	1	0	-	4103
	C2	0	4086	51	0		4137
	C3	0	0	4023	0		4023
	C4	4	0	21	4096		4121
	$\Sigma C_{.s.}$	4096	4096	4096	4096		16384
b	C1	3410	0	16	0	0	3426

	C2	0	3303	77	46	0	3426
	C3	6	1	2684	3	2	2696
	C4	6	0	26	3318	0	3350
	C5	35	0	60	0	3391	3486
	$\Sigma C_{.s.}$	3457	3304	2863	3367	3393	16384
c	C1	4058	17	38	0	-	4113
	C2	35	4071	0	6		4112
	C3	3	2	4043	8		4056
	C4	0	6	15	4082		4103
	$\Sigma C_{.s.}$	4096	4096	4096	4096		16384
d	C1	4.91	0	50	102	-	4243
	C2	1	4025	2	23		4051
	C3	0	60	4044	0		4104
	C4	5	11	0	3971		3986
	$\Sigma C_{.s.}$	4096	4096	4096	4096		16384

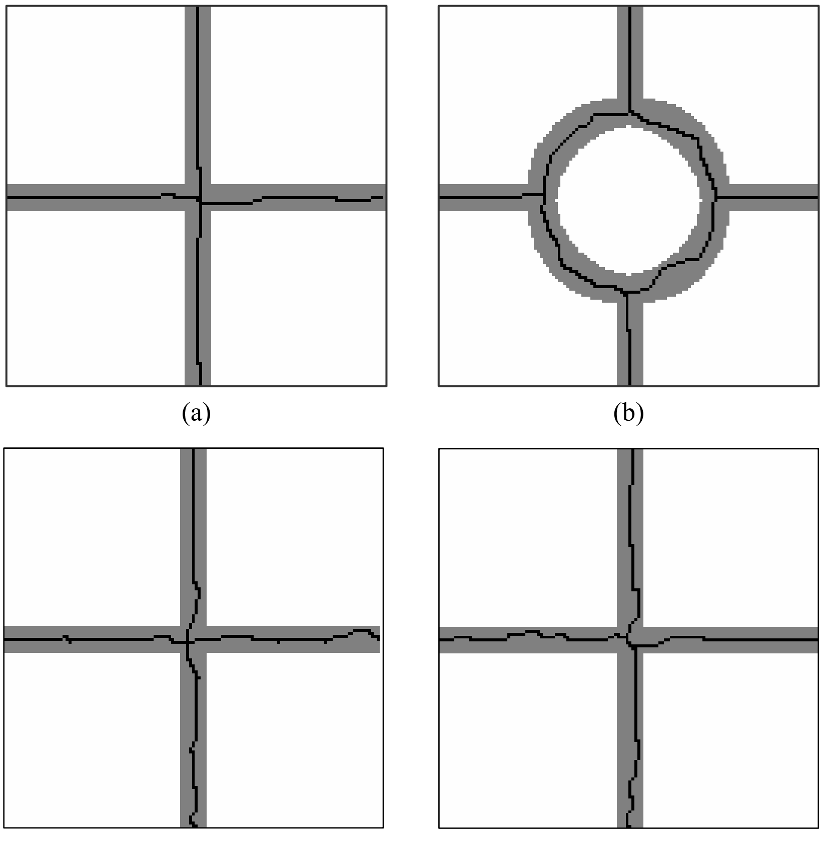
Based on the error matrix in Table 6.3, the producer's accuracy, user's accuracy, overall accuracy and Kappa coefficient can be calculated. Table 6.4 gives the values for the measures. From Table 6.4, it can been seen that both user's and producer's accuracy are similar for all real regions and all Kappa coefficients are up to 0.98, According to the general interpretation rules for thematic accuracy, the Kappa coefficients 0.81-1.00 can be interpreted as being almost perfect. As a result, the assessed accuracy demonstrates that the proposed algorithm for colour texture segmentation works well and is promising.

Table 6.4 Statistic measures for accuracy assessment.

		C1	C2	C3	C4	C5	Overall accuracy %	Kappa
a	Producer's accuracy %	99.90	99.76	98.22	100.00	-	99.47	0.99
	User's accuracy %	99.73	98.76	100.00	99.39			
b	Producer's accuracy %	98.75	99.97	93.75	98.54	99.94	98.30	0.98
	User's accuracy %	99.53	96.04	99.55	99.04	97.27		
c	Producer's accuracy %	99.07	99.39	98.71	99.66	-	99.21	0.98

	User’s accuracy %	98.66	99.00	99.68	99.49			
d	Producer’s accuracy %	99.88	98.27	98.73	96.94	-	98.46	0.98
	User’s accuracy %	96.42	99.36	98.54	99.62			

A buffer zone based scheme for accuracy assessment is used. Fig. 6.9 shows the extracted outlines lie on the buffer zones with 4 pixels around the real outlines for each homogenous region, in which the grey zones are buffer zones and the black lines are the outlines of the segmented homogenous regions. It can be observed that almost all extracted outlines lie on the buffer zones.



(a) (b)

(c) (d)

Fig. 6.9 Outlines delineating segmented regions in black and buffers around the real outlines of homogenous regions with 4 pixels in gray.

Table 6.5 gives the percentages of the extracted outlines on each buffer layer. It can be seen from the Table that over 75% of outlines of segmented regions are within the buffer zone with one pixel width around the outlines of real homogenous regions and that all outlines of segmented homogenous regions are on the buffer zone with three pixel width around the outlines of real homogenous regions

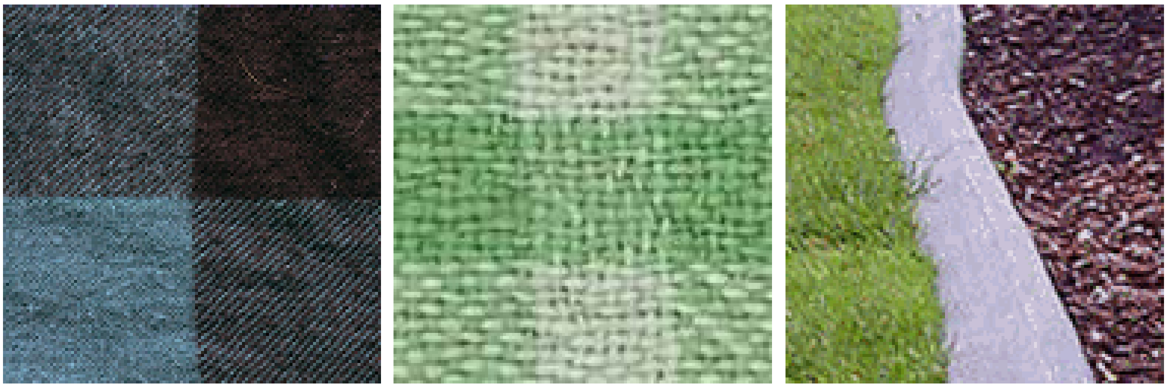
Table 6.5 Percents of the segmented outline matching the buffer zone of real outlines

Image	B_0	B_1 / ΣB_1	B_2 / ΣB_2	B_3 / ΣB_3
a	73.75	19.69/93.44	6.56/100.00	0.00/100.00
b	54.01	21.96/75.97	21.19/97.16	2.84/100.00
c	56.81	36.65/93.46	1.74/100.00	0.00/100.00
d	58.91	32.73/91.64	6.18/97.82	2.18/100.00
e	40.00	28.89/68.89	22.96/91.85	8.15/100.00

The results from the accuracy assessment scheme also point out that the proposed algorithm is efficient and effective for colour texture image segmentation.

B. Colour Texture Images

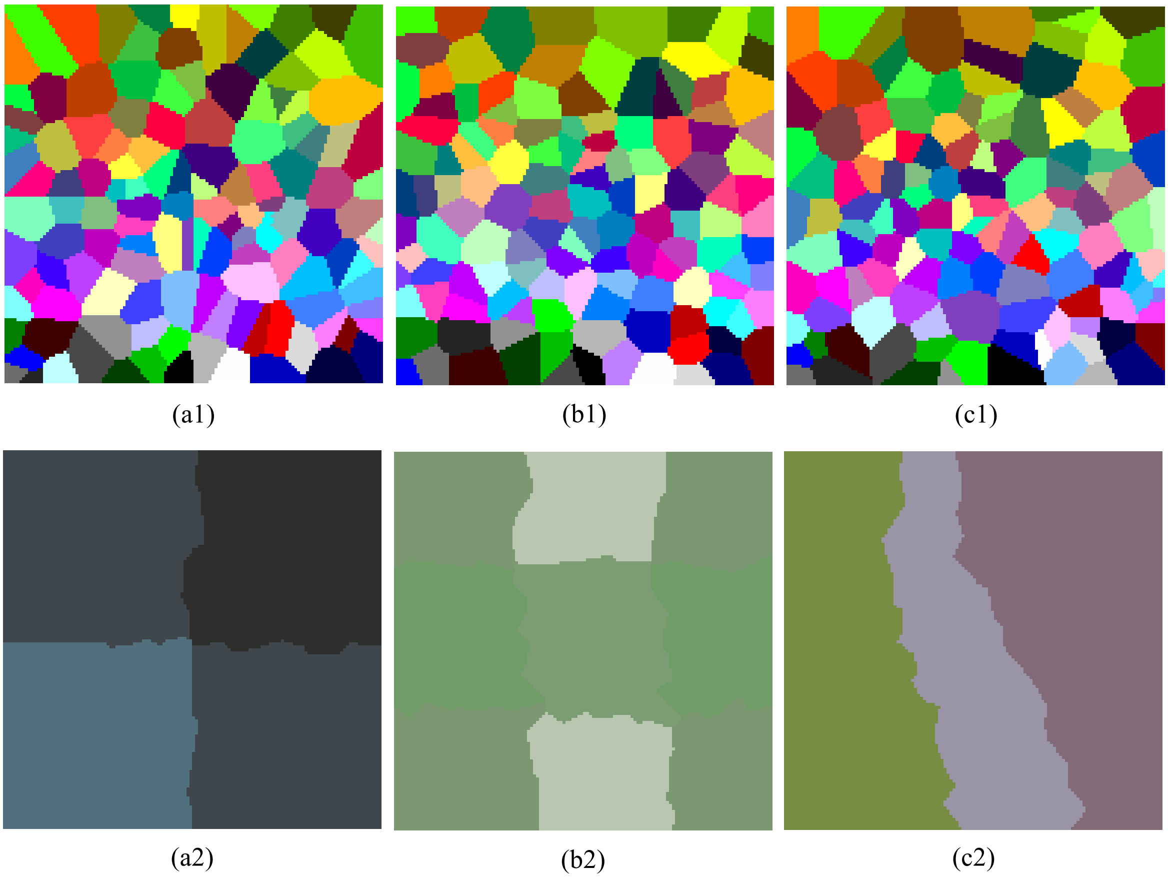
The proposed algorithm is tested by several colour texture images. Fig. 6.10 shows the testing images.



(a) (b) (c)

Fig. 6.10 Colour texture images for algorithm testing.

This experiment is carried out with 4,000, 6,000, and 8,000 iterations, 128 generating points for the partition of image domains, and 3, 4, 3 homogeneous regions respectively, for each testing colour texture image. Fig. 6.11 (a1)-(c1) show the optimal partitions obtained at 3,984th, 5,966th, and 7,989th out of 4,000, 6,000 and 8,000 iterations, respectively, and (a2)-(c2) shows the optimal segmentations corresponding to the optimal partitions in (a1)-(c2) where the segmented regions are represented by their estimated colour means.



(a1) (b1) (c1)

(a2) (b2) (c2)

Fig. 6.11 Optimal partitions and segmentations.

For the purposes of visual assessment accuracy, the outlines of the segmented region is overlaid on the original colour texture images, see Fig. 6.12, in which the outlines are represented in red. By visual inspection, the segmented regions fit their real regions well.

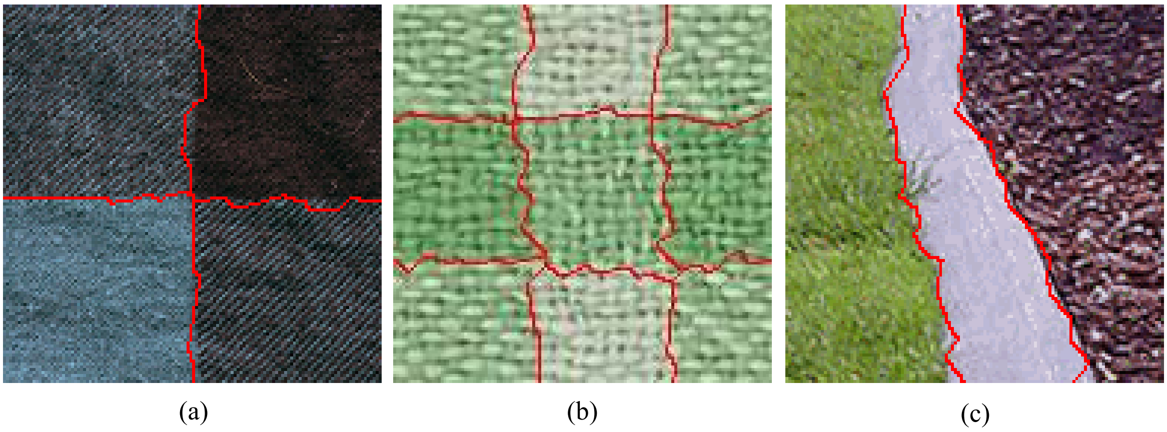


Fig. 6.12 Outlines of segmented regions overlaid on original images.

C. Colour IKONOS Images

To demonstrate the applicability of the proposed segmentation algorithm, it is explored with colour remote sensing images. Fig. 6.13 shows the colour IKONOS images with 1 *m* resolution, which reveal different colour textures representing different LULC classes. For example, in Fig. 6.13 (a) there are three LULC classes: grass, forest and farm land; Fig. 6.13 (b) includes grass, arable land, and uncovered land; Fig. 6.13(c) shows the forest with different species.

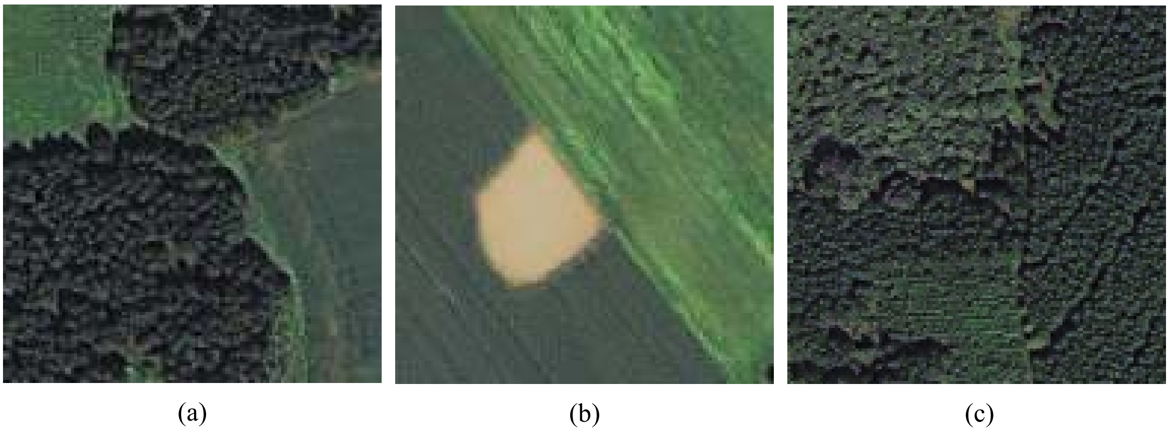
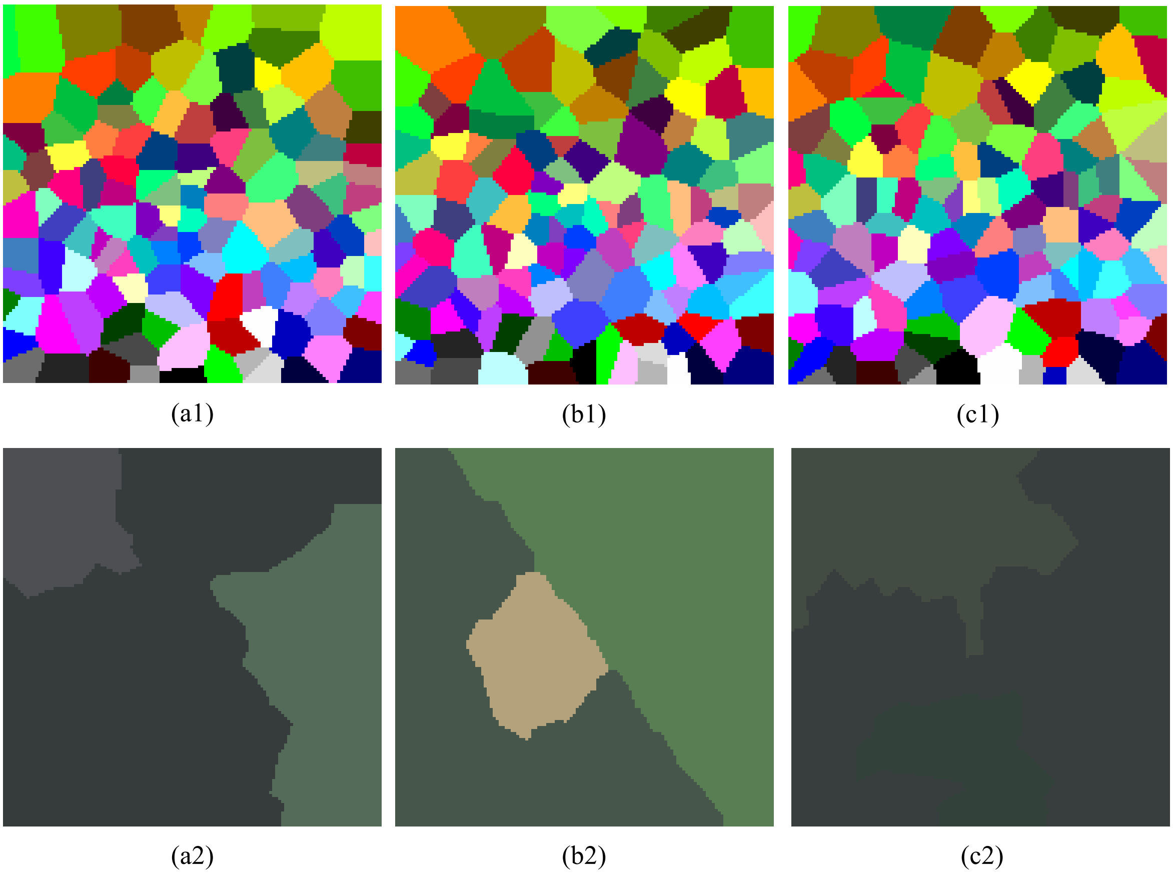

(a) (b) (c)

Fig. 6.13. Colour IKONOS images with 1 *m* resolution.

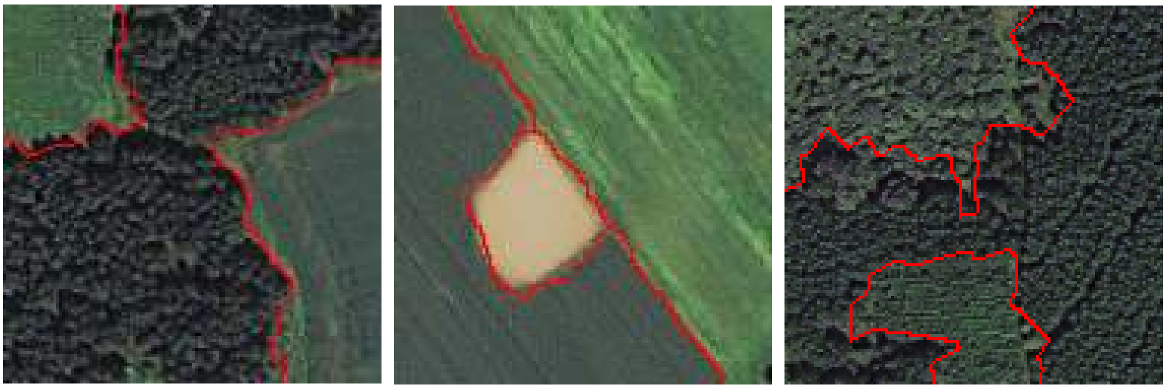
In this experiment, let $k = 3$, $m = 128$. Fig. 6.14 shows the optimal partitions and segmentations at the 3,994th, 3,887th and 3,856th out of 4,000 iterations. In Fig. 6.14 (a2)-(c2), the segmented regions are presented by the estimated mean colours.


(a1) (b1) (c1)

(a2) (b2) (c2)

Fig. 6.14 Optimal partitions and segmentations.

The outlines delineated by the segmented regions are also overlaid on the original images to demonstrate the accuracy of the results from the experiments, see Fig. 6.15. It can be concluded that the algorithm can visually captures the homogenous regions.



(a) (b) (c)

Fig. 6.15 Outlines (in red) of segmented regions overlaid on colour IKONOS images.

6.2 Texture Segmentation

6.2.1 Description of Algorithm for Texture Segmentation

A. Texture Model

Consider a texture image $\boldsymbol{Z} = \{Z_i;\, i = 1, \ldots, n\}$, Z_i is the intensity of the pixel at $(x_i, y_i) \in D$. Given two neighbouring pixels (x_i, y_i) and $(x_{i'}, y_{i'})$, the relationship of their intensities Z_i and $Z_{i'}$ can be characterized by a bivariate Gaussian distribution

$$p(\boldsymbol{Z}_{ii'}) = \frac{1}{(2\pi)\,|\,\boldsymbol{\Sigma}_{ii'}\,|}\exp\left\{-\frac{1}{2}\left[\boldsymbol{Z}_{ii'} - \boldsymbol{\mu}_{ii'}\right]^{T}(\boldsymbol{\Sigma}_{ii'})^{-1}\left[\boldsymbol{Z}_{ii'} - \boldsymbol{\mu}_{ii'}\right]\right\} \tag{6.16}$$

where $\boldsymbol{Z}_{ii'}$ = (Z_i, $Z_{i'}$) and $\boldsymbol{\mu}_{ii'}$ = $(\mu_i, \mu_{i'})^T$, μ_i and $\mu_{i'}$ are the means of Z_i and $Z_{i'}$, respectively and$\boldsymbol{\Sigma}_{ii'}$ is the covariance matrix

$$\boldsymbol{\Sigma}_{ii'} = \begin{bmatrix} \sigma_{ii} & \sigma_{ii'} \\ \sigma_{i'i} & \sigma_{i'i'} \end{bmatrix} \tag{6.17}$$

where the diagonal elements of the covariance matrix σ_{ii} and $\sigma_{i'i'}$ are the variances of Z_i and $Z_{i'}$. The correlation coefficient between Z_i and $Z_{i'}$ is defined as

$$\rho_{ii'} = \frac{\sigma_{ii'}}{\sqrt{\sigma_{ii}}\sqrt{\sigma_{i'i'}}} \tag{6.18}$$

This measure of association is symmetric in Z_i and $Z_{i'}$, that is, $\rho_{ii'} = \rho_{i'i}$. Under the partition in Eq. (3.4), $D = \{D_j, j = 1, \ldots, m\}$, the intensities of the pixels located on the polygon D_j, $\boldsymbol{Z}_j = \{Z_i\,(x_i, y_i), (x_i, y_i) \in P_j\}$, can be modeled by a BGMRF to characterize the regional structure of the sub-region D_j. To this end, assume that (1) Z_i's for all (x_i, y_i) $\in D_j$ have the same mean dependent on the label L_j, that is, μ_{Lj}; (2) the covariance matrix for a pair of neighbour pixels in D_j is determined by the orientation δ of the neighbouring pixel pair, where δ's are the same as ones in Fig. 5.1.

As a result, Eq. (6.16) can be rewritten as

$$p(\boldsymbol{Z}_{ii'} \mid L_i, \delta)$$

$$= \frac{1}{(2\pi) \mid \boldsymbol{P}_{ii'} \mid^{1/2} \sigma_{L_j}} \exp\left\{ -\frac{1}{2} \left[\boldsymbol{Z}_{ii'} - (\mu_{Lj}, \mu_{Lj}) \right]^T (\boldsymbol{P}_{ii'} \sigma_{L_j}{}^2)^{-1} \left[\boldsymbol{Z}_{ii'} - (\mu_{Lj}, \mu_{Lj}) \right] \right\} \tag{6.19}$$

where $\boldsymbol{P}_{ii'}$ is the correlation coefficient matrix and is expressed as

$$\boldsymbol{P}_{ii'} = \begin{bmatrix} 1 & \rho_{L_i\delta} \\ \rho_{L_i\delta} & 1 \end{bmatrix} \tag{6.20}$$

On the assumption that the $p(Z_i, Z_{i'})$'s for all neighbouring pixel pairs in D_j are independent, their joint probability distribution conditional on the label L_i can be obtained by

$$p(\boldsymbol{Z}_j \mid L_j) = \prod_{\substack{\delta_{ii'} \in \{1,2,3,4\} \\ (x_i, y_i), (x_{i'}, y_{i'}) \in D_j}} p(\boldsymbol{Z}_i, \boldsymbol{Z}_{i'}) \tag{6.21}$$

where $\delta_{ii'}$ indicates the direction of neighbour pixel i and i'.

For simplicity, assume the prior distributions for μ, and ρ to be independently truncated Gaussian distributions and the prior distribution for σ is a Gamma distribution, that is, μ_{L_j} ~ truncated $N(\mu_\mu, \sigma_\mu)$, $\rho_{L1,2,3,4}$ ~ truncated $N(\mu_\rho, \sigma_\rho)$, and σ_{L_j} ~ $\Gamma(\alpha_\sigma, \beta_\sigma)$. Thus, the join prior distributions for $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$ and $\boldsymbol{\rho}$ can be written

$$p(\boldsymbol{\mu}) = \prod_{l=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_{\mu}} \exp\left\{ -\frac{(\mu_l - \mu_{\mu})^2}{2\sigma_{\mu}^{\ 2}} \right\} \tag{6.22}$$

$$p(\boldsymbol{\sigma}) = \prod_{l=1}^{k} \frac{1}{\Gamma(\sigma_l)\beta_{\sigma}^{\ \alpha_{\sigma}}} (\sigma_l)^{\alpha_{\sigma}-1} \exp\left\{ -\frac{\sigma_l}{\beta_{\sigma}} \right\} \tag{6.23}$$

$$p(\boldsymbol{\rho}) = \prod_{l=1}^{k} \prod_{\rho_{l\delta} \in \{\rho_{l1}, \rho_{l2}, \rho_{l3}, \rho_{l4}\}} \frac{1}{\sqrt{2\pi}\sigma_{\rho}} \exp\left\{ -\frac{(\rho_{l\delta} - \mu_{\rho})^2}{2\sigma_{\rho}^{\ 2}} \right\} \tag{6.24}$$

By the Bayesian paradigm, the posterior distribution of ***L***, ***μ***, ***σ***, ***ρ***, and ***G*** given ***Z*** can be written as,

$$p(\boldsymbol{L}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\rho}, \boldsymbol{G} \mid \boldsymbol{Z}) \propto p(\boldsymbol{Z} \mid \boldsymbol{L}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\rho}, \boldsymbol{G}) p(\boldsymbol{\mu}) p(\boldsymbol{\sigma}) p(\boldsymbol{\rho}) p(L) p(G) =$$

$$\prod_{j=1}^{m} \prod_{\substack{\delta_{ii'} \in \{1,2,3,4\} \\ (u_i, v_i), (u_{i'}, v_{i'}) \in P_j}} \frac{1}{(2\pi)^{3/2} \mid \boldsymbol{P}_{L_j \delta_{ii'}} \mid^{1/2} \sigma_{L_j}} \exp\left\{ -\frac{1}{2} \left[(Z_i, Z_{i'}) - (\mu_{L_j}, \mu_{L_j}) \right]^T (\boldsymbol{P}_{L_j \delta_{ii'}} \sigma_{L_j}^{\ 2})^{-1} \left[(Z_i, Z_{i'}) - (\mu_{L_j}, \mu_{L_j}) \right] \right\} \times$$
$$\prod_{l=1}^{k} \frac{1}{\sqrt{2\pi}\sigma_{\mu}} \exp\left\{ -\frac{(\mu_{\gamma l} - \mu_{\mu})^2}{2\sigma_{\mu}^{\ 2}} \right\} \times \prod_{l=1}^{k} \frac{1}{\Gamma(\alpha_{\sigma})\beta_{\sigma}^{\ \alpha_{\sigma}}} (\sigma_l)^{\alpha_{\sigma}-1} \exp\left\{ -\frac{\sigma_l}{\beta_{\sigma}} \right\} \times$$
$$\prod_{l=1}^{k} \prod_{\theta_{l\delta} \in \{\theta_{l1}, \theta_{l2}, \theta_{l3}, \theta_{l4}\}} \frac{1}{\sqrt{2\pi}\sigma_{\rho}} \exp\left\{ -\frac{(\rho_{l\delta} - \mu_{\rho})^2}{2\sigma_{\rho}^{\ 2}} \right\} \times \prod_{j=1}^{m} \frac{\exp\left(c \sum_{D_{j'} \in ND_j} t(L_j, L_{j'}) \right)}{\sum_{l=1}^{k} \exp\left(c \sum_{D_{j'} \in ND_j} t(l, L_{j'}) \right)} \times \left(\frac{1}{\mid D \mid} \right)^m$$

(6.25)

To simulate the posterior distribution given in Eq. (6.25), the Metropolis-Hastings algorithm based scheme is designed. The moves proposed in the scheme include (a) updating the distribution parameters in $\boldsymbol{\Psi} = \{\boldsymbol{u}, \boldsymbol{\sigma}, \boldsymbol{\rho}\}$; (b) updating the labels in $\boldsymbol{L} = \{L_j\,; j = 1, \ldots m\}$; (c) updating the location of generating points in $\boldsymbol{G} = \{(u_j, v_j); j = 1, \ldots m\}$.

Move 1: Updating distribution parameters. Draw a proposal for the updating parameter, $\Psi_l^* \sim N(\Psi_l, \varepsilon)$ where $\varepsilon = \varepsilon_{\mu_l}, \varepsilon_{\sigma_l}, \varepsilon_{\rho}$ in terms of the type of Ψ_l. The acceptance probability for the proposal can be calculated by its marginal distribution,

$$r_{\Psi}(\Psi_l, \Psi_l^*) = \min\left\{1, \frac{p(\boldsymbol{Z} \mid \Psi_l^*)p(\Psi_l^*)}{p(\boldsymbol{Z} \mid \Psi_l)p(\Psi_l)}\right\} \tag{6.26}$$

Move 2: Updating texture class labels. Draw a proposal L_j^* from $\{1,\ldots, k\}$ conditional on $L_j^* \neq L_j$; The acceptance probability for the proposal can be calculated by its marginal distribution

$$r_L(L_j, L_j^*) = \min\left\{1, \frac{p(\boldsymbol{Z} \mid L_j^*)p(L_j^*)}{p(\boldsymbol{Z} \mid L_j)p(L_j)}\right\} \tag{6.27}$$

Move 3: Updating locations of generating point. Giving the proposal for the generating point by uniformly drawing a point from D_j, that is, $G_j^* = (u_j^*, v_j^*) \sim U(D_j)$. The acceptance probability for the proposal can be calculated by its marginal distribution

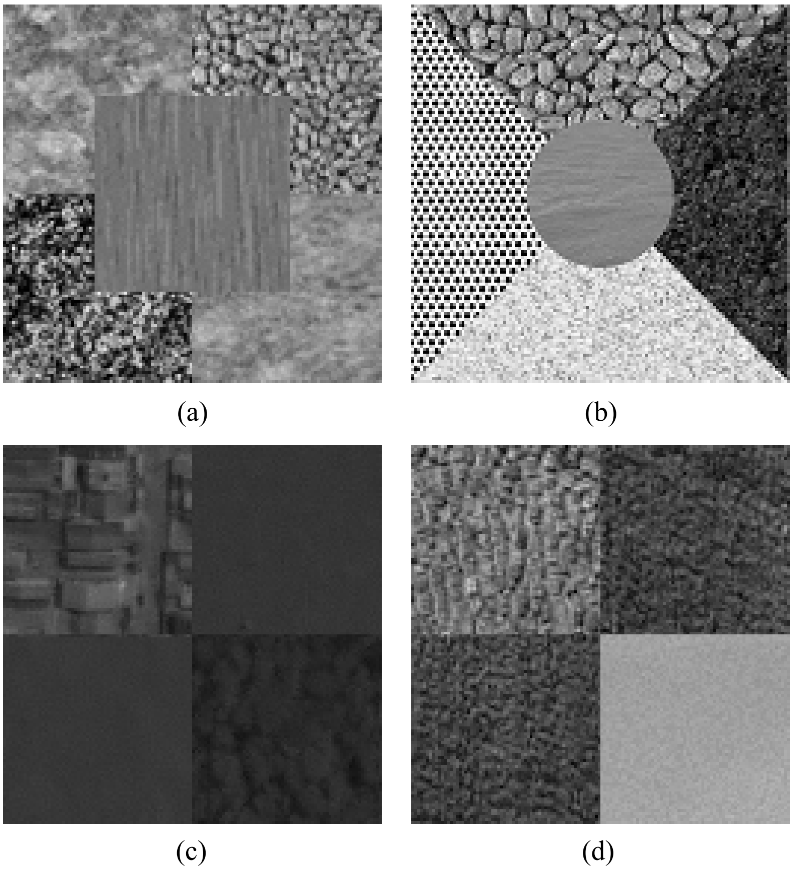
$$r_G(G_j, G_j^*) = \min\left\{1, \frac{p(\boldsymbol{Z} \mid G_j^*)p(G_j^*)}{p(\boldsymbol{Z} \mid G_j)p(G_j)}\right\} \tag{6.28}$$

The MAP criterion described in Section 3.6 is used to obtain the optimal segmentation.

6.2.2 Experimental Results on Texture Segmentation

A. Synthetic Texture Images

Fig. 6.16 show the synthetic images with size 128 × 128 pixels, and 5, 5, 4 and 4


(a)

(b)

(c)

(d)

Fig. 6.16 Synthetic texture images.

texture classes, respectively.

The constants in Eq. (6.25) for this experiment are listed in Table 6.6.

Table 6.6 Constants in Eq. (6.25).

m	c	ξ	κ	ω	ν
128	1.0	128	64	0.5	0.25
γ	δ	ε_μ	ε_θ	ε_τ	T
2.0	1.0	1.0	1/64	1/256	6000

Fig. 6.17 shows the optimal partitions obtained at 5,889th, 5,906th, 5,987th and 5,934th out of 6,000 iterations by the MAP estimation. Fig. 6.18 gives the optimal segmentations corresponding to the optimal partitions shown in Fig. 6.17, in which

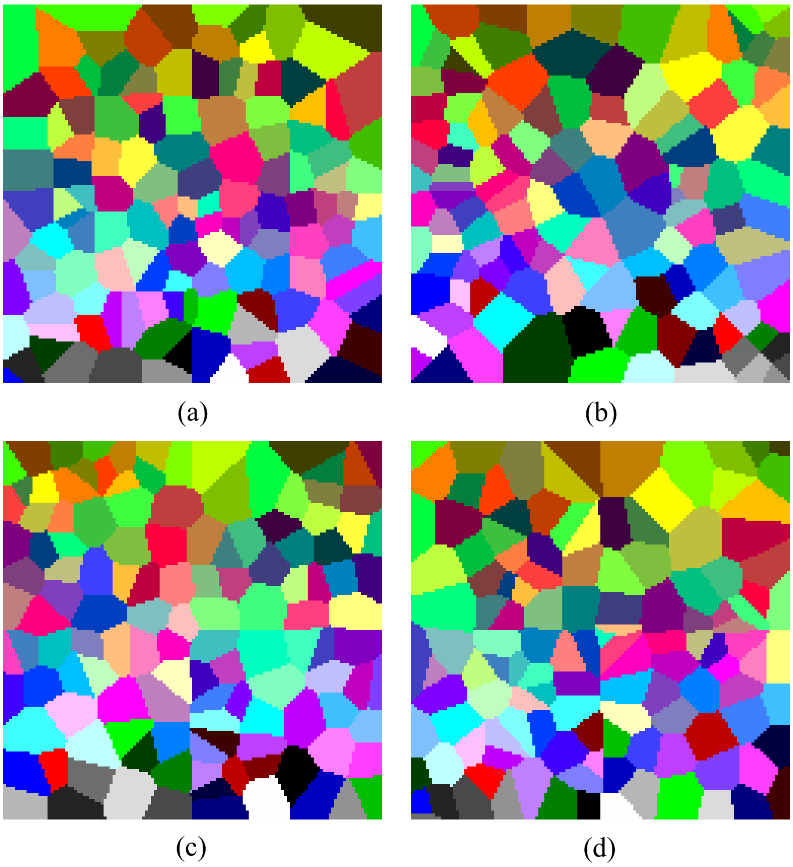
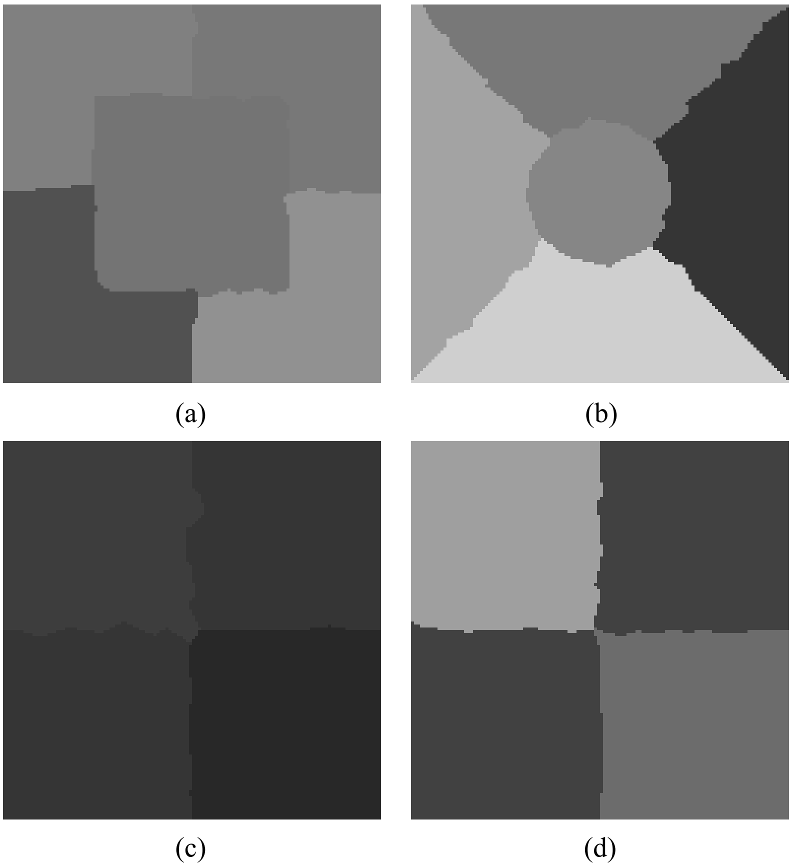


Fig. 6.17 Optimal partitions obtained during 6,000 iterations by MAP estimation.

the homogenous textures are represented by their estimated colour means.



(a) (b)

(c) (d)

Fig.6.18 Optimal partitions corresponding to the optimal partitions shown in Fig. 6.17.

To visually evaluate the segmentation results, the outlines of segmented texture regions are delineated, see Fig. 6.19, and overlaid on the synthetic texture images, see Fig. 6.20. In Fig. 6.20, the blue lines are the outlines of real texture regions and the red lines represent the outlines of the segmented regions, while the green ones correspond to the overlying parts of both outlines. It can be observed that both the outlines of segmented and real homogeneous regions are matched well.

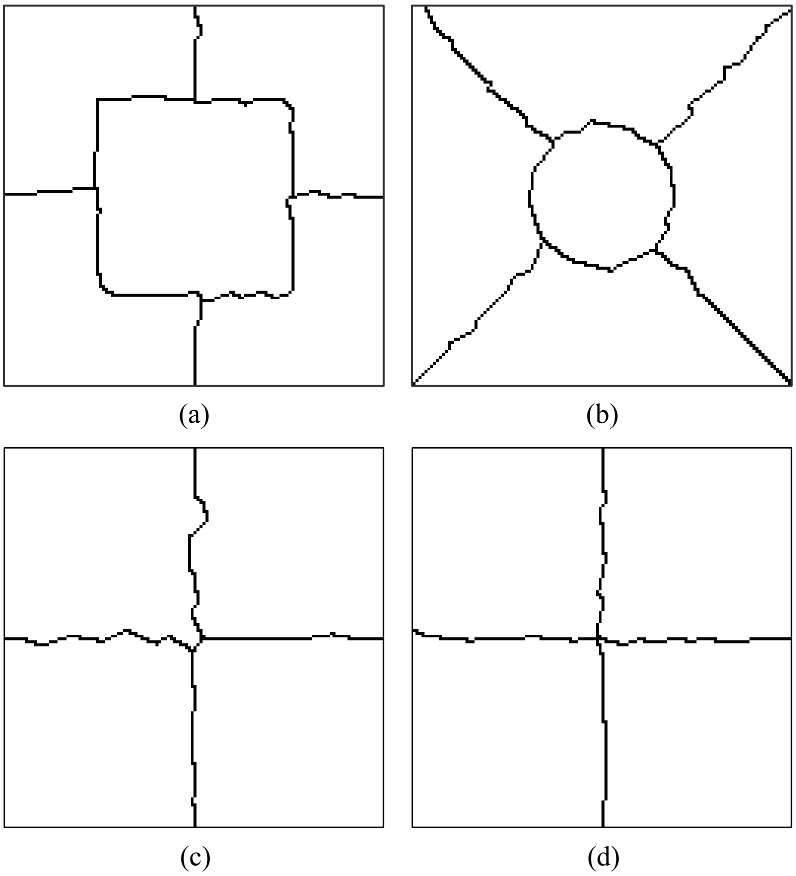


Fig. 6.19 Outlines of segmented texture regions show in Fig. 6.18.

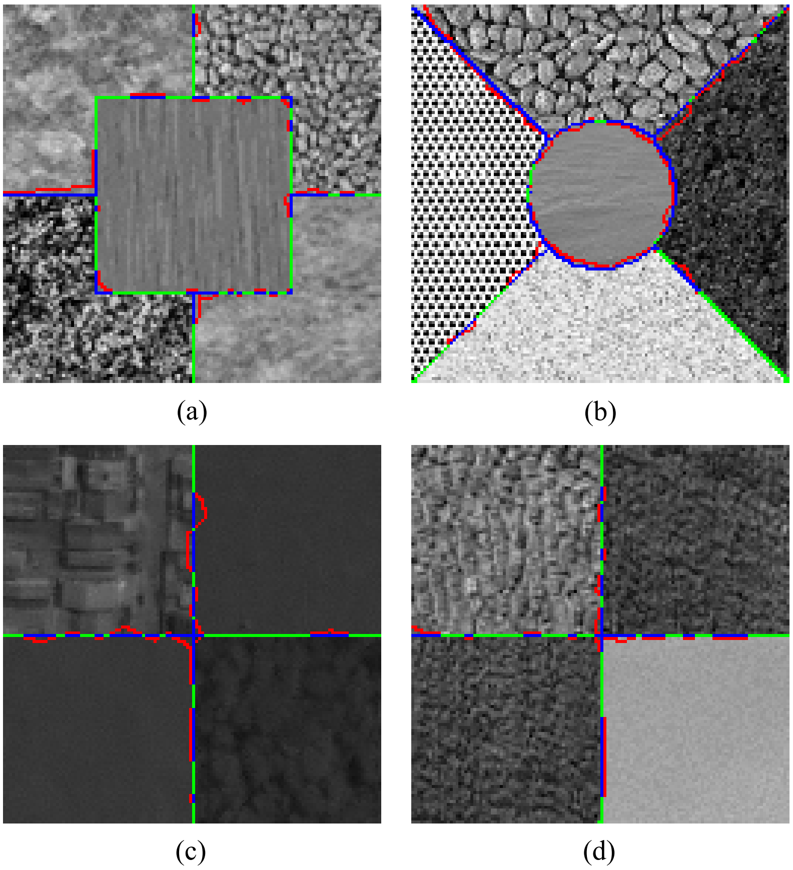
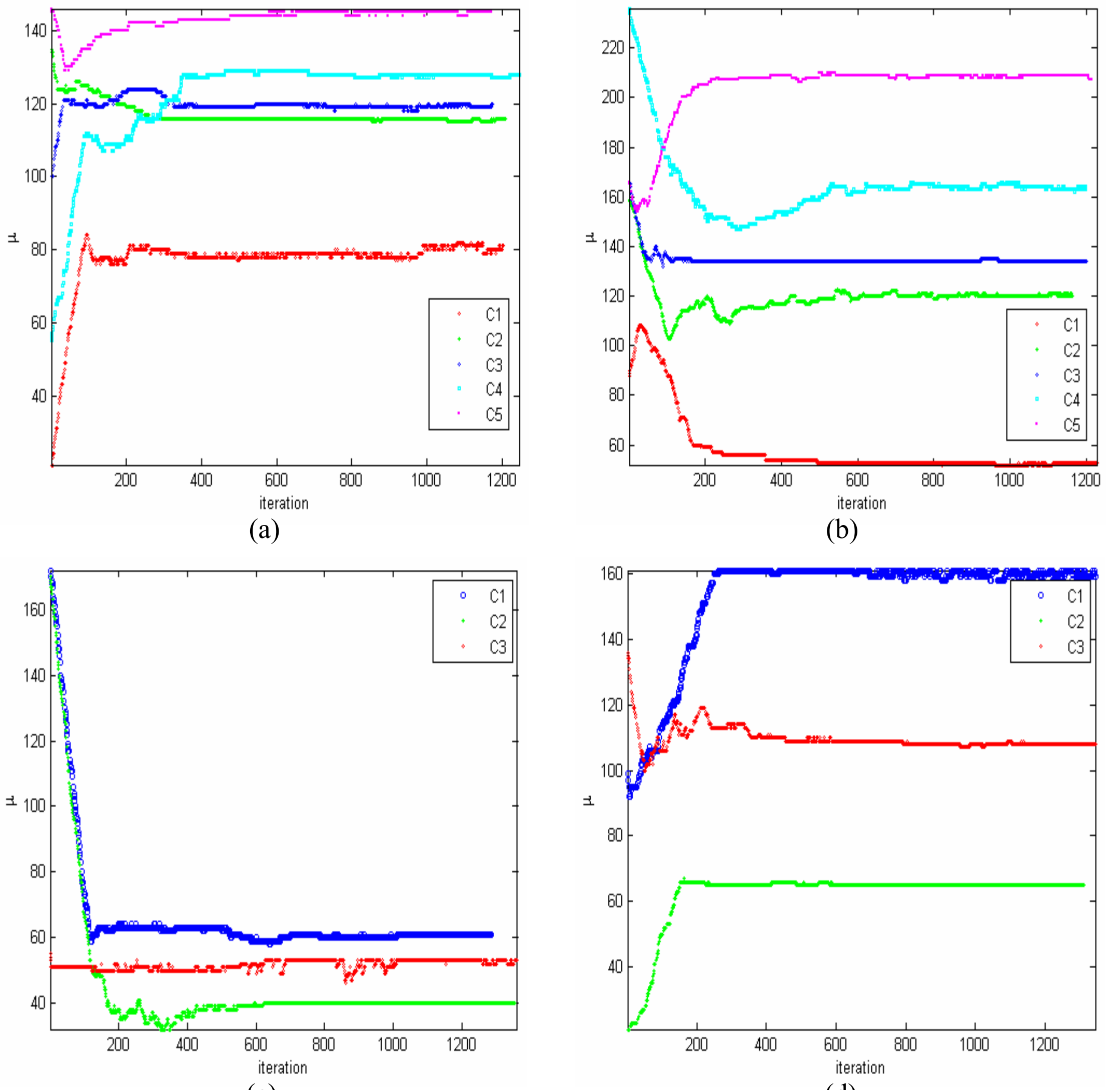


Fig. 6.20 Overlaying the outlines of segmented texture regions shown in Fig. 6.19 on testing images.

Fig. 6.21 shows the changes of the estimated intensity means during 6,000 iterations. It is obvious that these estimated values converge to their stable values finally.



(a) (b) (c) (d)

Fig.6.21 Changes of the estimated intensity means during 6000 iterations.

Table 6.7 lists the optimal estimated values of model parameters including intensity mean (μ), standard deviation of intensity (σ), and the correction coefficient at different directions, as well as the iteration at which they are obtained.

Table 6.7 Estimated model parameters in Eq. (6.25).

Image	Region	μ	τ	θ_1	θ_2	θ_3	θ_3	*ite*
a	C1	65	2.52	0.25	0.13	0.36	0.34	3987
	C2	114	2.90	0.27	0.27	0.49	0.47	
	C3	130	1.43	0.58	0.53	0.68	0.75	
	C4	144	1.00	0.55	0.52	0.62	0.74	
b	C1	81	3.02	0.26	0.19	0.40	0.46	5946
	C2	116	0.51	0.18	0.18	0.81	0.25	
	C3	120	2.67	0.20	0.22	0.47	0.43	
	C4	128	1.26	0.52	0.49	0.65	0.69	
	C5	145	1.08	0.56	0.56	0.67	0.75	
c	C1	53	1.68	0.15	0.05	0.33	0.23	5963
	C2	120	3.08	0.28	0.25	0.50	0.37	
	C3	134	0.43	0.36	0.21	0.25	0.76	
	C4	163	3.49	0.00	0.00	0.11	0.04	
	C5	207	1.50	0.03	0.09	0.07	0.15	
d	C1	61	0.62	0.60	0.68	0.79	0.74	5912
	C2	53	0.02	0.42	0.42	0.42	0.39	
	C3	40	0.15	0.56	0.61	0.75	0.64	
e	C1	159	0.14	0.34	0.32	0.40	0.40	5955
	C2	65	1.32	0.18	0.2	0.40	0.43	
	C3	108	3.42	0.32	0.23	0.56	0.54	

For assessing accuracy quantitatively, two evaluation techniques are employed: statistical and buffer zone based assessments. First of all, statistical measurements including error matrix, producer's accuracy, consumer's accuracy, overall and Kappa coefficient, are used for this purpose. Table 6.8 gives the error matrices for the segmented results shown in Fig. 6.18.

Table 6.8 Error matrices for the optimal segmentations shown in Fig. 6.18

Image	Region	$C_{1.}$	$C_{2.}$	$C_{3.}$	$C_{4.}$	$C_{5.}$	$\sum C_{+l}$
a	$C_{.1}$	3007	19	0	59	20	3105

	$C_{.2}$	0	4302	0	23	17	4342
	$C_{.3}$	0	21	2978	0	0	2999
	$C_{.4}$	0	0	12	2925	0	2937
	$C_{.5}$	0	14	17	0	2970	3001
	C_{l+}	3007	4356	3007	3007	3007	16384
b	$C_{.1}$	3556	7	34	0	2	3599
	$C_{.2}$	64	3480	34	39	0	3617
	$C_{.3}$	0	1	1846	0	0	1847
	$C_{.4}$	0	27	19	3542	7	3595
	$C_{.5}$	13	0	51	28	3634	3726
	C_{l+}	3633	3515	1984	3609	3643	16384
c	$C_{.1}$	4045	5	90	-	-	4140
	$C_{.2}$	0	4091	43			4134
	$C_{.3}$	51	0	8059			8110
	C_{l+}	4096	4096	8192			16384
d	$C_{.1}$	4030	7	1	-	-	4038
	$C_{.2}$	66	8170	43			8279
	$C_{.3}$	0	15	4052			4067
	C_{l+}	4096	8192	4096			16384

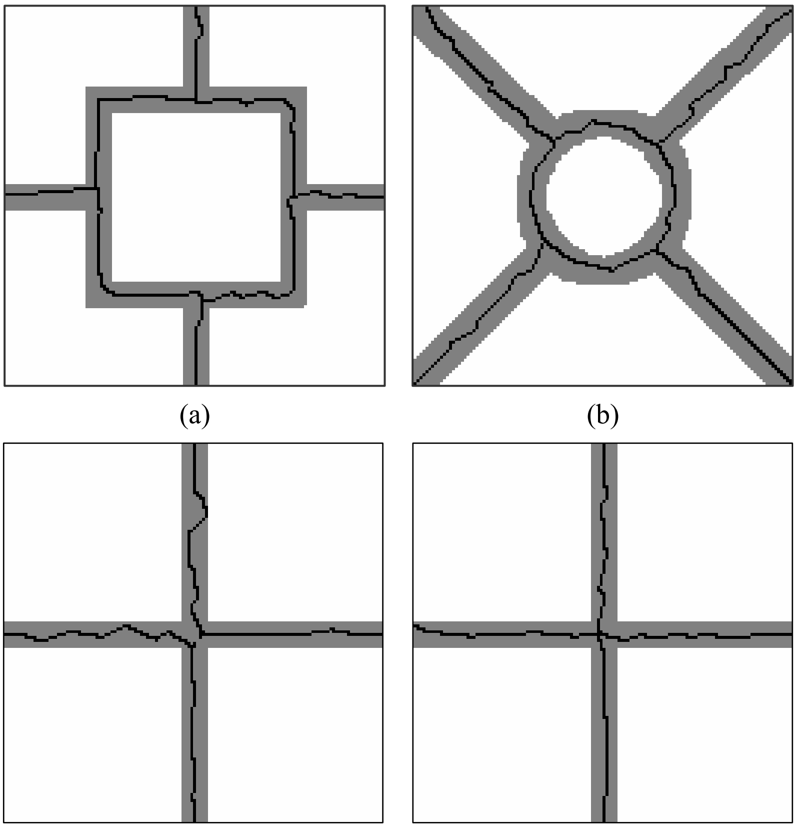
Based on the error matrices, the statistical measurements can be calculated, including Produce's accuracy, User's accuracy, overall accuracy and Kappa coefficient. Table 6.9 lists those values. The measurements indicate that the proposed algorithm segments texture images with a high accuracy.

Table 6.9 Statistical measures for testing images in Fig. 6.16.

Image	Region	Product's accuracy (%)	User's accuracy (%)	Overall accuracy (%)	Kappa
a	C1	100.00	96.84	98.77	0.985
	C2	98.76	99.08		
	C3	99.04	99.30		
	C4	97.27	99.59		
	C5	98.77	98.97		
b	C1	97.88	98.80	98.01	0.975
	C2	99.00	96.21		
	C3	93.04	99.95		
	C4	98.14	98.53		

	C5	99.75	97.53		
c	C1	98.75	97.71	98.85	0.982
	C2	99.88	98.96		
	C3	98.38	99.37		
d	C1	98.39	99.80	99.19	0.987
	C2	99.73	98.68		
	C3	98.93	99.63		

The buffer zone based assessment technique is also used. Fig. 6.22 shows the extracted outlines of segmented texture regions in black and the buffer zones with width of 4 pixels in grey. By visual inspection, the extracted outlines are completely laid on the buffer zones.



(a) (b)

(c) (d)

Fig. 6.22 Outlines of segmented regions shown in Fig. 6.18and buffer zones with the width of 4 pixels.

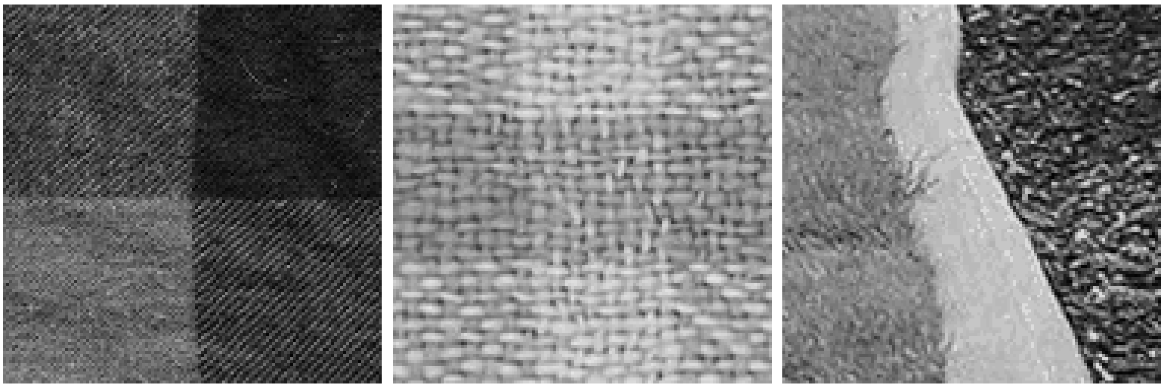
Table 6.10 gives the percentages of the extracted outlines on each buffer layer. As shown in Table 6.10 in the worst case more than 95% of the extracted outlines are with an error of less than 2 pixels and nearly 99% of the extracted outlines are with an error of less than 3 pixels.

Table 6.10 Percents of the extracted outlines on each buffer layer.

Image	B0	B1/∑1	B2/∑2	B3/∑3	B4/∑4
a	61.29	27.30/88.6	9.93/98.30	1.75/100.00	0.00/100.0
b	45.71	45.49/91.2	8.80/100.0	0.00/100.00	0.00/100.0
c	48.41	34.28/82.69	12.72/95.41	3.18/98.59	1.41/100.00
d	54.18	40.00/94.18	5.09/99.27	0.073/100.0	0.00/100.00

B. Real Texture Images

Three texture images are used to test the proposed algorithm for texture segmentation, see Fig. 6.23. The textures in these images illustrate the changes of either structures, or intensity, or both of them.



(a) (b) (c)

Fig. 6.23 Texture images for testing proposed algorithm.

Fig. 6.24 shows the optimal partitions and optimal segmentations which are obtained after 4,000 iterations and under the MAP scheme. In Fig. 6.24 (a2)-(c2), the segmented regions are represented by the estimated means for corresponding homogeneous regions.

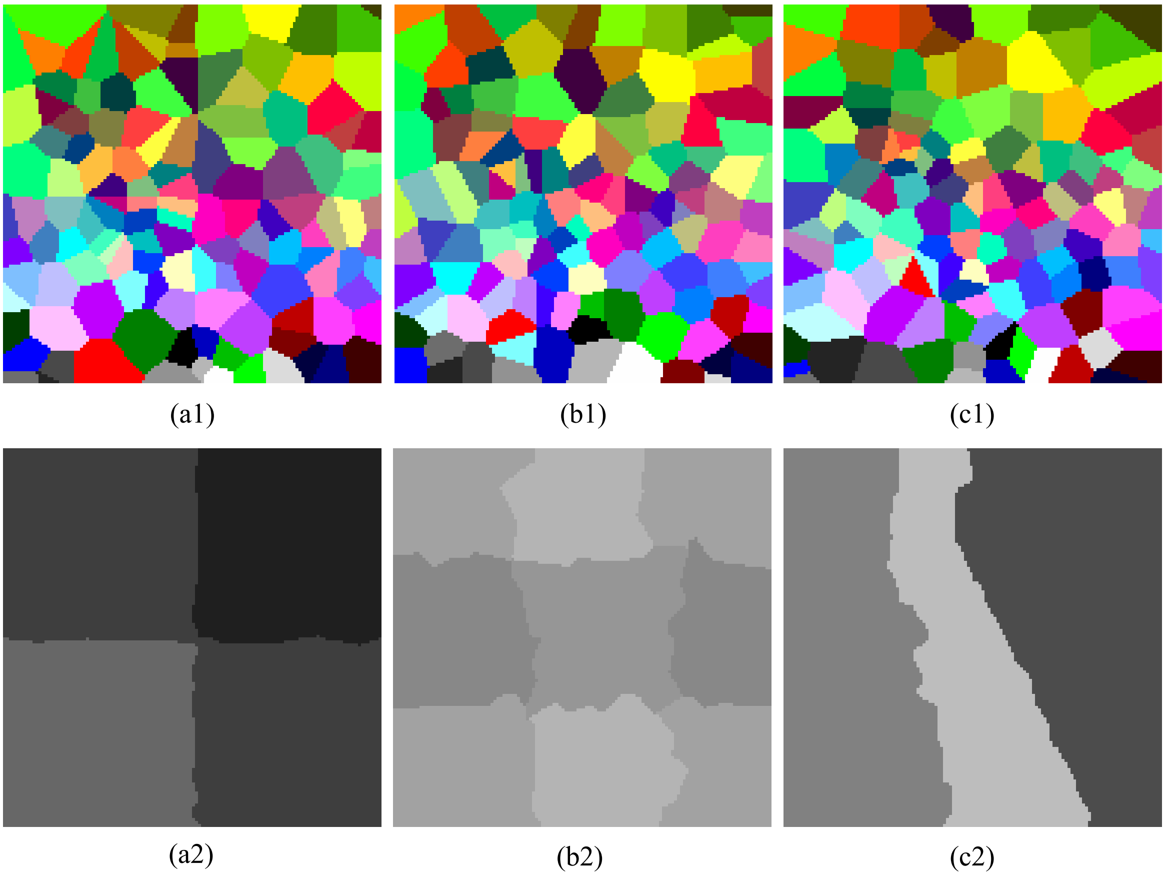
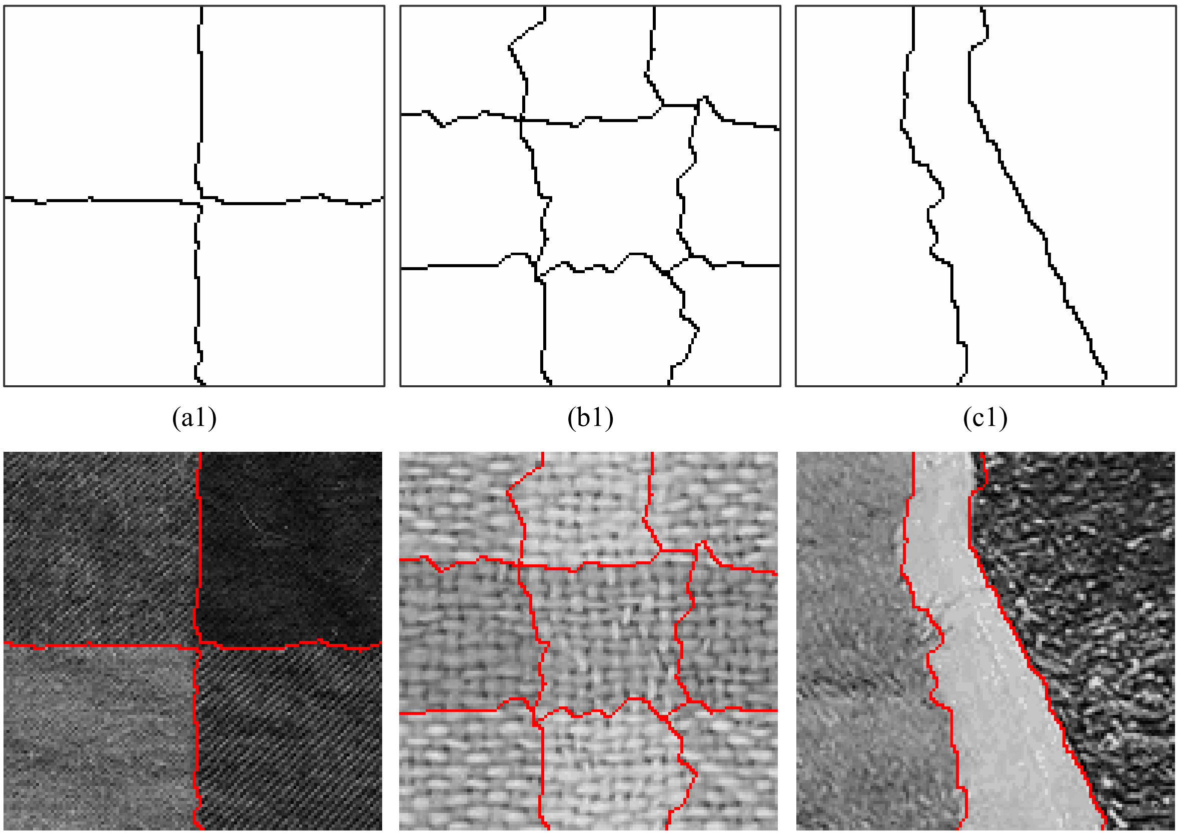


Fig.6.24 Optimal partitions and segmentations during 4,000 iterations.

Fig. 6.25 (a1)-(a3) shows the outlines of the segmented textured regions in Fig. 6. 24. In Fig. 6.25 (b1)-(b3), the outlines (in red) are overlaid on the original texture images shown in Fig. 6.23. It can be observed that the extracted outlines match the boundaries of the textured regions well.



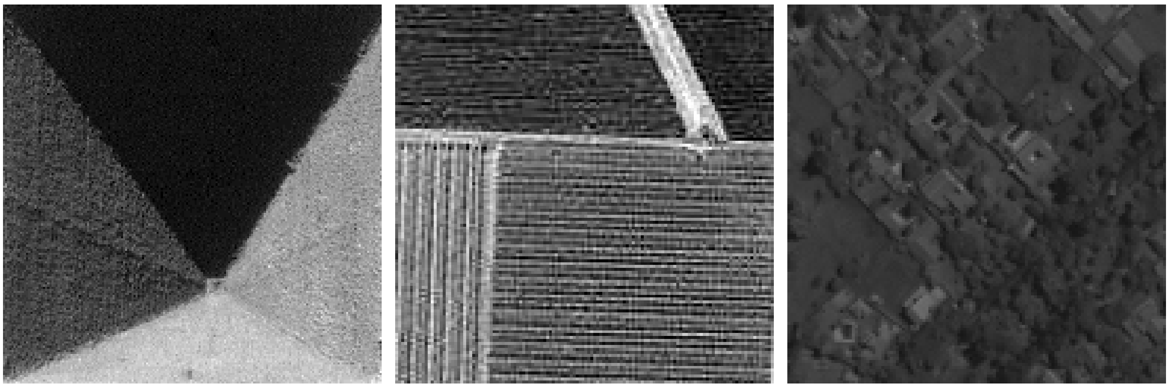
(a1) (b1) (c1)

(a2) (b2) (c2)

Fig. 6.25 Outlines of segmented textured regions (a1)-(c1) and Overlaid on original images (a2)-(c2).

C. IKONOS Images

Fig. 6.26 shows three pan IKONOS images with 1 *m* resolution and 128 × 128

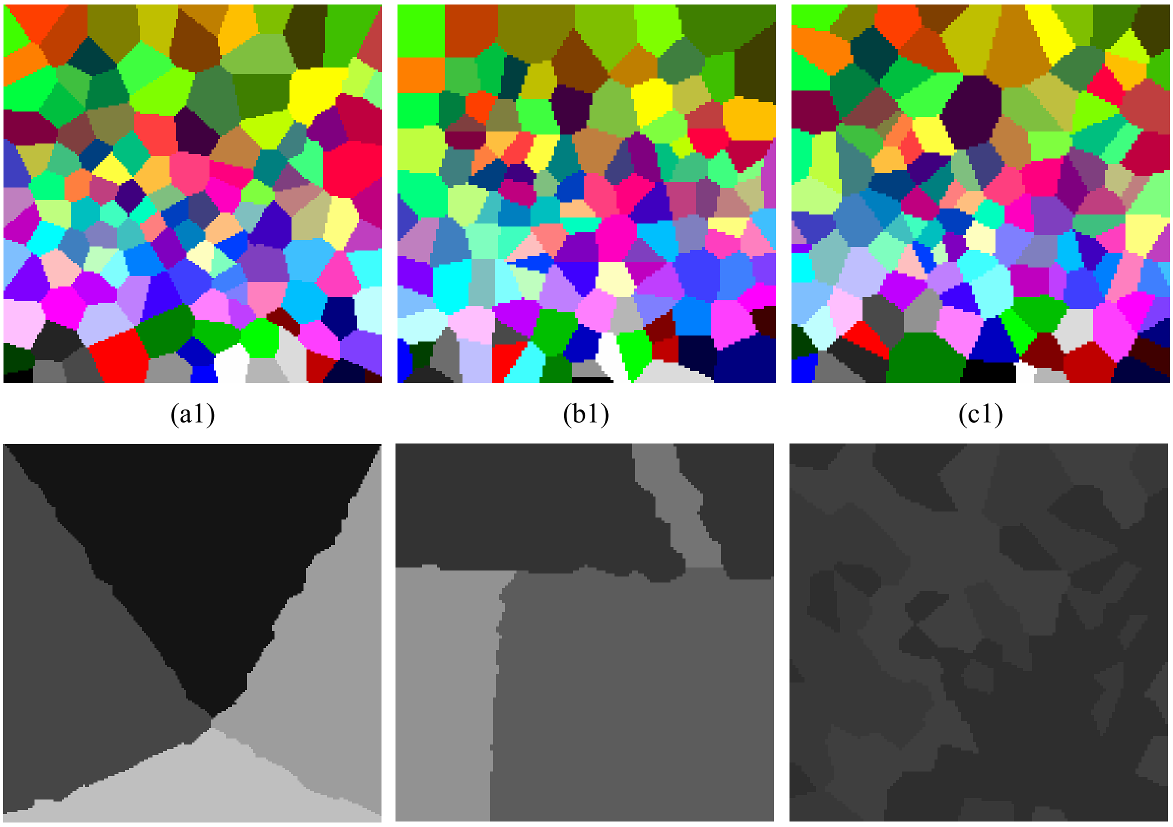


(a) (b) (c)

Fig. 6.26 IKONOS images with 1 *m* resolution and the size of 128×128 pixels.

pixels in size. Fig. 6.26 (a) is pictured from the top of the pyramid. According to the change in intensity, four texture classes are indicated during its segmentation. Fig. 6.26 (b) reveals agricultural area with four LULC classes. Fig. 6.26 (c) shows an urban area corresponding to three objects: trees, grass and buildings.

Fig. 6.27 gives the optimal partitions and optimal segmentations after 4,000 iterations. For simplicity, the number of generating points for the Voronoi tessellation is fixed, that is, *m* = 128. In segmented images, see Fig. 6.27 (a2)-(c2), the segmented texture regions are presented with the estimated means corresponding to the regions.



(a1) (b1) (c1)

(a2) (b2) (c2)

Fig. 6.27 Optimal partitions and optimal segmentations after 4,000 iterations.

Fig. 6.28 (a1)-(a3) show the outlines of the segmented texture regions. In Fig. 6.28 (b1)-(b3), they are overlaid on the testing images shown in Fig. 6.23.

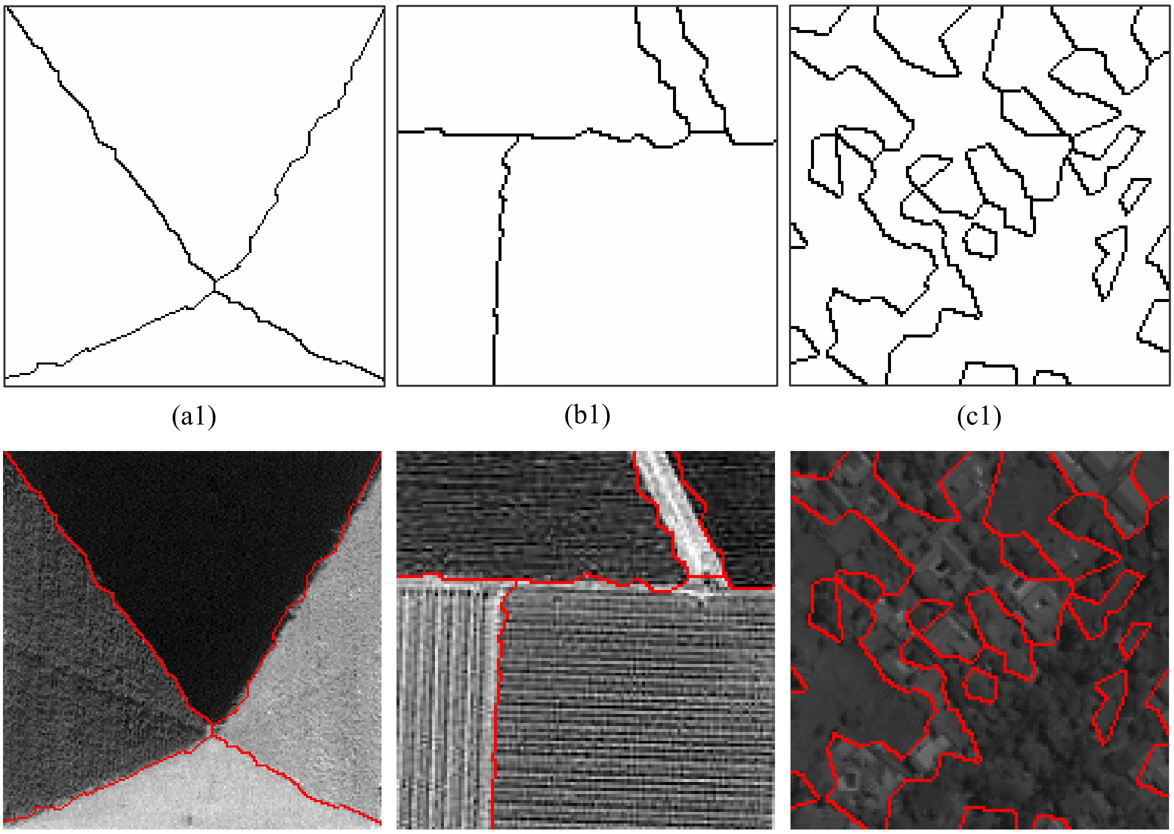


Fig. 6.28 Outlines of segmented textured regions (a1)-(c1) and their overlaying on original images (a2)-(c2).

6.3 Chapter Summary

In this chapter, an algorithm for colour texture segmentation is presented, which uses Voronoi tessellation to partition an image domain into polygons and two MRF models to characterize the colour texture structures intra- and inter-polygons. The algorithm has been evaluated by using synthetic colour texture images generated for accuracy assessment as well as real colour texture and remote sensing colour images. The experiments show the effectiveness of the proposed framework and developed algorithm for colour texture segmentation. Following above study, an algorithm for texture segmentation is also introduced.

The future work on this study includes (1) investigating the impact of the number of polygons by setting it as variable; (2) improving the current algorithm to automatically search the effective number of homogenous regions rather than indicate it *a priori*; (3) developing a general MGMRF model to characterize the spatial and spectral interactions from pixels in a given sub-region instead of using the MGMRF model which only considers the spatial and spectral interaction of neighbouring pixels. It would suffer from a difficulty since the positive definite requirement for the coefficient matrix in a multivariate Gaussian distribution; (4) developing a novel MCMC strategy for the Bayesian model.

Chapter 7

CONCLUSIONS AND RECOMMENDATIONS

7.1 Summary

In this thesis, a framework for remotely sensed data segmentation is introduced. In the proposed segmentation framework, a hierarchical data model is first developed. Based on a data domain partition, which can be carried out by Voronoi tessellation or marked point process, the developed data model uses the improved Potts model, strict stationary RF and MGMRF to characterize the global, regional and local structures hidden in remotely sensed data, respectively. By combining the hierarchical data model and general and scene-specific prior knowledge on geometries of LULC classes or objects and their spatial relationship, Bayesian inference is employed to provide a uniform framework for remotely sensed data segmentation. Finding model parameters and generating a corresponding segmentation of the remotely sensed data are achieved according to the MAP criterion. To facilitate this, MCMC schemes are formulated to allow the direct sampling of all the model parameters from the posterior distribution on the data.

Based on the segmentation framework, several algorithms for remotely sensed data segmentations and for object detections have been designed, including SAR intensity image segmentation, dark spot detection from SAR intensity images by marked point process, LiDAR point cloud segmentation, building detection from LiDAR point cloud by marked point process, and texture image segmentation.

7.2 Contributions

This thesis investigates the issues on remotely sensed data segmentation, the main contributions are:

- Proposing a hierarchical data model to characterize complex structures hidden in VHSR remotely sensed data. The advantages of the proposed data model include: (1) by attaching class labels to sub-regions, rather than to individual data points, a label field can be formed and used to characterize the global data characteristics caused by the homogeneity of measures for the same LULC class but with separated regions or the heterogeneity of measures for the different LULC classes; (2) by partitioning the data domain into sub-regions, the spatial-spectral correlation can be described at different scales: local (neighbouring) level and regional level. For example, the colour texture structure can be modeled with MGMRF. To characterize the local and regional colour texture structures, the MGMRF first models the spatial-spectral correlation of neighbouring pixels and then considers this kind of correlation for all neighbouring pixels in a sub-region to be statistically independent and identically distributed.

- Introducing stochastic geometry tools, such as Voronoi tessellation, marked point process into remotely sensed data analysis into data processing tasks for the purpose of data domain partition. The advantages of using theses tools lie in: (1) providing the ways to model geometric constrains on LULC or object classes, (2) by choosing different tools for data domain partition, different data processing tasks can be carried out, for example, remotely sensed data segmentation and object detection from remotely sensed data.

- Developing a uniform framework for remotely sensed data segmentation, which is easy to use for multiple data sources, such as SAR intensity imagery, LiDAR data and multi-spectral imagery.

- Under the proposed framework, designing new algorithms for remotely sensed data segmentations and object detection from remotely sensed data, including satellite SAR intensity image segmentation, dark spot detection satellite SAR intensity imagery, LiDAR point cloud segmentation, building detection from LiDAR point cloud, colour texture segmentation and texture segmentation.

7.3 Recommendations for Future Work

7.3.1 Unknown Number of Classes in Segmentation Problem

In remotely sensed data segmentation, automatically determining the number of LULC classes is very important, because the ground truth is not always available *a priori*.

Consider a remotely sensed dataset consisting of an unknown but bounded number *k* of homogeneous regions, *k* can be assumed to have a prior distribution. If *k* is a random variable, then the number of parameters in the data model is variable. Therefore, there are two issues that should be paid attention to, i.e., simulation scheme and label switching.

The RJMCMC algorithm (Green, 1995) is a good solution for its ability and flexibility in performing model selection and parameter estimation simultaneously. When *k* is variable the numerical labeling of the object classes is arbitrary. For example, the class labeled 1 at a certain point in time will usually represent a completely different texture at a later time. To overcome this problem, some form of identifibability is required, for example, Richardson and Green (1997) considered ordering the class labels in the relative order of the means or variables, and Stephens (2000) considered relabeling algorithms using decision theory.

7.3.2 Geometric Partition Problem

The purpose of the domain partition lies in fitting homogenous regions which are usually irregular by a collection of sub-regions. In this thesis, two kinds of partition schemes are used to divide a domain into sub-regions, Voronoi tessellation and marked point process. In practice, there are many schemes available for this purpose, including regular tessellation, Poisson tessellation (Chiles and Delfiner, 1999), dead leaves model (Lantuejoul, 2002), and so on. A regular tessellation means a tessellation made up of congruent regular polygons. The examples of regular tessellations in the Euclidean plane include, for example, triangles, squares and hexagons, see Fig. 7.1.

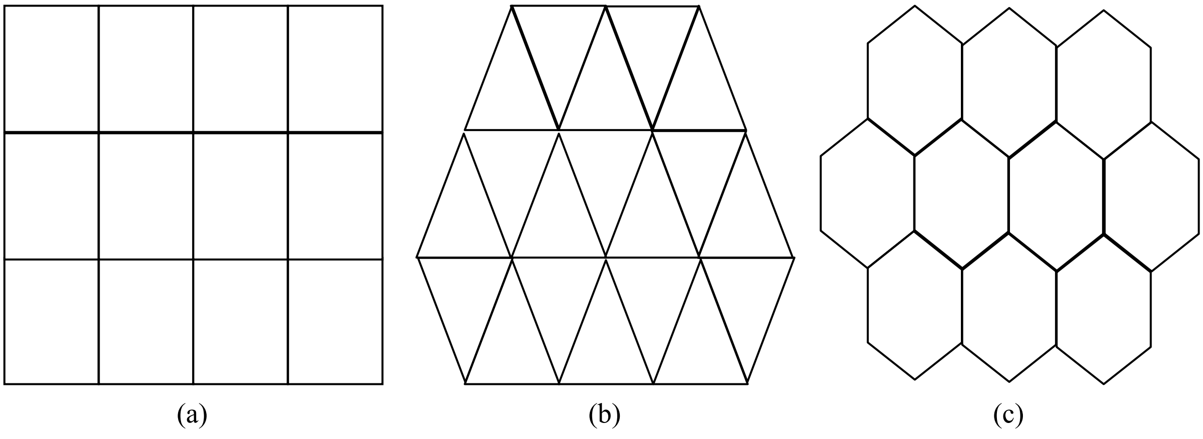


Fig. 7.1 Regular Tessellations: (a) squares, (b) Triangles, (c) hexagons.

Instead of the regular tessellation in which each cell has the same size, tessellations in which the cells are regular but of different sizes is recommended. Fig.7.2 shows an example of this kind of tessellation, in which cells are squares with different sizes.

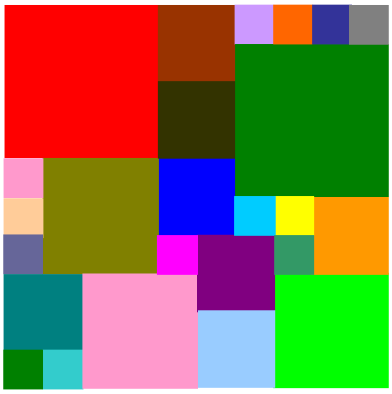


Fig. 7.2 Tessellation consists of squares with different sizes.

A line in R^2 is specified by two parameters $(a, d) \in [0, 2\pi) \times [0, \infty)$ where a is the direction of the unit vector orthogonal to the line and d is the distance from the line to the origin, see Fig. 7.3 (a). A Poison line network can be completely defined by the

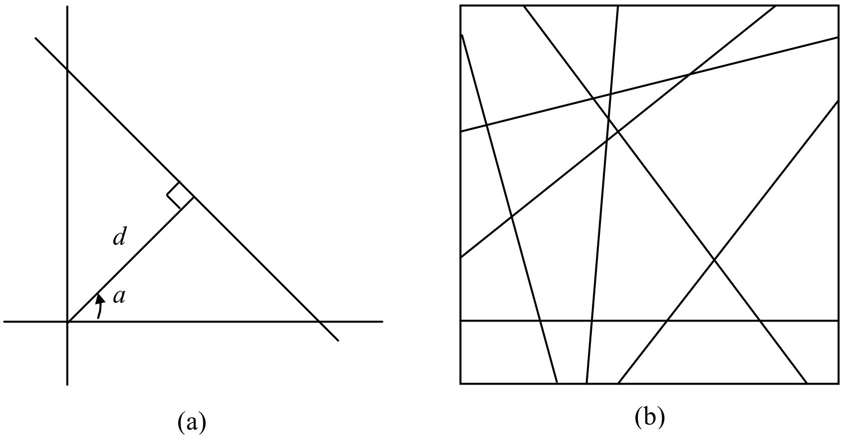


Fig. 7.3 (a) 2D line with parameter (a, d); (b) Poisson polygon tessellation.

intensity λ of the Poisson point process in the parameter space $[0, 2\pi) \times [0, \infty)$. The Poisson lines in the network hitting the domain D form a partition of D into convex polygons, see Fig. 7.3 (b).

As its name indicates, the dead leaves model has been devised to imitate falling leaves in fall. The dead leaves overlap and gradually tessellate the domain. Formally, the dead leaves model associates to each point $(x, y) \in R^2$. The colour $C(x, y)$ of the most recently fallen leaf that covers (x, y) (Lantuejoul, 2002) is shown in Fig. 7.4 as an example of the dead leaf model.

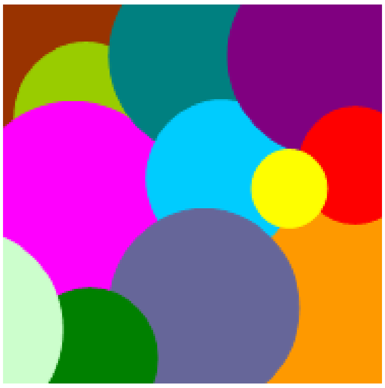


Fig. 7.4 Example of a dead leaf model.

The selection of a partition scheme should consider the following factors: its flexibility and its capability of fixing complex regions.

7.3.3 Optimal Segmentation Problem

Consider remotely sensed data $\boldsymbol{Z} = \{Z_i\ (x_i, y_i);\ i = 1, \ldots, n\}$ containing an unknown number k of homogenous regions. Assume that its domain D is partitioned into an unknown number m of sub-regions, that is, $D = \{D_j; j = 1, \ldots, m\}$ (here D denotes the domain of a given dataset and its partition). Associate each sun-region, say j^{th} sub-region, a label L_j is assigned to indicate the region to which it belongs. The labels for all sub-regions form a label filed, $\boldsymbol{L} = \{L_j;\ L_j \in \{1, \ldots, k\}$ and $j = 1, \ldots, m\}$. Let $\boldsymbol{\Theta}_M$ be parameter vector containing all parameters for prior distributions and data model and $\boldsymbol{\Theta}_S = \{\boldsymbol{D}, \boldsymbol{L}, k, m\}$. In fact, $\boldsymbol{\Theta}_S$ completely determinates the segmentation of the given data. Given posterior distribution $p(\boldsymbol{\Theta}_M, \boldsymbol{\Theta}_S \mid \boldsymbol{Z})$, the optimal segmentation is obtained by MAP estimate thought this thesis.

Instead of the MAP estimate, the marginal MAP (MMAP) estimate (Ripley, 1987) can be employed, that is,

$$\boldsymbol{\Theta}_S^{MMAP} = \arg\max p(\boldsymbol{\Theta}_S \mid \boldsymbol{Z}) \tag{7.1}$$

where

$$p(\boldsymbol{\Theta}_S \mid \boldsymbol{Z}) = \int_{\boldsymbol{\Theta}_M} p(\boldsymbol{\Theta}_S, \boldsymbol{\Theta}_M \mid \boldsymbol{Z}) d\boldsymbol{\Theta}_M \tag{7.2}$$

Unfortunately, approximating $\boldsymbol{\Theta}_M^{MMAP}$ is a complex problem since, in general, neither the maximization Eq. (7.1) nor the integration Eq. (7.2) can be preformed analytically.

7.3.4 Model Choice Problem

The segmentation problem studied in this thesis can be viewed as a model choice problem. Consider a countable collection of candidate models $\{M_k; k \in R\}$. Model M_k, which corresponds to a segmentation of a given data set $\boldsymbol{Z}$, has a vector $\boldsymbol{\theta}_k$ of unknown parameters in R^{d_k}, where the dimension d_k may vary from model to model. Bayesian inference about k and $\boldsymbol{\theta}_k$ will be based on the joint posterior distribution $p(k, \boldsymbol{\theta}_k \mid \boldsymbol{Z})$, that is,

$$p(k,\boldsymbol{\theta}_k \mid \boldsymbol{Z}) \propto p(k \mid \boldsymbol{Z}) p(\boldsymbol{\theta}_k \mid k,\boldsymbol{Z}) \tag{7.3}$$

Though the dimension of $\boldsymbol{\theta}_k$ may vary from model to model during a segmentation procedure, the distributions of parameters in $\boldsymbol{\theta}_k$ are the same from model to model. According to the experiments for the studies in this thesis, it is not always true. For example, in some SAR intensity images, the intensities in their homogenous regions needed to be segmented may satisfy both the Gamma and Gaussian distributions. It causes the variable parameter space and distributions in model choice. How to solve the problem will be challenging.

7.3.5 Distribution Choice Problem

Under the proposed framework for remotely sensed data representation, choosing proper distributions to characterize the data is very important. The followings lists some distribution models considered during the studies in this thesis.

It is well known that the distribution of multilook SAR intensity data for a homogenous region can be characterized with the Gamma, Gaussian, or K-distributions. Though all of them have been successfully used for SAR data analysis, they have difficulty in modelling the interaction among pixels and bands. Perhaps the Multivariate Gamma Distribution (MGD) is reliable for this purpose. Unfortunately, unlike multivariate Gaussian distribution which can be defined from univariate Gaussian directly, multivariate Gamma distribution can not be defined from the univariate Gamma distribution.

In practice, the MGDs on R^d have several non-equivalent definitions (Johnson et al., 1997). Among them, the Laplace transform based definition for MGD (Bernardoff, 2006) seems much more proper and easy to use for modeling interactions existing in SAR data. The following is an example of the definition for 2D MGB, Bivariate Gamma Distribution (BGB).

The BGB of random vector $\boldsymbol{Z} = (Z_1, Z_2)$ in R^2 is defined by its moment generating function or Laplace transform, which is characterized by an affine polynomial (Barndorff-Nielsen, 1980; Bernardoff, 2006). Given an affine polynomial,

$$P(\boldsymbol{\theta}) = 1 + \beta\theta_1 + \beta\theta_2 + \beta^2\rho\theta_1\theta_2 \qquad (7.4)$$

where the parameters satisfy the conditions: $\beta > 0$ and $1 > \rho > 0$, then the moment generating function of $\boldsymbol{Z}$ can be defined as

$$L(\boldsymbol{\theta}) = E\left(e^{\theta_1 Z_1 + \theta_2 Z_2}\right) = \left(P(-\boldsymbol{\theta})\right)^{-\alpha} = (1 - \beta\theta_1 - \beta\theta_2 + \beta^2\rho\theta_2\theta_2)^{-\alpha} \qquad (7.5)$$

where L is the Laplace transform and E is expectation operator.

From the defined generating function, it is obvious that Z_j, $j = 1, 2$ are distributed according to a univariate Gamma distribution with shape parameter α and scale parameter β, that is, $Z_j \sim Ga(\alpha,\beta)$, $j = 1, 2$. The probability density function can be expressed as follows,

$$f(Z_j;\alpha,\beta) = \frac{Z_j^{\alpha-1}}{\Gamma(\alpha)\beta^{\alpha}} \exp\left(-\frac{Z_j}{\beta}\right); \quad for \;\; j = 1,\ 2 \qquad (7.6)$$

where $\Gamma(\cdot)$ is the Gamma function.

The moments of the BGB can be obtained by differentiating Eq. (7.5). For example, the means and variances can be obtained as

$$\mathrm{E}[Z_j] = \left.\frac{\partial L(\boldsymbol{\theta})}{\partial \theta_j}\right|_{\boldsymbol{\theta}=0} = \alpha\beta\text{; For } j = 1, 2 \tag{7.7}$$

$$\mathrm{Var}[Z_1, Z_2] = \left.\frac{\partial^2 L(\boldsymbol{\theta})}{\partial \theta_1 \partial \theta_2}\right|_{\boldsymbol{\theta}=0} = \alpha\beta^2 \tag{7.8}$$

In Section 6.1, the MGMRF model is used to characterize the coefficient relations among colour bands for two neighbouring pixels. In theory, a general MGMRF model can be constructed to characterize the coefficient relations among colour bands for all pixels in a sub-region. The following shows an example for this model.

Given a Voronoi polygon D_j, the set of colour vectors $\boldsymbol{Z}_j$ can be rewritten as 3×n_j dimensional vector, that is, $\boldsymbol{Z}_j = \{Z_r(x_{(1)}, y_{(1)}), Z_g(\boldsymbol{x}_{(1)}, y_{(1)}), Z_b(\boldsymbol{x}_{(1)}, y_{(1)}), \ldots, Z_r(\boldsymbol{x}_{(i)}, y_{(i)}), Z_g(x_{(i)}, y_{(i)}), Z_b(x_{(i)}), \ldots, Z_r(x_{(n_j)}, y_{(n_j)}), Z_g(x_{(n_j)}, y_{(n_j)}), Z_b(x_{(n_j)}, y_{(n_j)}))\}$ where $n_j = \#\{(x_i, y_i); (x_i, y_i) \in D_j\}$ and the subscript i are the indexes of pixels in D_j. The joint distribution of $\boldsymbol{Z}_j$ is characterized by an MGMRF with the probability density function as follows

$$p(\boldsymbol{Z}_j) = \frac{1}{(2\pi)^{-\frac{3}{2}n_j} \, |\boldsymbol{\Sigma}_j|^{-\frac{1}{2}}} \exp\left\{-\frac{1}{2}\left(\boldsymbol{Z}_j - \boldsymbol{\mu}_{l_j}\right)^T \boldsymbol{\Sigma}_j^{-1}\left(\boldsymbol{Z}_j - \boldsymbol{\mu}_{l_j}\right)\right\} \tag{7.9}$$

where $\boldsymbol{\mu}_{L_j}$ is the expected value for colour vector with colour texture class label l_j

and expressed as

$$\boldsymbol{\mu}_{L_j} = \underbrace{(\mu_{L_jr}, \mu_{L_jg}, \mu_{L_jb}, \ldots, \mu_{L_jr}, \mu_{L_jg}, \mu_{L_jb})}_{3n_j} \tag{7.10}$$

where μ_{L_jr} , μ_{L_jg} , μ_{L_jb} are the means of red, green, and blue components, respectively. $\boldsymbol{\Sigma}_j$ is the covariance matrix and defined as

$$\boldsymbol{\Sigma}_j = \boldsymbol{B}_j \sigma_{L_j}{}^2 \tag{7.11}$$

where $\boldsymbol{B}_j$ is called the covariance matrix $\boldsymbol{B}_j = [B_{(i)(i')}]_{(3n_j)\times(3n_j)}$ and its elements $B_{(i)(i')}$ are called the spatial-spectral parameters which control the spatial (between pixels) and spectral (between bands) interactions. Generally speaking, the definition of $\boldsymbol{B}_j$ can be data-specific. The parameter space is restricted to those values which give a positive definite covariance matrix. Some researchers work on this issue when the domains are square lattices (Besag and Moran, 1995; Jian, 1979; Moura and Balram, 1992; Balram and Moura, 1993). As we know, there has been no work conducted on the issues in irregular domain cases.

7.3.6 Modeling Label Field Problem

Given a partition of data domain $D = \{D_j; j = 1, \ldots, m\}$, the label field $\boldsymbol{L} = \{L_j; j =$

1, …, m} can be constructed by assigning each sub-region D_j a label $L_j \in \{1, …, k\}$ indicating the homogeneous region to which D_j belongs.

It can be noted that the remotely sensed data usually reveal both local and global meaning, that is, they are characterized by invariance of certain local attributions that are distributed over the domain of the data. The label field ***L*** provides a chance to model the data structures at large scale. It will be challenging to determine how to use the label field for this purpose.

7.4 Chapter Summary

In this final chapter, the work and contributions on studies for this thesis are summarized. The work is motivated by the idea of developing approaches, based on sophisticated mathematical tools to remotely sensed data analysis to catch up with the increasing development of remote sensing techniques. Also, some research direction in future work is recommended. Some of them are in progress, and some are just ideas.

REFERENCES

Anderson, T.W., 2003. *An Introduction to Multivariate Statistical Analysis*, Wiley-Interscience.

Baddeley, A.J. and J. Mϕller, 1989. Nearest neighbour Markov point processes and random sets. *International Statistical Review*, 57: 89-121.

Ballard, D.H. and C.M. Brown, 1982. *Computer Vision*, Prentice Hall, Englewood Cliffs, NJ, USA.

Balram, N. and J.M.F. Moura, 1993. Noncausal Gauss Markov random fields: parameters structure and estimation. *IEEE Transactions on Information Theory*, 39(4): 1333-1355.

Barndorff-Nielsen, O.E., 1980. Conditional resolutions. *Biometrika*, 67: 293-310.

Bernardoff, P., 2006. Which multivariate Gamma distributions are infinitely divisible? *Bernoulli*, 12(1): 169-189.

Besag, J.E., 1974. Spatial interaction and the statistical analysis of lattice systems (with discussion). *Journal of the Royal Statistical Society*, *Series B*, 36(2): 192-326.

Besag, J.E., 1986. On the statistical analysis of dirty picture (with discussion). *Journal of the Royal Statistical Society*, *Series B*, 48(3): 259-302.

Besag, J.E., 1993. Towards Bayesian image analysis. *Journal of Applied Statistics*, 20(5): 107-119.

Besag, J.E. and P.A.P. Moran, 1975. On the estimation and testing of spatial interaction in Gaussian lattice processes. *Biometrika*, 62: 555-562.

Bezdek, J.C., R. Ehrlich, and W. Full, 1984. FCM: The fuzzy c-means clustering algorithm. *Computers and Geosciences*, 10(2-3): 191-203.

Blackwell, P.G. and J. Mϕller, 2003. Bayesian analysis of deformed tessellation models. *Advances in Applied Probability*, 35: 4-26.

Bruzzone, L. and L. Carlin, 2006. A multilevel context-based system for classification of very high spatial resolution images. *IEEE Transactions on Geoscience and Remote Sensing*, 44(9): 2587-2600.

Cannon, R.L., J.V. Dave, J.C. Bezdek, M.M Trivedi, 1986. Segmentation of a thematic mapper image using the fuzzy c-means clustering algorithm. *IEEE Transactions on Geoscience and Remote Sensing*, 24: 400-408.

Cao, Y. and X. Yang, 2007. Man-made object detection in aerial images using multi-stage level set evolution. *International Journal of Remote Sensing*, 28 (8): 1747-1757.

Chan, T.F. and J. Shen, 2005. *Image Processing and Analysis: Variational, PDE, wavelet, and stochastic methods*, the Society for Industrial and Applied Mathematics.

Chang, Y. L. and X. Li, 1994. Adaptive image region-growing. *IEEE Transactions on Image Processing*, 3(6): 868-872.

Chiles, J.P. and P. Delfiner, 1999. *Geostatistics: Modeling Spatial Uncertainty*, Wiley.

Congalton, R.G. and K. Green, 1999. *Assessing the Accuracy of Remote Sensing Data: Principles and Practices*, CRC Press.

Curran, P.J., 2001. Remote sensing: using the spatial domain. *Environmental and Ecological Statistics*, 8(4): 331-344.

de Jong, S.M. and F.D. van der Meer (eds), 2004. *Remote Sensing Image Analysis: Including the Spatial Domain*, Springer, 2004.

de Ves, E., J. Domingo, G. Ayala, and P. Zuccarello, 2006. A novel Bayesian framework for relevance feedback in image content-based retrieval system. *Pattern Recognition*, 39(9): 1622-1632.

D'Elia, C., G. Poggi, and G. Scarpa, 2003. A tree-structured Markov random field model for Bayesian image segmentation. *IEEE Transactions on Image Processing*, 12 (10): 1259-1273.

Ding, J., R. Ma, and S. Chen, 2008. A scale-based connected coherence tree algorithm for image segmentation. *IEEE Transactions on Image Process*, 17 (2): 204-216.

Dryden, I.L., M.R. Faghihi, and C.C. Taylor, 1997. Procrustes shape analysis of planar point subset. *Journal of the Royal Statistical Society*, *Series B*, 59: 353-374.

Dryden, I.L., M.R. Scarr, and C.C. Taylor, 2003. Bayesian texture segmentation of weed and crop images using reversible jump Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society, Series C* (*Applied Statistics*), 52(1): 31-50.

Dryden, I.L., R. Farnoosh, and C.C. Taylor, 2006. Image segmentation using Voronoi polygons and MCMC, with application to muscle fibre images. *Journal of Applied*

Statistics, 33(6): 609-622.

Dubes, R.C. and A.K. Jain, 1989. Random fields models in image analysis. *Journal of Applied Statistics*, 16 (2): 131-164.

Fleiss, J.L., 1975. Measuring agreement between two judges on present or absent of a trait. *Biometrics*, 31: 651-659.

Gaetano, R., G. Scarpa, and G. Poggi, 2009. Hierarchical texture-based segmentation of multiresolution remote-sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 47 (7): 2129-2141.

Gelman, A.J.B. Carlin, H.S. Stern, and D.B. Rubin, 2004. *Bayesian Data Analysis* (Second Edition), Chapman & Hall/CRC.

Geman, S. and D. Geman, 1984. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 6(6): 721-741.

Green, P.J., 1995. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4): 711-732.

Greenfeld, J.S., 1991. An operator-based matching system. *Photogrammetric Engineering and Remote Sensing*, 57(8): 1049-1055.

Hadamard, J., 1952. *Lectures on the Cauchy's Problem in Linear Partial Differential Equation*, Dover Publications, New York.

Hastings, W.K., 1970. Monte Carlo sampling method using Markov chain and their applications. *Biometrika*, 57(1): 97-109.

Horn, A.R. and C.R. Johnson, 1991. *Topics in Matrix Analysis*, Cambridge University

Press.

Jähne, B., 2005. *Digital Image Processing* (Sixth Revised and Extended Edition), Springer.

Jensen, J.R., 2006. *Introductory Digital Image Processing: A Remote Sensing Perspective* (Third Edition), Prentice Hall.

Johnson, N., S. Kotz, and N. Balakrishnan, 1997. *Continuous Multivariate Distributions*, Wiley, New York.

Kermad, C.D., and K. Chehdi, 2002. Automatic image segmentation system through iterative edge-region co-operation. *Image and Vision Computing*, 20(8):541-555.

Koch, K.R., 2007. *Introduction to Bayesian Statistics* (Second Edition), Springer.

Lantuejoul, C., 2002. *Geostatistical Simulation Models and Algorithms*, Springer.

Laprade, R.H., 1988. Split-and-merge segmentation of aerial photographs. *Computer Vision Graphics and Image Processing*, 44(1): 77-86.

Lee, J.S., I. Jurkevich, P. Dewaele, P. Wambacq and A. Oosterlinck, 1994. Speckle filtering of synthetic aperture radar images: a review. *Remote Sensing Reviews*, 8: 313-340.

Li, S.Z., 2009. *Markov Random Field Modeling in Image Analysis* (Third Edition), Springer.

Li, Y, J. Li, and Y. Lu, 2008. A fuzzy segmentation based approach to extraction of coastlines from IKONOS imagery. *Geomatica*, 62(4): 396-408.

Mardia, K.V. and T.J. Hainsworth, 1988. A spatial thresholding method for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10 (6): 919-927.

Marr, D., 1982. *Vision: A Computation Investigation into the Human Representation and Processing of Visual information*, W. H. Freeman, San Francisco.

Mehnert, A. and P. Jackway, 1997. An improved seeded region growing algorithm. *Pattern Recognition Letters*, 18(10): 1065-1071.

Metropolis, N., A.W. Rosenbluth, M.N.Rosenbluth, A.H.Teller, and E. Teller, 1953. Equation of state calculation by fast computing machines. *The Journal of Chemical Physics*, 21: 1087-10932.

Moura, J. M. F. and N. Balram, 1992. Recursive structure of noncausal Gauss-Markov random fields, *IEEE Transactions on Information Theory*, 38(2): 334-354.

Mϕller, J. and Φ. Skare, 2001. Bayesian image analysis with coloured Voronoi tessellation and a view to applications in reservoir modelling. *Statistical Modelling*, 1:213-232.

Mueller, M., K. Segl, and H. Kaufmann, 2004. Edge- and region-based segmentation technique for extraction of large, man-made objects in high-resolution satellite imagery. *Pattern Recognition*, 37(8): 1619-1628.

Okabe, A., B. Boots, and K. Sugihara, 1992. *Spatial tessellations: Concepts and applications of Voronoi diagrams*, John Wiley & Sons, Chichester, UK.

Pal, N.R. and J.C. Bezdek, 1995. On cluster validity for the fuzzy c-means model. *IEEE Transactions on Fuzzy Systems*, 3(3): 370-379.

Poggio, T., V. Torre, and C. Koch, 1985. Computational vision and regularization theory. *Nature*, 317: 314-319

Qiu, P., 2005. *Image Processing and Jump Regression Analysis*, A John Wiley & Sons.

Rekik, A., M. Zribi, A.B. Hamida, and M. Benjelloun1, 2007. Review of satellite image segmentation for an optimal fusion system based on the edge and region approaches. *International Journal of Computer Science and Network Security*, 7 (10): 242-250.

Richards, J.A. and X. Jia, 2006. *Remote Sensing Digital Image Analysis: An Introduction* (Fourth Edition), Springer.

Richardson, S. and P.J. Green, 1997. On Bayesian analysis of mixtures with an unknown number of components (with discussion). *Journal of the Royal Statistical Society*, *Series B*, 59(4): 731-792.

Rignot, E.J.M, R. Kowk, J.C. Curlander, and S.S. Pang, 1991. Automated multisensor registration: requirements and techniques. *Photogrammetric Engineering and Remote Sensing*, 57(8): 1029-1038.

Ripley, B.D., 1987. *Stochastic Simulation*, Wiley, New York.

Rosenfeld, A., 1970. Connectivity in digital pictures. *Journal of ACM*, 17: 146-156.

Russ, J.C., 2008. *The Image Processing Handbook* (Fifth Edition), CRC Press.

Ryan, T., 1985. Image segmentation algorithms. In: *Architectures and Algorithms for Digital Image Processing II*, SPIE, 534(2): 172-178.

Stephens, M., 2000. Dealing with label switch in mixture models. *Journal of the Royal Statistical Society*, *Series B*, 62(4): 795-809.

Stoyan, D., W.S. Kendall, and J. Mecke, 1995. *Stochastic Geometry and Its*

Applications (Second Edition), Wiley, New York.

Strauss, D.J., 1977. Clustering on coloured lattices. *Journal of Applied Probability*, 14(1): 135-143.

Tremeau, A. and N. Borel, 1997. A region growing and merging algorithm to color segmentation. *Pattern Recognition*, 30 (7): 1191-1203.

Udupa, J.K. and S. Samarasekera, 1996. Fuzzy connectedness and object definition: theory, algorithms, and applications in image segmentation. *Graphical Models and Image Processing*, 58 (3): 246-261.

Winkler, G., 1995. *Image Analysis, Random Fields and Dynamic Monte Carlo Methods: a mathematical introduction*, Springer-Verlag.

Yu, Q., P. Gong, N. Clinton, G. Biging, M. Kelly, and D. Schirokauer, 2006. Object-based detailed vegetation classification with airborne high spatial resolution remote sensing imagery. *Photogrammetric Engineering and Remote Sensing*, 72 (7): 799-811.

Zucker, S.W., 1976. Region growing: childhood and adolescence. *Computer Graphics and Image Processing*, 5:382-399.