SINGLE TREE DETECTION FROM AIRBORNE LASER SCANNING DATA: A STOCHASTIC APPROACH

JUNJIE ZHANG

A DISSERTATION SUBMITTED TO

THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

GRADUATE PROGRAMME IN EARTH AND SPACE SCIENCE

YORK UNIVERSITY

TORONTO, ONTARIO

MAY 2015

ABSTRACT

Characterizing and monitoring forests are of great scientific and managerial interests, such as understanding the global carbon circle, biodiversity conservation and management of natural resources. As an alternative or compliment to traditional remote sensing techniques, airborne laser scanning (ALS) has been placed in a very advantageous position in forest studies, for its unique ability to directly measure the distribution of vegetation materials in the vertical direction, as well as the terrain beneath the forest canopy. Serving as basis for tree-wise forest biophysical parameter and species information retrieval, single tree detection is a very motivating research topic in forest inventory.

The objective of the study is to develop a method from the perspective of computer vision to detect single trees automatically from ALS data. For this purpose, this study explored different aspects of the problem. It starts from an improved pipeline for canopy height model (CHM) generation, which alleviates the distortion of tree crown shapes presented on CHMs resulted from conventional procedures due to the shadow effects of ALS data and produces pit-free CHM. The single tree detection method consists of a hybrid framework which integrates low-level image processing techniques, i.e. local maxima filtering (LM) and marker-controlled watershed segmentation (MCWS), into a high-level probabilistic model. In the proposed approach, tree crowns in the forest plot are modelled as a configuration of circular objects. The configuration containing the best possible set of detected tree objects is estimated by a global optimization solver in a probabilistic framework. The model features an accelerated optimization process compared with classical stochastic models, e.g. marked point processes. The parameter estimation is another issue: the study investigated both a reference-based supervised and an Expectation-Maximization (EM) based unsupervised method to estimate the parameters in the model. The model was tested in a temperate mature coniferous forest in Ontario, Canada, as well as simulated coniferous forest plots with various degrees of crown overlap. The experimental results showed the effectiveness of our proposed method, which was capable of reducing the commission errors produced by local maxima filtering based methods, thus increasing the overall detection accuracy by approximately 10% on all of the datasets.

TO MY PARENTS
MY WIFE
& MY DAUGHTER

ACKNOWLEDGEMENTS

This thesis would not be possible without the support of many people. I would like to extend my sincere appreciation to them for their help and support throughout my research.

My sincere gratitude goes to my supervisor, Dr. Gunho Sohn, for his guidance, encouragement, academic support, and all the positive energy he has given during my doctoral study. Not only his wealth of knowledge and expertise on LiDAR, but more than that, his personality as an extraordinary human being, helped me though many challenges on research and even in personal life. His advice always inspire me to grow into a better scientist. My PhD would not be done if I did not have great support from him.

I also wish to express my deep appreciation to my supervision committee Dr. Costas Armenakis and Dr. Baoxin Hu, for their scientific suggestions, valuable insights, warm encouragements, and continuous support on my research journey in the Department of Earth and Space Science and Engineering, York University. I would like to thank also Dr. James Elder and Dr. Franz Rottensteiner, my examination committee members, for carefully reviewing and providing comments concerning different aspects of this research. Your thoughtful suggestions definitely contributed to the quality of my work.

Many thanks goes to Dr. Tarmo Remmel, for your professional and organized teaching skills I have learned. It is such a great opportunity to TA your course; To Dr. James Elder, for your instructions, which enriched my knowledge and skills in the field of computer vision.

I enjoyed the warm support and company of all the GeoICT members and fellow students in the Department. It is difficult to name all of you here who assisted me in one way or another, but please accept my sincere thanks as well if you have not been mentioned. I would like to give my

thanks to Yoonseok Jwa, Brian Kim, Connie Ko, Bruce Jung, Larry Wang, Jili Li, Wen Zhang, Yu Gao, Chao Luo, Ravi Persad and Julien Li-Chee-Ming. I especially thank my colleagues Jili Li, Kongwen Zhang, Damir Gumerov and Connie Ko, who assisted in the fieldwork for the research.

I appreciate all the help and support received from the faculty and staff members in the Department of Earth and Space Science and Engineering. Special thanks goes to Paola Panaro and Marcia Gaynor for many administrative support during my study here from the first day I came. Their kind and valuable work made my stay and study a comfortable experience.

I am also indebted to the funding bodies of GeoDigial International Inc., Ontario Centres for Excellence (OCE), and a Discovery Grant from the Natural Sciences and Engineering Research Council of Canada (NSERC). I would like to thank Dr. Richard Pollock, Konstantin Lisitsyn, Doug Parent and Yulia Lazukova of GeoDigital Internatioanl Inc. for their assistant in the collection of data.

To my uncles and aunts, there is no words to express my appreciation. I thank you for your infinite support to my family, both physically and mentally, while I was away. The support of my friends who are far away is never forgotten. You are all precious treasure in the course of my life.

Above all, my heartfelt and everlasting gratitude goes to my family, especially my wife, Sharon, and my most lovely daughter, Iris, for their patience, encouragement and unconditional love. Now, it is time for me to repay all your sacrifices.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

3D	Three dimensional
ALS	Airborne Laser Scanning
CHM	Canopy Height Model
DBH	Diameter at Breast Height
DEM	Digital Elevation Model
DSM	Digital Surface Model
EM	Expectation-Maximization
GPS	Global Positioning System
IMU	Inertial Measurement Unit
Laser	Light Amplification by Simulated Emission of Radiation
LiDAR	Light Detection And Ranging
LM	Local Maxima
MAP	Maximum a Posterior
MCMC	Markov Chain Monte Carlo
MCWS	Marker-Controlled Watershed Segmentation
MLS	Mobile Laser Scanning
MPP	Marked Point Process
MRF	Markov Random Field
nDSM	normalized Digital Surface Model
PGMCMC	Prior-Guided Markov Chain Monte Carlo
RJMCMC	Reversible Jump Markov Chain Monte Carlo
TIN	Triangulated Irregular Network
TLS	Terrestrial Laser Scanning

Chapter 1
Introduction

1.1 Research Context

1.1.1 Remote sensing of forestry

Forests play an important role in the global carbon cycle. They are recognized as one of the largest reserves of terrestrial carbon and can influence the climate by modulating fluxes of carbon and water near the Earth's surface through nature (e.g., photosynthesis and respiration of plants) and anthropogenic (e.g., deforestation and aforestation) processes (Baldocchi et al., 1996; Brown et al., 1996; Dixon et al., 1994; Patenaude et al., 2005). Forests not only function as carbon stock, they are also of great importance in providing renewable raw materials and energy, maintaining biodiversity, protecting land and water resources (FAO, 2015). Quantification of forest structure characteristics (e.g., forest biomass) is essential to provide us with information to ensure sustainable forestry (Parresol, 1999).

To effectively manage the valuable forest resources, remote sensing techniques have become an integral part of forest inventory. They have greatly promoted the performance and efficiency to obtain information of forest and tree characteristics at different scales to support the practices of forest management and planning (Dubayah and Drake, 2000; Naesset, 2004a; Tomppo et al., 2002; Wulder, 1998; Xie et al., 2008). During the last century, statistical sampling approaches based on field measurements have been used to collect information regarding the state of forest resources for large areas (Jonsson et al., 1993). The conventional field measure, although accurate (Andersen, 2003), is proven to be very costly and time consuming (Gier and Roy, 2003; Lu, 2006),

not to mention the subjective sampling of forest stands over large areas. Furthermore, due to the high cost, the sampling intensity in those inventory programs is limited, making the estimation over large areas highly susceptible to sampling error. In contrast, remote sensing data acquired from an airborne or spaceborne platform, such as aerial photographs or satellite images, are much cheaper and able to provide information about forest resources at various temporal and spatial scales. Manual photo interpretation, for example, has been used to delineate forest stands and determine forest variables such as tree species, tree height, and stem volume (Mora et al., 2010; Rhody, 1965). Satellite imageries, combined with field measurements of sample plots, are used to automatically produce wall-to-wall estimate of forest variables (Nilsson, 1997).

One of the biggest challenges faced by the passive optical remote sensing, however, is its limited ability in recording the vertical information of forests, which are structurally complex systems in three-dimensional space. It relies on the solar illumination reflected from the outer canopy in the visible, near and middle infrared portion of the electromagnetic spectrum (~0.4 to 2.5 μm), thus only provide a two-dimensional view of the forest canopy. To sum up, the illumination geometry and intensity are not controlled for the acquisition of the passive optical remote sensing data. For the analysis of the passive optical data, it further requires an inference from two-dimensions to three-dimensions.

1.1.2 LiDAR remote sensing for forest inventory

With the development of new sensors and positioning devices, the emergence of a new generation of active optical remote sensing technologies has given rise to a source of remote sensing data that is well-suited for analysis of forest structure. In particular, small-footprint airborne laser scanning

(ALS), also known as airborne LiDAR, enables direct measuring the 3D structural information of trees and the elevation of the terrestrial surface under the canopy in forests. The direct measurement of vegetation structures enables a more accurate estimation of tree or stand variables. For example, Naesset (2004a) noted a 10% - 20% estimated root mean square error (RMSE) accuracies for total volume at stand level in the Nordic countries. The unique capability makes ALS an alternative to traditional passive optical remote sensing, or even the preferred method, to derive certain forest parameters, such as tree height, crown dimensions, stand volume, basal area, and above-ground biomass (Bortolot and Wynne, 2005; Hyypp äand Inkinen, 1999; Means et al., 2000; Næsset, 1997; Naesset, 2002).

The promising nature of LiDAR has inspired research scientist to explore the applicability of it to forest inventory. The first studies started around 1980 (Aldred and Bonnor, 1985; Maclean and Krabill, 1986; Nelson et al., 1984; Solodukhin et al., 1977) and focused on using a profiling system for forest height, stand density, tree species, and biomass estimation. The basics of using LiDAR for forest inventory were established at that time. ALS gained its popularity in forest inventory quickly after the first promising studies in the mid-1990s (e.g., Hyypp äand Inkinen (1999); Naesset (1997a, 1997b); Naesset (2002); Nilsson (1996)). ALS is currently the most applied LiDAR system in forestry. Other types of LiDAR systems, namely terrestrial laser scanning (TLS) and mobile laser scanning (MLS), are mainly used in applications, such as the acquisition of ground truth or small-area monitoring, for experimental purposes (Vastaranta, 2012).

1.1.3 Methods of airborne laser scanning for forest inventory

Characterization of forest resources using ALS can be broadly categorized into area-based approaches (ABAs) and individual-tree-based approaches (ITDs) (Hyyppä et al., 2008). ABAs rely on the statistical principle and predicts forest attributes based on parametric regression or nonparametric imputation models built between field measured variables and features derived from ALS data (Maltamo et al., 2006; Naesset, 2002). ABAs can perform under a low ALS point density, and is the method currently applied in operational forest inventory to provide a wall-to-wall estimation of forest attributes (Naesset, 2004c; White et al., 2013a). ITDs measure or predict tree-level variables on the basic unit of the individual trees from ALS data and then aggregate them to obtain stand-level forest inventory results (Hyyppä et al., 2012).

Despite the added costs and amount of information to store and process high-density ALS data, ITDs are of significant interest in forest inventory and is a motivating research topic. The primary advantage of ITDs over ABAs is the supply of tree lists and the ability to directly derive the true stem distribution series, which would result in better prediction for timber assortments (Vastaranta et al., 2011a). Generally, this information is invaluable in forest planning-related simulation and optimization, logging operation planning and wood supply logistics (Vastaranta et al., 2011b), e.g., detection of harvest trees and forest growth determination (Yu et al., 2004). Another advantage of ITDs is that they can reduce the amount of or potentially replace the expensive fieldwork required for ABAs (Hyyppä et al., 2008; Vastaranta et al., 2012). Additionally, tree species classification based on ITD has been reported in recent studies (Brandtberg, 2007; Heinzel and Koch, 2011; Orka et al., 2009; Suratno et al., 2009), which could potentially improve the prediction of species-specific forest attributes (Heurich, 2008; Yao et al., 2012; Yu et al., 2010). Furthermore,

the combination of ITD and ABA, called the semi-ITD method, to improve the estimation accuracy has also been viewed as a future method for forest inventory (Breidenbach et al., 2010; Hyyppä et al., 2012; Vastaranta et al., 2012). Therefore, individual tree detection techniques are still of significant importance from the practical forestry viewpoint.

1.2 Scope of Research

The application of ITDs depends on the successful detection of single trees and delineation of corresponding tree crowns from ALS data. The methods and workflows on single tree detection from ALS data have been more established since the first trials, e.g., (Hyyppä and Inkinen, 1999). Tree variables are then directly measured or predicted using derived ALS features for each tree. Serving as basis of tree modeling and other related applications mentioned above, single tree detection has been a topic receiving significant interests from the remote sensing community over the past decades.

Researches on ALS-based ITD can be broadly divided into two groups. The first group comprises researches focused on algorithms for single tree detection, e.g., Koch et al. (2006); Leckie et al. (2003); Lin et al. (2011); Reitberger et al. (2009). The other group addresses the estimation of single-tree or stem parameters. This thesis aims to develop a novel method to detect single trees from ALS data, with the attempt to fit a piece into the big puzzle of ALS-based forest inventory. Thus, this research falls within the former group. The subsequent derivation of forest variables and evaluation of the bias is another topic that requires extra research effort, which is out of the scope of our presented work. However, the motivation of the work within the first group is

clear: improved ITD algorithms can increase the tree detection accuracy, which could in turn result in better performance in the prediction of forest parameters (Yao et al., 2012; Yu et al., 2011).

The aim of the research describe in the thesis is to develop a method from the perspective of computer vision to detect single trees from ALS data. The model proposed intend to take advantages of both low-level image processing techniques and high-level probabilistic models. To evaluate the detection accuracy of the developed algorithm, a study area is selected and fieldwork is done in northern Ontario, Canada, which is dominated by temperate mature coniferous forests.

To test the performances of the model proposed more thoroughly in different forest conditions, we also use simulated ALS data to serve as a complement to the real ALS data. With a procedure we proposed, we simulate ALS data of mature coniferous forest plots with increasing tree density and degrees of crown overlap, which are assumed to be critical factors influencing the performance of ITD algorithms (Kaartinen and Hyypp ä 2008; Vauhkonen et al., 2012). Here please note the simulation does not emphasize on the synthesis of realistic ALS data under different flight and instrumental configurations, with sophisticated radiative transfer model and complex tree geometric models, like the ones presented in Morsdorf et al. (2009) and Disney et al. (2010). The purpose is to provide a controlled environment to test the robustness of our proposed single tree detection model, investigate the model configuration, i.e. parameter setting, with respect to forest conditions and evaluate the parameter estimation methods involved in the model.

1.3 Research Methodology

In this section, methodology is described step by step according to its order of appearance presented in the thesis. The four steps described are: 1) ALS data pre-processing, 2) ALS data simulation of forest plots, 3) a probabilistic model for single tree detection, and 4) parameter estimation and model optimization.

1.3.1 ALS data pre-processing

ALS data pre-processing serves as the starting point for single tree detection. Most single tree detection methods apply on the canopy height model (CHM) generated from ALS data, which provides an accurate representation of the outer surface of the tree canopy. The peaks and valleys on the CHM generated from high-density ALS data are better estimations of treetop positions and crown edges than can be obtained from aerial photographs or satellite imageries.

However, similar to aerial photographs and satellite imageries, ALS data also suffer from the shadow effects, in a way that there is a lack of returns from the ground in the shadowed areas of tree crowns. The geometric shapes of tree crowns presented on the CHMs can be greatly distorted, if the ALS data pre-processing does not take the shadow-effect into consideration. Another problem in ALS data pre-processing is that there are usually pits on the CHM generated, which can hamper the segmentation and delineation of tree crowns. In this study, we designed an improved pipeline of ALS data pre-processing to tackle those problems, which is detailed in **Chapter 3**.

1.3.2 Simulation of ALS data of different forest plots

Single tree detection using small-footprint airborne LiDAR data has been extensively studied over the past few years. Researchers have proposed many different methods for this purpose, however, few of the methods have a chance to get thoroughly tested under different forest conditions, either because of lack of remote sensing data, or no reference data.

In **Chapter 4**, we present a method to simulate airborne LiDAR point clouds of forest plots with different degrees of overlap. Firstly, point process theory is used to generate forest plots in which "points" are tree locations and "marks" the crown width. The interactions between trees are modelled as a hard-core process and three plots with increasing crown overlap degrees are produced. Then, point clouds of single trees are selected from a tree template library prepared from airborne LiDAR data acquired from forest areas, and are placed in the plots after being scaled according to the size of the "marks". The prepared point clouds of forest plots with different degrees of crown overlap are then used as reference data to test the performance of the single tree detection algorithms under different forest conditions.

In all the experiments carried out, simulated data serves as a complement to the real ALS data. Our main purpose of simulation is to produce ALS point clouds of coniferous forest plots with higher stem density than real forest plots, and of different degrees of crown overlap, with which we can test the robustness of the proposed model more thoroughly. It is also ideal to validate the parameter estimation method we proposed (described in **Chapter 5** and **Chapter 6**), as it provides a chance to observe the behavior of the proposed methods estimated results regarding different degrees of crown overlap.

1.3.3 A probabilistic model for single tree detection

In **Chapter 5**, we present a hybrid framework for single tree detection from airborne laser scanning (ALS) data by integrating low-level image processing techniques (local maxima filtering and marker-controlled watershed segmentation) into a high-level probabilistic framework. The proposed approach models tree crowns in a forest plot as a configuration of circular objects. We take advantage of low-level image processing techniques to generate candidate configurations from the canopy height model (CHM): the treetop positions are sampled within the over-extracted local maxima via local maxima filtering, and the crown sizes are derived from marker-controlled watershed segmentation using corresponding treetops as markers. The configuration containing the best possible set of detected tree objects was estimated by a global optimization solver. To achieve this, we introduced a Gibbs energy, which contains a data term that judges the fit of the objects with respect to the data, and a prior term that prevents severe overlaps between tree crowns, on the configuration space. The energy was then embedded into a Markov Chain Monte Carlo (MCMC) dynamics coupled with a simulated annealing to find its global minimum. Parameter estimation is another issue: in this chapter, we introduced a Monte Carlo-based sampling method for estimating key parameters of the model. We tested the proposed model on a temperate mature coniferous forest in Ontario, Canada and on simulated coniferous forest plots with different degrees of crown overlap. The experimental results showed the effectiveness of our proposed method, which was capable of reducing the commission errors produced by local maxima filtering, thus increasing the overall detection accuracy by approximately 10% on all of the datasets.

1.3.4 Parameter estimation and model optimization

In **Chapter 6**, we have a further investigation of parameter estimation and optimization methods of the proposed model.

In the previous chapter, we have introduced a Monte Carlo based sampling method for parameter estimation. This reference-based parameter estimation method is trained on reference data, which are laborious, time-consuming and expensive to collect in forest environment. It is of great value to develop a method that can adjust the parameters automatically according to the processed data without using any reference data. This would finally facilitate an unsupervised detection of trees in forest of a large scale with our proposed model.

In this prospect, we developed an automatic parameter estimation method based on an Expectation Maximization (EM) algorithm, which does not rely on reference data. In addition, in this study, we further improve the estimation procedure by modeling the feature distributions with logistic regression directly, which avoids the empirical selections of distributions in approximating of the likelihood of features in the previous method. We also further investigate how initial condition of the EM procedure might influence the parameter estimation results.

Furthermore, we investigated in this chapter a modified optimization method resembling Reversible Jump Markov Chain Monte Carlo (RJMCMC) used in classical marked point process; we call it as prior-guided MCMC (PGMCMC), in which the prior information about tree density of the forest plot is used. We carried out a comparative study of the two optimization algorithms

on simulated and real forest plots. We also study how different cooling schedules influence the detailed behaviors of the two algorithms.

1.4 Outline of the Thesis

This thesis is organized in seven chapters. **Chapter 1** presents an introduction of the thesis with research context, scope of research and an overview of research methodology. Background information about LiDAR technology and a review of related research on single tree detection are described in **Chapter 2**. **Chapter 3** introduces the study area and data collection, along with a preprocessing procedure designed to alleviate the influence of shadow effects of ALS data on the shapes of tree crowns and produce pit-free CHMs. Simulation of ALS date of coniferous forest plots with different tree density and degrees of crown overlap is given in **Chapter 4**. **Chapter 5** describes a hybrid stochastic model combining low-level image processing techniques and high-level probabilistic models to detect single trees from ALS. **Chapter 6** visits some new parameter estimation and model optimization strategies for possible improvements of the original model. Finally, a thesis conclusion with achievements and discussion on the possibilities of future research is presented in **Chapter 7**. The structure of the main research themes and their organization in the dissertation is shown in **Figure 1.1**.

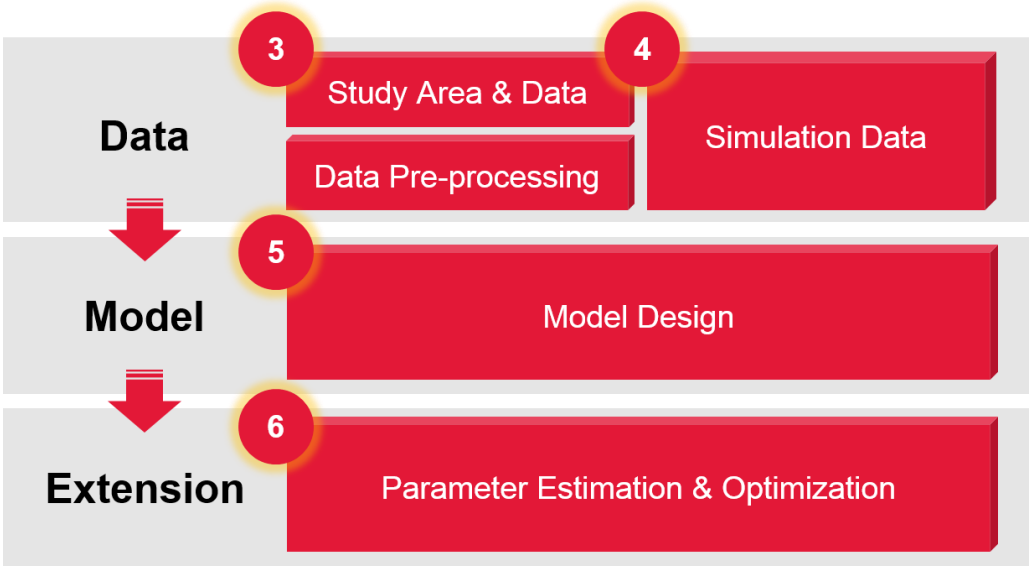


Figure 1.1 Structure of the research themes and their organization in the dissertation. Each number represents the chapter covering the specific topic(s).

1.5 List of Publications

Below shows a list of publications associated with the PhD research:

A. Peer-viewed Publications

Zhang, J., Sohn, G., & Brédif, M. (2014). A hybrid framework for single tree detection from airborne laser scanning data: A case study in temperate mature coniferous forests in Ontario, Canada. *ISPRS Journal of Photogrammetry and Remote Sensing, 98*, 44-57.

Zhang, J., Sohn, G., & Brédif, M. (2013). Single Tree Detection from Airborne Laser Scanning Data Using a Marked Point Process Based Method. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, II-3/W1*, 41-46.

Zhang, J., de Gier, A., Xing, Y. and Sohn, G., 2011. Full Waveform-based Analysis for Forest Type Information Derivation from Large-footprint Spaceborne LIDAR data. *Photogrammetric Engineering & Remote Sensing*, 77(3): 281-290.

Xing, Y., de Gier, A., **Zhang, J.** and Wang, L., 2010. An improved method for estimating forest canopy forest canopy height using ICESat-GLAS full waveform data over sloping terrain: A case study in Changbai mountains, China. *International Journal of Applied Earth Observation and Geoinforamtion*, 12(5): 385-392

Zhang, J. and Sohn, G., 2010. A Markov Random Field Model for Individual Tree Detection form Airborne Laser Scanning Data. *ISPRS Achieves of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XXXVIII, Part 3A*, 120-125.

B. Conference Contributions

Jwa, Y., Sohn, G., Pierre, T. S., Sarthou, C., & **Zhang, J.** (2014). 3D Recontruction of Power Line Models from Airborne Image-Based Point Cloud Data. *the joint International Geoscience and Remote Sensing Symposium (IGARSS 2014) / 35th Canadian Symposium on Remote Sensing (35th CSRS)*. Québec, Canada. (**Poster Presentation**)

Zhang, J., Sohn, G., & Brédif, M. (2013). A Hybrid Framework for Single Tree Detection from Airborne Point Cloud Data with Application in Boreal Forests, Ontario, Canada. *Silvilaser 2013, the 13th International Conference on LiDAR Applications for Assessing Forest Ecosystems*. Beijing, China. (**Oral Presentation**)

Zhang, J., Sohn, G., & Brédif, M. (2013). Single Tree Detection from Airborne Laser Scanning Data Using a Marked Point Process Based Method. *VCM 2013, The ISPRS Workshop on 3D Virtual City Modeling, 28 May 2013*, Regina, Canada. (**Oral Presentation**)

Zhang, J., & Sohn, G. (2012). A Marked Point Process for Single Tree Detection from Airborne Laser Scanning Data. *Silvilaser 2012, the 12th International Conference on LiDAR Applications for Assessing Forest Ecosystems*. Vancouver, Canada. (**Poster Presentation**)

Zhang, J. and Sohn G., 2011. Performance of Single Tree Detection Algorithms with the Effects of Crown Overlapping Based on Simulated ALS data. *Advanced LiDAR Data Processing*

and Applications, the 6th ISPRS Student Consortium and WG VI/5 Summer School, Fayetteville State University, North Carolina, USA, July 30 - August 6, 2010. (**Oral Presentation**)

Zhang, J. and Sohn, G., 2010. A comparison of single tree detection algorithms using simulated and real tree samples. *Silvilaser 2010, the 10th International Conference on LiDAR Applications for Assessing Forest Ecosystems*, Freiburg, Germany, September 14-17, 2010. (**Oral Presentation**)

Zhang, J. and Sohn, G., 2010. A Markov Random Field Model for Individual Tree Detection form Airborne Laser Scanning Data. In: *Photogrammetric Computer Vision (PCV) 2010, ISPRS Commission III Symposium*, Paris, France, September 1-3, 2010. (**Oral Presentation**)

Zhang, J., de Gier, A., Xing, Y. and Sohn, G., 2009. Derivation of forest type information from large-footprint spaceborne LiDAR waveforms. *Silvilaser 2009: the 9th International Conference on lidar applications for assessing forest ecosystems*, Texas A&M University, Texas, USA, October 14-16, 2009. (**Oral Presentation**)

Xing, Y., **Zhang J.**, 2008. Estimating canopy height of montane cool temperate forest using large-footprint spaceborne LIDAR. *ISPRS 2008: Proceedings of the XXI congress: Silk road for information from imagery: the International Society for Photogrammetry and Remote Sensing*, Beijing, China, July 3-11, 2008. (**Oral Presentation**)

Chapter 2
Background

2.1 LiDAR Technology

2.1.1 Principle of LiDAR

LiDAR (Light Detection And Ranging) is an active remote sensing technique using a laser beam as the sensing carrier (Wehr and Lohr, 1999). LiDAR systems measure the time elapsed from a pulse of the laser energy (usually in the near-infrared, for vegetation studies) generated from the sensor and reflected back from target. A distance measure between the sensor and the target can be calculated with respect to the speed of light:

$$R = c\frac{t}{2} \tag{2.1}$$

where t is the time interval between the emission and reception of a laser pulse, and c is the speed of light (approximately 30 cm/nanosecond). With known position of the sensor and precise orientation of the range measurements between the sensor and the target, the position (x, y, z) of the target can be determined. The principle of LiDAR measurements is the same regardless of the platform.

LiDAR system can be classified as profiling (recording only along a single narrow line of return directly below the sensor) or scanning systems (recording across a wide swath on either side of the sensor) (Dubayah and Drake, 2000; Lefsky et al., 2002; Lim et al., 2003). In forest mapping

and inventory, the most commonly used LiDAR system is airborne laser scanning (ALS), which are scanning laser systems mounted on airborne platforms, such as fixed-wing aircrafts or helicopters (**Figure 2.1**). A typical ALS system usually includes four components: 1) a laser ranging unit; 2) an opto-mechanical scanner; 3) data control, monitoring, and recording unit; 4) a positioning unit, including an kinematic global positioning system (GPS) receiver and an inertial measurement unit (IMU) (Wehr and Lohr, 1999).

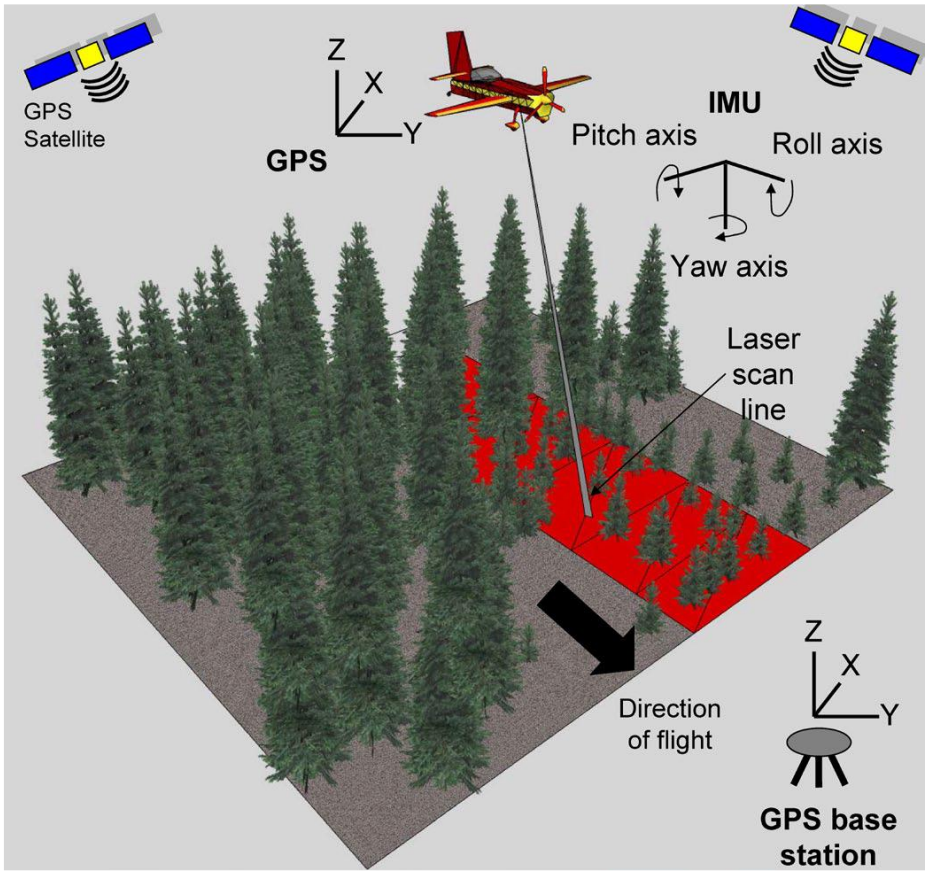


Figure 2.1. Schematic of an ALS system. Extracted from White et al. (2013b).

The laser ranging unit measures the time-of-flight of the laser pulse between the sensor and the target (round trip). The opto-mechanical scanner directs laser to the ground across the flight path, within a user-specified angle, such that many pulses are transmitted and received. The kinematic GPS is used to accurately measure the position of the platform, while IMU determines the three-axis orientation (i.e., roll, pith, and yaw) of the platform. The processing solution then determine the geographic location of each LiDAR pulse reflection integrating the information collected by different components, i.e. the laser instrument, GPS and IMU. This usually results in a set of points that gives the 3D location of each record LiDAR return in earth-reference coordinates such as Universal Transverse Mercator Projection (UTM easting, northing, elevation in meters) or longitude, latitude and elevation.

2.1.2 Discrete return and waveform ALS data

LiDAR instruments used in ALS systems can be categorized as full waveform or discrete return. Full waveform systems digitized the entire reflected or backscattered energy for each laser pulse as a single, continuous signal. In contrast, discrete return systems record single or multiple returns from a give laser pulse, usually done by converting the waveform data into return targets referenced in time and space (**Figure 2.2**). Common criteria for detection of a discrete return are when intensity value reaches a maximum (i.e., signal peak), when the intensity value exceeds a defined threshold of leading edge or when the intensity value of a peak exceeds a fraction of the peak maximum in which case the received signal must be saved temporarily (Stilla and Jutzi, 2008).

The intensity of LiDAR return is derived from the radar equation (Wagner et al., 2006; Wehr and Lohr, 1999). Assuming a diffuse reflecting surface equal to or larger than the laser footprint, the received optical power P_r is described by **Equation (2.2)**:

$$P_r = P_T \times \tau_{Total} \times \frac{1}{\omega} \times \frac{D^2}{R^4} \times \sigma_{cross} \tag{2.2}$$

where P_T is the transmitted power, τ_{Total} is the total transmission (i.e., the transmission of the receiver objective, the optical interference filter, and the scanning device as well as the two-way transmission of the atmosphere), ω is the divergence of the laser beam, D is the diameter of the receiving aperture, R is the distance from the laser scanning system to the reflecting surface, and σ_{cross} is the cross-section of the reflecting surface.

The cross-section σ_{cross} is proportional to the product of the reflectance ρ and the illuminated area A_s of the reflecting surface. The reradiation pattern is in general complex, but if we assume that the incoming radiation is scattered uniformly into a cone of solid angle Ω for simplicity (Wagner et al., 2006), the backscatter cross-section σ_{cross} can be further written as **Equation (2.3)**:

$$\sigma_{cross} = \frac{4\pi}{\Omega} \rho A_s \tag{2.3}$$

From equations **(2.2)** and **(2.3)**, we can see the received waveform depends mainly on the emitted pulse and the reflection surface, assuming that the influence of the receiver and amplifier as well as the atmosphere is constant. Considering the emitted pulse is not infinitely short, the received waveform will be the convolution between the emitted pulse and the surface properties (Wagner, 2010).

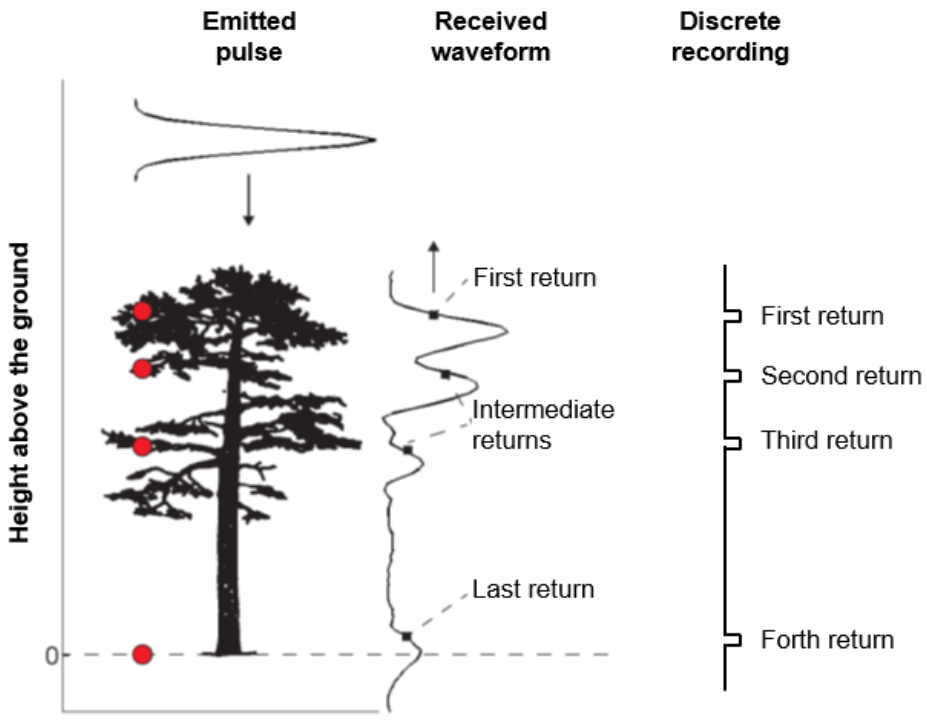


Figure 2.2. Recorded waveform and discrete returns of a laser pulse reflected from different surfaces. Modified from Lindberg (2012).

2.1.3 ALS characteristics for forest inventory

The most commonly used ALS are discrete return systems in forest inventory. In a forest environment, a laser pulse usually pass through the gaps in the canopy cover when it hits the forest canopy and intercepts different parts of the canopy, such as the trunk, branches and leaves before reaching the ground. This results in multiple returns for a single laser pulse. Current instruments are capable of recording up to five returns (White et al., 2013b). The first returns are mainly assumed to come from the outer surface of the canopy and the last returns from the ground, which is further utilized for extracting the terrain surface. Multiple returns produce useful information regarding the 3D forest structure (Hyyppä et al., 2008).

The characteristics of ALS data acquired in forest inventory can be influenced by the flight and instrumental configurations, such as the flying attitude and speed, the scan angle, the scanning mechanism of the laser scanner, the pulse repetition frequency, the pulse duration, and the wavelength of the laser (Petrie and Toth, 2009). In practice, the laser system specification and flight configuration play important roles in how the laser pulse interacts with the forest canopy, and in turn the derivation of forest variables. A larger scan angle or a higher flying altitude or speed will result in lower point density, while a higher pulse repetition frequency will result in a higher measurement density. Research also shows, for example, ground returns decreases as the scanning angle increases (Disney et al., 2010; Lovell et al., 2005). For the above reasons, the ALS system used for forest inventory purposes typically has short emit (3-10 ns), narrow-beam width (0.15-2.0 mrad), and infrared (0.80-1.55 μm) laser pulse at near-nadir incidence angle (<25 °) with high pulse repetition frequencies (50-200 kHz) (White et al., 2013b).

Post-processing of the data acquired by an ALS system produces a "point cloud" of the forest canopy structures, as well as the ground beneath. The point cloud is usually first filtered to differentiate vegetation returns and ground returns, and then interpolated into different surface model products, such that digital elevation model (DEM) from the classified ground returns and a digital surface model (DSM) from the non-ground returns (**Figure 2.3**). A Canopy Height Model (CHM), representing the height of canopy above ground level, is generated by subtracting the DEM from the DSM. The DEM can also be used to normalize the ALS points to heights above ground level. It is common to have a raster size of 0.25×0.25 m or 0.5×0.5 m resolution of CHM for single tree detection purpose depending on the density of the ALS data.

It has been reported that ALS measurements tend to underestimate tree height (Hyypp äand Inkinen, 1999; Lefsky et al., 2002; Nelson et al., 1984). Firstly, first returns reflect more often from the shoulder of the tree instead of the top. Although a laser pulse hits the top, the reflection may not be strong enough to be a recordable signal. On the other hand, dense understory causes overestimation in the DEM. Mainly for these two reasons, the CHM is usually underestimated. Increase of point density and reuse of full-waveform data for better terrain detection could improve the accuracy of tree height estimation.

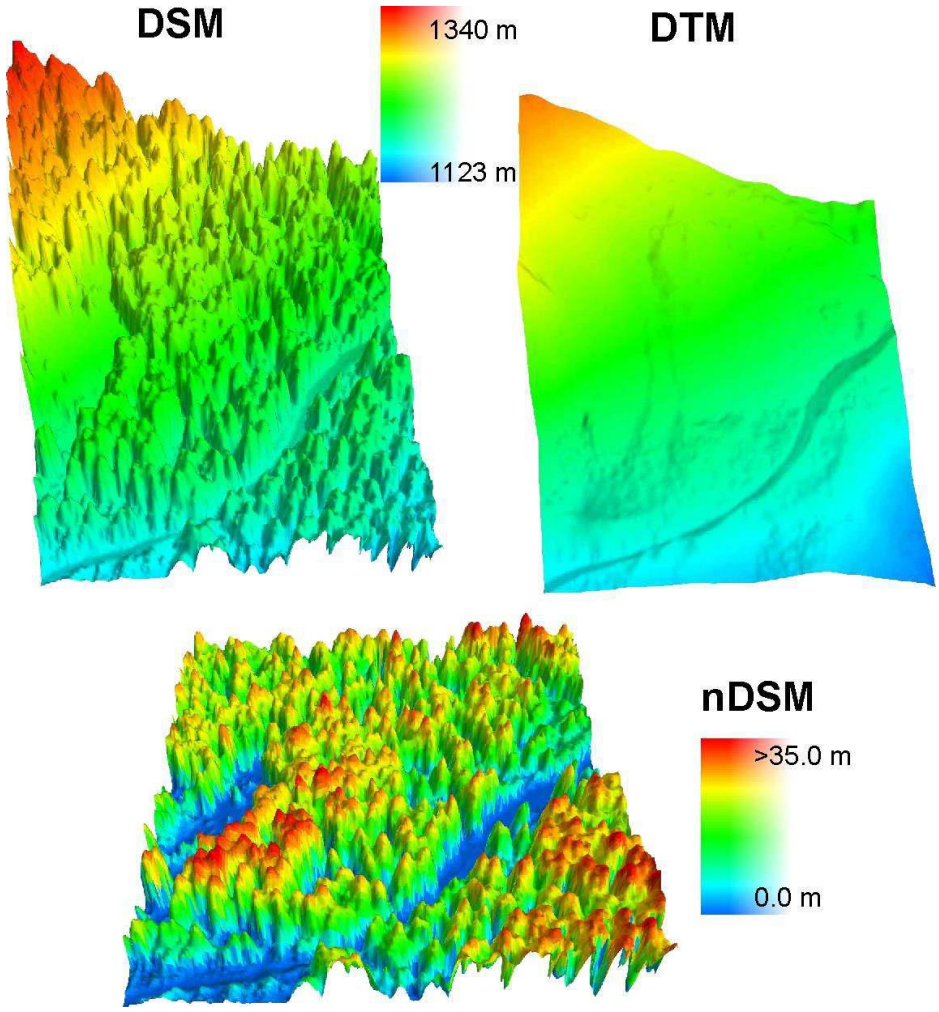


Figure 2.3. Basic products generated from the ALS point cloud: a Digital Surface Model (DSM) represents heights of objects above the ground surface; a Digital Terrain Model (DTM), also referred as a Digital Elevation Model (DEM), represents ground elevations; a Normalized DSM (nDSM), or a Canopy Height Model (CHM) in a forest environment, represents the normalized above-ground heights of non-ground objects, generated by subtracting the DTM from the DSM.

2.1.4 Other types of Lidar systems for forest inventory

Laser systems may be mounted on other different platforms, such as satellites (spaceborne LiDAR), tripods (terrestrial LiDAR), or moving vehicles (mobile LiDAR). Different types of LiDAR system has different applications in forest inventory according to platform specifications and data characteristics.

Spaceborne LiDAR

The Geoscience Laser Altimeter System (GLAS) onboard the Ice, Cloud, and land Elevation Satellite (ICESat) was the first large-footprint spaceborne full waveform profiling LiDAR. One of the scientific objective of the ICESat is the global measurement of canopy height (Zwally et al., 2002). The footprint of the waveforms are elliptical with the size of 95×52 m on average (Harding and Carabajal, 2005), and the spacing of footprints along track is 175 m with horizontal geo-location accuracy of 3.7 m.

Forest canopy metrics can be generated from the GLAS waveforms and these metrics were mainly used to estimate the canopy height and aboveground biomass (Lefsky, 2010; Lefsky et al., 2007; Rosette et al., 2008; Xing et al., 2010). Several recent studies also used GLAS data to generate large-area estimates of volume (Nelson et al., 2009) or derive forest type information (Zhang et al., 2011). More information about ICESat/GLAS can be found on (NASA, 2015).

Terrestrial Laser Scanning (TLS)

While ALS is the most frequently applied laser system in forestry, TLS is more feasible in the acquisition of ground truth or in small area monitoring. TLS usually operates by a laser scanner mounted on a tripod. TLS provides highly accurate 3D-locations of the targets surrounding the laser

scanner. “Phase-shift” scanners are mainly used in measuring individual trees or field plots, while pulse scanners can be used to map larger areas with a maximum distance of around one kilometer from the scanner (Vastaranta, 2012).

TLS produces a dense point cloud from the surrounding trees, e.g. 25 000 points/m^2. From the dense point clouds, tree and stand variables such as location, height, crown coverage, species, and stem curves can be measured (Hopkinson et al., 2004; Liang et al., 2007; Pfeifer and Winterhalder, 2004), thus TLS provides a means of objectively collecting various tree and forest variables that are laborious to acquire with traditional field measurements. The major drawback of TLS is that tree located in the shadow or blind spots behind the nearby trees to the scanner cannot be measured in single-scan mode. Automatic processing and forest variable measurements are currently being developed for TLS applications, e.g., integrating several scans of a plot resulting from multi-scan mode into a single point cloud.

Mobile Laser Scanning (MLS)

MLS can be seen as a method falling between ALS and TLS. MLS mounts a laser scanner on a moving vehicle such as a car or a logging machine. The application of MLS in forestry is being actively studied (Holopainen et al., 2011; Lin et al., 2012). MLS is also seen as a practical means to produce tree maps or inventories in urban forest environments (Holopainen et al., 2011). It is anticipated a combination of MLS and a logging machine could enable the automatic selection of

harvestable trees and enhancement in stem bucking[1]. However, MLS is still far from a widely used practical application in forestry. The situation may change due to the rapid development of automatic MLS and TLS data processing (Vastaranta, 2012).

2.2 Related Works on Single Tree Detection/Delineation

Single trees can be detected from high density ALS data and used to estimate the stem distribution and lists of trees based on the delineated tree crowns. The parameters that can be directly derived from ALS of individual trees, such as tree height, diameter, shape of the crown, and height profiles, can be further used to predict other important forest variables, including Diameter at Breast Height (DBH), tree volume, tree species, and biomass. The information on individual tree level is required by modern forest management and planning. Single tree detection is therefore essential to provide unbiased estimates of those pieces of information.

2.2.1 Surface model based methods

Accordingly, numerous methods have been proposed to detect single trees from ALS data. Most of the methods focus on the generation of a canopy height model (CHM), which provides an accurate representation of the outer surface of the tree canopy. The peaks and valleys on the CHM generated from high-density ALS data are better estimations of treetop positions and crown edges than can be obtained from aerial photographs or satellite imageries. Therefore, many studies have extended

[1] Stem bucking refers to the activity in forestry of cutting tree stems into shorter logs. The resulting logs are suitable for further processing as saw logs, pulp logs, poles and other products.

methods developed for passive optical imageries to detect single trees from ALS data. Those methods include, but are not limited to, local maxima filtering (Popescu et al., 2002; Wulder et al., 2000), region growing (Erikson, 2003; Solberg et al., 2006), valley following (Gougeon, 1995; Leckie et al., 2003), template matching (Korpela et al., 2007; Pollock, 1996), watershed segmentation and its variant marker-controlled watershed segmentation (MCWS) (Chen et al., 2006; Pyysalo and Hyyppa, 2002; Wang et al., 2004), and multi-scale segmentation (Brandtberg and Walter, 1998; Brandtberg et al., 2003; Jing et al., 2012).

In the literature, the concepts of tree detection and crown delineation are clearly defined (Hyyppa et al., 2008; Ke and Quackenbush, 2011). Tree detection refers to the process of finding treetops or locating trees, and crown delineation as the defining of crown outlines. From this point of view, the methods of local maxima filtering and template matching can be regarded as tree detection. While some other methods, such as region growing and marker-controlled watershed segmentation, involve the two consecutive processes. For methods falling within this category, treetops or tree locations are first detected using 'tree detection' algorithms, and then used as input 'seeds' or 'markers' in the subsequent image processing techniques to delineate the tree crown boundaries. Some references treat tree detection as equivalent to crown delineation, as individual trees are detected as the crowns are delineated, which is the case of valley following and multi-scale segmentation. We can see from the literature analysis that the processes of tree detection and crown delineation are intertwined in many applications. Therefore, some articles (Kaartinen and Hyypp ä, 2008; Kaartinen et al., 2012; Larsen et al., 2011) use individual tree detection (ITD) or single tree detection as a general term to describe any methods involving either or both of the two processes. In this

paper, we adopt the later definition of single tree detection as a general description of such approaches. Below is a brief introduction of some of the most commonly used single tree detection methods.

Local maxima filtering

Compared with passive optical images, ALS generated CHMs provide a more accurate outer surface of forest canopies. The "peaks" and "valleys" detected on CHMs are better representations of treetops and crown edges. Treetops can be detected by finding the local maxima with in fixed or variable window sizes in a CHM (Wulder et al., 2000). This method only provide locations for each crown (Hyyppäet al., 2001), but it is used as part of other methods that do define crown boundaries or as the base algorithm for further processing. Many examples can be found in Morsdorf et al. (2004), Chen et al. (2006), Zhao et al. (2009), Reitberger et al. (2009), Dalponte et al. (2011), Palenichka et al. (2013), etc..

Valley following algorithm

The valley following was originally presented by (Gougeon, 1995) for delineation of trees in a mature coniferous forest stand in Canada using aerial imagery. Gougeon (1995) considered that the shaded gaps between the mountain-like tree crowns could be represented by valleys. Instead of searching for local maxima as treetops, Gougeon's algorithm finds local minima as valley bottoms and follows the valleys to separate trees and applied a rule-based approach to further refine and outline tree boundaries. The algorithm was later applied to isolate individual trees from CHM of coniferous forest plots (Leckie et al., 2003).

Region growing algorithm

Region growing is a pixel-based image segmentation approach used to separate regions and recognize objects within an image. This approach to segmentation starts at some seed points. It examines neighboring pixels one at a time and adds to the growing region if they are sufficiently similar to the seed pixel. When a significant boundary is found, these pixels are labelled as belonging to the region specific to the seed pixel. For tree crown delineation, treetops or tree location pixels are often used as seed points, and the differences between tree crowns and the background used to determine the criteria to stop gowoing (Ke and Quackenbush, 2011).

Watershed segmentation and MCWS

The watershed segmentation can be classified as region-based segmentation approach. Firstly proposed by Beucher and Lantuejoul (1979), it is a well-known image segmentation method that imposes the advantages of other segmentation methods such as region growing and edge detection (Soille, 2003). The intuitive idea underlying this method comes from geography: it is that of a landscape or topographic relief which is flooded by water, watersheds being the divide lines of the domains of attraction of rain falling over the region (Serra, 1982).

Meyer and Beucher (1990) introduced marker-controlled watershed segmentation to overcome the over-segmentation problem of ordinary watershed segmentation. The idea is to perform watershed segmentation around user-specified markers rather than the local maxima in the input image (Chen et al., 2006). In MCWS, the image indicating the locations of markers is called a marker function and the image for producing watersheds is called a segmentation function. With

appropriate marker and segmentation functions, marker-controlled watershed segmentation can be used to delineate the boundaries of individual crowns.

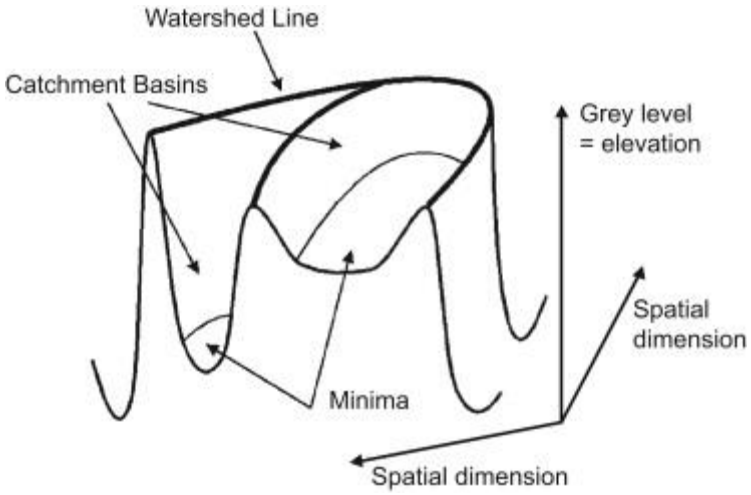


Figure 2.4. Topographic representation of a one-brand image. Extracted from Tarabalka et al. (2010).

2.2.2 Three dimensional methods

There is another group of methods to detect single trees directly from 3D ALS point clouds, attempting to bypass some inherent drawbacks of surface model based methods that trees located below the dominant canopy are typically missed.

Morsdorf et al. (2003) presented a two-stage procedure algorithm to retrieve individual trees in a coniferous forest plot. In this approach, tree locations were first detected for CHM as LM, and then trees were delineated in the three-dimensional point clouds with a k-means clustering algorithm using LM derived from CHM as seeds. This method is not advantageous compared with surface model based methods in terms of detection quality as it still relies on CHM derived LM.

Reitberger et al. (2009) developed a voxel-based method to segment individual trees in ALS point clouds. What is particular is that this method could recover part of the trees below the dominant tree layer by detecting tree stems in the dense point clouds produced from full-waveform ALS data. The detected stems were combined with LM detected from CHM, and a normalized cut was applied with these seed points to segment voxels derived from ALS data. For each voxel, features, i.e. the mean intensity and mean width of echo, are derived from calibrated waveform ALS data. Normalized cut divides the voxels into groups based on the feature dissimilarity between different groups as well as feature similarity within the groups. The method relies on that tree stems can be clearly distinguished in the ALS point cloud. It could be problematic if there is abundance of bushes and shrubs growing in the understory, which leads to difficulty in detecting stems from ALS data.

Wang et al. (2008) used a top-to-down growing method to segment individual trees in a voxel structure of ALS data. A hierarchical morphological algorithm is first applied to generate crown region at different height intervals. Tree crowns are then reconstructed by grouping neighboring crown regions generated at the same height intervals. The method is able to identify both canopy and over-topped trees, but it is sensitive to both the voxel scale and the size of morphological elements (Wang et al., 2008).

Ferraz et al. (2012) developed another clustering approach based on a mean-shift algorithm. The segmentation is applied to pre-defined forest layers: ground vegetation, understory and overstory. The method showed very promising results in well-stratified vegetation layers, with detection rate ranging from 98.6% for dominant trees to 12.8% for suppressed in the test plots. But

According to the authors, a more sophisticated method is required for processing more complex forest structures (Ferraz et al., 2012).

Currently, three dimensional methods for single tree detection are mainly in experimental stage and not applied in practical inventory. One reason is it very high computational cost in clustering dense ALS point clouds into individual trees. Also, there is a lack of comparison between the clustering methods with the surface model based methods. Furthermore, the ALS data specifications, such as point density, full-waveform or discrete return, required by the developed three dimensional methods varies from study to study, which also prevents wider application of those methods. Therefore, the work in this thesis focuses on making improvements on the existing surface model based methods.

2.2.3 Performances of common single tree detection algorithms

A variety of single tree detection algorithms has been developed, along with a range of evaluation methods applied. The evaluation methods differ in terms of the source of reference data and the procedure followed. From the review, it turns out the evaluation of tree detection results mainly focused on the proportion of tree detected correctly, while some included further evaluation on how well the delineated trees represent the actual tree crowns (Chen et al., 2006; Jing et al., 2012). Caution needs to be taken when interpreting and comparing their results due to the difference in study sites, forest conditions, data used, and evaluation methods (Hu et al., 2014). This is evident from **Table 2.1**, which summarizes the performances of the common single tree detection algorithms mentioned in above sections.

Table 2.1. Summary of the performances of the common single tree detection algorithms

Categorize	Examples	Algorithms	Data	Forest conditions	Detection Results
Surface model based methods	Popescu et al., 2002	Local Maxima Filtering	Point density: 0.7 pts/m2 CHM resolution: 1.5m	Decidous, coniferous and mixed stands of varying age classes and settings typical of the southeastern US	Detection quality not reported
	Solberg et al., 2006	Region growing	Point density: 5 pts/m2 CHM resolution: not given	Primeval heterogenous forest dominated by Norway spruce	Detection quality between 44% and 69% on test data for different processing settings
	Leckie et al., 2003	Valley following	Point density: 2 pts/m2 CHM resolution: 0.5m	Even aged (55 years old) Douglas-fir plots on the west coast of Canada	Detection quality not reported; Accuracy of good matches between 66% - 84%
	Chen et al., 2006	MCWS	Point density: 9.5 pts/m2 CHM resolution: 0.2m	Open oak savanna woodland with a scattered, clumped distribution of blue oaks	Percentage of correct detection between 56% and 65% depending on parameter settings
	Jing et al., 2012	Multi-scale segmentation	Point Density: 45 pts/m2 CHM resolution: 0.15m	Three temparate forest plots of mixed woods, decideous and coniferous trees in South Ontario, Canada	Detection accuracy range from 65% to 73%
3D methods	Morsdorf et al., 2003	K-means clustering	Point Density: 20-30 pts/m2 CHM resolution: 0.5-1m	Ofenpass coniferous forests dominanted by pine trees	Detection quality not reported
	Reitburger et al., 2009	3D segmentation	Point density: 10-25 pts/m2; CHM resolution: 0.5m	Four test sites containing alpine spruce forests, mixed mountain forests and spruce forests as the three major forest types	Detection accuracy range from about 45% to 60% with different paramenter settings

Effects haven been made by Kaartinen and Hyyppä (2008), Kaartinen et al. (2012) and Vauhkonen et al. (2012), to test and compare the accuracy of single tree detection algorithms. They have reported a high variation in the quality of the published methods under different forest types and with varying laser point densities. The results obtained for single tree detection have varied significantly from study to study and percentage of correctly detected trees has ranged from 40% to 93% (Kaartinen et al., 2012). Vauhkonen et al. (2012) tested several algorithms under different types of forests: Eucalyptus plantation in Brazil, coniferous and deciduous plots in Germany and mainly coniferous plots in Norway and Sweden. The reported tree detection rate varied between

54% and 86%. Results shown in those comparative studies have indicated the success of tree detection not only dependent on the algorithms themselves, but also on forest types and conditions, i.e. tree density and clustering.

2.2.4 Low-level image processing techniques in this research

Among the proposed surface model based methods, local maxima filtering (LM) and marker-controlled watershed segmentation (MCWS) are the most commonly used and are ready for operational application because of their rapid implementation while maintaining the capability to produce relatively accurate results (Kaartinen et al., 2012). Popescu et al. (2002) have been the first to test a variable window local maxima filtering on the CHMs, attempting to overcome errors of omission and commission associated with fixed window local maxima filtering (Hyyppä et al., 2001).

Once the treetops are detected, MCWS is well suited to delineate the tree crown segments from the CHM. In our application, user-specified markers, i.e. the pre-extracted local maxima are used as the marker function to perform the segmentation, while the original CHM is used as the segmentation function. For additional details of MCWS, please see Gonzalez and Woods (2008). In the resultant segmentation, there will be one segment corresponding to each marker; in the case of single tree detection, one tree crown will be captured by one treetop. This result indicates the detection accuracy of MCWS, subject to the accuracy of the pre-determined local maxima as true treetops in the previous stage.

The issue with LM is the selection of the filter window size and the determination of the relationship between the crown size and the tree height. In a comparison of tree detection algorithms

(Kaartinen et al., 2012), the local maxima-based approach tends to produce high commission errors, i.e. several local maxima are detected on a tree crown, and a single tree is segmented into several segments, thus increasing the number of detected trees (Kaartinen et al., 2012; Pouliot et al., 2002). Especially in coniferous forests, spurious treetops are detected within the tree crowns from large branches. In other cases, local maxima filtering produces a low commission error, and the omission error often increases because small tree crowns are more likely to be undetected (Gebreslasie et al., 2011).

2.3 Probabilistic Models in Image Analysis

Probabilistic methods represent another branch of powerful tools in image analysis. These methods have proven to hold great promise in solving inverse problems, including image segmentation, image restoration, and feature extraction (Descombes and Zerubia, 2002). In particular, stochastic models have evolved from random fields to object processes, and the work has shifted from an early focus on 'low-level' tasks that aim to de-noise, sharpen, and segment images to solving 'high-level' tasks of feature recognition, i.e., describing an image by its content (Van Lieshout, 2009). Additional details on low-level and high-level image analysis tasks can be found in Sonka et al. (2008).

2.3.1 Markov Random Fields and Marked Point Processes

Markov Random Fields (MRFs) were introduced into computer vision community by the seminal works of Besag (1986) and Geman and Geman (1984). A MRF models an image as a realization of a collection of random variables associated with each pixel. The appeal of a MRF is that it provides a probabilistic framework to encode contextual constrains into the prior probability, making it robust with respect to noise in the image (Li, 2009). However, the local definition of pixelwise

constrains in MRF makes it difficult to incorporate more global and strong geometric constrains (Descombes and Zerubia, 2002).

Marked point process models can be seen as an extension of MRF (Lafarge et al., 2010; Mallet et al., 2010; Ortner et al., 2008), such that random variables are associated not with each pixels in the image but with random configurations of geometric objects or shapes describing the image. This means marked point processes could model higher level geometrical primitives more naturally, while inheriting the merit of inclusion of priori knowledge on spatial patterns of features. Additionally, although there are high-level MRF models designed beyond pixel level (Li, 1994), the number of nodes and their relations need to be fixed before model optimization. The marked point process model, however, is more flexible in terms of random structure representation. The number of nodes can be changed and relations between nodes can be modified during the optimization process.

Marked point processes (Van Lieshout, 2000) are among the most efficient stochastic models used to exploit the random variables whose realizations are configurations of geometric objects or shapes. Generally, in these processes, after a probability distribution measuring the quality of each object configuration is defined in the configuration space, the maxima density estimator is searched for by the Markov Chain Monte Carlo (MCMC) sampler (Hastings, 1970) coupled with conventional simulated annealing (Metropolis et al., 1953). This process has led to convincing experimental results in various image analysis and feature extraction applications, such as road networks extraction (Lacoste et al., 2005), road mark detection (Tournaire and Paparoditis, 2009), and 3D building reconstruction (Lafarge et al., 2008; Ortner et al., 2008; Tournaire et al., 2010).

2.3.2 Application of probabilistic models in single tree detection

Likewise, several stochastic models have been proposed to detect tree crowns from remote sensing data. Descombes and Pechersky (2006) have presented a three-state Markov Random Field (MRF) model to detect the tree crowns from aerial imagery. This approach addressed the problem as an image segmentation problem and works on the pixel level. Each pixel is assigned to one of the following three states: (i) *vegetation*, (ii) *background*, and (iii) *center of trees*. Although the MRF was defined on the pixel level, the label update was performed on the object level using elliptical templates of crowns.

Furthermore, Perrin (2005, 2006) has employed marked point processes to detect tree crowns in plantations from color infrared (CIR) aerial imageries. Tree crowns in the remote sensing image are modeled as a configuration of discs or ellipses. In both of the studies, tree crowns were detected by maximizing a Bayesian criterion, such as *Maximum A Posteriori* (MAP), which became an energy minimization problem and was solved in a simulated annealing framework.

These stochastic models provide a powerful framework to allow the inclusion of spatial interactions between objects in the prior while enabling a measure of consistency between objects and the image in the data term. However, the inherited property of stochastic models requires exploration of a large configuration space searching for the optimal configuration, especially for non-data-driven models, which do not employ any low-level information that can be extracted from the images. The optimization process is typically lengthy and computationally expensive.

2.4 Summary

In the chapter, we have given an overview of LiDAR systems and their applications in forest inventory. The general principle of LiDAR system is first presented and the characteristics of ALS is described in more detail as ALS is the primary LiDAR systems applied in forest inventory. LiDAR mounted on other platforms and the state-of-art of their application in forestry is also briefly included. Then, related works on single tree detection/delineation methods using ALS data are reviewed, followed by an introduction of probabilistic models, especially MRF and MPP, and their applications in the field of image processing.

In this study, we intend to present a hybrid framework to detect single trees from ALS data by integrating the low-level image processing techniques, i.e., LM and MCWS, into a high-level probabilistic model. The proposed model aims to improve the detection accuracy compared with traditional LM. Moreover, this model potentially accelerates the optimization process compared with classical stochastic models, e.g., marked point processes, by sampling in a reduced configuration space by utilizing image features extracted by LM and MCWS. The parameter estimation of the stochastic model is another issue. In most cases, the parameters are tuned by trial and error. We will address the problem of parameter estimation by a Monte Carlo based sampling method. The details of this hybrid framework is presented in detail in **Chapter 5**.

Chapter 3
Materials and Data Pre-processing

3.1 Study Area and Data

3.1.1 Study area

The study area is a temperate mature coniferous forest located in the Great Lakes-St. Lawrence region approximately 60 km east of Sault Ste. Marie, Ontario, Canada (**Figure 3.1 (a)**). The natural vegetation dominant in the coniferous forest is eastern white pine (*Pinus strobus*) and jack pine (*Pinus banksiana*), mixed with some red pine (*Pinus resinosa*) and black spruce (*Picea mariana*). The forest has an intermediate dense canopy with some open space. The canopy height is homogenous with an average height of approximately 20 m. There are some small white pines and shrubs growing in the understory with a height of approximately 2-3 m (**Figure 3.1 (b) and (c)**).

3.1.2 Field survey

To test the proposed single tree detection model, three plots with sizes of 82 × 95 m^2, 50 × 50 m^2 and 80 × 80 m^2 were selected, and a field survey was conducted in August 2009. The forest mensuration campaign determined the tree height (h_i, m) with a Vertex hypsometer and the Diameter at Breast Height (DBH) with a DBH tape. The positions of trees with a height greater than 5m ($h_i \geq 5$) were determined using GPS and a total station. The crown width and species were also measured and recorded. The stem densities of trees with a value of $h_i \geq 5$ are 154/ha, 160/ha and 190/ha, with increasing values for the three study plots.

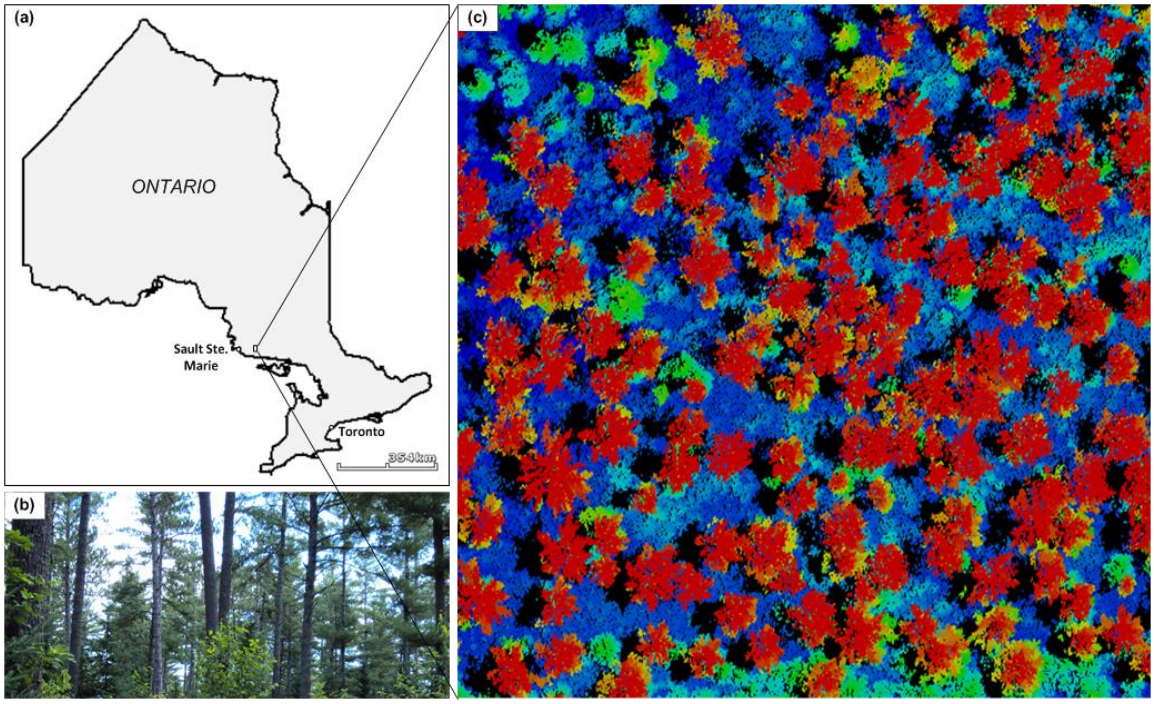


Figure 3.1: (a) Location of the study area in the Province of Ontario, Canada; (b) a photo and (c) ortho view of the ALS data of a forest plot in the study area rendered by height.

3.1.3 Airborne laser scanning data

The ALS data were acquired over the study area by a Riegl LMS-Q560 laser scanner during the same period as the field work. The flight was performed at a height of approximately 300 m above the ground with a maximum scanning angle of 22.5 º, rendering a swath width of approximately 300 m. The flight line was designed to pass over the planned forest plots; therefore, they were located in the middle part of the swath, and the shadow effect of the crowns can be minimized for the plots of interest. The device recorded full-waveforms that were processed into discrete point clouds with up to 5 returns per pulse. The data collection configuration yielded a high point density of approximately 30 points per m^2 over the forested area. The returns were classified as ground and

vegetation points using TerraScan software (TerraSolid Ltd, Helsinki, Finland). The CHM with a resolution of 0.5 m was derived as the difference between the digital surface model (DSM) and the digital elevation model (DEM), interpolated from vegetation points and ground points, respectively (Hyyppä et al., 2001). Details on ALS data pre-processing are described in **Section 3.2**.

3.1.4 Auxiliary data (Simulated data)

Vauhkonen et al. (2012) noted that the performance of the ITD algorithms typically depends on the tree density and the spatial distribution of trees, i.e., clustering patterns. To test the robustness of the proposed model more thoroughly, simulated ALS data of coniferous forest plots with a higher stem density than real forest plots and different degrees of crown overlap were also prepared in our study. First, three forest plots, each with a size of 100×100 m^2, were generated with a hard-core process in which the crown overlap was controlled by the interaction distance specified in the hard-core process. The smaller the interaction distance in the hard-core process, the more likely the tree objects will be overlapped in the resultant plots. **Figure 3.2(a)-(c)** show the three resulting point processes. With an increasing degree of crown overlap, the tree density in the plots also increases. The stem densities of trees with a value of $h_i \geq 5$ in the three forest plots are 186/ha, 234/ha and 261/ha, respectively.

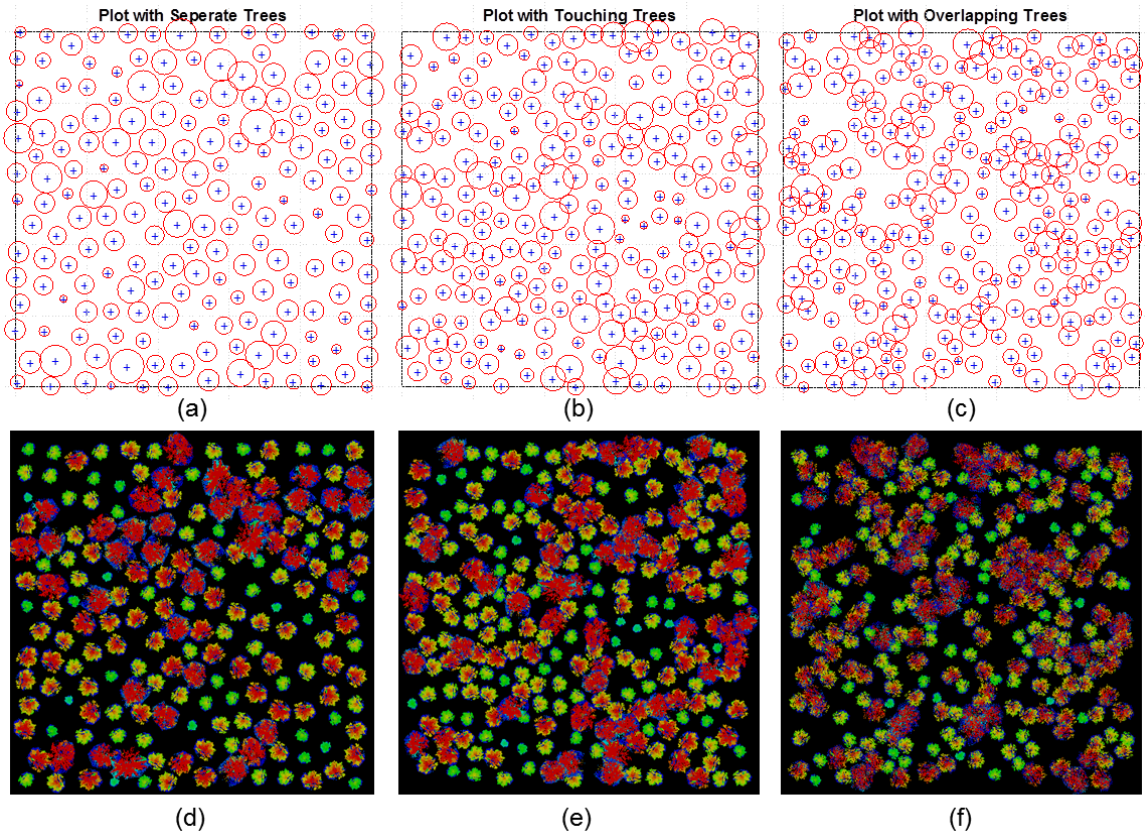


Figure 3.2: (a)-(c) Point process simulated forest plots with different degrees of crown overlap: (a) plot with separated crowns; (b) plot with tree crowns slightly touching each other; (c) plot with overlapping crowns. (d)-(f) the corresponding ALS point clouds of the three forest plots generated.

ALS point clouds of individual trees were then selected according to the crown size from a coniferous tree template library and placed in each position to synthesize the ALS data of the forest plot. The tree template library was prepared from ALS data acquired from the study area we surveyed. The detailed procedure is described in **Chapter 4**. The generated ALS point clouds viewed in nadir direction are shown in **Figure 3.2(d)-(f)**. The plots from left to right show forest plots with separated, touching and overlapping tree crowns, respectively.

In the simulated forest plots, the tree position, height and crown size are exactly known, therefore providing ideal reference data to examine the performance of our proposed model under different forest conditions. The simulated ALS data can also be used to validate the parameter estimation method proposed in **Chapter 5** and to investigate the influence of crown overlap has on the parameter settings and the robustness of the proposed model.

3.2 ALS Data Pre-processing

The general procedure of ALS data pre-processing for surface model based single tree detection methods includes: ALS data acquisition and processing, filtering, surface model generation and CHM pre-processing. In this section, we will have a description of each step of the ALS data pre-processing chain, especially the improvements on the last two steps of the workflow: surface model generation and CHM pre-processing. The improved pipeline can alleviate the shadow-effects of ALS data, which have great influence on the crown representation on the CHMs produced by conventional procedures, and produce pit-free CHMs.

3.2.1 ALS data acquisition and processing

The ALS data in this study were acquired by GeoDigital International Inc. in August 2009 using a Riegl LMS-Q560 instrument. The initial processing of ALS data was carried out by the company carried out the laser survey campaign. In this stage, Laser rangers, GPS and IMU measurements are combined together in this stage based on a time stamp attached to each data source.

The small-footprint airborne laser scanner collects full-waveform data at a pulse frequency of 111 kHz (Wagner et al., 2006). Waveform processing was performed using RiAnalyze© Software from Riegl (Riegl, Austria) based on a Gaussian pulse estimation technique. 3D points were derived from the gravity centres of the Gaussian components fitted on each echo pulse, describing the position of targets having interacted with the laser pulse (Chauve et al., 2009). Before applying ALS filtering algorithms in next step, we de-noised the point clouds by removing isolated high points and low points from the raw point cloud data. Isolated points are defined as the points having less than 10 neighbors within the spherical space with radius of 5m centered by themselves.

3.2.2 ALS data filtering

In this step, we classified the ALS point clouds into returns from ground and returns from vegetation. The common approach is to filter out ground returns from the point clouds first. The rest of them are assumed to be returns from vegetation in a forest environment. In some applications, vegetation points can be further classified based on the distance from the ground.

Different algorithms have been proposed to filter ground returns from ALS point clouds so as to generate DSM (Axelsson, 1999; Elmqvist, 2001; Kraus and Pfeifer, 1998; Vosselman and Maas, 2001). The method applied in this study comes from Axelsson (1999) that has been implemented in Terrascan software. In this method, a progressive TIN densification approach is used to classify group returns from ALS point clouds. A sparse TIN is first generated from neighbourhood minima. Then more points are added to progressively densify the TIN if they fall within the defined thresholds in each iteration. The ALS points are classified as ground returns and non-ground returns

when termination conditions are met. As mentioned earlier, non-ground returns are labelled as vegetation.

3.2.3 Surface model generation

Surface models, including DEM, DSM, and CHM, were then generated from the classified ALS data. The generation of the DEM is straightforward. Ground return points were interpolated into a regular grid with a 0.5 m resolution using TerraScan software (TerraSolid Ltd., Helsinki, Finland).

Usually, a DSM is generated in a similar way as the DEM. Typically, non-ground returns were used to produce the DSM. Each DSM cell was assigned with the maximum height value of the points within it. However, the DSM generated with this method suffers from the shadow effects of laser scanning on the side of a swath. There is usually no points collected in the areas shadowed by tree crowns. This will results in gentle slopes on the shadow side of tree crowns on the DSM. The geometric shapes of tree crowns eventually present in the CHM will be distorted, which could affect the subsequent tree detection and the accuracy of extracted crown features.

To alleviate the influence of shadow effects on the geometric shapes of tree crowns presented on the CHM, we propose an improved procedure to generate the DSM. We firstly exported the DEM generated in the previous step as lattice points. Then, these DEM lattice points were combined with vegetation points to go through the conventional DSM generation method. By adding the DEM lattice points in the non-ground points, the shadow areas will be filled up with points should potentially returned from the ground. A CHM at 0.5 m spatial resolution was finally calculated by subtracting the DEM from the DSM.

3.2.4 CHM pre-processing

The ALS produced CHMs are commonly affected by pit problem, i.e. holes or pixels with a much lower digital number than their immediate neighbors, due to factors such as sampling design, point density, and post-processing of the ALS data including point classification and interpolation (Ben-Arie et al., 2009). Large gaps between branch clusters of some tree species, such as coniferous trees, is also a reason why there are pits on CHMs. These pits can hamper the segmentation and delineation of tree crowns, and correct extraction of forest variables from the CHM.

Previous studies recommended applying smoothing methods such as a median filter or a Gaussian filter to smooth the CHM and reduce the data pits. However, smoothing can result in missing of small trees and modifies the CHM in a way leading to subsequent misinterpretation of biophysical tree parameters (Solberg et al., 2006).

We employed an automatic hole-filling algorithm to remove the pits from the CHMs similar with the one developed by (Véga and Durrieu, 2011). The algorithm first examines if a given pixel value is lower than a given threshold (fixed to 2m in our study) compared to its 8-connected neighborhood. If yes, the pixel value is replaced by the mean value of its 8 neighbours. Otherwise, the algorithm tests if the condition could be satisfied by 4-connected neighborhood (either vertical cross or oblique cross). Again, if the condition is met, the pixel value is replaced by the mean value of those 4 connected pixels. The hole-filled CHM is used as the starting point for the single tree detection method we describe in **Chapter 5**.

Chapter 4
ALS data simulation of forest plots

4.1 Introduction

Single tree detection using small-footprint airborne LiDAR data has been extensively studied over the past few years. It is important for precise forest management and serves as basis for single tree based analysis and forest characteristics extraction, but it has been proven as a challenging task by numerous studies. Different researches test their methods on datasets with totally different forest types and conditions, which makes it even harder to tell the performance variances is a result of the algorithm applied itself, or the influence of different forest conditions (e.g., stem density, crown overlap, and crown story layers, etc.) or dataset properties (e.g., pulse density and number of returns per pulse), not to mention examining more precisely about how forest condition plays a role on the performance of a method, and how the method should be adjusted to cope with certain forest condition.

Even in some situations, the validation of the single tree detection algorithms is problematic, either it is because of lack of ALS data, or the unavailability or relatively small size of field-surveyed reference data, let alone testing them under different forest conditions. Therefore, before diving into proposing our own single tree detection method, in this chapter, we want to address this problem by simulating ALS data of forest plots with different condition, which has the following merits:

(1) Making it possible to test the method in a controlled manner by simulating ALS point clouds of different forests conditions, to see the how the algorithms works under different conditions, such as forest with separate and overlap crowns.

(2) Without having to collect field data. Field surveys in the forests are usually expensive and time-consuming. We can get more precise and controlled reference data without having to spend a lot of money and time in the field.

(3) Making use of the existing data set. It is able to get more value from the existing data set. The method we propose intends to make use of the point clouds of single trees collected in the existing ALS data. Although there are a few commercial software can generate tree models, it is hard to validate how close to reality the model they produce are as one does not know how they do this due to intellectual property reasons. Furthermore, producing point clouds from those software generated tree model requires additional knowledge and technology which may be complex or not publicly available.

Point process theory in applied statistics has been applied in forestry for describing and analyzing the spatial variability of forest stands as well as for simulating the forest pattern and predicting its dynamics (Stoyan and Penttinen, 2000). It could provide an ideal tool for simulating the forest plots by which the degree of interaction of trees can be modelled. As such, the study tries to employ this method and simulate forest plots of different degrees of canopy overlap. The simulated forest plots will be used throughout in this research to test the performance of our proposed

model for single tree detection, and also serve as a great dataset to validate the parameter estimation method and to observe the behavior of the estimated parameters in different forest condition.

The specific objectives of this study are: (1) to simulate ALS point clouds of forest plots with different degrees of canopy overlap using a point process; and (2) we will also showcase with the simulated dataset on how the performances of LM approach are influenced by forest condition with different degrees of crown overlap.

4.2 Methodology

4.2.1 Point processes in forestry statistics

This section details point processes in forestry statistics and how they can be used to simulate forest plots with different of crown overlap.

In forestry statistics, the distribution of trees in a forest plot can be regarded as a point process or as a marked point process. The "points" are the location of the trees and the "marks" are tree characteristics such as crown diameter. Point processes have been applied in forestry statistics with two perspectives: for analyzing the spatial variability of forest stands to understand and quantify ecological relationship; and for simulating the forest pattern and predicting its dynamics (Stoyan and Penttinen, 2000).

Broadly speaking, there are two types of point process models. The basic "reference" model of a point process is the uniform Poisson point process, or homogeneous Poisson process. It corresponds to the hypothesis of Complete Spatial Randomness (CSR), which assumes points are

distributed uniformly in space and they are independent of each other. Such a model is implausible in the ecological system, as in a typical forest, trees growing in the neighborhood would competing for common resources and influence the growth and mortality of each other (Grabarnik and S ärkk ä 2009). Although clustered patterns usually appear at the initial stage of forest development, it is observed that there is always a tendency towards regularity with the evolution of a forest (Comas and Mateu, 2007; Stoyan and Penttinen, 2000).

All the other point process models fall within the other type of point process: inhomogeneous point process. In modern forestry statistics, two classes of such process are of particular interest: Cox and Gibbs processes. Cox processes used to be a popular class of point process models for forestry. A cox process is also called "double stochastic Poisson process" as the process can be seen as the result of a two-stage random mechanism (Stoyan and Penttinen, 2000). Matérn's cluster process and Thomas process are two common Cox models. The drawback of the Cox models is that they do not guarantee a hard-core distance between trees and therefore too variable at short distances, which means it may produce locally severely clustered points.

Gibbs point process can model the pairwise interaction in the spatial pattern, which is proven to be very suitable for modeling the competition between neighboring trees. This type of process can be defined by the probability density function by **Equation (4.1)**:

$$f(X) = \frac{1}{Z}\exp\big(-U(X)\big) \tag{4.1}$$

where X represent the point process, Z is a normalizing constant and $U(X)$ denotes an energy function (see **Equation (4.2)**) which contains a pairwise interaction function.

$$U(X) = n\alpha + \sum\sum_{i<j} \phi\left(\left\|x_i - x_j\right\|\right) \tag{4.2}$$

where α is a local term, n is the number points, and $\left\|x_i - x_j\right\|$ denotes the distance between point i and j and $\phi(\cdot)$ the pair-wise potential function. $\phi(\cdot)$ defines the pairwise of neighboring points.

According to the design of the interaction function, a number of models are further defined within Gibbs point process, such as the hard-core process, the Strauss process, the Strauss hard-core process and the soft-core process. In a hard-core process, there will be no two points located within a specified distance of each other.

The interaction between points in the Gibbs point processes mentioned above are all symmetric. However, in the real case, the interactions between different trees are not symmetric. For example, bigger trees have more influence on the small trees in their neighborhood. Marked Gibbs point processes can take the "marks", which could be both qualitative and quantitative characteristics of forests in this case, into account and provide a more powerful way to include the asymmetric competition between plants in a model with greater flexibility (Comas and Mateu, 2007; Cressie, 1992). For detailed definition of those point processes, one can reference to Neeff et al. (2005) or Comas and Mateu (2007).

4.2.2 Simulation of forest plots using point process

As mentioned, a point process is employed to simulate forest plots with different extent of canopy overlap, which is useful for testing how the single tree detection algorithms work under different scenarios.

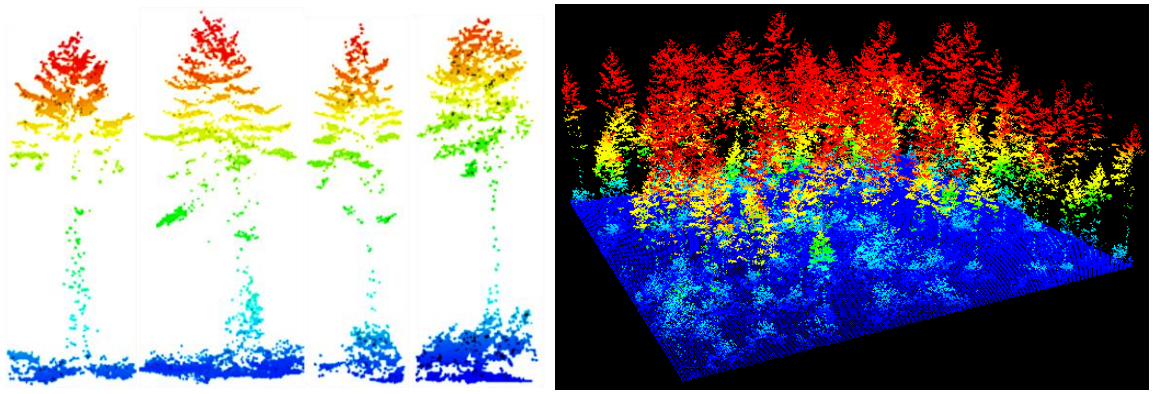
For this purpose, the "marks" of the point process are the crown diameters and we consider only the horizontal interaction of neighboring trees in the simulation, in this case the canopy overlap. There are at least two choices for modelling the "points" - the location of trees and the "marks" - the crown diameters: they can be modelled jointly or considered as independent processes and modelled randomly, and different choices typically lead to different stochastic models. We here choose the latter as a relative simple but effective way to do the simulation.

We first model the location of the "points". As rigidly quantification as to what extent the crowns overlap is not necessary and sometimes unrealistic for the events and extents of crown overlapping are considered totally random, we choose the hard-core process to model the spatial distribution of tree locations, and then attach the "marks", which adhere to the empirical distribution, to the "points". The empirical distribution of crown width is acquired from the field survey, and marks that adhere to this distribution can be generated. In the point process simulation, Markov Chain Monte Carlo (MCMC) method combined with Hastings-Metropolis and spatial birth-and-death algorithms are used to simulate the target point process. In the model, the interaction between trees is controlled by the distance specified. If the distance is set large, for example, as the mean plus the standard derivation of the crown width, the chance the tree crowns overlap will be small and vice versa. Three scenarios are simulated in this way, by setting the interaction distance equal

to the mean plus the standard derivation of the crown width, the mean of the crown width and the mean minus the standard derivation of the crown width, representing forest plots with separated canopies, canopies slightly touching each other and overlapping canopies respectively. The simulation is implemented in R using spatial statistics package spatstat (Baddeley and Turner, 2005).

4.2.3 Point clouds generation for simulated forest plots

Now we have simulated forest plots with information about tree position (x, y) and crown width r for all the trees in the plots. To simulate LiDAR point clouds of a forest plot, a tree cloud template library is firstly prepared. The library contains separate files of point clouds of single trees, which can be clearly identified and manually selected and clipped from multiple return, high density, small-footprint ALS data acquired in a coniferous forest area as shown in **Figure 4.1(a)**.



(a) (b)

Figure 4.1. (a) Four tree templates clipped from ALS point clouds acquired in forest plots for illustration; (b) Oblique view of ALS point cloud of a simulated forest plot using point process and the tree template library.

Then tree characteristics of each tree template in the library, such as tree height and the crown width, are extracted. Afterward, the point cloud for the plot is generated as the algorithm shown in **Table 4.1**.

Table 4.1. Point cloud generation algorithm with simulated forest plots and tree library

Algorithm: Point cloud generation for a simulated forest plot

Input: Position (x, y) and crown width r of all trees in the plot

Tree point cloud template

Output: Point cloud file the forest plot

repeat:

- Pick i-th tree in the plot with position (x_i, y_i) and crown width r_i
- Select the tree in the template of smallest crown width difference, with crown width r'
- Calculate the scale ratio of the template $R_s = \frac{r_i}{r'}$
- Generate a random angle $\theta \in [0, 360[$
- For all the points in the tree template, scale at ratio R_s, and rotate with angle θ, then translate to position (x_i, y_i)
- Append the processed point clouds to file

until: all trees in the plot visited;

In the simulated forest plots (see **Figure 4.1(b)**), all the tree parameters are exactly known and are used as reference data to test the single tree detection algorithm.

4.2.4 LM approach for singe tree detection

The LM approach is used here as a showcase to demonstrate how the performance of single tree detection is influenced by different tree density and crown overlap using the simulated ALS data.

The local maxima (LM) approach has been extensively used to detect single trees from remote sensing imagery (Pouliot et al., 2002; Wulder et al., 2000) and airborne LiDAR data (Chen et al., 2006; Popescu et al., 2002; Reitberger et al., 2009). A local maximum filter passes over an image to find the pixels having a larger value than all the pixels in the window and these local maxima are identified as tree tops.

Researches of Wulder et al. (2000), Popescu et al. (2002) and Chen et al. (2006) all suggested that the determination of the LM filter window size has influence of the detection results, and variable size window performs better than static size window in term of reducing both commission and omission errors, considering the relationship between the tree height and canopy width.

In our research, the way in which the window size affects the LM detection results on LiDAR data is further explored using the three simulated forest plots. A circular LM filter is used and the ratio of the LM window size to the pixel value will be set larger than, nearly equal to and smaller than the mean crown width to tree height ratio of the tree templates, to see its effect on the detection rate.

4.3 Results

4.3.1 Simulated ALS data of forest plots

The ALS data used in this study was described in **Section 3.1.3**, from which we collected the tree templates for the simulation. Field survey data was used to estimate the empirical distribution of tree characteristics, i.e. crown width, which obeys a Gaussian distribution with mean of about 5.25m and the standard deviation of 1.25m. The fieldwork was also helpful for identifying suitable

trees as templates from the ALS data. Three coniferous forest plots were generated as shown in **Figure 3.2**.

The interaction distance used in the hard-core process to simulated forest plot with separate trees was set as the mean plus the standard derivation of crown width. As we can see from **Figure 3.2 (a)**, the trees are mostly separated and have no overlaps. The interaction distances used in the hard-core process were set smaller and from **Figure 3.2 (b)-(c)**, more trees were overlapping and the degree of overlap increased. **Figure 3.2 (d)-(e)** show the corresponding point clouds of the three forest plot. A summary of simulated ALS data is described in **Section 3.1.4**.

4.3.2 LM detection results

The minimum ratio of the crown width to tree height of the entire tree templates was 0.156. In this case, three ratios of variable size window to pixel value were used to test the performances of LM filter under three forest conditions with increasing degree of overlapping.

Table 1 shows the LM detection results in the three forest plots. The three ratios used were 0.25, 0.20 and 0.15. The third ratio 0.15 is below the minimum crown width to tree height ratio, which means the LM filter window size would always be smaller than the actual crown size, and we can see from the table, a small window size produces a great number of commission errors; while for the first ratio 0.25 which is largest among the three, we can see the omission error it produced is the largest. Among the three forest plots, the detection rate of LM on the plot with separating canopy crowns is best. The omission errors were low, which means without canopy occlusion from surrounding trees, trees can be detected relatively easily. When the overlap degree

increases, it is very likely that shorter trees occluded by taller trees in their neighborhood can't be detected. For the same window size ratio, in terms of omission error, LM always performs worse on the plot which has a higher degree of overlap. What can be concluded as well is that a smaller window size is not always good. When the window size ratio of LM filter drops beyond a certain degree, the omission error is reduced really slowly, but meanwhile, the commission error can increase exponentially and greatly degrade the overall detection quality. For instance in the separated plots, when the window size ratio drops from 0.20 to 0.15, the omission error hardly changes, while the commission error increases from 37% to 209% and the overall quality decreases from 72.2% to 32.1%.

Table 4.2. LM detection results of three forest plots with ratios 0.25, 0.20 and 0.15 of variable size window.

Ratio of Variable	Detected	Correct		Commission		Omission		Overall
Size Window	Trees	no	%	no	%	no	%	Quality
Plot 1 - separating stand; 186 trees								
0.25	192	184	98.9%	8	4.3%	2	1.1%	94.8%
0.20	253	184	98.9%	69	37.1%	2	1.1%	72.2%
0.15	575	185	99.5%	390	209.7%	1	0.5%	32.1%
Plot 2 - touching stand; 234 trees								
0.25	221	215	91.9%	6	2.6%	19	8.1%	89.6%
0.20	312	220	94.0%	92	39.3%	14	6.0%	67.5%
0.15	674	227	97.0%	447	191.0%	7	3.0%	33.3%
Plot 3 - overlapping stand; 261 trees								
0.25	218	214	82.0%	4	1.5%	47	18.0%	80.8%
0.20	311	239	91.6%	72	27.6%	22	8.4%	71.8%
0.15	697	245	93.9%	452	173.2%	16	6.1%	34.4%

4.4 Conclusion

Point process theory provides a powerful tool to the fields where the spatial pattern of complex forest ecosystems needs to be modelled. We have shown in this chapter that employing a point process to simulate ALS point cloud data of forest plots with different degrees of crown overlap. The simulated forest plots provide excellent reference dataset and fully controlled environment to test single tree detection algorithms, which allows us to have a better understanding of how they work under different forest conditions. An experiments with the simulated ALS data showed that LM based approaches are both sensitive to filter window sizes as well as specific forest conditions, i.e. crown overlap degrees in this study, which may result in different commission and omission errors.

Chapter 5
A stochastic model for Single Tree Detection

5.1 Introduction

In this chapter, we introduce a novel stochastic model to automatically extract single trees from ALS data generated canopy height models (CHMs). In particular, we propose to model tree crowns in a forest plot as a configuration of circular objects. A stochastic process is then employed to search the configuration containing the best possible set of tree object with respect to the underlying CHM.

A stochastic model, in general, involves sampling from a huge set of random variables. Our model is novel in that it utilizes low-level image process techniques, i.e., local maxima filtering (LM) and marker-controlled watershed segmentation (MCWS), to extract object parameters, such as locations and crown sizes, and integrates them into the stochastic process; this is why we call it a "hybrid" framework. By integrating the low-level image processing into the stochastic process, the hybrid framework enables a discrete configuration space to sample the optimal configuration, which could potentially accelerate the optimization process when compared with classical stochastic models, e.g., marked point processes. In addition, the parameters of the model are learned from the data through a Monte Carlo based sampling method.

We first introduce the generic mathematical framework of energy modeling for stochastic models in Section **5.2** and present an overview of the proposed framework in Section **5.3**. Our

model features the construction of a discrete configuration space by integrating the low-level image processing into a stochastic model. We give details about the model definition in Section **1.1**. We show in this section how candidate configurations are generated from the CHM: the treetop positions are sampled within the over-extracted local maxima via local maxima filtering, and the crown sizes were derived from marker-controlled watershed segmentation using corresponding treetops as markers. We also present the design of a Gibbs energy, which contains a data term that judges the fitness of the objects with respect to the data, and a prior term that prevents severe overlapping between tree crowns in a configuration.

Parameter estimation of the stochastic model is another issue. We address the parameter estimation in Section **5.5**. In most cases, the parameters are set empirically and tuned by trial and error. With the increase of model complexity, it becomes more difficult and time consuming to set appropriate parameters of such stochastic models. We propose a Monte Carlo based sampling method to estimation the parameters in the model.

Once the model has been properly defined and parameters estimated, the energy is embedded into a Markov Chain Monte Carlo (MCMC) dynamics coupled with a simulated annealing to find its global minimum (See Section **1.1** Model Optimization for details). Section **5.7** briefly describes the accuracy assessment method.

We test the proposed model on a temperate mature coniferous forest in Ontario, Canada, as well as three simulated coniferous forest plots with different degrees of crown overlap. The

experimental results on the parameter estimation and tree detection in Section **5.8** show the effectiveness of our proposed model. It is capable of reducing the commission errors produced by local maxima filtering based method, and achieve our aim to improve the detection accuracy of traditional LM based methods. We discuss the proposed model and the achieved results in Section **5.9**. We conclude and give an outlook on future work in Section **5.10**.

5.2 General Framework of Energy Modeling for Stochastic Models

As a preclude to the construction of our proposed model in the next section, we start by outlining the general framework of energy modeling that underlies probabilistic feature extraction or object detection from remote sensing imagery based on a stochastic framework.

5.2.1 Stochastic models for feature extraction

Since the early work of Besag (1974), Bayesian approaches have played a leading role in image analysis. They have proven to be powerful tools to solve various image analysis tasks, including image segmentation, image restoration, and feature extraction (Descombes and Zerubia, 2002; Li, 2009).

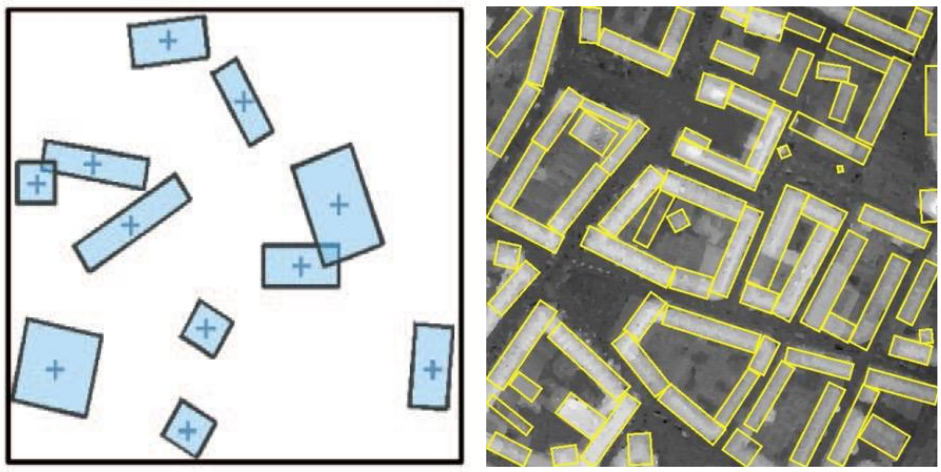
In a probabilistic framework, feature extraction or object detection from remotely sensed data can be viewed as an inverse problem. In particular, stochastic models link a set of random variables **x**, be it at pixel level (e.g., in low-level MRFs) or a set of geometric shapes (e.g., in marked point processes), with the image **y**, leading to defining two random variables X and Y (see **Equation (5.1)**).

In object-based stochastic models, a set of geometric shapes or objects representing features of interests is usually called a configuration, denoted by **x** in the rest of the thesis. To find the best configuration **x** based on the observed data **y** (the image), we must find the configuration $\hat{\mathbf{x}}$ maximizing the posterior probability, according to the following equation:

$$\hat{\mathbf{x}} = arg \max_{\mathbf{x} \in \Omega} \mathbb{P}(X = \mathbf{x} | Y = \mathbf{y}) \tag{5.1}$$

where Ω is the configuration space in which configuration **x** resides.

An example of a marked point process for rectangle detection from a multiview based photogrammetric DSM is shown in **Figure 5.1**.



(a) (b)

Figure 5.1: (a) A marked point process of rectangles, and (b) the MPP-extracted rectangles from a multiview based photogrammetric DSM. Figures are extracted from Lafarge et al. (2010) and Brédif et al. (2013).

5.2.2 Gibbs energy and energy minimization problem

The probability of the model can be specified in the form of an energy U, commonly called *Gibbs energy*, as shown in Equation **(5.2)**:

$$\mathbb{P}(X = \mathbf{x}|Y = \mathbf{y}) = \frac{1}{Z} e^{-U(\mathbf{x})} \tag{5.2}$$

where Z is a normalizing constant such that $Z = \int_{\mathbf{x}\in\Omega} e^{-U(\mathbf{x})}$. Moreover, the energy $U(\mathbf{x})$ can be expressed as a weighted sum of a prior energy $U_p(\mathbf{x})$ that favours specific spatial structures in configuration $\mathbf{x}$, and a data energy $U_d(\mathbf{x})$ which quantifies the quality of the configuration with respect to the data. The design of the two energy terms of our model is detailed in Section **1.1**.

The problem is then reduced to an energy minimization problem of finding the configuration minimizing the Gibbs energy $U(.)$, i.e., $\hat{\mathbf{x}} = arg\min_{\mathbf{x}\in\Omega} U(\mathbf{x})$, which is equivalent to finding the *Maximum A Posteriori* estimator $\hat{\mathbf{x}} = arg\max_{\mathbf{x}\in\Omega} \mathbb{P}(X = \mathbf{x}|Y = \mathbf{y})$.

5.2.3 Estimator and MCMC

Finding the global minimum of the Gibbs energy on Ω is not straightforward. In the general case, it is not possible to derive an analytical expression of $\hat{\mathbf{x}}$. As stated previously, the density of the model is defined up to an unknown normalizing constant. However, in a Markovian framework, it is possible to reduce the global optimization problem to local optimization problems: $\mathbf{x}$ is limited to only depend on elements in its neighbourhood (Tournaire and Paparoditis, 2009).

Within this framework, one only has to build a discrete Markov Chain which converges in probability to the desired law. Transitions between states of the Markov Chain are defined as a set of simple modifications of the current configuration. The energy optimization is then achieved by the Markov Chain Monte Carlo (MCMC) sampler embedded in a simulated annealing with a logarithmic cooling schedule (Salamon et al., 2002b). In practice, we use a geometric decrease to allow a faster relaxation while giving a very good solution close to the optimal one. The optimization process is particularly interesting because the complex computation of the normalizing constant Z is avoided.

5.3 Overview of the Proposed Model

The flow chart of the proposed method is shown in **Figure 5.2**. As our primary contribution, the blue blocks show the process how we construct a constrained configuration space for tree detection, by taking advantages of low-level image processing techniques, which is detailed in Section **5.4.1**. The red block involves techniques of energy formulation and parameter estimation, which are covered in Section **5.4.2** and Section **5.5.2**, respectively. The optimization process illustrated by the yellow blocks is described in Section **1.1**.

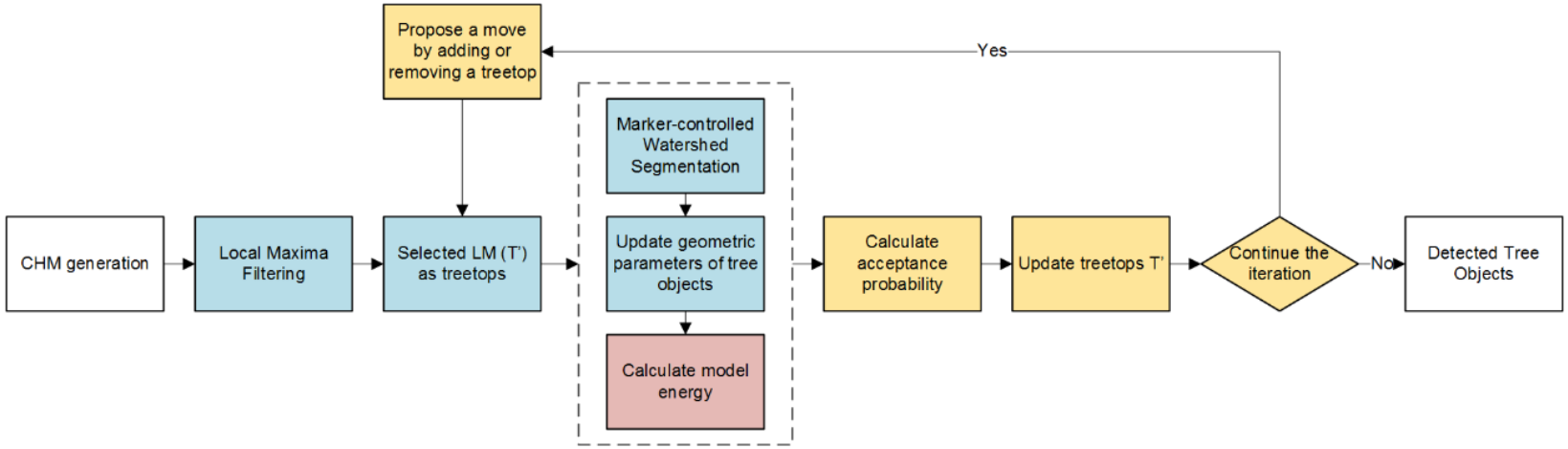


Figure 5.2: Flow chart of the proposed model.

5.4 Model Definition

5.4.1 Configuration space definition of the proposed model

Our proposed model resembles marked point processes in the aspects of object modelling and energy formulation. However, it proposes to build a constrained configuration space when compared with marked point processes, by integrating low-level image processing techniques in its framework to derive object parameters from underlying images.

Marked point processes

Let us first recall the configuration space definition in the marked point process. In remote sensing images, the distribution of tree crowns in forests can be represented by a marked point process of disks. The associated space $\mathcal{S}$ can be written according to the following equation:

$$\mathcal{S} = \mathcal{P} \times \mathcal{M} = [0, X_M] \times [0, Y_M] \times [r_m, r_M] \tag{5.3}$$

where X_M and Y_M are the width and height of the image $\mathcal{I}$, respectively, and (r_m, r_M) are the minimum and maximum radii of the disks in the configuration, respectively. Note that $x = (p, r) \in \mathcal{S}$ is a tree object, where $p \in \mathcal{P}$ is its position and $r \in \mathcal{M}$ its radius. The configuration space Ω of the marked point process of the tree crowns can be written according to the following equation:

$$\Omega = \bigcup_{n=0}^{\infty} \Omega_n \,, \Omega_n = \{\{x_1, \dots, x_n\} \subset \mathcal{S}\} \tag{5.4}$$

that contains all of the configurations of a finite number of tree objects x_i of $\mathcal{S}$(Van Lieshout, 2000).

Proposed model

In this study, we seek to construct a constrained configuration space $\Omega_T \subset \Omega$ in which the optimal or near optimal configuration resides. We will then limit the search for the optimal configuration in the constrained space Ω_T, which could significantly reduce the computation demand of random sampling in Ω in the optimization process.

We begin by constructing a CHM image, representing the height of the tree crowns above ground from the classified ALS data. Then, we extract the local maxima as potential treetops from the CHM using local maxima filtering with a variable window size method adapted from Popescu et al. (2002). Our rule is to detect as many true treetops as possible and reduce omission errors in the first stage. Therefore, the filters of the LM are set to relative small sizes empirically based on the *priori* knowledge about the plots to over-populate initial 'treetops'. Let T represent the set of extracted local maxima: $T = \{t_1, \dots, t_N\}, \forall i \in \{1, \dots, N\}, t_i \in \mathcal{P}$, where N is the total number of local maxima extracted. The true treetops within the set of local maxima T are denoted by $T^o \subset T$.

Given any subset of local maxima $C \subset T$, they can be used as markers in marker-controlled watershed segmentation to obtain a partition $S(C) = \{s_{C_1}, \dots, s_{C_{n(C)}}\}$ of the CHM, where s_{C_i} is the corresponding segment of the local maxima $t_{C_i} \in C$. $S(C)$ is a low-level presentation of the CHM image, and the set of segments are assumed to be a reasonable approximation of the tree crowns with respect to the set of local maxima C, where $n(c)$ is the number of local maxima in C.

A tree object $x_{C_i} = (t_{C_i}, r_{C_i})$ is then defined by its location and radius on the segment s_{C_i}, where the tree location is the corresponding local maximum t_{C_i}, and the radius r_{C_i} is calculated as the average radius of the segment s_{C_i}. A configuration $\mathbf{x}(C) = \{x_{C_1}, \ldots, x_{C_{n(C)}}\}$ is then constructed from the set of local maxima C. The entire procedure of configuration construction is illustrated in **Figure 5.3**.

We note all of the configurations generated from the subsets of local maxima T as $\Omega_T = \{\mathbf{x}(C), C \subset T\}$. Apparently, Ω_T is a discrete subspace of the configuration space Ω, which cardinality is $\mathrm{card}(\{\mathbf{x}(C), C \subset T\}) = \mathrm{card}(\{C, C \subset T\}) = 2^{\mathrm{card}(T)}$. In this manner, we build a constrained configuration space Ω_T from which to sample the optimal configuration.

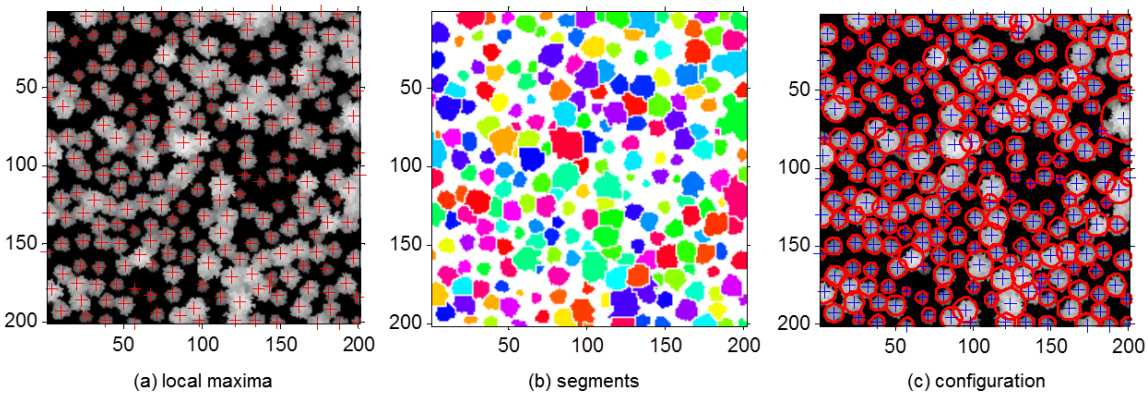


Figure 5.3: An example showing the configuration construction from a CHM by a set of local maxima. (a) a subset of local maxima. Local maxima are shown as red crosses; (b) a marker-controlled watershed segmentation of the CHM using local maxima in (a) as the marker function; (c) the configuration constructed from the local maxima. Radii of the tree crowns are extracted from the corresponding segments in (b).

5.4.2 Energy formulation

As previously mentioned, the Gibbs energy $U(\mathbf{x})$ is defined on the configuration space to measure the goodness or cost of each object configuration. The Gibbs energy can be further expressed as a weighted sum of a prior term $U_p(\mathbf{x})$ that favors a specific spatial pattern in configuration **x** and a data term $U_d(\mathbf{x})$ that quantifies the quality of the configuration with respect to the data, according to the following equation:

$$U(\mathbf{x}) = \alpha U_d(\mathbf{x}) + (1-\alpha)U_p(\mathbf{x}) \tag{5.5}$$

where $\alpha \in [0,1]$ specifies the relative weights of the two energy terms.

We intend to make simple and effective choices for the design of each energy term. The basic assumptions are the geometric properties of trees in mature coniferous forests in which treetops are typically located in the central part of tree crowns, and tree crowns are of a circular shape when viewed from the nadir direction (Chen et al., 2006; Gleason and Im, 2012). We also tend to penalize certain patterns in the configurations in the prior term that tree crowns should not severely overlap.

Data Term

The data term is in accordance with the aforementioned assumption, indicating the likelihood of the tree objects relative to the low-level segments obtained from the CHM image. Certain geometric features are extracted from the underlying segment of each object, and energy functions are proposed to measure how well those features support the object as a plausible tree.

We incorporate the following two energy functions to reflect the assumption: symmetric function $U_d^s(x)$ and area ratio function $U_d^a(x)$. The data term is a weighted sum of the two energy functions, subject to a hard constraint on the object radii, according to the following equation:

$$U_d(\mathbf{x}) = \begin{cases} \sum_{x \in \mathbf{x}} \left(w_1 U_d^s(x) + (1 - w_1) U_d^a(x) \right) & if\ r(x) \in [r_m, r_M] \\ +\infty & otherwise \end{cases} \tag{5.6}$$

where w_1 is the weight regulating the relative importance of the symmetric and area ratio functions in the data term.

Each energy function is defined on a geometric feature R^* of a tree object extracted from the corresponding crown segment. The single energy function U_d^* takes the form of a sigmoid function $\boldsymbol{s}$ with width λ and turning point at $x = \mu$, scaled to the range $\boldsymbol{k}$, as shown in **Equation (5.7)**:

$$\boldsymbol{s}(x, \mu, \lambda, \boldsymbol{k}) = \frac{k_{max} - k_{min}}{1 + \exp - \left(\frac{x - \mu}{\lambda} \right)} + k_{min} \tag{5.7}$$

where x will be substituted by the geometric feature R^* of a tree object in the actual calculation.

Symmetric Function $\boldsymbol{U_d^s(x)}$

A symmetric function is defined as a measure of how well a treetop is located in the central part of the crown and the degree to which the tree crown is of a symmetric circular shape.

We first derive an asymmetric ratio R_{sym} for each tree object. For a given tree object x with corresponding segment s_x, the radii from the treetop point T to the edge of the segment in 8 directions with constant angular intervals $\overline{TP_i}$ $(i = 1, \dots, 8)$ are first extracted (see **Figure 5.4**). The average and standard deviation of the 8 radii are noted as $r(x)$ and $\Delta r(x)$, respectively. The asymmetric ratio $R_{sym}(x) \in [0, 1]$ of object x is calculated as the coefficient of variance of the radii according to the following equation:

$$R_{sym}(x) = \frac{\Delta r(x)}{r(x)} \tag{5.8}$$

A sigmoid function $\boldsymbol{s}\left(R_{sym}(x), \mu_s, \lambda_s, -1, 0\right)$ is then used to define the symmetric function to penalize asymmetric tree crowns given by **Equation (5.9)**:

$$U_d^s(x) = \boldsymbol{s}\left(R_{sym}(x), \mu_s, \lambda_s, -1, 0\right) = \frac{1}{1 + \exp - \left(\frac{R_{sym}(x) - \mu_s}{\lambda_s}\right)} - 1 \tag{5.9}$$

where μ_s and λ_s are parameters set to control the position and slope of the sigmoid function, respectively. The larger the asymmetric ratio $R_{sym}(x) \in [0, 1]$, the higher the symmetric function score $U_d^s(x) \in [-1, 0]$, which indicates that the treetop is more likely to be a false treetop.

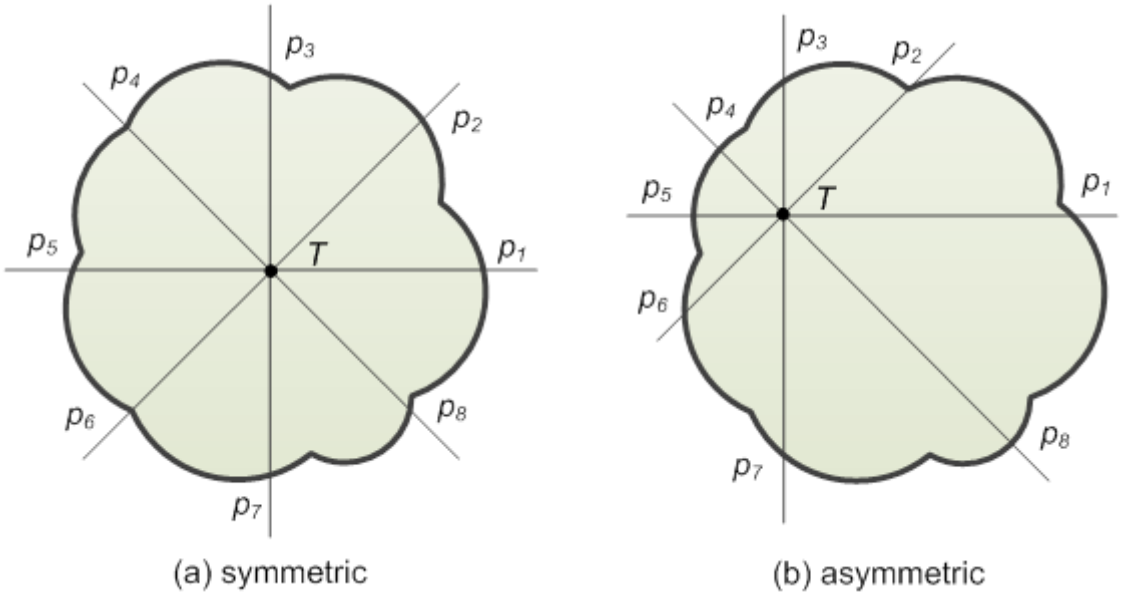


Figure 5.4: Asymmetric ratio calculation for (a) symmetric and (b) asymmetric tree crowns.

Area Ratio Function $U_d^a(x)$

Another term, the area ratio term $U_d^a(x)$, is included to re-enforce the assessment of the geometric features of the objects in the configuration.

As before, an area ratio $R_{area} \in [0, 1]$ is first calculated. The ratio is computed as the proportion of the intersection of object x and the underlying segments s_x to the area of the segments $A(s_x)$ by **Equation (5.10)**. As the area ratio increases, the degree of the geometric feature of the object increases, in accordance with the hypothesis (see **Figure 5.5**).

$$R_{area}(x) = \frac{A(x \cap s_x)}{A(s_x)} \tag{5.10}$$

Based on the area ratio of the object, the area ratio function is defined according to the following equation:

$$U_d^a(x) = \boldsymbol{s}(R_{area}(x), \mu_a, -\lambda_a, -1, 0)$$
$$= \frac{1}{1 + \exp - \left(\frac{R_{area}(x) - \mu_a}{-\lambda_a}\right)} - 1 \tag{5.11}$$

where μ_a and λ_a are used to control the position and slope of the sigmoid function, respectively.

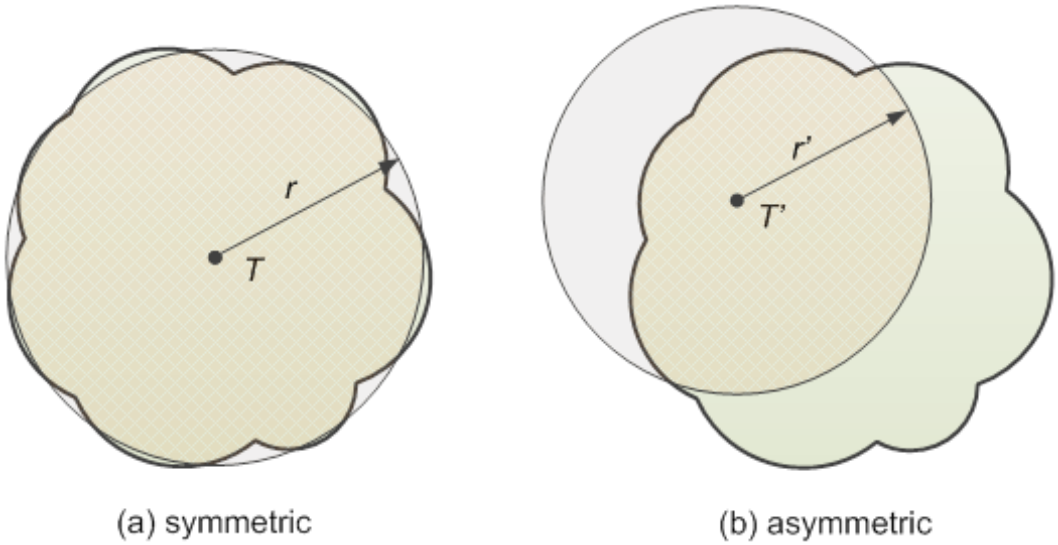


Figure 5.5: Area ratio calculation for tree objects with (a) symmetric and (b) asymmetric tree crowns.

As a note, this area ratio function serves as a compliment to the symmetric function, to better penalize situations as illustrated in **Figure 5.6**. Think about two neighbouring trees corresponding to treetops T and T' in the configuration (**Figure 5.6**(a)). The crown of tree T' will be merged with tree T in marker-controlled watershed segmentation if T' is removed from the configuration (**Figure 5.6**(b)). Although tree T becomes more asymmetric after the merge in this undesirable situation, the symmetric ratio of tree T stays the same before and after the merge because of the limitation of the radius sampling method we used to calculate asymmetric ratio. In this case, the symmetric function fails to penalize this situation. This is why the area ratio function comes in. As can be seem in **Figure 5.6**(c) and **Figure 5.6**(d), the area ratio of tree T will be considerably decreased after the two tree crowns are merged. This will result in an increase in the area ratio score to penalize this situation.

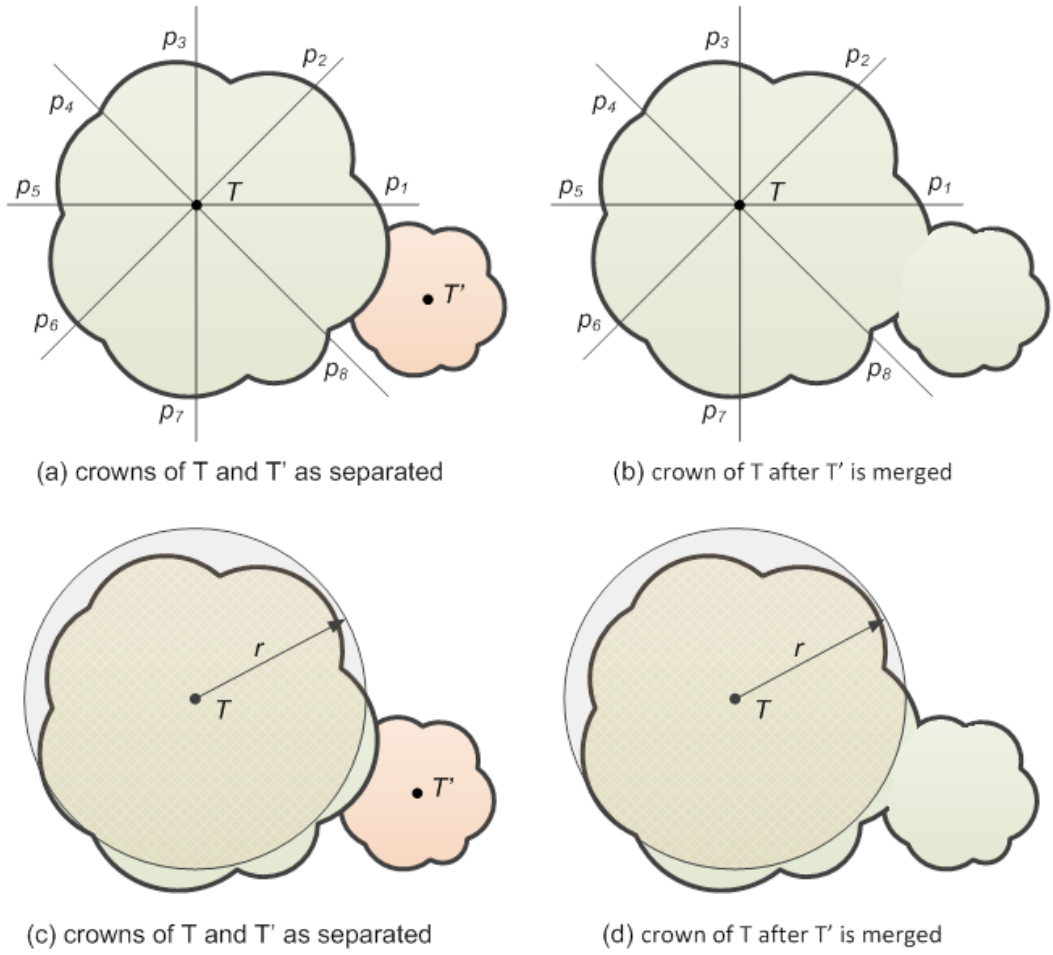


Figure 5.6: The situation is not penalized by symmetric function while area ratio function works.

Prior Term

The prior term introduces a *priori* knowledge concerning the layout of the objects. In most mature coniferous forest stands, tree crowns will not overlap too severely. However, overlap between objects should not be totally prohibited. A repulsive term is then defined as a soft penalizing function to penalize severe overlaps in the configuration.

The energy functions in the prior term also take the form of sigmoid functions $\boldsymbol{s}$ as introduced in the data term. The difference is that they are defined on pairwise interaction features between neighboring tree objects.

(i) Overlap Function $U_p^o(\mathbf{x})$

To define the overlap function, we first introduce a symmetric neighborhood relationship between objects. We say two objects $x_i = (t_i, r_i)$ and $x_j = (t_j, r_j)$ are overlapping if the distance between them is smaller than the sum of their radii, noted as $d(t_i, t_j) < r_i + r_j$, and we write $x_i \sim x_j$. Then, an overlap ratio $R_{ove} \in [0, 1]$ is calculated as the ratio of the overlap area between the two objects normalized by the area of the smaller object, according to the following equation (see **Figure 5.7**):

$$R_{ove}(x_i, x_j) = \frac{A(x_i \cap x_j)}{\min\left(A(x_i), A(x_j)\right)} \tag{5.12}$$

The overlap score $O(x_i, x_j)$ on $x_i \sim x_j$ is then given according to the following equation:

$$O(x_i, x_j) = \boldsymbol{s}(R_{ove}(x_i, x_j), \mu_o, \lambda_o, 0, 1) = \frac{1}{1 + \exp -\left(\frac{R_{ove}(x_i, x_j) - \mu_o}{\lambda_o}\right)} \tag{5.13}$$

where μ_o and λ_o are set to control the position and slope of the sigmoid function, respectively.

The overlap function of configuration **x** can be expressed according to the following equation:

$$U_p^o(\mathbf{x}) = \sum_{x_i \sim x_j} O(x_i, x_j), \forall x_i, x_j \in \mathbf{x}, i \neq j \tag{5.14}$$

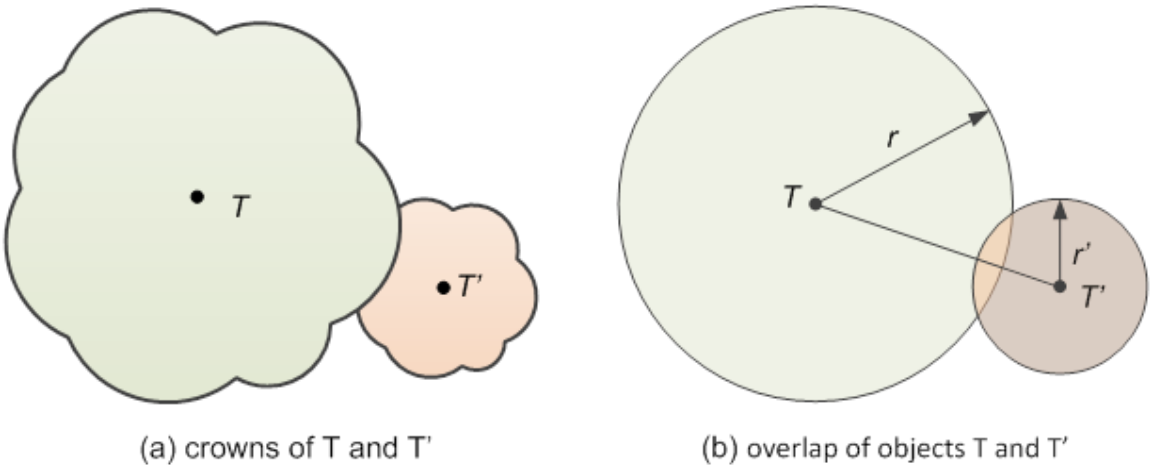


Figure 5.7: Overlap ratio calculation of overlapping tree crowns.

Compared with a classical Marked Point Process, limiting the search space to configurations generated from a subset of a finite set of seed points T (the pre-extracted local maxima) prevents multiple detection problems. The global energy does not have to be designed to prevent the selection of multiple instances of the same tree because duplicated trees are not part of the search space. Thus, the prior term only contains the overlap function and is written according to the following equation:

$$U_p(\mathbf{x}) = U_p^o(\mathbf{x}) \tag{5.15}$$

5.5 Parameter Estimation

5.5.1 Parameter settings

The parameters that need to be specified in the model can be categorized into three groups: physical parameters (bounds), weights and thresholds.

The physical parameters (bounds) r_m and r_M are size constraints specifying the range of the tree crown radius in the forest plots (see Equation **(5.6)**). They are set as 1.0 m and 6.0 m, respectively, according to the range of tree sizes in the test sites.

The weights α and w_1 are assigned to tune the relative importance that we want to grant to different energy terms or functions in the combination (see **Equation (5.5)** and **(5.6)**). Both α and w_1 are set to 0.5 because we place equal importance on those functions in all of our experiments.

The threshold pairs (μ^*, λ^*) of the sigmoid functions (see Equation **(5.9)**, **(5.11)**, and **(5.13)**) in the energy terms control the tolerance and slope of the sigmoid functions respectively, which plays a significant role in the model. To reduce hand-set parameters and avoid a "trial-and-error" test for parameter setting in most practices, we propose a parameter estimation method to estimate the threshold pairs (μ^*, λ^*) in the sigmoid functions in the energy terms.

Figure 5.8 illustrates how parameters μ and λ control the behavior of the sigmoid function. We can easily observe from the exemplary graphs that, for example, if we set a smaller μ_s value in the symmetric function (**Equation (5.9)**), trees with asymmetric crowns will be penalized more effectively; while a smaller value of λ_s results in a steeper slope, and the associated energy function has an increased discriminative behavior of a step function.

In many practices, these critical parameters are tuned by trial and error. This procedure is usually long and complex, nevertheless, the empirical solution cannot guarantee the optimum is reached. Moreover, parameters tuned for one scene are likely to inapplicable to another. When the scene or data changes, the parameters need to be tuned again.

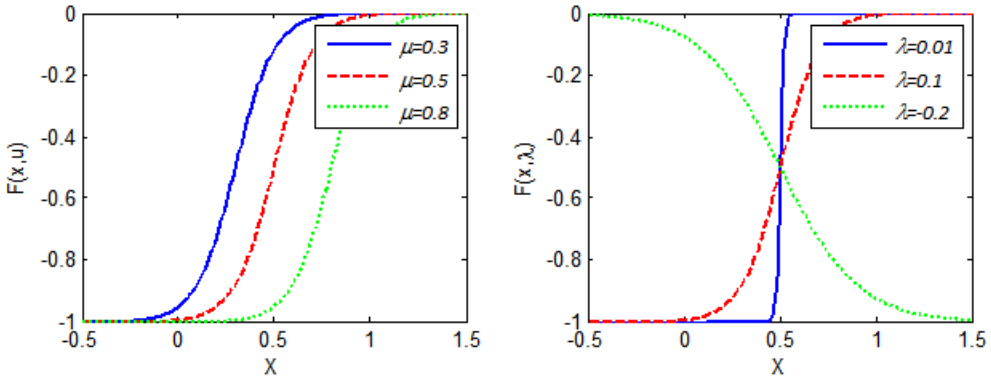


Figure 5.8: Plots of the sigmoid function $F(x) = 1/(1 + \exp -(x - \mu)/\lambda) - 1$ with respect to different values of μ and λ. In the left plot, λ is set to 0.2 for all three curves. In the right plot, μ is set to 0.5 for all three curves.

To reduce hand-set parameters and avoid a "trial-and-error" test for parameter setting in most practices, we propose a parameter estimation method to estimate the threshold pairs (μ^*, λ^*) in the sigmoid functions in the energy terms, which is detailed in the following section.

5.5.2 Parameter estimation of threshold pairs $(\boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$

We address two issues in the parameter estimation for the sigmoid functions in the energy terms.

First, the sigmoid functions are designated in the data term and prior term to penalize false tree objects or implausible spatial patterns in the configurations. In other words, false tree objects or implausible spatial patterns in the configurations should receive high energy scores. For this purpose, we can fit the sigmoid functions to the posterior probability distribution of geometric features derived from false tree objects or implausible pairwise interactions between objects.

Let us denote the unary features or the binary feature[2] extracted in our model as D. A random variable $Z = \{0, 1\}$ takes the value of 1 if $d \in D$ is derived from a *true* tree object or a plausible pairwise interaction, or 0 otherwise. Given an observed feature d, the probability of it being derived from a false tree object or an implausible pairwise interaction can be given by the posterior probability:

[2] In our model, unary features refer to asymmetric ratio $R_{sym}(x_i)$ and area ratio $R_{area}(x_i)$ of tree object x_i; binary feature refers to the overlap ratio $R_{over}(x_i, x_j)$ between neighboring tree objects $x_i \sim x_j$. See Section 5.4.2 for their definitions.

$$p(Z = 0|d) = \frac{p(d|Z = 0)p(Z = 0)}{p(d)} \tag{5.16}$$

The higher the posterior probability of the object being a false tree, the higher the energy we assign to the object through the energy functions.

According to the Bayesian theorem, the posterior probability can be rewritten as the following:

$$p(Z = 0|d) = \frac{p(d|Z = 0)p(Z = 0)}{p(d|Z = 0)p(Z = 0) + p(d|Z = 1)p(Z = 1)} \tag{5.17}$$

$$p(Z = 0|d) = \frac{1}{1 + L_i^o P_i^o} \tag{5.18}$$

where L_i^o is the likelihood ratio, and P_i^o is the prior ratio, according to the following equations:

$$L_i^o = \frac{p(d|Z = 1)}{p(d|Z = 0)} \tag{5.19}$$

$$P_i^o = \frac{p(Z = 1)}{p(Z = 0)} \tag{5.20}$$

The likelihood ratio L_i^o can be calculated by modeling the likelihood distributions of features derived from true and false tree objects, or plausible and implausible interactions. **This is the second issue** we need to tackle: how could we obtain enough samples to model the likelihood distribution of the features with label $Z = 1$ and $Z = 0$?

A Monte Carlo sampling method is utilized to estimate the likelihood distribution of features in **Equation (5.19)**. We can generate a number of random subset of local maxima $\mathcal{T} = \{T_1, \dots, T_n\}$ based on the full set of local maxima T we have extracted, where $T_i \in T, \forall i \in \{1, \dots, n\}$. For each local maxima subset T_i, a configuration $\mathbf{x}(T_i)$ can be produced with respect to the given CHM. Configuration $\mathbf{x}(T_i)$ is then compared with the reference configuration $\mathbf{x}(T^o)$, and each tree object $x_j^{T_i}, j = 1, \dots, n(T_i)$ in configuration $\mathbf{x}(T_i)$ can be labeled as *true* or *false*. Unary features collected from the *true* and *false* tree objects in all generated configurations $\boldsymbol{x} = \{\mathbf{x}(T_1), \mathbf{x}(T_2), \dots, \mathbf{x}(T_n)\}$ can be labeled with $Z = 1$ and $Z = 0$.

Collection of binary features is carried out analogic to that of unary features, except that three interaction types exist between *true* and *false* trees objects in a configuration as *true-true*, *true-false*, and *false-false*[3]. Only features derived from the *true-true* interaction type take the label

[3] The interaction type *"true-true"* ("*false-false*") means the two neighboring tree objects are both *true (or false)* tree objects, while "*true-false*" implies one of the two neighboring tree object is *true* and the other is *false*.

of $Z = 1$, and otherwise (*true-false* and *false-false* types) take the label $Z = 0$. We repeat this process for n ($n = 50$) times in our experiments, to collect sufficient samples of features.

The Monte Carlo-based method produces a pool of samples sufficient to model the likelihood distributions of different features with labels $Z = 1$ and $Z = 0$. The likelihood distributions of features, asymmetric ratio, area ratio and overlap ratio, with label $Z = 1$ are modelled with Log-logistic, Weibull and exponential distribution, respectively; while features with label $Z = 0$ are all modelled with normal distribution. The maximum likelihood method is employed to model all those distributions.

In practice, we set the prior ratio P_i^o to 2 empirically, based on the general detection accuracy achieved by LM-based approaches. The modeled distributions and fitted functions are shown in **Figure 5.9**.

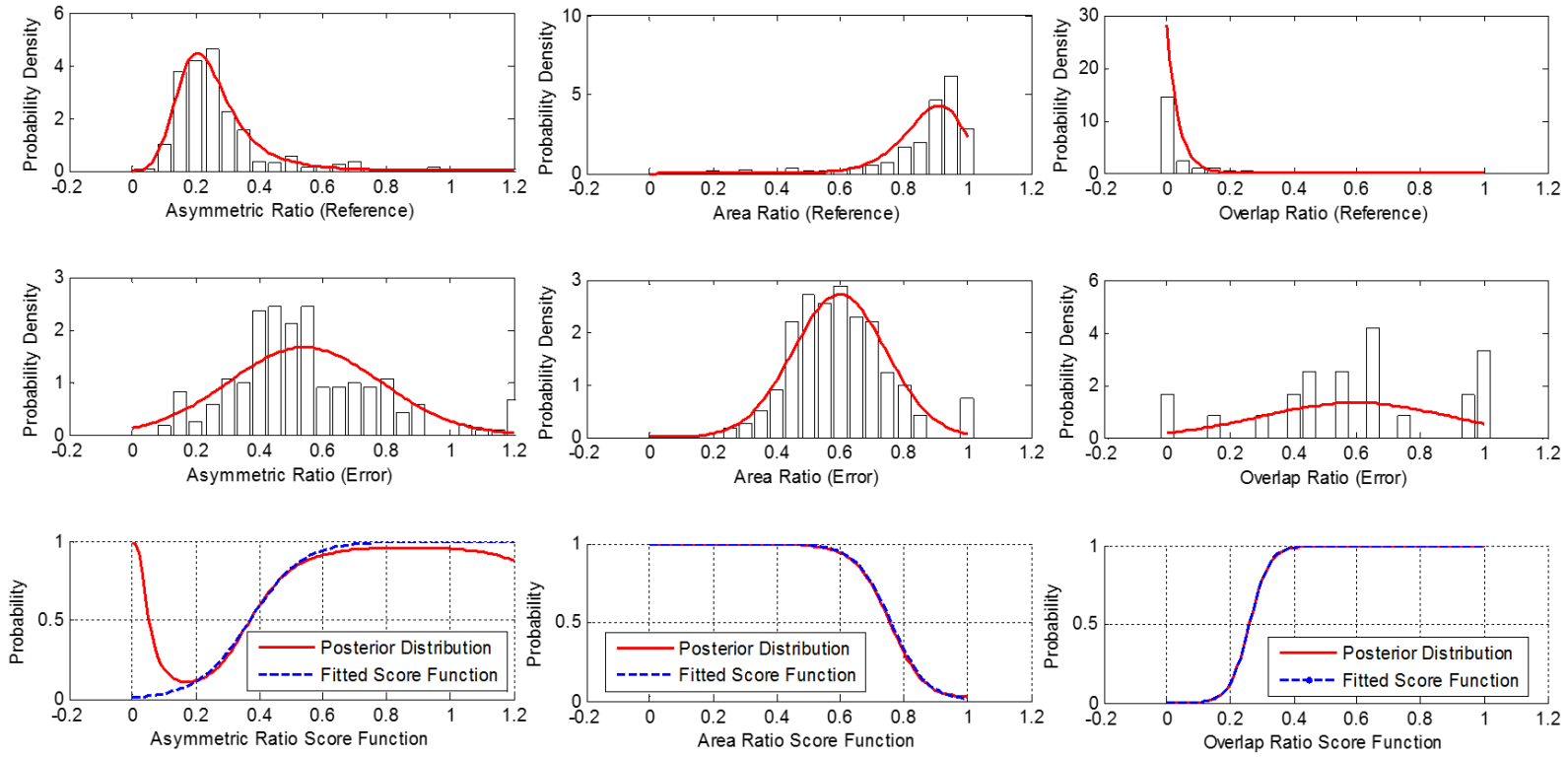


Figure 5.9: Likelihood distributions, posterior probability and fitted sigmoid functions for the asymmetric ratio, area ratio and overlap ratio. Row 1: Likelihood models of those ratios for the reference group; Row 2: Likelihood models of those ratios for the error group; Row 3: Posterior probabilities (red lines) for those ratios for the error group and the fitted sigmoid functions (blue dashed lines).

5.6 Model Optimization

The aim of model optimization is to find the optimal configuration of objects **x,** which minimizes the global energy $U(\mathbf{x})$ in **Equation (5.5)**, from the configuration space Ω_T we have proposed. The discrete space can be effectively explored using a Markov Chain Monte Carlo sampler coupled with simulated annealing as we specify in the following.

5.6.1 MCMC sampler

A MCMC sampler is adapted for the problem by simulating a discrete Markov chain $(X_t), t \in \mathbb{N}$ on the configuration space Ω_T, which converges towards an invariant measure specified by the energy $U(\mathbf{x})$. The sampler performs transitions between different states respecting the reversibility assumption of the Markov chain. The transition can be managed by a set of proposition kernels denoted by Q_m. If a configuration **x** transit to **y** according to a probability $Q_m(\mathbf{x} \to \mathbf{y})$, the move is accepted with the following probability:

$$\min\left(1, \frac{Q_m(\mathbf{y} \to \mathbf{x})}{Q_m(\mathbf{x} \to \mathbf{y})} \exp -\big(U(\mathbf{y}) - U(\mathbf{x})\big)\right) \tag{5.21}$$

As detailed in **Section 5.4.1**, a configuration of trees $\mathbf{x}(T^k)$ can be solely determined by a subset of local maxima $T^k \subset T$ given the CHM image. Once treetops are set as the local maxima T^k, the tree sizes are decided and directly derived from the corresponding marker-controlled watershed segments. Therefore, finding the optimal configuration of trees $\mathbf{x}(T^*)$ is equivalent to determining the optimal set of local maxima $T^* \subset T$. Simply put, the search space is thus the set of

the $2^{|T|}$ possible subsets of T, which can be encoded as a Boolean vector of size $|T|$, which states whether each element of T is part of the solution T^*.

In our application, a "birth-and-death" kernel is defined to perform moves between different configurations. A local maximum is added or removed from the current set of local maxima, to generate a new configuration **y** from a previous configuration **x**. We assume a move between configuration **x** and **y** to be symmetric, which gives $Q_m(\mathbf{x} \to \mathbf{y}) = Q_m(\mathbf{y} \to \mathbf{x})$. Theoretically, the kernel is sufficient for the chain to visit the whole configuration space. Specifically, the move from a configuration $\mathbf{x}(T^k)$ to $\mathbf{x}(T^{k+1})$ is realized by the two following processes:

- In a birth process, a local maximum u is randomly selected from $T \backslash T^k$ and added to the current local maxima set T^k to generate a new configuration $\mathbf{x}(T^{k+1})$, with $T^{k+1} = T^k \cup \{u\}$.

- In a death process, a local maximum v is randomly selected and removed from the current local maxima set T^k to generate a new configuration $\mathbf{x}(T^{k+1})$, with $T^{k+1} = T^k \backslash \{v\}$.

This birth-and-death process may be efficiently implemented by maintaining the Boolean selection vector of size $|T|$. The auto-adjoint move boils down to sampling uniformly an element u of T and inverting its selection, thus considering the symmetric difference $T' = T^k \Delta u = (T^k \backslash u) \cup (u \backslash T^k)$ (Wikipedia, 2015).

The move between the configurations is then accepted with the following probability:

$$\min\left(1, \exp - \left(U\left(\mathbf{x}(T^{k+1})\right) - U\left(\mathbf{x}(T^{k})\right)\right)\right) \tag{5.22}$$

Otherwise, the previous set of local maxima is kept: $T^{k+1} = T^{k}$.

5.6.2 Simulated annealing

A simulated annealing is then embedded in the MCMC to find the optimal configuration with the minimum global energy $U(\mathbf{x})$. To perform the simulated annealing, the Gibbs energy $U(\mathbf{x})$ is replaced with $U_{T_t} = U(\mathbf{x})/T_t$. T_t is the temperature parameter, which tends toward zero as t approaches ∞. A logarithmic decrease ensures the convergence to the global optimum for any initial configuration $\mathbf{x}_0$. In practice, a geometric cooling scheme is preferred to accelerate the process and to give an approximate solution close to the optimal one, for example, use $T_t = T_0\alpha^t$ with α close to 1, typically $\alpha = 0.98$. During the cooling procedure, the process explores the configuration space and becomes more and more selective, and eventually corresponds to local adjustments of the configuration.

To sum up the optimization process, if at k, $X_k = \mathbf{x}(T^k)$

1. Choose randomly an element u of T and set $T' = T^k \Delta u$;
2. Generate a new configuration $\mathbf{y} = \mathbf{x}(T')$ based on current configuration $\mathbf{x} = \mathbf{x}(T^k)$;
3. Compute the acceptance ratio $\alpha = \exp - \left(\frac{U(\mathbf{y}) - U(\mathbf{x})}{T_k}\right)$;

4. Take $T^{k+1} = T'$ with a probability $\min(1, \alpha)$ and take $T^{k+1} = T^k$ otherwise;
5. Repeat the processes 1-4 until the energy has converged.

5.7 Accuracy Assessment

To evaluate the performances of the proposed model, the detected trees are compared to the reference data. The comparison results of all of the aggregated trees from the detected trees and the reference data can be classified into the following three categories: the correctly detected trees (correct), trees in the detection results that have no corresponding reference tree (commission) and trees in the reference data not detected (omission). A correctly detected tree means the overlapping area between the ground truth or reference crown polygon and the segment overlaying with it is within the range of $(1 \pm 10\%) * S(r)$, where $S(r)$ is the area of the reference crown polygon.

Commission/Omission statistics and the overall detection accuracy are used to quantify the detection results. The calculation of the commission error, omission error and overall accuracy is based on a conventional method of error matrix assessment (Girard, 2003), as shown by **Equation (5.23) - (5.25)**:

$$Commission\ error = \frac{N_{det} - N_{cor}}{N_{det}} \times 100\% \tag{5.23}$$

$$Omission\ error = \frac{N_{ref} - N_{cor}}{N_{ref}} \times 100\% \tag{5.24}$$

$$Overall\ quality\ = \frac{N_{cor}}{N_{ref} + (N_{det} - N_{cor})} \times 100\% \tag{5.25}$$

where N_{cor} is the number of correctly detected trees, N_{det} is the total number of detected trees by the algorithm, and N_{ref} is the number of reference trees.

5.8 Results

5.8.1 Parameter estimation results

Table 5.1 displays the parameters estimated for the energy functions of the proposed model. We then performed experiments with the estimated parameters on real and simulated forest plots to test the robustness of the model.

The parameter μ_s is the threshold in the symmetric function used to penalize tree crowns with high asymmetric ratios. In a forest in which most tree crowns are of regular circular shapes, the value of μ_s can be set relatively small to more effectively penalize crowns with asymmetric ratios that exceed this threshold. The threshold μ_a works conversely. Because a larger area ratio indicates a more circular shaped crown, it must be set to a larger value to better penalize tree crowns of a non-circular shape. Parameter μ_o in the overlap function is set to penalize an overlapping situation that exceeds a certain degree, which works similarly to the μ_s parameter. The greater the degree of crown overlap in a forest plot, the larger the μ_o value should be set.

The results shown in **Table 5.1** support this reasoning for parameter setting in which the more the tree crowns in the plot are of symmetric circular shape, the smaller the estimated value of μ_s, whereas the larger the value of μ_a. This reasoning is more explicitly evidenced by the simulated forest plots in which the shape irregularity of the tree crowns increases with the increasing degree of canopy overlap from separated to overlapping, which in turn causes an increase in the value of

μ_s from 0.32 to 0.45 and the value of μ_o from 0.08 to 0.40, whereas the value of μ_a decreases correspondingly from 0.82 to 0.72. This result also confirms the rationality of our proposed method for parameter estimation. We also notice that the smaller the overlap degree of a plot, the smaller the estimated λ in the sigmoid function, which indicates a better "threshold" behavior of the associated energy function. This relationship is well in line with the assumption that the simpler the plot situation, the easier the *true* tree crowns and the *false* tree crowns can be distinguished.

From the estimation results of the real and simulated forest plots, we also conclude that the degrees of crown overlap of the real forest plots are between the touch and overlap situations in the simulated forest plots. This can be observed from the ranges of the estimated values of μ_s and μ_o of the real forest plots, which are between the parameters estimated for the touch and overlap simulated forest plots.

Table 5.1: Parameter estimation results of the proposed model for all of the forest plots.

		Parameter Estimation					
		Real Forest Plots			**Simulated Forest Plots**		
		Plot 1	**Plot 2**	**Plot3**	**Separate**	**Touch**	**Overlap**
Symmetric	μ_s	0.43	0.39	0.45	0.32	0.37	0.45
Function	λ_s	0.10	0.11	0.13	0.06	0.08	0.15
Area Ratio	μ_a	0.69	0.68	0.67	0.82	0.76	0.72
Function	λ_a	-0.07	-0.07	-0.11	-0.03	-0.06	-0.14
Overlap	μ_o	0.28	0.32	0.38	0.08	0.26	0.40
Function	λ_o	0.04	0.05	0.05	0.01	0.03	0.05

5.8.2 Detection results of real forest plots

We first applied the proposed model with the estimated parameters to the ALS data of the three real forest plots. The detection results of local maxima filtering with a variable window size (also

referred to as LM) and the proposed model are illustrated in **Figure 5.10**, which shows a good visual assessment of the performance of the two methods.

The LM results are displayed in the first row (**Figure 5.10(a)-(c)**). In these images, the red circles with blue crosses in the center represent the corrected detected tree crowns, whereas the green and cyan circles represent the commission and omission errors, respectively. **Figure 5.10** clearly shows that the LM method is prone to produce commission errors in those coniferous forest plots. This problem is particularly noted in plot 1 and plot 3 in which numerous false treetops occur on the edge of tree crowns because of the branching structure of the pine tree species growing in those plots. Plot 2 is a forest with relatively sparser trees, and commission errors primarily occur near the plot boundaries caused by incomplete crown segments and a lack of reference data.

The corresponding images in the second row (**Figure 5.10(d)-(f)**) show the detection results using the proposed model. As can be easily interpreted, most green circles were successfully removed, indicating that the proposed model could effectively reduce the commission errors. We noticed that a small number of yellow dot line circles appear, which indicate trees over-pruned by the proposed model. From the three images, we can observe that the omission errors produced by the proposed model are primarily trees with small crowns and are severely overlapped by their neighboring larger trees. We also noticed that many commission errors occur at the edge of the plots where crowns are shown incomplete or the reference data are missing.

Table 5.2 depicts the detailed quantitative assessment of the detection results of the LM and the proposed model. There is an obvious improvement in the results of the proposed model

over the LM method on which it is based. The commission errors of the three forest plots significantly decreased, with the largest extend in plot 1, decreasing from 36.2% to 10.3%, whereas the omission errors before and after the application of the proposed model remain at similar levels. On average, the overall detection accuracy increased by approximately 15%, comparing results of the proposed model with those of the LM method.

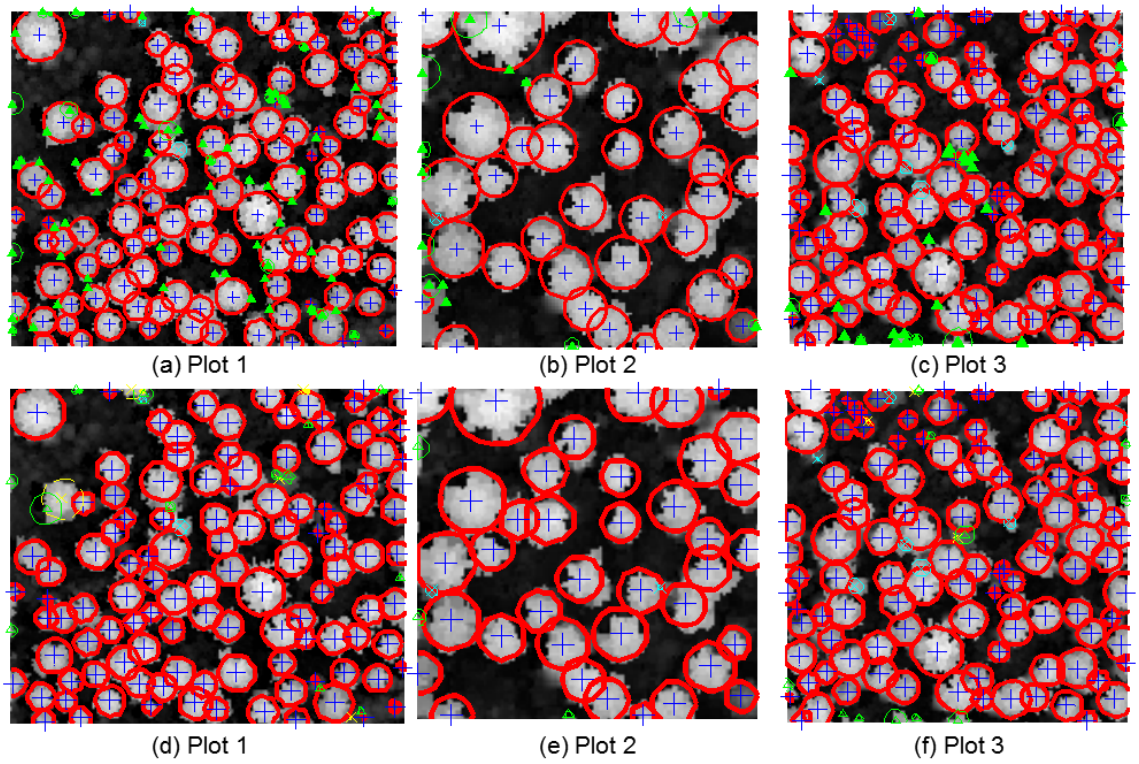


Figure 5.10: Detection results of the proposed model with estimated parameters compared with traditional local maxima filtering on real coniferous forest plots. (a)-(c) show the local maxima filtering results; (d)-(f) show the detection result of the proposed model using the corresponding local maxima filtering detection as the initial configuration. (the green circles with triangles in the center represent the commission errors; the cyan dot line circles represent the omission errors resulting from the LM; the yellow circles represent the omission errors produced by the proposed model.)

Table 5.2: Detection results of the proposed model with estimated parameters compared with local maxima filtering (LM) on the real coniferous forest plots.

	Detected Trees	Correct		Commission		Omission		Overall Accuracy
		No.	%	No.	%	No.	%	
Plot 1 - 120 trees								
LM	185	118	63.8%	67	36.2%	2	1.7%	63.1%
Proposed Model	126	113	89.7%	13	10.3%	7	5.8%	85.0%
Plot 2 - 40 trees								
LM	51	38	74.5%	13	25.5%	2	5.0%	71.7%
Proposed Model	41	38	92.7%	3	7.3%	2	5.0%	88.4%
Plot 3 - 122 trees								
LM	141	115	81.6%	26	18.4%	7	5.7%	77.7%
Proposed Model	123	112	91.1%	11	8.9%	10	8.2%	84.2%

5.8.3 Detection results of simulated forest plots

The proposed model with the estimated parameters applied to the simulated forest plots exhibited similar detection results to those of the real forest plots. The proposed model significantly reduced the commission errors resulting from the LM method in the three simulated forest plots. **Figure 5.11** shows a clear contrast in the detection results of the LM and the proposed model.

Similarly, by comparing the corresponding images in **Figure 5.11(a)-(c)** and **Figure 5.11(d)-(f)**, it can be observed that nearly all of the green circles (commission errors) in the LM detection results were removed by the proposed model in the three simulated forest plots. Meanwhile, there is only a negligible increase in the number of yellow dot line circles (omission errors). On average, the proposed model increases the overall detection accuracy by approximately 10% compared with the LM method in all of the cases.

Table 5.3 gives the exact detection results of the LM method and the proposed model on the three simulated plots. It is interesting to examine the influence of the crown overlap degree on the single tree detection results of the LM method. The overall detection accuracy decreases by approximately 10% across the three simulated forest plots with an increasing degree of crown overlap from separated to overlapping. This result is primarily because of the increase in the number of omission errors with the increase in the crown overlap. Trees growing by taller trees are more likely to be missed in the LM detection when crowns are more overlapped. However, the commission errors are less affected by the degree of crown overlap, which remains at a similar level for the three forest plots.

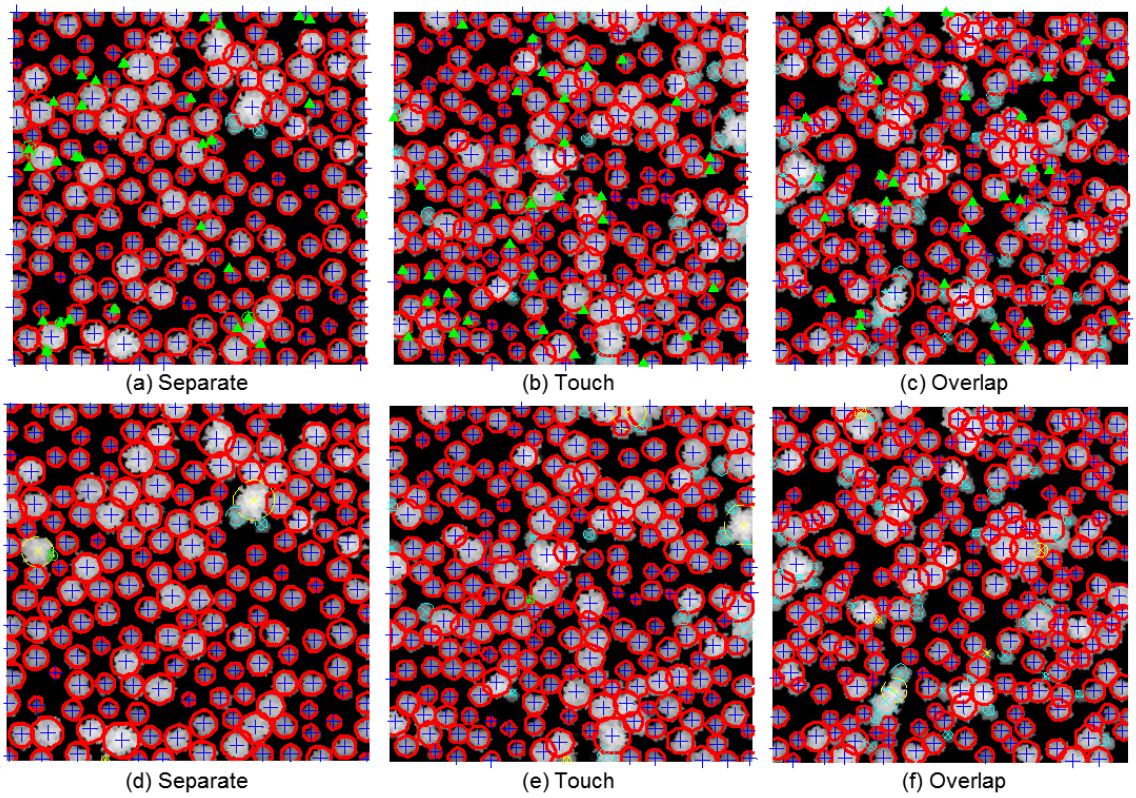


Figure 5.11: Detection results of the proposed model with estimated parameters compared with local maxima filtering on simulated forests. (a)-(c) show the local maxima filtering

detection on the three simulated forest plots; (d)-(f) show the proposed model detection results using the corresponding local maxima filtering detection as the initial configuration. (the green circles with triangles in the center represent the commission errors; the cyan dot line circles represent the omission errors resulting from the LM; the yellow circles represent the omission errors produced by the proposed model.)

Table 5.3: Detection results of the proposed model with estimated parameters compared with local maxima filtering (LM) on the simulated forest plots.

	Detected Trees	Correct		Commission		Omission		Overall Accuracy
		No.	%	No.	%	No.	%	
Separate Plot - 186 trees								
LM	213	184	86.4%	29	13.6%	2	1.1%	85.6%
Proposed Model	182	181	99.5%	1	0.5%	5	2.7%	96.8%
Touching Plot - 234 trees								
LM	252	218	86.5%	34	13.5%	16	6.8%	81.3%
Proposed Model	216	215	99.5%	1	0.5%	19	8.1%	91.5%
Overlapping Plot - 261 trees								
LM	256	226	88.3%	30	11.7%	35	13.4%	77.6%
Proposed Model	221	221	100.0%	0	0.0%	40	15.3%	84.7%

5.8.4 Optimization process

Figure 5.12 presents the statistics associated with the optimization process, using a simulated forest plot with a touching crown as an example. The plots are at the same abscissa scale to simplify the observation of the optimization process. The iteration index is consistently represented on this axis. In all of the experiments, the temperature decrease coefficient α is set to 0.98, and the temperature is updated every 500 iterations. For a plot with approximately 200 trees, it takes approximately $1.2e+5$ iterations for the energy to converge, which is significantly fewer than the total number of configurations ($2^{200} \approx 1.6e+60$) in the entire configuration space. The program takes approximately 3 hours to run in Matlab on a processor with a 2.83 GHz frequency.

The first plot (**Figure 5.12(a)**) shows the evolution of the temperature in accordance with a geometric cooling scheme, as described in **Section 5.6.2**. **Figure 5.12(b)** represents the acceptance rate associated with the "birth-and-death" kernel. The move acceptance rates are high at the beginning of the process and tend to progressively decrease to and stabilize near 0. Finally, **Figure 5.12(c)** plots the global energy. Variations are the highest during the first iterations, and the energy slowly decreases. The decrease becomes faster as the iterations progress and tends to converge slowly to its minimum.

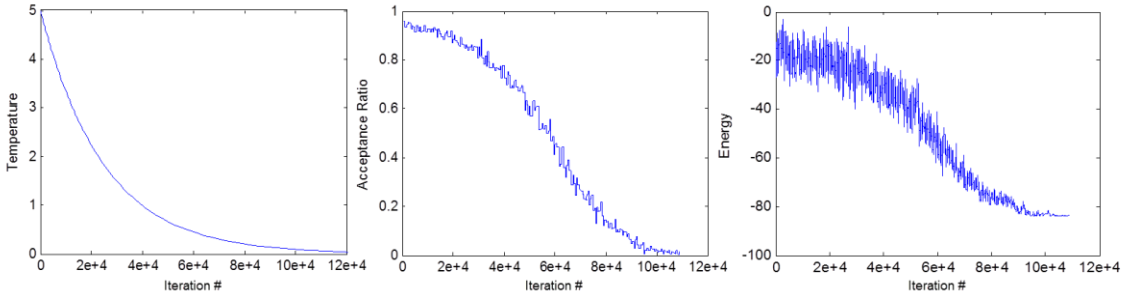


Figure 5.12: Statistics associated with the optimization process of the simulated forest plot with touching crowns: (a) Temperature; (b) Acceptance rate; (c) Global energy.

5.9 Discussion

In this study, we present a hybrid framework to improve the performance of single tree detection from ALS data by taking advantage of low-level image processing techniques and a high-level probabilistic model. The proposed model is applied to the ALS data of real and simulated coniferous forest plots. The results show the feasibility of our approach, and the detection quality is superior to that obtained by the local maxima filtering based method.

The proposed method has been proven to be effective in reduce the commission errors that are introduced by LM in all coniferous forest plots. The LM approach requires a *priori* knowledge about the relationship between the tree height and the crown size, and the detection accuracy can be significantly influenced by the specification of the relationship. In many cases, this relationship is either hard to obtain or different from study to study because it depends on certain factors, such as tree species, tree age, tree density, crown overlapping, and species composition of the forest plot. Moreover, Falkowski et al. (2006) noted that the relationship between the tree height and the crown size can be weak under certain forest conditions, which is coherent with our case. In this case, when a relationship is designated between the tree height and the crown size, the parameters set for the LM are simply a trade-off between commission and omission errors. We suggested a relative small window size for the LM to over-extract initial 'treetops' at the first stage, and the embedded probabilistic model showed its potential in excluding the false treetops from the final configuration through stochastic inference by considering the spatial layouts and geometric characteristics of the trees in the forest plots.

Simulation of forest plots and ALS data provide a valuable tool to examine the performance of tree detection methods under the influence of stem densities and degrees of crown overlap. The detection results evidence the higher the stem density, the more likely the tree crowns are overlapped in the plot, causing smaller trees growing nearby larger trees not easily be detected. The results obtained are coherent with those reported in other studies that denser plots give less accurate results than sparse plots. The simulated data also provides a fully controlled environment to observe the behavior of the estimated parameters in the designed energy functions with respect to the factor of crown overlap. The increase in crown overlap results in more asymmetric crowns in CHM, which are noted by the estimated parameters and further validate the rationality of the parameter estimation method we proposed. The simulation in our study is intended to test our proposed model under certain key forest variables, i.e., the tree density and crown overlap in our case. Additional sophisticated simulations of forest structure and ALS returns can be found in Morsdorf et al. (2009) and Disney et al. (2010).

The detection of single trees from remote sensing data using marked point processes was first performed by Andersen et al. (2002) in an attempt to directly detect trees of a coniferous plot from ALS point clouds using the marked point process in a Bayesian framework. The results have indicated that the algorithm is generally successful in identifying structures associated with individual tree crowns within the forest plot but appears to be sensitive to complex point cloud data. Perrin (2005, 2006) has employed marked point processes to detect tree crowns from CIR aerial imageries of plantations, which leads to a continuous search space for the tree objects, in contrast to the proposed method.

The stochastic model we proposed is the first to integrate low-level image processing techniques and a high-level probabilistic model into a hybrid framework for single tree detection. The model resembles marked point processes in terms of object modeling and energy formulation. However, in the model, the parameters of the tree objects are directly derived from low-level representations of LiDAR images produced by traditional image processing techniques rather than random sampling in classical marked point processes. Thus, the model generates a constrained discrete configuration space, in which we sample for the global optimum that contains the final set of detected trees. In this manner, the computation cost is significantly reduced, and the optimization process can be significantly accelerated.

The design of proper energy terms is an important issue we attempt to address due to the different types of data we used and the specific manner in which we constructed a configuration. The models used to detect tree crowns in aerial imageries (Perrin et al., 2005, 2006) make use of the distinctive pixel values between the illuminated area near the center of the tree crowns and that of the backgrounds or valleys between the crowns. The contrast between the tree crowns and the background, or treetop areas and valleys between them, can be exaggerated by shadows and stretched spectral or radiometric characteristics in the optical images. However, the elevation differences between those parts in the CHM images are much milder and more complex to model than the contrasts in optical imageries. This fact is also the reason we chose a Gibbs energy to measure the morphological characteristics of the tree objects in a configuration, other than a Bayesian framework to model height distributions, considering the complexity required to design a height model valid for all of the trees of various heights and crown forms in the forest area.

Parameter estimation is another challenging task in most stochastic models. In this study, we proposed a Monte Carlo-based method to estimate certain key parameters in our model. The Monte Carlo simulation was used to generate random configurations and to create a sufficient number of samples of true and false tree crowns, which enabled the modeling of feature distributions of true and false tree crowns to estimate thresholds in the energy terms. The experimental results on all of the datasets, especially the simulated ones, suggested that the parameter estimation method works reasonably well.

The proposed method has certain inherited drawbacks detecting trees from the rasterized canopy height model, which is incapable of finding suppressed trees under dominant crowns (Hyyppä et al., 2012). The method is designed to detect trees in the dominant layers in the coniferous forest plots of interest. Exploiting 3D information from the ALS point cloud to detect small trees in the lower forest layer is a possible direction to overcome this disadvantage (Ferraz et al., 2012; Reitberger et al., 2009). Another limitation of the method is that it is unable to recover the omission error produced by local maxima filtering on which it is based. Because tree positions are constrained within the pre-extracted local maxima, the model experienced a reduced ability in the classical marked point process to sample the configuration space more thoroughly. However, experimental results on real and simulated forest plots still suggest that the proposed model is a good compromise regarding complexity, efficiency and accuracy.

5.10 Conclusion

We propose a hybrid framework to detect single trees from ALS data by combining the low-level image processing techniques of LM and MCWS with a high-level probabilistic model. More specifically, in this model, tree crowns in an ALS recovered CHM are modeled as objects and are considered as a configuration of circles. The probabilistic model enables the consideration of the geometric characteristics and the pair-wise interactions of objects in the configuration. The LM and MCWS are employed to produce a low-level representation of the image, which provides a constrained configuration space for the probabilistic model to sample for the optimal configuration. We also propose a Monte Carlo-based method to estimate important parameters in the proposed model. The model is proven effective when applied to real and simulated coniferous forest plots. The results show that the proposed model has a distinct improvement in the detection quality over the traditional local maxima filtering based approach by approximately 10% on all of the datasets.

Future studies should involve a further examination of the optimization methods. An important benefit we gained from our proposed model is that the configuration space is significantly reduced by incorporating features extracted from the CHM image through low-level image processing techniques. However, there remains a significant requirement to accelerate the optimization process. A prior-guided MCMC or a steepest gradient descent algorithm are possibilities we will examine to accelerate the search for the optimal configuration within the discrete configuration space. Second, we will investigate automatic parameter estimation methods, which could minimize human intervention and enable the realization of unsupervised single tree detection without using any reference data.

Chapter 6
Parameter Estimation and Model Optimization

6.1 Introduction

In the previous chapter, we have introduced a probabilistic model to detect single trees from airborne laser scanning (ALS) data by integrating low-level image processing techniques into a high-level probabilistic framework. Within this framework, we treat the tree crowns in a forest plot as a configuration of circular objects, and the density of the model is expressed as a combination of energy terms in Gibbs form: first, a data term, which judges the fitness of the objects with respect to the underlying data; and second, a prior term which encodes geometrical constraints on the configuration that tree crowns should not be severely overlapped. The configuration containing the best possible set of tree objects is then estimated by a global optimization solver, or in our case, an energy minimum estimator.

There are two essential elements in the specification of a proposed model: the energy function definition and the parameter involved. While formulating the forms of objective functions, e.g. the posterior distribution, has long been a subject of research in image and vision analysis, estimating the parameters involved has a much shorter history (Li, 2009). The parameters have to be tuned properly before the system can perform successfully. A common practice is to choose such parameters manually by trial and error. Such empirical methods have always been criticized for their ad hoc nature.

In our specific application, estimation of the threshold parameters incorporated in each energy function is especially critical, as the setting of those thresholds is closely related to characteristics of tree objects to be detected, which are scene dependant. To tackle this problem, a Monte Carlo based sampling method has been proposed in our previous research to estimate those parameters.

This method relies on reference data to categorize the collected features into two groups of "*true*" and "*false*", which enables approximating the likelihood of feature distribution of *true* and *false* tree objects, and further the estimation of parameters in each energy function. It is of great value to develop a method that can adjust the parameters automatically according to the processed data without using any reference data, since collection of reference data in forests is usually laborious, time-consuming and expensive, and even impossible sometimes in remote and inaccessible areas. This would finally facilitate an unsupervised detection of trees in forest of a large scale with our proposed model.

In this prospect, we aim to develop an automatic parameter estimation method based on an Expectation Maximization (EM) procedure. In addition, in this study, we further improve the estimation procedure by modeling the feature distributions with logistic regression directly, which avoids the empirical selection of distributions for approximating of the likelihood of features in the previous method. We also further investigate how initial condition of the EM procedure might influence the parameter estimation results. We explain the whole automatic parameter estimation procedure in detail in **Section 6.2**.

Furthermore, in this chapter we test a modified optimization method resembling RJMCMC used in classical marked point processes (see **Section 6.3** for the details); we call it as prior-guided MCMC (PGMCMC), in which the prior information about tree density of the forest plot is used. We carry out a comparative study of the two optimization algorithms on simulated and real forest plots. We also study how different cooling schedules influence the detailed behavior of the two algorithms.

6.2 Parameter Estimation

6.2.1 Modified parameter estimation method with reference data

We first present an improvement on the previous Monte Carlo sampling based parameter estimation with reference data. In this method, Monte Carlo sampling enables producing of sufficient number of tree samples, while reference data helps to categorize those generated tree samples: the combination of them thus lead to the successful modeling of likelihood distribution of features labeled as "*true*" and "*false*", and further estimation of threshold parameters in the energy function.

All elements in this method seems to integrate perfectly, except one thing that we want to avoid: the likelihood distribution of different features are selected empirically in the modeling. To eliminate this empirical element, we propose to model the posterior distribution of features directly using logistic regression.

Let us denote by $x = \{x_1, \dots, x_n\}$ as all the tree objects in configurations generated by the Monte Carlo method, where n is the number of tree objects. For a feature d_i extracted from a tree

object x_i, we have a random variable $z_i \in \{0,1\}$ indicating the label of the feature d_i. We then note $\mathbf{z} = (z_1, \dots, z_n)$ be the label vector for corresponding features $\mathbf{d} = (d_1, \dots, d_n)$.

In case of one of the data energy terms, we define z so that $z_i = 1$ if feature d_i is extracted from *true* tree object, or $z_i = 0$ if corresponding tree object is *false*. In case of the binary energy term, the overlap ratio, as previously specified in **Section 5.5.2**, the ones derived from tree object pairs with *true-true* interaction type take the label $z_i = 1$, or otherwise labelled as $z_i = 0$ if they are derived from *true-false* and *false-false* interaction types. For simplicity, we specify features with $z_i = 1$ as *reference* features, and features with $z_i = 0$ as *error* features.

When reference data is available, the estimation of the parameters $\boldsymbol{\theta} = (\mu, \lambda)$ in each energy function in the previous chapter can be summarized in the following three steps:

1) Categorize the features **d** into *reference* and *error* groups;
2) Model the likelihood distributions for *reference* and *error* groups, i.e., $p(d|z = 1)$ and $p(d|z = 0)$, to calculate the likelihood ratio (see Equation **(5.19)**), and further the posterior probability $p(z = 0|d)$ (see Equation **(5.18)**);
3) Fit the sigmoid function to the posterior probability distribution to estimate the parameter $\boldsymbol{\theta} = (\mu, \lambda)$.

In step 2), the likelihood distribution of features are selected empirically, especially for *reference* features, i.e., Log-logistic distribution is used to model *reference* asymmetric ratio, Weibull distribution for *reference* area ratio, and exponential distribution for *reference* overlap ratio.

We propose in this study to model the posterior probability using logistic regression model directly to avoid the empirical selection of distributions for different features. According to logistic regression model, the posterior probability of a feature d_i having the label $z_i = 0$ takes the form of

$$p(z_i = 0|d_i) = \frac{1}{1 + \exp -(\beta_0 + \beta_1 d_i)} \tag{6.1}$$

The **Equation (6.1)** actually takes the same form of the sigmoid function $\sigma(d_i, \boldsymbol{\theta})$:

$$\sigma(d_i, \boldsymbol{\theta}) = \frac{1}{1 + \exp -\left(\frac{d_i - \mu}{\lambda}\right)} \tag{6.2}$$

It can be observed that the coefficient (β_0, β_1) in the logistic function is equivalent to the parameter pair $(-\mu/\lambda\,, 1/\lambda)$ in the sigmoid function.

Further assume the observations are independent of each other, the conditional likelihood function of all features **d** with corresponding labels **z** can be written as

$$p(\mathbf{z}|\mathbf{d}, \boldsymbol{\theta}) = \prod_{i=1}^{n} \sigma(d_i, \boldsymbol{\theta})^{1-z_i}\big(1 - \sigma(d_i, \boldsymbol{\theta})\big)^{z_i} \tag{6.3}$$

The parameter $\boldsymbol{\theta}$ is then estimated by maximizing the likelihood of $p(\mathbf{z}|\mathbf{d}, \boldsymbol{\theta})$

$$\widehat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} p(\mathbf{z}|\mathbf{d}, \boldsymbol{\theta}) \tag{6.4}$$

In another word, the maximum likelihood estimation entails finding the parameter $\boldsymbol{\theta}$ for which the probability of the observed data is greatest.

The improved parameter estimation method can be summarized as follows:

1) Categorize the features **d** into *reference* and *error* groups, given reference data;
2) Estimate the parameter $\boldsymbol{\theta} = (\mu, \lambda)$ by maximizing the likelihood of $p(\mathbf{z}|\mathbf{d}, \boldsymbol{\theta})$ (see **Equation (6.3)** and **(6.4)**).

6.2.2 Automatic parameter estimation based on Expectation-Maximization

To automatically estimate the parameters without any reference data, the main difficulty lies in that the corresponding labels **z** for collected features **d** is unknown. The problem falls within the framework of "missing data"[4]. Relating to our model, the complete data is considered to consist of two parts: the actual values of collected features **d** (the observed part) and the corresponding labels **z** (the missing part or unobserved part).

[4] More precisely, our case falls within the framework of "incomplete data", where the configuration of the objects to be extracted is unknown. This framework is more general than the "missing data". The "incomplete" part is compensated using a Monte Carlo sampling method we proposed (see Section 5.5.2) to approximate the feature likelihoods.

In such a situation, the Expectation-Maximization (EM) algorithm offers an appropriate framework for estimating the parameters. The EM algorithm, introduced by (Dempster et al., 1977), is a general technique for finding maximum likelihood estimates in problems where some variables are unobserved.

The EM algorithm estimates the parameters iteratively, starting from some initial guess. Each iteration consists of two steps: the Expectation (E) step estimates the distribution of the unobserved variables, given the known values of observed data and the current estimate of the parameters, while the Maximization (M) step re-estimates the parameters by maximum likelihood estimator, with the assumption that the distribution found in E step is correct.

The parameters can be approximated with local optimal solution using a variant of the EM algorithm (Neal and Hinton, 1998). In our model, the unobserved variables take a discrete binary value of $\{0, 1\}$, indicating the labels of the observed features as "false" or "true". The E-step minimizes the energy of the model (**Equation (5.5)**) with respect to the distribution over the unobserved variables, and updates the labels of features; the M-step uses the updated labels to estimate the parameters through maximum likelihood of the logistic regression models (**Equation (6.3)**) on features with updated labels. The partial E-step in the EM is a "winner-take-all" version according to Neal and Hinton (1998), in which the distributions over unobserved variables are restricted to those in which a single value has probability one or zero. The use of EM is somewhat usual, but the method is still valid. A detailed description of this procedure is given below:

A. The E-Step: Find the minimum energy estimator, and update the label vector

With the current parameter $\boldsymbol{\theta}^{(t)}$, E-step first finds the best configuration of tree objects $\mathbf{x}^{(t)}$ through optimization as minimum energy estimator:

$$\mathbf{x}^{(t)} = arg\min_{\mathbf{x}\in\Omega} U_{\boldsymbol{\theta}^{(t)}}(\mathbf{x}) \tag{6.5}$$

We treat the obtained optimal configuration as reference data to update the labels $\mathbf{z}^{(t)}$ of collected features **d** using the procedure we developed in **Section 5.5.2**.

B. The M-Step: Parameter estimation through maximum likelihood estimation

Once we have updated label vector for the extracted features, the M-step estimates the parameter $\boldsymbol{\theta}^{(t+1)}$ in sigmoid functions by modeling them with logistic regression model we developed in **Section 6.2.1**:

$$\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} p\left(\mathbf{z}^{(t)}\middle|\mathbf{d},\boldsymbol{\theta}\right) \tag{6.6}$$

C. First E-Step: Initial parameter setting

To get the algorithm started, we need to set initial parameters to run the optimization process to find the minimum energy estimator. The initial parameter setting of EM algorithm is important as we do not want it get stuck in a local optimum. We want to investigate in this study the influences of initial parameter settings on the performance of the EM algorithm with respect to

parameter estimation results. More specifically, we aim to answer two questions through this experiment: 1) whether and how initial parameter setting can affect the estimation results, and 2) find out which parameter setting can produce the best results, under which forest condition.

6.3 Model Optimization

In the previous chapter, we have shown a Markov Chain Monte Carlo (MCMC) embedded simulated annealing could be applied to search for the optimal configuration with minimum global energy in the configuration space. In that application, the transition of the Markov Chain was managed by a birth-and-death process, in which a treetop is either added or removed in the set of seed points to generate a new configuration.

This birth-and-death process is implemented by maintaining the Boolean selection vector size $|T|$, where T is the set of all extracted local maxima. The auto-adjoint move boils down to sampling uniformly an element u of T and inverting its selection. In this perspective, we treated the transition between configurations as symmetric in our previous experiment.

In this study, we attempt to view the transitions as jumps between configurations of different dimensions and resort to the Reversible Jump MCMC (RJMCMC) algorithm to explore the configuration space. We will compare the performances of RJMCMC with MCMC and investigate the influence of different cooling schedules of simulated annealing (SA) on the optimization performances of both algorithms.

6.3.1 Prior-Guided MCMC algorithm

The RJMCMC algorithm (Green, 1995) allows us to build a Markov Chain (X_t), $t \in \mathbb{N}$ which jumps between the difference dimensions of a configuration space Ω. At each step, the transition of the chain is managed by a set of proposition kernels denoted by $Q_m(\mathbf{x}, .)$, which propose a transformation of the current configuration $\mathbf{x}$ into a new configuration $\mathbf{y}$. The move is accepted with a probability $\alpha = \min\big(1, R(\mathbf{x}, \mathbf{y})\big)$, where:

$$R(\mathbf{x}, \mathbf{y}) = \frac{\mathcal{P}(d\mathbf{y}) Q_m(\mathbf{y}, d\mathbf{x})}{\mathcal{P}(d\mathbf{x}) Q_m(\mathbf{x}, d\mathbf{y})} \tag{6.7}$$

is called the Green ratio. This will ensure the transitions between different states respecting the reversibility assumption of the Markov Chain, with $\mathcal{P}(d\mathbf{x})$ as equilibrium distribution. The procedure of the algorithm is illustrated in **Figure 6.1**.

The number of trees growing in a forest plots often follows a Poisson distribution in practice. In our implementation of the PGMCMC, we use it as a reference measure of our stochastic model. The sampling of the Markov Cain is then guided by a prior information of tree density of the forest plot, which converges ergodically to the reference distribution[5]; this is why we call the sampling algorithm as prior-guided MCMC (PGMCMC).

[5] Reference distribution refers to the Poisson distribution, with $v(.)$ as the intensity measure, proportional to the Lebesgue measure on space $\mathcal{S}$ w.r.t. the forest plot.

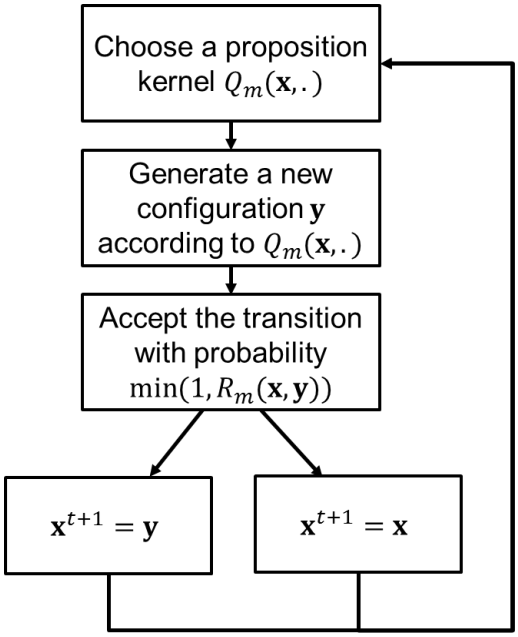


Figure 6.1. Green's algorithm.

In PGMCMC algorithm, similar to the previous MCMC, only birth and death kernels are used in the transition of the Markov Chain. However, when the prior information on tree density of the forest plot is taken into consideration, the calculation of acceptance ratio needs to be modified.

For each move, we define the probability of selecting a birth kernel or a death kernel to be equal: with probability $\frac{1}{2}$ propose to add a local maximum u from T/T^k to the current local maxima set T^k, and with probability $\frac{1}{2}$ propose to remove a local maximum v from the current local maxima set T^k. For a given state $X_t = \mathbf{x}(T^k)$,

- In case of a birth, $\mathbf{y} = \mathbf{x}(T^{k+1}) = \mathbf{x}(T^k \cup \{u\})$, the Green's ratio is[6]:

$$R(\mathbf{x}, \mathbf{y}) = \frac{\mathcal{P}(\mathbf{y})v(\mathcal{S})}{\mathcal{P}(\mathbf{x})(n(\mathbf{x}) + 1)} = \frac{v(\mathcal{S})}{|T^k| + 1} \exp - \big(U(\mathbf{y}) - U(\mathbf{x})\big) \tag{6.8}$$

where $n(.)$ is the number of objects in a configuration, and $v(.)$ is the intensity measure, proportional to the Lebesgue measure on space $\mathcal{S}$.

- In case of a death, $\mathbf{y} = \mathbf{x}(T^{k+1}) = \mathbf{x}(T^k \backslash \{v\})$, the Green's ratio is:

$$R(\mathbf{x}, \mathbf{y}) = \frac{\mathcal{P}(\mathbf{y})n(\mathbf{x})}{\mathcal{P}(\mathbf{x})v(\mathcal{S})} = \frac{|T^k|}{v(\mathcal{S})} \exp -\big(U(\mathbf{y}) - U(\mathbf{x})\big) \tag{6.9}$$

where $n(.)$ is the number of objects in a configuration and $v(.)$ is the intensity measure of the associated space S.

With probability $\alpha = \min(1, R(\mathbf{x}, \mathbf{y}))$, we accept the proposition $X_{t+1} = \mathbf{y}$. Otherwise, the configuration stays the same, in which case $X_{t+1} = \mathbf{x}$.

---

[6] For simplicity, in equations (6.8) and (6.9), we write the current state $\mathbf{x}(T^k)$ as $\mathbf{x}$, corresponding to the denotation of the new configuration $\mathbf{x}(T^{k+1})$ as $\mathbf{y}$.

We will put the PGMCMC algorithm to test in comparison of MCMC to see their performances under different optimization settings, which is detailed in next section.

6.3.2 Cooling schedules of simulated annealing

Model optimization is still achieved by embedding the PGMCMC algorithm in a simulated annealing scheme. The optimization procedure of SA is detailed in **Section 5.6.2**.

The evolution of the temperature parameter during the optimization process is called a cooling schedule. Mostly, we used an exponential, also called geometrical, cooling schedule for the SA optimization, which takes the form:

$$T_t = T_0 * a^t \tag{6.10}$$

with $\alpha < 1$ and very close to 1, and t is the number of iterations. A slight adaption of the schedule was made in which the temperature is updated every k iterations of the algorithm, which is also called the fixed length plateau cooling schedule. This allows the Monte Carlo sampler to explore the configuration space more thoroughly to reach the global minimum. In the previous experiments, the temperature decrease coefficient α is set to 0.9, and the plateau length k set to 500.

Alternatively, the temperature can be updated in each iteration, i.e. $k = 1$, if we set α to be small enough, e.g. 0.9998~0.99995. This also enables the Markov Chain to have more time to reach its equilibrium distribution, while reducing the parameters controlling the cooling schedule to only the temperature decrease coefficient α. Another factor influencing the convergence of the Markov

Chain is the initial temperature. It need to be large enough to allow the sampler to jump out of local minima and finally reach global minimum.

We intend to investigate in this study the influences of the those factors on the optimization performances of the two algorithms, MCMC and PGMCMC, attempting to test the robustness and sensitivity of the algorithms with respect to the cooling parameters, and more importantly, suggesting more appropriate cooling schedules for the SA optimization.

6.4 Results and Discussion

In this section, we summarize results obtained from different experiments we proposed above on parameter estimation and model optimization. Experiments were carried out consistently on both real and simulated forest plots.

6.4.1 Parameter estimation results

Modified parameter estimation with reference data

Figure 6.2 illustrates an example of modeling the posterior distributions of all three features (asymmetric ratio, area ratio and overlap ratio) with logistic regression models. The vertical axis gives a measure of the probability that a feature extracted comes from the error group based on observations. The blue dash line is the modeled logistic regression curve. As we can see from the graph, the logistic regression model plays a role very similar to the designated energy function. In effect, their equivalence in form makes the direct estimation of parameters possible.

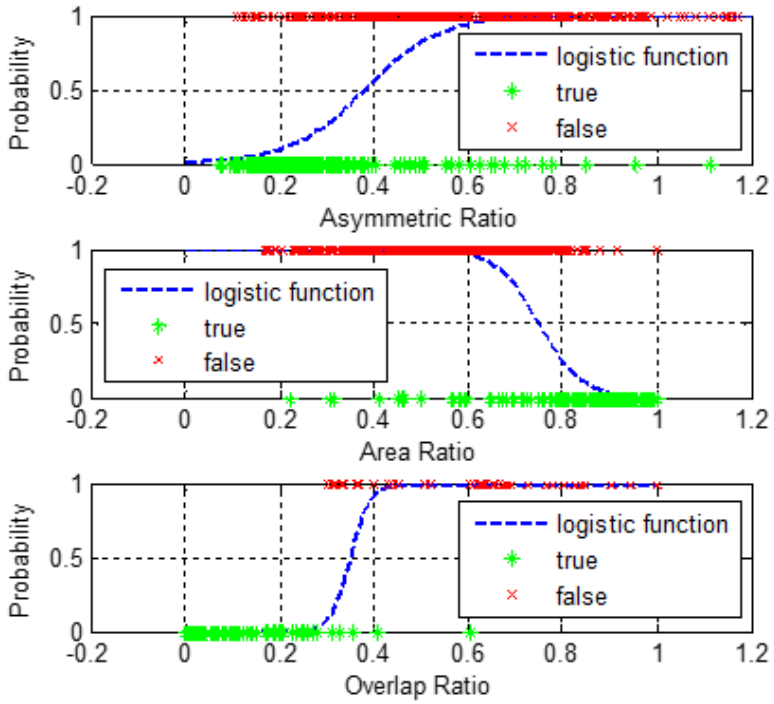


Figure 6.2. Logistic regression models are used to estimate parameters in sigmoid energy functions directly.

An advantage of this method is that it makes no assumptions about distribution of classes in feature space, and avoids the empirical selections of distributions for different features. The logistic regression model provides a natural probabilistic view of class prediction, which can be directly utilized as energy functions in similar cases. The method is thus more intuitive contrasting with the previous method as can be observed from **Figure 5.9**. This method is also effective in computation and gives good accuracy, as it is resistant to over fitting by avoiding empirical distribution selections as mentioned above.

However, it is worth noting that maximum likelihood can exhibit severe over-fitting for data sets that are linearly separable (Bishop, 2006). In this case, the logistic sigmoid function becomes infinitely steep in feature space, corresponding to a Heaviside step function, so that in binary classification, every training point from each class is assigned a posterior probability of 0 or 1. Therefore, this method should be process with caution according to the characteristics of forest plots. If the trees in the plot are all of regular shapes and there are few interactions between each other, this method will tend to produce a very steep regression curve, which means a very narrow "transition" zone between "true" and "false" features, resulting in similar configurations less distinguishable from their measures in energy. In such situations, a "softer" energy function curve is usually favored, to better guides the transition of Markov Chain in optimization process to find the optimal configuration. Another measure we should take to reduce such effect is that in the Monte Carlo sampling procedure, we need to produce sufficient random configurations so as to ensure features collected fill the whole range of their representative distributions.

Automatic parameter estimation based on Expectation-Maximization

(1) Initial Parameter Settings

To answer the research questions, we set up five groups of parameters as initial setting to test the performance of the EM algorithm in different forest plots (see **Table 6.1**). Each group corresponds to a typical parameter setting of our model under certain forest condition. In our experiment, we specify five different forest conditions with crown overlap from separate to overlap: extremely separate (ES), separate (S), touch (T), overlap (O) and extreme overlap (EO). The parameter values in each group are set with reference to the parameters estimated from actual forest plots with different degrees of crown overlap with reference data.

Table 6.1: Initial parameter settings for EM parameter estimation

		Initial Parameter Settings Five crown conditions from separate to overlapping						
		ES	S	T	O	EO	Ave.	Δx
Symmetric Function	μ_s	0.20	0.30	0.40	0.50	0.60	0.40	0.10
	λ_s	0.03	0.05	0.10	0.15	0.20	0.10	0.05
Area Ratio Function	μ_a	0.90	0.80	0.70	0.60	0.50	0.70	-0.10
	λ_a	-0.02	-0.03	-0.07	-0.12	-0.15	-0.06	-0.04~-0.01
Overlap Function	μ_o	0.05	0.10	0.35	0.60	0.70	0.28	0.05~0.25
	λ_o	0.01	0.01	0.03	0.05	0.05	0.03	0~0.02

* Note: ES - Extreme Separate; S - Separate; T - Touch; O - Overlap; EO - Extreme Overlap; Ave. - Average; Δx - increamentals;

(2) Overall performances of the EM method

Table 6.2 gives a summary of the iteration numbers the EM algorithm runs when reaching convergence with different initial parameter setting. Generally, the EM shows its effectiveness in estimating the parameters and finding the corresponding optimal configuration. For the six forest plots, the EM typically converges in 2-3 iterations for all different initial parameter settings. In some extreme cases of the experiments we carried out, the EM runs up to 4 iterations before it reaches terminal condition.

Table 6.3 shows the detection qualities of the EM for different initial parameters for both real and simulated forest plots. We can notice there are some variations in the detection quality of the EM w.r.t. different initial parameter settings. The EM with initial parameter settings T, O, and EO commonly produce higher detection qualities than that with the initial parameter settings ES and S. The cells with darker color in the tables show worse detection qualities.

Table 6.4 shows the comparison of the detection qualities produced by the EM method with different initial parameter settings and that from the standard optimization (MCMC) with parameters estimated with reference data (the last row in the table highlighted in yellow color, referred to as ***REF*** hereafter). The negative values in percentage show a degradation in detection quality of the EM in compare with ***REF***. It shows that ***REF*** has the best performances in all cases. This coincides with our expectation as the parameters learned from the reference data best fits the characteristics of the forest plot to detect trees from. The EM with initial parameter settings T, O, and EO, produce very close performances with ***REF***, if not the same. The slight decrease in detection quality corresponds to only one to two trees' differences in detection quantity-wise, which indicates the effectiveness of the EM as an unsupervised detection method. ES and S are the initial parameter settings producing worst detect qualities among the five. Detailed detection results can be found in **Appendix A Table 7.1-7.6**.

The results give evidence that EM is influenced by the initial condition. Although the EM is very efficient at reaching terminal condition and converges quickly, its performances in detection quality varies with the five pre-set initial parameter settings, which corresponds to typical crown overlap situations from extremely separated (ES) to extremely overlap (EO). The parameter settings T and O yield best performances among the five, which are very close to ***REF***, while ES and S result in much degraded detection quality, which indicates the EM may be trapped in some local minima. The parameter settings of EO has a performance in between. This may be explained by the fact that most plots in our experiments have degrees of crown overlap between touching and overlapping situation. We will have a more in-depth analysis about the reason behind the differences in the EM performances from the detailed parameter estimation results we will present in

next Section. However, the results have already indicated some best candidates to initialize the EM, as it can be noticed that the initial parameter settings T and O always produce best results across all plots with different degrees of crown overlap.

Table 6.2. Number of iterations the EM algorithm runs to estimate the parameters with different initial parameter settings for the real and simulated forest plots.

Parameter Settings	No. of EM Iterations					
	Real Forest Plots			Simulated Forest Plots		
	Plot 1	Plot 2	Plot 3	Separate	Touch	Overlap
ES	3	2	3	3	2	2
S	2	2	2	2	2	3
T	3	3	3	3	2	2
O	3	3	3	3	3	3
EO	3	3	3	3	3	3

Table 6.3. Detection qualities of EM algorithm w.r.t different initial parameter settings for both real and simulated forest plots

Parameter Settings	Detection Quality					
	Real Forest Plots			Simulated Forest Plots		
	Plot 1	Plot 2	Plot 3	Separate	Touch	Overlap
ES	56.2%	70.0%	55.0%	94.2%	63.6%	49.4%
S	73.5%	83.3%	72.9%	96.8%	88.0%	68.4%
T	85.0%	86.1%	81.2%	96.8%	91.5%	84.7%
O	84.9%	86.1%	82.0%	97.9%	91.5%	84.7%
EO	87.1%	81.4%	82.0%	97.9%	91.5%	84.7%

Table 6.4. Comparison of the detection qualities resulted from the EM method with different initial parameter settings and that from MCMC with the parameters estimated with reference data.

Parameter Settings	Detection Quality Compared with Reference					
	Real Forest Plots			Simulated Forest Plots		
	Plot 1	Plot 2	Plot 3	Separate	Touch	Overlap
ES	*-29.5%*	*-18.4%*	*-29.2%*	*-3.8%*	*-27.9%*	*-36.4%*
S	*-12.1%*	*-5.1%*	*-11.3%*	*-1.1%*	*-3.5%*	*-17.4%*
T	*-0.6%*	*-2.4%*	*-3.0%*	*-1.1%*	*0.0%*	*-1.1%*
O	*-0.7%*	*-2.4%*	*-2.3%*	*0.0%*	*0.0%*	*-1.1%*
EO	*1.5%*	*-7.0%*	*-2.3%*	*0.0%*	*0.0%*	*-1.1%*
REF	*85.6%*	*88.4%*	*84.2%*	*97.9%*	*91.5%*	*85.8%*

(3) Summaries of parameter estimation results

Table 6.5 and **Table 6.6** summarize parameter estimation results w.r.t. different initial parameter settings for the simulated and real forest plots respectively. To better illustrate how the parameters estimated by EM compared with those estimated using reference data, we produce **Table 6.7** and **Table 6.8**. In the comparison tables, the positive values show how much the EM estimated parameters are greater than that estimated with reference data, while the negative values indicate the opposite. The cells in the tables are rendered with colors according to the scale of offsets between the EM estimated parameters and reference parameters[7].

[7] In Table 6.7 and Table 6.8, positive offsets are rendered in red color, while negative in green. The larger the offsets, the brighter the color.

As can be observed, the offsets in the estimated parameters correspond to the detection qualities presented earlier. The larger the offsets, the greater the decrease in detection quality compared with reference. Generally, the initial parameter settings T, O, and EO have the best performances with small offsets in estimated parameters compared with reference. Initial parameter setting S follows in terms of performance, while ES performs worst with the largest offsets. Among them, T and O perform best and their estimated parameters are almost the same. It is interesting to notice that, there is a big contrast in the performances of the two extreme initial parameter settings EO and EO: ES always produces the greatest offsets, while EO has very close performance with T and O. This can be explained by that ES sets very strict constraint on shape regularity of crowns and overlap between them, and its low tolerance resulting in much reduced resolution in 'not-so-good' crowns and their interactions, which cannot be reflected well on the energy of the configuration. These two properties are controlled by the parameters μ and λ in the sigmoid function, respectively. The ES will thus more likely to retain more 'bad' objects in the optimal configurations reached especially in its initial iterations, which results in much biased distributions of collected features and in turn parameters estimated with larger offsets. Therefore, the EM with ES is more likely to be trapped in some local minima.

Table 6.5. Parameter estimation results of the EM method w.r.t. different initial parameter settings for simulated forest plots.

Plots	Parameter Setttings	Estimated Parameters					
		Asymmetric		Area Ratio		Overlap	
		μ'	λ'	μ'	λ'	μ'	λ'
Separate	**ES**	0.33	0.05	0.76	-0.02	0.78	0.00
	S	0.34	0.06	0.77	-0.02	0.07	0.01
	T	0.33	0.06	0.77	-0.02	0.07	0.01
	O	0.33	0.06	0.78	-0.02	0.07	0.01
	EO	0.33	0.06	0.78	-0.02	0.07	0.01
	REF	0.33	0.06	0.78	-0.02	0.09	0.01
Touch	**ES**	0.39	0.06	0.73	-0.03	0.09	0.01
	S	0.38	0.07	0.74	-0.03	0.22	0.02
	T	0.38	0.07	0.75	-0.04	0.32	0.02
	O	0.38	0.07	0.75	-0.04	0.32	0.02
	EO	0.38	0.07	0.75	-0.04	0.32	0.02
	REF	0.38	0.08	0.75	-0.05	0.35	0.02
Overlap	**ES**	0.48	0.13	0.66	-0.10	0.07	0.01
	S	0.41	0.09	0.72	-0.06	0.71	0.21
	T	0.41	0.09	0.72	-0.06	0.38	0.04
	O	0.41	0.09	0.72	-0.06	0.37	0.04
	EO	0.41	0.09	0.72	-0.06	0.37	0.04
	REF	0.44	0.14	0.71	-0.12	0.60	0.02

Table 6.6. Parameter estimation results of the EM method w.r.t. different initial parameter settings for real forest plots.

Plots	Parameter Setttings	Estimated Parameters					
		Asymmetric		Area Ratio		Overlap	
		μ'	λ'	μ'	λ'	μ'	λ'
Plot 1	**ES**	0.49	0.15	0.61	-0.09	0.03	0.01
	S	0.45	0.11	0.65	-0.06	0.58	0.01
	T	0.44	0.11	0.67	-0.06	0.35	0.02
	O	0.43	0.11	0.68	-0.06	0.35	0.02
	EO	0.44	0.12	0.68	-0.07	0.43	0.02
	REF	0.43	0.10	0.67	-0.06	0.28	0.04
Plot 2	**ES**	0.55	0.17	0.59	-0.08	0.02	0.002
	S	0.40	0.10	0.70	-0.05	0.20	0.04
	T	0.41	0.11	0.69	-0.06	0.21	0.02
	O	0.41	0.11	0.69	-0.06	0.21	0.02
	EO	0.40	0.10	0.70	-0.05	0.22	0.02
	REF	0.39	0.11	0.69	-0.07	0.32	0.05
Plot 3	**ES**	0.56	0.18	0.58	-0.10	0.06	0.01
	S	0.51	0.13	0.63	-0.08	0.62	0.003
	T	0.49	0.12	0.66	-0.07	0.32	0.03
	O	0.49	0.12	0.66	-0.08	0.29	0.04
	EO	0.49	0.13	0.66	-0.08	0.29	0.04
	REF	0.46	0.13	0.66	-0.11	0.59	0.03

Table 6.7. Comparison of parameters estimated from EM method w.r.t. different initial parameter settings with that estimated with reference data for simulated forest plots.

Plots	Parameter Setttings	Estimated Parameters Compared with Reference					
		Asymmetric		Area Ratio		Overlap	
		$\Delta\mu'$	$\Delta\lambda'$	$\Delta\mu'$	$\Delta\lambda'$	$\Delta\mu'$	$\Delta\lambda'$
Separate	ES	0.00	-0.01	-0.02	0.00	0.69	-0.01
	S	0.01	0.00	-0.01	0.00	-0.02	0.00
	T	0.00	0.00	-0.01	0.00	-0.02	0.00
	O	0.00	0.00	0.00	0.00	-0.02	0.00
	EO	0.00	0.00	0.00	0.00	-0.02	0.00
	REF	0.33	0.06	0.78	-0.02	0.09	0.01
Touch	ES	0.01	-0.02	-0.02	0.02	-0.26	-0.01
	S	0.00	-0.01	-0.01	0.02	-0.13	0.00
	T	0.00	-0.01	0.00	0.01	-0.03	0.00
	O	0.00	-0.01	0.00	0.01	-0.03	0.00
	EO	0.00	-0.01	0.00	0.01	-0.03	0.00
	REF	0.38	0.08	0.75	-0.05	0.35	0.02
Overlap	ES	0.04	-0.01	-0.05	0.02	-0.53	-0.01
	S	-0.03	-0.05	0.01	0.06	0.11	0.19
	T	-0.03	-0.05	0.01	0.06	-0.22	0.02
	O	-0.03	-0.05	0.01	0.06	-0.23	0.02
	EO	-0.03	-0.05	0.01	0.06	-0.23	0.02
	REF	0.44	0.14	0.71	-0.12	0.60	0.02

Table 6.8. Comparison of parameters estimated from EM method w.r.t. different initial parameter settings with that estimated with reference data for real forest plots.

Plots	Parameter Setttings	Estimated Parameters Compared with Reference					
		Asymmetric		Area Ratio		Overlap	
		$\Delta\mu'$	$\Delta\lambda'$	$\Delta\mu'$	$\Delta\lambda'$	$\Delta\mu'$	$\Delta\lambda'$
Plot 1	ES	0.06	0.05	-0.06	-0.03	-0.25	-0.03
	S	0.02	0.01	-0.02	0.00	0.3	-0.03
	T	0.01	0.01	0.00	0.00	0.07	-0.02
	O	0.00	0.01	0.01	0.00	0.07	-0.02
	EO	0.01	0.02	0.01	-0.01	0.15	-0.02
	REF	0.43	0.1	0.67	-0.06	0.28	0.04
Plot 2	ES	0.16	0.06	-0.10	-0.01	-0.30	-0.05
	S	0.01	-0.01	0.01	0.02	-0.12	-0.01
	T	0.02	0.00	0.00	0.01	-0.11	-0.03
	O	0.02	0.00	0.00	0.01	-0.11	-0.03
	EO	0.01	-0.01	0.01	0.02	-0.10	-0.03
	REF	0.39	0.11	0.69	-0.07	0.32	0.05
Plot 3	ES	0.10	0.05	-0.08	0.01	-0.53	-0.02
	S	0.05	0.00	-0.03	0.03	0.03	-0.03
	T	0.03	-0.01	0.00	0.04	-0.27	0.00
	O	0.03	-0.01	0.00	0.03	-0.30	0.01
	EO	0.03	0.00	0.00	0.03	-0.30	0.01
	REF	0.46	0.13	0.66	-0.11	0.59	0.03

6.4.2 Model optimization results

In this study, we carried out a series of experiments to compare the performances of algorithms MCMC and PGMCMC in our energy minimization problem and investigate the influence of different cooling schedules on the optimization performances.

Cooling Schedules for the Comparative Study

Table 6.9 shows the three cooling schedules we set for the comparative study. Parameter setting I acts as the reference group, with initial temperature T_0 and cooling parameter α having an equivalent cooling effect as the cooling schedule we used in **Chapter 5**. Parameter setting II has the same initial temperature as the reference, but a smaller cooling parameter, which will accelerate the temperature decrease. Parameter setting III has the same cooling parameter, but a much smaller initial temperature. Both of the two settings could increase the probability of the Monte Carlo sampler being trapped in local minima.

Table 6.9. Three optimization parameter settings for SA to test the performances of MCMC and PGMCMC.

	Optimization Parameters		
	I	II	III
Initial Temparature T_0	5	5	0.5
Cooling parameter α	0.99995	0.9998	0.99995
Iterations	8.75E+04	2.15E+04	4.15E+04

In the end, we will also test the algorithms using a series of initial temperatures in descending order from 5 to 0.1, with the same cooling parameter. All experiments are designed to test the robustness and sensitivity of the algorithms with respect to different parameters.

Optimization Results

(1) Results of three cooling Schedules

Table 6.10 and **Table 6.11** summarize the optimization statistics, showing the performance of the algorithms, MCMC and PGMCMC, with different cooling schedules on simulated and real forest plots, respectively.

For MCMC, we can see it can always find the optimal configuration for all plots with parameter setting I. The optimal energies reached for each plot are highlighted in bold in the tables. This indicates the initial temperature and cooling parameter of this cooling schedule allow MCMC to sample the configuration space sufficiently and reach global minimum, which justifies why we use it as "reference" cooling schedule. When we reduce the search time (or iterations) in the configuration space, either by reducing the cooling parameter (from 0.99995 to 0.9998 in II) or initial temperature (from 5.0 to 0.5 in III), we can see a degradation in the optimization performance of MCMC. Furthermore, the comparison also demonstrates the more time the Markov Chain spends in the searching space, the higher the chance it will can find the global optimal. Iteration number of cooling schedule III (~42500) is about half of the reference one (~87500), while about twice as much as that of cooling schedule II (~21500). The tables shows that, four out of six plots reached the global minimum for cooling schedule III, while only 2 out of 6 plots reached global minimum with cooling schedule II.

The results is in accordance with our expectation. It is often suggested in the literature to set the initial temperature T_0 as about twice the standard derivation of the energy $U(\mathbf{x})$ of random configurations (White, 1984). We calculated this value to be around 2.7~3.3 in different forest plots with infinite temperature. It has been shown in our experiments that it is enough to set the initial temperature T_0 as 5.0. For temperature decrement, as we only do one iteration at each temperature, we decrease the temperature very slowly by setting $\alpha = 0.99995$ in the reference cooling schedule. It should be noted that when the scale of forest plots increase, we should further increase the cooling parameter so that the configuration space can be sufficiently explored to find the global optimum. Generally speaking, in this discrete problem, we should always take into account the energy to be optimized, its scale, its landscape, the number and the size of local minima in the cooling schedule (Salamon et al., 2002a).

However, it is interesting to find out that PGMCMC does not present superior performance to MCMC, although when we proposed this method, we assumed PGMCMC could lead to more effective sampling of the configuration space in the more feasible regions by taking the prior information on tree density into consideration.

As PGMCMC is based on the assumption that the configuration dimensions obey certain distribution, the configuration subspace with higher probability will be sampled more intensively. Given this prior distribution on the configuration dimensions, PGMCMC is expected to be able to find the global minimum with less searching time than MCMC. In the experiments, although PGMCMC showed close performance with MCMC, it failed to reach the global minimum in two

cases with reference cooling schedule where MCMC was able to. This may due to that the sampling of PGMCMC is constrained in configuration space with much narrowed dimensions, whereas MCMC is more flexible in jumping between dimensions of a wider range, which results in a greater chance to find the global minimum.

Table 6.10. Tree detection results of MCMC and PGMCMC w.r.t. different cooling schedules in SA for simulated forest plots.

	Separate Plot					
	MCMC	PGMCMC	MCMC	PGMCMC	MCMC	PGMCMC
Parameter Setting	I		II		III	
Optimal Energy	**-82.3526**	**-82.3526**	-81.3873	-82.3085	-81.8704	**-82.3526**
Detection Quality	97.8%	97.8%	95.7%	98.4%	96.8%	97.8%
Number of objects	182	182	182	183	182	182
Correct	182	182	180	183	181	182
Commision Removed	29	29	27	29	28	29
Omission Produced	2	2	4	1	3	2
	Touch Plot					
	MCMC	PGMCMC	MCMC	PGMCMC	MCMC	PGMCMC
Parameter Setting	I		II		III	
Optimal Energy	**-88.7214**	**-88.7214**	-88.2760	-87.6737	**-88.7214**	**-88.7214**
Detection Quality	91.5%	91.5%	90.7%	89.9%	91.5%	91.5%
Number of objects	216	216	216	216	216	216
Correct	215	215	214	213	215	215
Commision Removed	33	33	32	31	33	33
Omission Produced	3	3	4	5	3	3
	Overlap Plot					
	MCMC	PGMCMC	MCMC	PGMCMC	MCMC	PGMCMC
Parameter Setting	I		II		III	
Optimal Energy	**-79.2539**	-78.8313	-78.9854	-78.5994	**-79.2539**	-78.9854
Detection Quality	85.8%	85.1%	85.1%	84.4%	85.8%	85.1%
Number of objects	224	224	224	224	224	224
Correct	224	223	223	222	224	223
Commision Removed	30	29	29	28	30	29
Omission Produced	2	3	3	4	2	3

Table 6.11. Tree detection results of MCMC and PGMCMC w.r.t. different cooling schedules in SA for real forest plots.

	Plot 1					
	MCMC	PGMCMC	MCMC	PGMCMC	MCMC	PGMCMC
Parameter Setting	I		II		III	
Optimal Energy	**-46.8015**	-46.3513	-46.7376	-46.6159	-46.7942	-46.6227
Detection Quality	85.6%	83.6%	86.3%	85.2%	84.3%	85.0%
Number of objects	125	126	124	130	127	126
Correct	113	112	113	115	113	113
Commision Removed	55	53	56	52	53	54
Omission Produced	5	6	5	3	5	5
	Plot 2					
	MCMC	PGMCMC	MCMC	PGMCMC	MCMC	PGMCMC
Parameter Setting	I		II		III	
Optimal Energy	**-14.6482**	**-14.6482**	**-14.6482**	**-14.6482**	**-14.6117**	**-14.6482**
Detection Quality	88.4%	88.4%	88.4%	88.4%	88.4%	88.4%
Number of objects	41	41	41	41	41	41
Correct	38	38	38	38	38	38
Commision Removed	10	10	10	10	10	10
Omission Produced	0	0	0	0	0	0
	Plot 3					
	MCMC	PGMCMC	MCMC	PGMCMC	MCMC	PGMCMC
Parameter Setting	I		II		III	
Optimal Energy	**-43.9913**	**-43.9913**	**-43.9913**	-43.8263	**-43.9913**	-43.9805
Detection Quality	84.2%	84.2%	84.2%	82.8%	84.2%	82.8%
Number of objects	123	123	123	123	123	123
Correct	112	112	112	111	112	111
Commision Removed	15	15	15	14	15	14
Omission Produced	3	3	3	4	3	4

The evolutions of the tree detection at different steps of the two algorithms, MCMC and PGMCMC, with reference cooling schedule are shown in **Figure 6.3** and **Figure 6.4**. For MCMC, at the beginning of the process, the detected configuration shows lots of false alarms in yellow and cyan circles (see **Figure 6.3**). 50,000 iterations later, it shows obvious improvements on the detected configuration with much less false alarms. Then, as long as the temperature decreases, more and more tree objects are correctly detected, and finally, the algorithm converges with optimal configuration having lowest global energy.

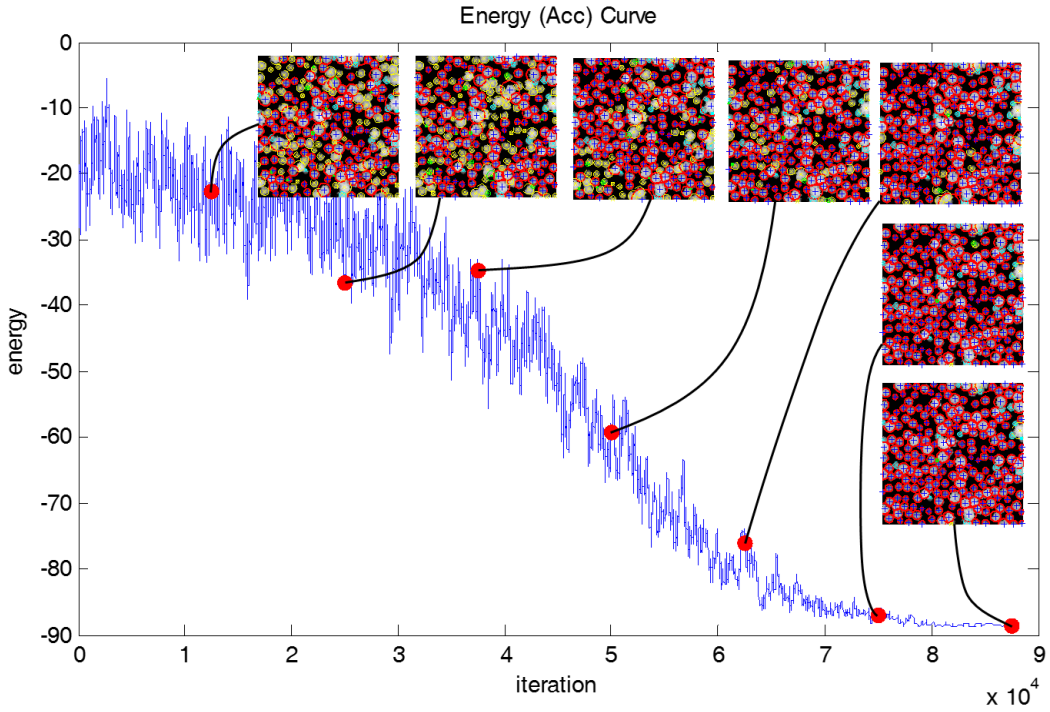


Figure 6.3. Evolution of MCMC at different temperatures. The curve shows the evolution of the energy. The samples denoted by dots correspond to the superimposed configurations. (Yellow circles in the plots show tree objects in the detected configuration as commission errors, while cyan circles represent omission errors. Red circles are corrected detected tree objects.)

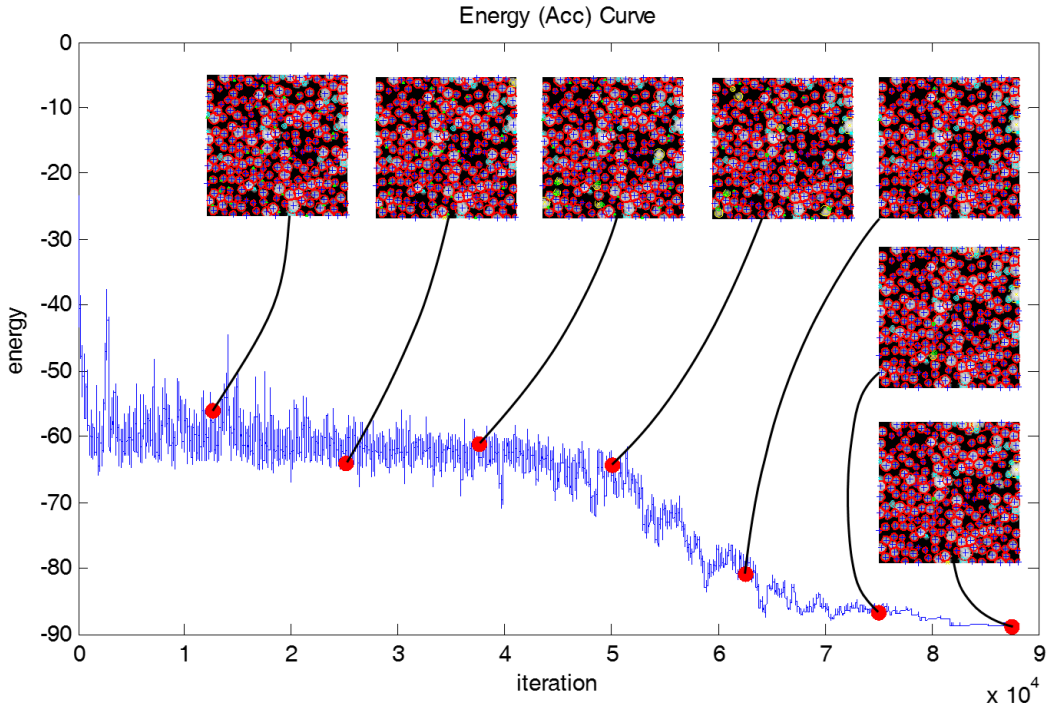


Figure 6.4. Evolution of PGMCMC at different temperatures. The curve shows the evolution of the energy. The samples denoted by dots correspond to the superimposed configurations. (Yellow circles in the plots show tree objects in the detected configuration as commission errors, while cyan circles represent omission errors. Red circles are corrected detected tree objects.)

PGMCMC, guided by the prior information on tree density, do show a much quicker convergence speed at the beginning stage of the optimization, which is clearly presented in **Figure 6.4**. The energy of model dropped very quickly for the first few thousands of iterations, and then entered into a stable process with an average energy of -60 until about 50,000 iterations. Afterwards, similar to MCMC, the algorithm went into a more distinct convergence process until global optimal was reached.

This interesting property of PGMCMC could lead to a compound strategy which accelerates the optimization process. As PGMCMC could enter into a stable stage very quickly, we may reduce the time it stays in the stable zone by decreasing the temperature more rapidly. As it enters the final convergence stage, MCMC may be adopted as it has more constant performance in finding the global minimum.

(2) Initial Temperature Influence

We would like to assess the influence of the initial temperature and observe the behaviors of the two algorithms in greater details. As can be seen in **Table 6.12**, we gradually reduce the initial temperature from 5.0 to 0.1, i.e., from about twice the standard deviation to twice the final temperature. It seems 0.5 is a turning point. With initial temperature greater than that, both algorithms can reach the optimal energy of -88.7214. PMCMC could achieve that with initial temperature of 0.2 as well. When temperature decreases, we lost more time in searching the configuration space. When the cooling starts with 0.1, it is "too late" and end up with configuration in local minima.

A more vivid impression on the behaviors of the two algorithms can be seen in **Figure 6.3** and **Figure 6.4**. As pointed out in the section above, PGMCMC could reach a stable stage in a very short period of time. Again, it can be observed that 0.5 can be regarded as a critical temperature for PGMCMC to turn from a "stable" stage into a distinct "convergence" stage. Those properties could be utilized in the design of a more effective optimization strategy.

Table 6.12. Detection results of MCMC and PGMCMC under different initial temperatures for touch plot.

	Initial Temperature					
	5.0	2.0	1.0	0.5	0.2	0.1
Cooling Parameter	0.99995					
Iterations	8.75E+04	6.90E+04	5.55E+04	4.15E+04	2.30E+04	9.50E+03
	MCMC					
Optimal Energy	**-88.7214**	**-88.7214**	**-88.7214**	**-88.7214**	-88.2760	-85.8614
Number of objects	216	216	216	216	216	212
Correct	215	215	215	215	214	208
Detection Quality	**91.5%**	**91.5%**	**91.5%**	**91.5%**	90.7%	87.4%
Commision Removed	33	33	33	33	32	30
Omission Produced	3	3	3	3	4	10
	PGMCMC					
Optimal Energy	**-88.7214**	**-88.7214**	**-88.7214**	**-88.7214**	**-88.7214**	-86.7509
Number of objects	216	216	216	216	216	216
Correct	215	215	215	215	215	211
Detection Quality	**91.5%**	**91.5%**	**91.5%**	**91.5%**	**91.5%**	88.3%
Commision Removed	33	33	33	33	33	29
Omission Produced	3	3	3	5	3	7

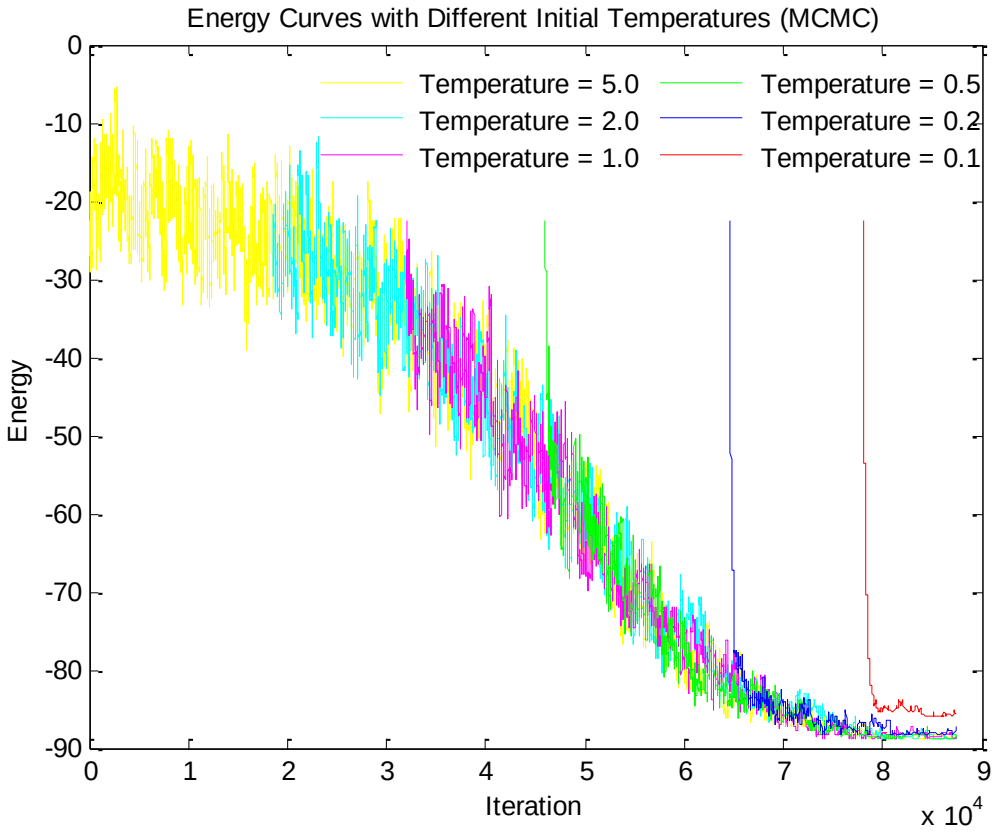


Figure 6.5. Energy curves of MCMC with different initial temperatures for touch plot.

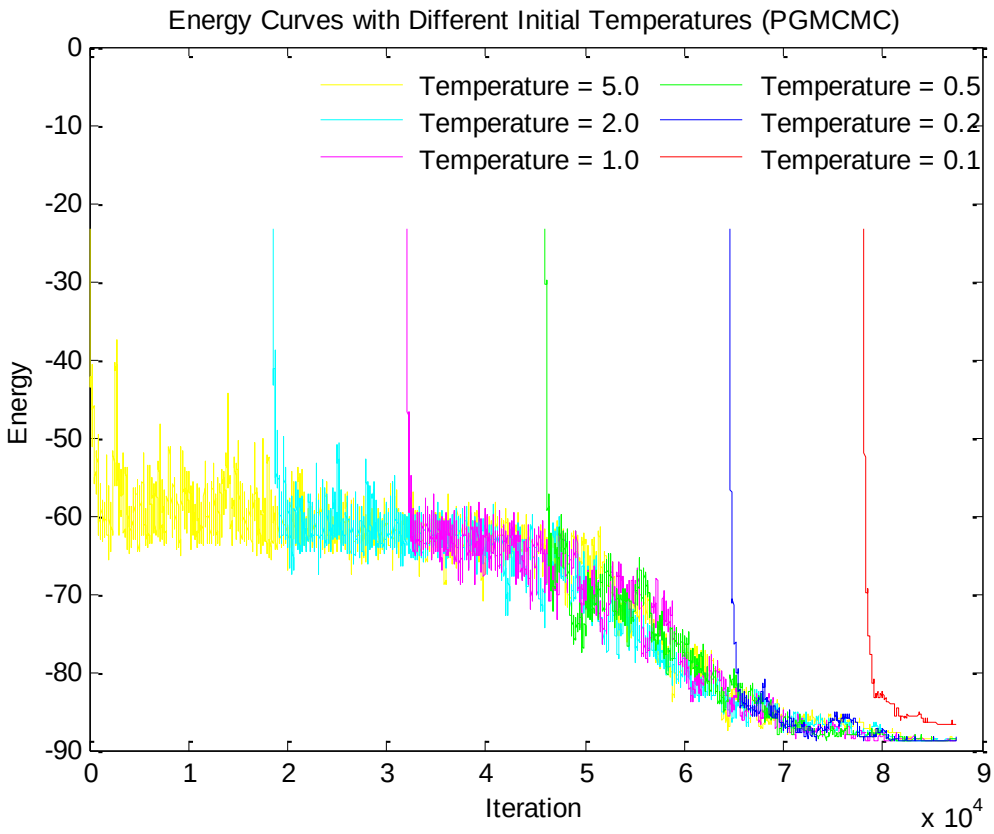


Figure 6.6. Energy curves of PGMCMC with different initial temperatures for touch plot.

6.5 Conclusion

In this chapter, we address the problems of parameter estimation and model optimization for the probabilistic model we proposed for single tree detection from ALS data.

We first presented a modified parameter estimation method when reference data is available. In this method, parameters are estimated by modelling the feature distributions with logistic regression directly, which avoids the empirical selections of distributions when approximating feature likelihoods in the previous method. With its intuitive nature, this method was integrated in an Expectation-Maximization procedure to estimate parameters automatically without using any reference data. This procedure minimizes human intervention and enables an unsupervised way for single tree detection.

Then, we proposed a modified optimization method resembling RJMCMC used in classical marked point process. The algorithm makes use of the prior information about tree density of the forest plot, which is why we call it prior-guided MCMC (PGMCMC). The algorithm is supposed to be able to sample more effectively the feasible regions of configuration space. Comparative studies have been carried out to test the performances of the two algorithms, MCMC and PGMCMC. We also studied how different cooling schedules influence the behaviors of the two algorithms, with the aim of designing more effective optimization strategy.

Chapter 7
Summary and Conclusions

7.1 Summary and Contributions

General work in computer vision aims to recover models or scenarios from degraded sensor information. Accurate detection of single tree detection from CHMs generated from ALS data is considered to be a difficult problem in both forestry and computer vision. The CHM images of forests have some distinguishing features which are different from that of urban areas. *First*, objects of interest are mostly of irregular shapes which are hard to be fitted with a geometrical model. However, objects presented in urban scenes are mostly man-made objects with regular shapes, such as roads, building, etc. *Second*, the presences of multiple objects of interest (e.g., cluster of trees), the variation of object size, shapes and relative locations limit the effective use of existing detection methods, which results in commission or omission errors. *Third*, in dense forest areas tree crows overlap and some trees are growing beneath the dominant crown layer, making them hard to be distinguished from each other or detected from above. Let alone the fact that the CHMs generated from ALS data with some conventional workflow are often problematic, with distorted crown shapes and pits distributed in tree crowns. All those factors make it not ideal to handle the tree detection and delineation task using the conventional image segmentation methods, whose main function is to partition the image into homogeneous regions. This work takes attempt to address those difficulties in single tree detection using ALS data.

In **Chapter 3**, we have dealt with the problem of degraded sensor data, or degradation of data during the pre-processing stage before single tree detection is actually carried. An improved

workflow of CHM generation from ALS point cloud is proposed, to eliminate the problem of crown shape distortions caused by shadow effects on ALS data, which often presents when using a conventional CHM generation workflow. The idea is to resample the terrain points of crown shadowed areas from DEM model generated in the first place and add them to the original ALS point cloud before DSM generation. When the shadowed areas are filled with repopulated terrain points, the large slopes previously presented on the shadow side of tree crowns will no longer be produced in the triangulation and interpolation procedure of DSM generation, which eliminates the crown shape distortions presented on CHM. Other than smoothing the raw CHM generated by a mean or Gaussian filter directly in the following step, a simple but effective pit-filling algorithm is applied to the raw CHM to produce pit-free CHM image, to reserve the geometric characteristics of CHM as much as possible. The CHM generated with the improved workflow is then used as the starting point for single tree detection.

Before proceeding to single tree detection method we developed, in **Chapter 4**, we present a method to generate simulated ALS data of forest plots. In this method, we make use of point process theory to model forest plots with certain spatial pattern, i.e. increasing degrees of tree density and crown overlap. Simulated ALS data are great in providing reference datasets without the time-consuming collection of expensive field data. They offer a fully controlled environment to test single tree detection algorithms. The simulated ALS data are used throughout the research (i) to examine the performance of our proposed single tree detection model, (ii) to validate the parameter estimation method of the model and observe the behavior of the estimated parameters, and (iii) to suggest the best initial parameter settings in EM-based automatic parameter estimation method, under forest conditions with increasing degrees of crown overlap.

Chapter 5 presents our main contributions. We propose a method from the perspective of computer vison to detect single trees from airborne laser scanning data in temperate mature coniferous forests. The hybrid framework we proposed attempts to take advantages of both low-level image processing techniques and high-level probabilistic models.

The probabilistic models in computer vision provide a systematic framework allowing inclusion of prior knowledge about spatial interaction between objects, while enabling the measure of the consistency between the model and the underlying image. For this advantage, probabilistic models have been adapted in various image processing problems to achieve solution with reduced errors w.r.t. deterministic models. Meanwhile, we make use of image features extracted by low-level image processing techniques, i.e. LM and MCWS, to generate a reduced configuration space, to accelerate the optimization process of the probabilistic model. To author's best knowledge, this is the first report integrating low-level image processing techniques and high-level probabilistic model into a hybrid framework for singe tree detection.

The novelty of the model also lies in the statistically sound parameter estimation method we proposed, which is another contribution to the probabilistic model. This method learns important parameters based on characteristics of crowns from the data of interest, whereas in many low-level image processing techniques based approaches or high-level models, parameters are usually set empirically.

Experimental results in this chapter showed the effectiveness of the single tree detection model we have proposed. The detection quality of the proposed method is superior to that obtained by local maxima filtering based approach by about 10%. The evaluation of results on both single tree detection and estimated parameters again proved the rationale of the parameter estimation method we proposed.

Chapter 6 makes further extensions on the model we introduced in **Chapter 5**, w.r.t. parameter estimation and model optimization. The extension on parameter estimation aims at developing an automatic method with which parameters in the model can be estimated according to the characteristics of data, a procedure that minimizes human intervention and ultimately achieves an unsupervised way for single tree detection without using any reference data for model training. With its intuitive nature, an Expectation-Maximization procedure is utilized for this purpose. Given the data and an initial condition, the EM algorithm recalculates the feature distributions for true and false tree objects, and re-estimates the parameters in each iteration until convergence.

Another enhancement of the parameter estimation method lies in how the parameters in the energy functions are estimated. In this upgraded version, parameters are estimated by modeling the posterior probabilities of features with logistic regression directly. This avoids the empirical selections of distributions when approximating feature likelihoods in the previous method, thus further improves the robustness of the parameter estimation methods, and in turn the applicability of the proposed probabilistic model.

While the EM procedure is reported to be influenced by initial condition and might be trapped in a local minimum, we also further investigate in this chapter the performance of the EM-based parameter estimation method with different of initial parameter settings corresponding to typical forest conditions. The tests also give insight and suggestion on which initial parameter setting(s) performs better and should be used when applying this method on ALS data collected from other forest conditions.

Furthermore, we tested another optimization method called prior-guided MCMC, which takes the prior information about tree density of the forest plots into consideration. This method resembles RJMCMC used in classical marked point process and supposed to be able to sample more effectively the feasible regions in the configuration space. We compared the performances of the two optimization algorithms, MCMC and PGMCMC, and studied how different cooling schedules influence the behaviors of the two algorithm. The comparative studies we have carried out give a suggestion for designing a more effective optimization strategy.

7.2 Recommendations for Future Work

As a recommendation for future work, post-processing will be introduced to recover omission errors from the detection results. Although the proposed model was proven effective in reducing commission errors, the tree positions are constrained in the predetermined set of the local maxima extracted by local maxima filtering. It is possible to recover a portion of the omitted trees from the detected results because those missed crowns will result in more geometrically irregular segments. Second, the developed single tree method is expected to be applied in various forest variable extraction applications where single trees are concerned. Finally, automated segmentation of forest

stands into homogenous areas with similar forest conditions can be introduced to help train parameters of the proposed model of representative regions and make the model applicable to larger areas.

Bibliography

Aldred, A., Bonnor, G. (1985). *Application of airborne lasers to forest surveys*.

Andersen, H.-E. (2003). *Estimation of critical forest structure metrics through the spatial analysis of airborne laser scanner data.* (Ph.D. 3079202), University of Washington, Washington, United States.

Andersen, H.-E., Reutebuch, S.E., Schreuder, G.F. (2002). *Bayesian object recognition for the analysis of complex forest scenes in airborne laser scanner data*. Paper presented at the International Archives of Photogrammetry and Remote Sensing.

Axelsson, P.E. (1999). Processing of laser scanner data - algorithms and applications. *ISPRS Journal of Photogrammetry and Remote Sensing, 54*(2-3), 138-147.

Baddeley, A., Turner, R. (2005). Spatstat: an R package for analyzing spatial point patterns. *J Stat Softw, 12*(6), 1-42.

Baldocchi, D., Valentini, R., Running, S., Oechel, W., Dahlman, R. (1996). Strategies for measuring and modelling carbon dioxide and water vapour fluxes over terrestrial ecosystems. *Global Change Biology, 2*(3), 159-168.

Ben-Arie, J.R., Hay, G.J., Powers, R.P., Castilla, G., St-Onge, B. (2009). Development of a pit filling algorithm for LiDAR canopy height models. *Comput Geosci-Uk, 35*(9), 1940-1949.

Besag, J. (1974). Spatial Interaction and the Statistical Analysis of Lattice Systems. *Journal of the Royal Statistical Society. Series B (Methodological), 36*(2), 192-236.

Besag, J. (1986). On the Statistical-Analysis of Dirty Pictures. *J Roy Stat Soc B Met, 48*(3), 259-302.

Beucher, S., Lantuejoul, C. (1979). *Use of watersheds in contour detection*. Paper presented at the International Workshop on Image Processing, Realtime Edge and Motion Detection/Estimation, Rennes, France.

Bishop, C.M. (2006). *Pattern recognition and machine learning*. New York: Springer.

Bortolot, Z.J., Wynne, R.H. (2005). Estimating forest biomass using small footprint LiDAR data: An individual tree-based approach that incorporates training data. *ISPRS Journal of Photogrammetry and Remote Sensing, 59*(6), 342-360.

Brandtberg, T. (2007). Classifying individual tree species under leaf-off and leaf-on conditions using airborne lidar. *ISPRS Journal of Photogrammetry and Remote Sensing, 61*(5), 325-340.

Brandtberg, T., Walter, F. (1998). Automated delineation of individual tree crowns in high spatial resolution aerial images by multiple-scale analysis. *Machine Vision and Applications, 11*(2), 64-73.

Brandtberg, T., Warner, T.A., Landenberger, R.E., McGraw, J.B. (2003). Detection and analysis of individual leaf-off tree crowns in small footprint, high sampling density lidar data from the eastern deciduous forest in North America. *Remote Sens. Environ., 85*(3), 290-303.

Brédif, M., Tournaire, O., Vallet, B., Champion, N. (2013). Extracting polygonal building footprints from digital surface models: A fully-automatic global optimization framework. *ISPRS Journal of Photogrammetry and Remote Sensing, 77*(0), 57-65.

Breidenbach, J., Næsset, E., Lien, V., Gobakken, T., Solberg, S. (2010). Prediction of species specific forest inventory attributes using a nonparametric semi-individual tree crown approach based on fused airborne laser scanning and multispectral data. *Remote Sens. Environ., 114*(4), 911-924.

Brown, S., Sathaye, J., Cannell, M., Kauppi, P.E. (1996). Mitigation of carbon emissions to the atmosphere by forest management. *The Commonwealth Forestry Review, 75*(1), 80-91.

Chauve, A., Vega, C., Durrieu, S., Bretar, F., Allouis, T., Deseilligny, M.P., Puech, W. (2009). Advanced full-waveform lidar data echo detection: Assessing quality of derived terrain and tree height models in an alpine coniferous forest. *International Journal of Remote Sensing, 30*(19), 5211-5228.

Chen, Q., Baldocchi, D., Gong, P., Kelly, M. (2006). Isolating individual trees in a savanna woodland using small footprint lidar data. *Photogrammetric Engineering and Remote Sensing, 72*(8), 923-932.

Comas, C., Mateu, J. (2007). Modelling forest dynamics: a perspective from point process methods. *Biometrical Journal, 49*(2), 176-196.

Cressie, N. (1992). Statistics for spatial data. *Terra Nova, 4*(5), 613-617.

Dalponte, M., Bruzzone, L., Gianelle, D. (2011). A System for the Estimation of Single-Tree Stem Diameter and Volume Using Multireturn LIDAR Data. *IEEE Transactions on Geoscience and Remote Sensing, 49*(7), 2479-2490.

Dempster, A., Laird, N., Rdin, D. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society, Series B, 39*(1), 1-38.

Descombes, X., Pechersky, E. (2006). *Tree Crown Extraction using a Three State Markov Random Field*.

Descombes, X., Zerubia, J. (2002). Marked point process in image analysis. *IEEE Signal Processing Magazine, 19*(5), 77-84.

Disney, M.I., Kalogirou, V., Lewis, P., Prieto-Blanco, A., Hancock, S., Pfeifer, M. (2010). Simulating the impact of discrete-return lidar system and survey characteristics over young conifer and broadleaf forests. *Remote Sens. Environ., 114*(7), 1546-1560.

Dixon, R.K., Solomon, A.M., Brown, S., Houghton, R.A., Trexier, M.C., Wisniewski, J. (1994). Carbon pools and flux of global forest ecosystems. *Science, 263*(5144), 185-190.

Dubayah, R.O., Drake, J.B. (2000). Lidar remote sensing for forestry. *J Forest, 98*(6), 44-46.

Elmqvist, M. (2001). *Ground estimation of laser radar data using active shape models.* Paper presented at the OEEPE workshop on airborne laserscanning and interferometric SAR for detailed digital elevation models.

Erikson, M. (2003). Segmentation of individual tree crowns in colour aerial photographs using region growing supported by fuzzy rules. *Canadian Journal of Forest Research, 33*(8), 1557-1563.

Falkowski, M.J., Smith, A.M.S., Hudak, A.T., Gessler, P.E., Vierling, L.A., Crookston, N.L. (2006). Automated estimation of individual conifer tree height and crown diameter via two-dimensional spatial wavelet analysis of lidar data. *Can J Remote Sens, 32*(2), 153-161.

FAO. (2015, March 23, 2015). FAO Forestry Department. Retrieved March 25, 2015, from http://www.fao.org/forestry/en/

Ferraz, A., Bretar, F., Jacquemoud, S., Gonçalves, G., Pereira, L., Tomé, M., Soares, P. (2012). 3-D mapping of a multi-layered Mediterranean forest using ALS data. *Remote Sens. Environ., 121*(0), 210-223.

Gebreslasie, M.T., Ahmed, F.B., Van Aardt, J.A.N., Blakeway, F. (2011). Individual tree detection based on variable and fixed window size local maxima filtering applied to IKONOS imagery for even-aged Eucalyptus plantation forests. *International Journal of Remote Sensing, 32*(15), 4141-4154.

Geman, S., Geman, D. (1984). Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-6*(6), 721-741.

Gier, A.d., Roy, P. (2003). A new approach to woody biomass assessment in woodlands and shrublands. *Geoinformatics for tropical ecosystems*, 161-198.

Girard, M.C. (2003). *Processing of Remote Sensing Data*. London: Taylor & Francis.

Gleason, C.J., Im, J. (2012). A Fusion Approach for Tree Crown Delineation from Lidar Data. *Photogrammetric Engineering and Remote Sensing, 78*(7), 679-692.

Gonzalez, R.C., Woods, R.E. (2008). *Digital image processing*, 3rd ed. Upper Saddle River, NJ: Pearson/Prentice Hall.

Gougeon, F.A. (1995). A crown-following approach to the automatic delineation of individual tree crowns in high spatial resolution aerial images. *Can J Remote Sens, 21*(3), 274-284.

Grabarnik, P., Särkkä, A. (2009). Modelling the spatial structure of forest stands by multivariate point processes with hierarchical interactions. *Ecological Modelling, 220*(9), 1232-1240.

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

Harding, D.J., Carabajal, C.C. (2005). ICESat waveform measurements of within - footprint topographic relief and vegetation vertical structure. *Geophysical research letters, 32*(21).

Hastings, W.K. (1970). Monte-Carlo Sampling Methods Using Markov Chains and Their Applications. *Biometrika, 57*(1), 97-&.

Heinzel, J., Koch, B. (2011). Exploring full-waveform LiDAR parameters for tree species classification. *International Journal of Applied Earth Observation and Geoinformation, 13*(1), 152-160.

Heurich, M. (2008). Automatic recognition and measurement of single trees based on data from airborne laser scanning over the richly structured natural forests of the Bavarian Forest National Park. *Forest Ecology and Management, 255*(7), 2416-2433.

Holopainen, M., Vastaranta, M., Kankare, V., Hyyppa, J., Liang, X., Litkey, P., Yu, X., Kaartinen, H., Kukko, A., Kaasalainen, S. (2011). *The use of ALS, TLS and VLS measurements in mapping and monitoring urban trees.* Paper presented at the Urban Remote Sensing Event (JURSE), 2011 Joint.

Hopkinson, C., Chasmer, L., Young-Pow, C., Treitz, P. (2004). Assessing forest metrics with a ground-based scanning lidar. *Can J Forest Res, 34*(3), 573-583.

Hu, B., Li, J., Jing, L., Judah, A. (2014). Improving the efficiency and accuracy of individual tree crown delineation from high-density LiDAR data. *International Journal of Applied Earth Observation and Geoinformation, 26*(0), 145-155.

Hyyppa, J., Hyyppa, H., Leckie, D., Gougeon, F., Yu, X., Maltamo, M. (2008). Review of methods of small-footprint airborne laser scanning for extracting forest inventory data in boreal forests. *International Journal of Remote Sensing, 29*(5), 1339-1366.

Hyyppä, J., Hyyppä, H., Leckie, D., Gougeon, F., Yu, X., Maltamo, M. (2008). Review of methods of small-footprint airborne laser scanning for extracting forest inventory data in boreal forests. *International Journal of Remote Sensing, 29*(5), 1339-1366.

Hyyppä, J., Inkinen, M. (1999). Detecting and Estimating Attributes for Single Trees Using Laser Scanner. *The Photogrammetric Journal of Finland, 16*(2), 27-42.

Hyyppä, J., Kelle, O., Lehikoinen, M., Inkinen, M. (2001). A segmentation-based method to retrieve stem volume estimates from 3-D tree height models produced by laser scanners. *IEEE Transactions on Geoscience and Remote Sensing, 39*(5), 969-975.

Hyyppä, J., Yu, X., Hyyppä, H., Vastaranta, M., Holopainen, M., Kukko, A., Kaartinen, H., Jaakkola, A., Vaaja, M., Koskinen, J., Alho, P. (2012). Advances in Forest Inventory Using Airborne Laser Scanning. *Remote Sensing, 4*(5), 1190-1207.

Jing, L., Hu, B., Li, J., Noland, T. (2012). Automated Delineation of Individual Tree Crowns from Lidar Data by Multi-Scale Analysis and Segmentation. *Photogrammetric Engineering and Remote Sensing, 78*(12), 1275-1284.

Jonsson, B., Jacobsson, J., Kallur, H. (1993). The Forest Management Planning Package. Theory and application. *Studia Forestalia Suecica*(189), 56.

Kaartinen, H., Hyyppä, J. (2008). *EuroSDR/ISPRS Project Commission II, Tree Extraction, Final Report*. EuroSDR. European Spatial Data Research, Official Publication No. 53, p. 56.

Kaartinen, H., Hyyppä, J., Yu, X., Vastaranta, M., Hyyppä, H., Kukko, A., Holopainen, M., Heipke, C., Hirschmugl, M., Morsdorf, F., Næsset, E., Pitkänen, J., Popescu, S., Solberg, S., Wolf, B.M., Wu, J.C. (2012). An International Comparison of

Individual Tree Detection and Extraction Using Airborne Laser Scanning. *Remote Sensing, 4*(4), 950-974.

Ke, Y.H., Quackenbush, L.J. (2011). A review of methods for automatic individual tree-crown detection and delineation from passive remote sensing. *International Journal of Remote Sensing, 32*(17), 4725-4747.

Koch, B., Heyder, U., Weinacker, H. (2006). Detection of individual tree crowns in airborne lidar data. *Photogrammetric Engineering and Remote Sensing, 72*(4), 357-363.

Korpela, I., Dahlin, B., Schäfer, H., Bruun, E., Haapaniemi, F., Honkasalo, J., Ilvesniemi, S., Kuutti, V., Linkosalmi, M., Mustonen, J. (2007). *Single-tree forest inventory using lidar and aerial images for 3D treetop positioning, species recognition, height and crown width estimation*. Paper presented at the International Archieves of Photogrammetry and Remote Sensing and spatial information sciences.

Kraus, K., Pfeifer, N. (1998). Determination of terrain models in wooded areas with airborne laser scanner data. *ISPRS Journal of Photogrammetry and Remote Sensing, 53*(4), 193-203.

Lacoste, C., Descombes, X., Zerubia, J. (2005). Point processes for unsupervised line network extraction in remote sensing. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 27*(10), 1568-1579.

Lafarge, F., Descombes, X., Zerubia, J., Pierrot-Deseilligny, M. (2008). Automatic building extraction from DEMs using an object approach and application to the 3D-city modeling. *ISPRS Journal of Photogrammetry and Remote Sensing, 63*(3), 365-381.

Lafarge, F., Gimel'Farb, G., Descombes, X. (2010). Geometric feature extraction by a multimarked point process. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 32*(9), 1597-1609.

Larsen, M., Eriksson, M., Descombes, X., Perrin, G., Brandtberg, T., Gougeon, F.A. (2011). Comparison of six individual tree crown detection algorithms evaluated

under varying forest conditions. *International Journal of Remote Sensing, 32*(20), 5827-5852.

Leckie, D., Gougeon, F., Hill, D., Quinn, R., Armstrong, L., Shreenan, R. (2003). Combined high-density lidar and multispectral imagery for individual tree crown analysis. *Can J Remote Sens, 29*(5), 633-649.

Lefsky, M.A. (2010). A global forest canopy height map from the Moderate Resolution Imaging Spectroradiometer and the Geoscience Laser Altimeter System. *Geophysical Research Letters, 37*.

Lefsky, M.A., Cohen, W.B., Parker, G.G., Harding, D.J. (2002). Lidar Remote Sensing for Ecosystem Studies. *BioScience, 52*(1), 19-30.

Lefsky, M.A., Keller, M., Pang, Y., de Camargo, P.B., Hunter, M.O. (2007). Revised method for forest canopy height estimation from Geoscience Laser Altimeter System waveforms. *Journal of Applied Remote Sensing, 1*.

Li, S.Z. (1994). *A Markov random field model for object matching under contextual constraints.* Paper presented at the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1994.

Li, S.Z. (2009). *Markov random field modeling in image analysis*, 3rd Edition ed: Springer-Verlag New York Inc.

Liang, X., Hyyppä, J., Matikainen, L. (2007). Deciduous-coniferous tree classification using difference between first and last pulse laser signatures. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 36*(3/W52).

Lim, K., Treitz, P., Wulder, M., St-Onge, B., Flood, M. (2003). LiDAR remote sensing of forest structure. *Progress in physical geography, 27*(1), 88-106.

Lin, C., Thomson, G., Lo, C.S., Yang, M.S. (2011). A multi-level morphological active contour algorithm for delineating tree crowns in mountainous forest. *Photogrammetric Engineering and Remote Sensing, 77*(3), 241-249.

Lin, Y., Hyyppä, J., Jaakkola, A., Yu, X. (2012). Three-level frame and RD-schematic algorithm for automatic detection of individual trees from MLS point clouds. *International Journal of Remote Sensing, 33*(6), 1701-1716.

Lindberg, E. (2012). *Estimation of canopy structure and individual trees from laser scanning data.* (Doctoral Doctoral thesis), Swedish University of Agricultural Sciences. (8888)

Lovell, J.L., Jupp, D.L.B., Newnham, G.J., Coops, N.C., Culvenor, D.S. (2005). Simulation study for finding optimal lidar acquisition parameters for forest height retrieval. *Forest Ecology and Management, 214*(1-3), 398-412.

Lu, D. (2006). The potential and challenge of remote sensing‐based biomass estimation. *International journal of remote sensing, 27*(7), 1297-1328.

Maclean, G.A., Krabill, W. (1986). Gross-merchantable timber volume estimation using an airborne LIDAR system. *Can J Remote Sens, 12*(1), 7-18.

Mallet, C., Lafarge, F., Roux, M., Soergel, U., Bretar, F., Heipke, C. (2010). A Marked Point Process for Modeling Lidar Waveforms. *Image Processing, IEEE Transactions on, 19*(12), 3204-3221.

Maltamo, M., Malinen, J., Packaln, P., Suvanto, A., Kangas, J. (2006). Nonparametric estimation of stem volume using airborne laser scanning, aerial photography, and stand-register data. *Can J Forest Res, 36*(2), 426-436.

Means, J.E., Acker, S.A., Fitt, B.J., Renslow, M., Emerson, L., Hendrix, C.J. (2000). Predicting forest stand characteristics with airborne scanning lidar. *Photogrammetric Engineering and Remote Sensing, 66*(11), 1367-1371.

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E. (1953). Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics, 21*(6), 1087-1092.

Meyer, F., Beucher, S. (1990). Morphological segmentation. *Journal of visual communication and image representation, 1*(1), 21-46.

Mora, B., Wulder, M.A., White, J.C. (2010). Segment-constrained regression tree estimation of forest stand height from very high spatial resolution panchromatic imagery over a boreal environment. *Remote Sens. Environ., 114*(11), 2474-2484.

Morsdorf, F., Meier, E., Allgöwer, B., Nüesch, D. (2003). Clustering in airborne laser scanning raw data for segmentation of single trees. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 34*(part 3), W13.

Morsdorf, F., Meier, E., Kotz, B., Itten, K.I., Dobbertin, M., Allgower, B. (2004). LIDAR-based geometric reconstruction of boreal type forest stands at single tree level for forest and wildland fire management. *Remote Sens. Environ., 92*(3), 353-362.

Morsdorf, F., Nichol, C., Malthus, T., Woodhouse, I.H. (2009). Assessing forest structural and physiological information content of multi-spectral LiDAR waveforms by radiative transfer modelling. *Remote Sens. Environ., 113*(10), 2152-2163.

Næsset, E. (1997). Determination of mean tree height of forest stands using airborne laser scanner data. *ISPRS Journal of Photogrammetry and Remote Sensing, 52*(2), 49-56.

Naesset, E. (1997a). Determination of mean tree height of forest stands using airborne laser scanner data. *ISPRS Journal of Photogrammetry and Remote Sensing, 52*(2), 49-56.

Naesset, E. (1997b). Estimating timber volume of forest stands using airborne laser scanner data. *Remote Sens. Environ., 61*(2), 246-253.

Naesset, E. (2002). Predicting forest stand characteristics with airborne scanning laser using a practical two-stage procedure and field data. *Remote Sens. Environ., 80*(1), 88-99.

Naesset, E. (2004a). Accuracy of forest inventory using airborne laser scanning: Evaluating the first Nordic full-scale operational project. *Scand J Forest Res, 19*(6), 554-557.

Naesset, E. (2004c). Practical large-scale forest stand inventory using a small-footprint airborne scanning laser. *Scand J Forest Res, 19*(2), 164-179.

NASA. (2015, 2015-03-23). NASA: GLAS. Retrieved March, 23, 2015, from http://icesat.gsfc.nasa.gov/icesat/glas.php

Neal, R.M., Hinton, G.E. (1998). A view of the EM algorithm that justifies incremental, sparse, and other variants, *Learning in graphical models*: Springer, pp. 355-368.

Neeff, T., Biging, G.S., Dutra, L.V., Freitas, C.C., Dos Santos, J.R. (2005). Markov point processes for modeling of spatial forest patterns in Amazonia derived from interferometric height. *Remote Sens. Environ., 97*(4), 484-494.

Nelson, R., Krabill, W., MacLean, G. (1984). Determining forest canopy characteristics using airborne laser data. *Remote Sens. Environ., 15*(3), 201-212.

Nelson, R., Ranson, K.J., Sun, G., Kimes, D.S., Kharuk, V., Montesano, P. (2009). Estimating Siberian timber volume using MODIS and ICESat/GLAS. *Remote Sens. Environ., 113*(3), 691-701.

Nilsson, M. (1996). Estimation of tree heights and stand volume using an airborne lidar system. *Remote Sens. Environ., 56*(1), 1-7.

Nilsson, M. (1997). *Estimation of forest variables using satellite image data and airborne Lidar.* Swedish University of Agricultural Sciences, Umeå, Sweden.

Orka, H.O., Naesset, E., Bollandsas, O.M. (2009). Classifying species of individual trees by intensity and structure features derived from airborne laser scanner data. *Remote Sens. Environ., 113*(6), 1163-1174.

Ortner, M., Descombes, X., Zerubia, J. (2008). A marked point process of rectangles and segments for automatic analysis of digital elevation models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on, 30*(1), 105-119.

Palenichka, R., Doyon, F., Lakhssassi, A., Zaremba, M.B. (2013). Multi-Scale Segmentation of Forest Areas and Tree Detection in LiDAR Images by the Attentive Vision Method. *Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of, 6*(3), 1313-1323.

Parresol, B.R. (1999). Assessing tree and stand biomass: a review with examples and critical comparisons. *Forest Sci, 45*(4), 573-593.

Patenaude, G., Milne, R., Dawson, T.P. (2005). Synthesis of remote sensing approaches for forest carbon estimation: reporting to the Kyoto Protocol. *Environmental Science & Policy, 8*(2), 161-178.

Perrin, G., Descombes, X., Zerubia, J. (2005). *Point processes in forestry: an application to tree crown detection*. INRIA.

Perrin, G., Descombes, X., Zerubia, J. (2006). *A non-Bayesian model for tree crown extraction using marked point processes*.

Petrie, G., Toth, C.K. (2009). Introduction to Laser Ranging, Profiling, and Scanning, in: Shan, J., Toth, C.K. (Ed.), *Topographic Laser Ranging and Scanning: Principles and Processing*. Boca Raton, FL: CRC Press/Taylor & Francis Group.

Pfeifer, N., Winterhalder, D. (2004). Modelling of tree cross sections from terrestrial laser scanning data with free-form curves. *International Archives of Photogrammetry, remote sensing and spatial information sciences, 36*(Part 8), W2.

Pollock, R. (1996). *The automatic recognition of individual trees in aerial images of forests based on a synthetic tree crown image model.* (PhD), University of British Columbia. (148157)

Popescu, S.C., Wynne, R.H., Nelson, R.F. (2002). Estimating plot-level tree heights with lidar: local filtering with a canopy-height based variable window size. *Comput Electron Agr, 37*(1-3), 71-95.

Pouliot, D.A., King, D.J., Bell, F.W., Pitt, D.G. (2002). Automated tree crown detection and delineation in high-resolution digital camera imagery of coniferous forest regeneration. *Remote Sens. Environ., 82*(2–3), 322-334.

Pyysalo, U., Hyyppa, H. (2002). Reconstructing tree crowns from laser scanner data for feature extraction. *International Archieves of Photogrammetry and Remote Sensing and spatial information sciences, 34*(3/B), 218-221.

Reitberger, J., Schnörr, C., Krzystek, P., Stilla, U. (2009). 3D segmentation of single trees exploiting full waveform LIDAR data. *ISPRS Journal of Photogrammetry and Remote Sensing, 64*(6), 561-574.

Rhody, B.F. (1965). Photointerpretation and mapping for forestry purposes. *Unasylva, 19*(2).

Rosette, J.A.B., North, P.R.J., Suarez, J.C. (2008). Vegetation height estimates for a mixed temperate forest using satellite laser altimetry. *International Journal of Remote Sensing, 29*(5), 1475-1493.

Salamon, P., Sibani, P., Frost, R. (2002a). *Facts, conjectures, and improvements for simulated annealing*. Society for Industrial and Applied Mathematics, Philadelphia, PA: SIAM.

Salamon, P., Sibani, P., Frost, R. (2002b). *Facts, Conjectures, and Improvements for Simulated Annealing*: Society for Industrial and Applied Mathematics.

Serra, J. (1982). *Image Analysis and Mathematical Morphology*. London: Academic Press.

Soille, P. (2003). *Morphological Image Analysis: Principles and Applications*, 2 ed. Berlin, Germany: Springer-Verlag.

Solberg, S., Naesset, E., Bollandsas, O.M. (2006). Single tree segmentation using airborne laser scanner data in a structurally heterogeneous spruce forest. *Photogrammetric Engineering and Remote Sensing, 72*(12), 1369-1378.

Solodukhin, V., Zukov, A., Mazugin, I. (1977). Possibilities of laser aerial photography for forest profiling. *Lesnoe Khozyaisto (Forest Management), 10*, 53-58.

Sonka, M., Hlavac, V., Boyle, R. (2008). *Image processing, analysis, and machine vision*, 3rd ed. Toronto: Thompson Learning.

Stilla, U., Jutzi, B. (2008). Waveform analysis for small-footprint pulsed laser systems. *Topographic laser ranging and scanning: Principles and processing*, 215-234.

Stoyan, D., Penttinen, A. (2000). Recent applications of point process methods in forestry statistics. *Statistical Science*, 61-78.

Suratno, A., Seielstad, C., Queen, L. (2009). Tree species identification in mixed coniferous forest using airborne laser scanning. *ISPRS Journal of Photogrammetry and Remote Sensing, 64*(6), 683-693.

Tarabalka, Y., Chanussot, J., Benediktsson, J.A. (2010). Segmentation and classification of hyperspectral images using watershed transformation. *Pattern Recognition, 43*(7), 2367-2379.

Tomppo, E., Nilsson, M., Rosengren, M., Aalto, P., Kennedy, P. (2002). Simultaneous use of Landsat-TM and IRS-1C WiFS data in estimating large area tree stem volume and aboveground biomass. *Remote Sens. Environ., 82*(1), 156-171.

Tournaire, O., Brédif, M., Boldo, D., Durupt, M. (2010). An efficient stochastic approach for building footprint extraction from digital elevation models. *ISPRS Journal of Photogrammetry and Remote Sensing, 65*(4), 317-327.

Tournaire, O., Paparoditis, N. (2009). A geometric stochastic approach based on marked point processes for road mark detection from high resolution aerial images. *ISPRS Journal of Photogrammetry and Remote Sensing, 64*(6), 621-631.

Van Lieshout, M.N.M. (2000). *Markov point process and their applications*. London: Imperial College Press.

Van Lieshout, M.N.M. (2009). Applications of Stochastic Geometry in Image Analysis, in: Kendall, W.S., Molchanov, I. (Eds.), *New Perspectives in Stochastic Geometry*: Oxford University Press, pp. 427-450.

Vastaranta, M. (2012). *Forest mapping and monitoring using active 3D remote sensing.* (Ph.D Doctoral dissertation (article-based)), University of Helsinki, Helsinki, Finland.

Vastaranta, M., Holopainen, M., Yu, X., Haapanen, R., Melkas, T., Hyyppä, J., Hyyppä, H. (2011a). Individual tree detection and area-based approach in retrieval of forest inventory characteristics from low-pulse airborne laser scanning data. *Photogram. J. Fin, 22*, 1-13.

Vastaranta, M., Holopainen, M., Yu, X.W., Hyyppa, J., Makinen, A., Rasinmaki, J., Melkas, T., Kaartinen, H., Hyyppa, H. (2011b). Effects of Individual Tree Detection Error Sources on Forest Management Planning Calculations. *Remote Sensing, 3*(8), 1614-1626.

Vastaranta, M., Kankare, V., Holopainen, M., Yu, X., Hyyppä, J., Hyyppä, H. (2012). Combination of individual tree detection and area-based approach in imputation of forest variables using airborne laser data. *ISPRS Journal of Photogrammetry and Remote Sensing, 67*(1), 73-79.

Vauhkonen, J., Ene, L., Gupta, S., Heinzel, J., Holmgren, J., Pitkanen, J., Solberg, S., Wang, Y.S., Weinacker, H., Hauglin, K.M., Lien, V., Packalen, P., Gobakken, T., Koch, B., Naesset, E., Tokola, T., Maltamo, M. (2012). Comparative testing of single-tree detection algorithms under different types of forest. *Forestry, 85*(1), 27-40.

Véga, C., Durrieu, S. (2011). Multi-level filtering segmentation to measure individual tree parameters based on Lidar data: Application to a mountainous forest with heterogeneous stands. *International Journal of Applied Earth Observation and Geoinformation, 13*(4), 646-656.

Vosselman, G., Maas, H.-g. (2001). *Adjustment and filtering of raw laser altimetry data.* Paper presented at the Proceedings of OEEPE Workshop on Airborne Laserscanning and Interferometric SAR for Detailed Digital Terrain Models, Stockholm, Sweden.

Wagner, W. (2010). Radiometric calibration of small-footprint full-waveform airborne laser scanner measurements: Basic physical concepts. *ISPRS Journal of Photogrammetry and Remote Sensing, 65*(6), 505-513.

Wagner, W., Ullrich, A., Ducic, V., Melzer, T., Studnicka, N. (2006). Gaussian decomposition and calibration of a novel small-footprint full-waveform digitising airborne laser scanner. *ISPRS Journal of Photogrammetry and Remote Sensing, 60*(2), 100-112.

Wang, L., Gong, P., Biging, G.S. (2004). Individual tree-crown delineation and treetop detection in high-spatial-resolution aerial imagery. *Photogrammetric Engineering and Remote Sensing, 70*(3), 351-358.

Wang, Y., Weinacker, H., Koch, B., Sterenczak, K. (2008). Lidar point cloud based fully automatic 3D single tree modelling in forest and evaluations of the procedure. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 37*(Part B6b), 45-51.

Wehr, A., Lohr, U. (1999). Airborne laser scanning - an introduction and overview. *ISPRS Journal of Photogrammetry and Remote Sensing, 54*(2-3), 68-82.

White, J., Wulder, M., Vastaranta, M., Coops, N., Pitt, D., Woods, M. (2013a). The Utility of Image-Based Point Clouds for Forest Inventory: A Comparison with Airborne Laser Scanning. *Forests, 4*(3), 518-536.

White, J.C., Wulder, M.A., Varhola, A., Vastaranta, M., Coops, N.C., Cook, B.D., Pitt, D., Woods, M. (2013b). *A best practices guide for generating forest inventory attributes from airborne laser scanning data using an area-based approach*, p. 50.

White, S.R. (1984). *Concepts of Scale in Simulated Annealing*. Paper presented at the the 1984 International Conference on Computer Design.

Wikipedia. (2015, April 11, 2015). Symmetric difference. Retrieved May 11, 2015, from http://en.wikipedia.org/wiki/Symmetric_difference

Wulder, M. (1998). Optical remote-sensing techniques for the assessment of forest inventory and biophysical parameters. *Progress in physical geography, 22*(4), 449-476.

Wulder, M., Niemann, K.O., Goodenough, D.G. (2000). Local maximum filtering for the extraction of tree locations and basal area from high spatial resolution imagery. *Remote Sens. Environ., 73*(1), 103-114.

Xie, Y., Sha, Z., Yu, M. (2008). Remote sensing imagery in vegetation mapping: a review. *Journal of Plant Ecology, 1*(1), 9-23.

Xing, Y.Q., de Gier, A., Zhang, J.J., Wang, L.H. (2010). An improved method for estimating forest canopy height using ICESat-GLAS full waveform data over sloping terrain: A case study in Changbai mountains, China. *International Journal of Applied Earth Observation and Geoinformation, 12*(5), 385-392.

Yao, W., Krzystek, P., Heurich, M. (2012). Tree species classification and estimation of stem volume and DBH based on single tree extraction by exploiting airborne full-waveform LiDAR data. *Remote Sens. Environ., 123,* 368-380.

Yu, X., Hyyppä, J., Vastaranta, M., Holopainen, M., Viitala, R. (2011). Predicting individual tree attributes from airborne laser point clouds based on the random forests technique. *ISPRS Journal of Photogrammetry and Remote Sensing, 66*(1), 28-37.

Yu, X.W., Hyyppa, J., Holopainen, M., Vastaranta, M. (2010). Comparison of Area-Based and Individual Tree-Based Methods for Predicting Plot-Level Forest Attributes. *Remote Sensing, 2*(6), 1481-1495.

Yu, X.W., Hyyppa, J., Kaartinen, H., Maltamo, M. (2004). Automatic detection of harvested trees and determination of forest growth using airborne laser scanning. *Remote Sens. Environ., 90*(4), 451-462.

Zhang, J., de Gier, A., Xing, Y., Sohn, G. (2011). Full waveform-based analysis for forest type information derivation from large footprint spaceborne lidar data. *Photogrammetric Engineering and Remote Sensing, 77*(3), 281-290.

Zhao, K.G., Popescu, S., Nelson, R. (2009). Lidar remote sensing of forest biomass: A scale-invariant estimation approach using airborne lasers. *Remote Sens. Environ., 113*(1), 182-196.

Zwally, H., Schutz, B., Abdalati, W., Abshire, J., Bentley, C., Brenner, A., Bufton, J., Dezio, J., Hancock, D., Harding, D. (2002). ICESat's laser measurements of polar ice, atmosphere, ocean, and land. *Journal of Geodynamics, 34*(3), 405-445.

Appendix A
Expectation-Maximization Based Parameter Estimation Results

Table 7.1. EM-based parameter estimation and tree detection results with different initial parameter settings for separate plot.

Plots	Parameter Settings	Iterations	Parameter Estimation						Detection Results								
			Asymmetric		Area Ratio		Overlap		Reference	Detected	Correct		Commission		Omission		Overall
			μ'	λ'	μ'	λ'	μ'	λ'	Trees	Trees	No.	%	No.	%	No.	%	Accuracy
Separate	ES	initial	0.20	0.03	0.90	-0.02	0.05	0.01	186	179	177	98.9%	2	1.1%	9	4.8%	94.15%
		iter1	0.33	0.05	0.76	-0.02	0.76	0.0008	186	179	177	98.9%	2	1.1%	9	4.8%	94.15%
		iter2	0.33	0.05	0.76	-0.02	0.78	0.0009	186	179	177	98.9%	2	1.1%	9	4.8%	94.15%
		iter3	0.33	0.05	0.76	-0.02	0.78	0	186	179	177	98.9%	2	1.1%	9	4.8%	94.15%
	S	initial	0.30	0.05	0.80	-0.03	0.10	0.01	186	182	181	99.5%	1	0.5%	5	2.7%	96.79%
		iter1	0.34	0.06	0.77	-0.02	0.07	0.009	186	182	181	99.5%	1	0.5%	5	2.7%	96.79%
		iter2	0.34	0.06	0.77	-0.02	0.07	0.009	186	182	181	99.5%	1	0.5%	5	2.7%	96.79%
		iter3	/	/	/	/	/	/									
	T	initial	0.40	0.10	0.70	-0.07	0.35	0.03	186	186	182	97.8%	4	2.2%	4	2.2%	95.79%
		iter1	0.33	0.06	0.77	-0.02	0.13	0.013	186	182	181	99.5%	1	0.5%	5	2.7%	96.79%
		iter2	0.33	0.06	0.77	-0.02	0.07	0.008	186	182	181	99.5%	1	0.5%	5	2.7%	96.79%
		iter3	0.33	0.06	0.77	-0.02	0.07	0.008	186	182	181	99.5%	1	0.5%	5	2.7%	96.79%
	O	initial	0.50	0.15	0.60	-0.12	0.60	0.05	186	186	183	98.4%	3	1.6%	3	1.6%	96.83%
		iter1	0.33	0.05	0.79	-0.02	0.13	0.013	186	182	182	100.0%	0	0.0%	4	2.2%	97.85%
		iter2	0.33	0.06	0.78	-0.02	0.07	0.008	186	182	182	100.0%	0	0.0%	4	2.2%	97.85%
		iter3	0.33	0.06	0.78	-0.02	0.07	0.008	186	182	182	100.0%	0	0.0%	4	2.2%	97.85%
	EO	initial	0.60	0.20	0.50	-0.15	0.70	0.05	186	186	183	98.4%	3	1.6%	3	1.6%	96.83%
		iter1	0.33	0.05	0.79	-0.02	0.13	0.013	186	182	182	100.0%	0	0.0%	4	2.2%	97.85%
		iter2	0.33	0.06	0.78	-0.02	0.07	0.008	186	182	182	100.0%	0	0.0%	4	2.2%	97.85%
		iter3	0.33	0.06	0.78	-0.02	0.07	0.008	186	182	182	100.0%	0	0.0%	4	2.2%	97.85%

Table 7.2. EM-based parameter estimation and tree detection results with different initial parameter settings for touch plot.

Plots	Parameter Settings	Iterations	Parameter Estimation						Detection Results								
			Asymmetric		Area Ratio		Overlap		Reference	Detected	Correct		Commission		Omission		Overall
			μ'	λ'	μ'	λ'	μ'	λ'	Trees	Trees	No.	%	No.	%	No.	%	Accuracy
Touch	ES	initial	0.20	0.03	0.90	-0.02	0.05	0.01	234	158	155	98.1%	3	1.9%	79	33.8%	65.40%
		iter1	0.40	0.07	0.72	-0.04	0.08	0.009	234	152	150	98.7%	2	1.3%	84	35.9%	63.56%
		iter2	0.39	0.06	0.73	-0.03	0.09	0.010	234	152	150	98.7%	2	1.3%	84	35.9%	63.56%
		iter3	/	/	/	/	/	/									
	S	initial	0.30	0.05	0.80	-0.03	0.10	0.01	234	206	206	100.0%	0	0.0%	28	12.0%	88.03%
		iter1	0.38	0.07	0.74	-0.03	0.22	0.016	234	206	206	100.0%	0	0.0%	28	12.0%	88.03%
		iter2	0.38	0.07	0.74	-0.03	0.22	0.015	234	206	206	100.0%	0	0.0%	28	12.0%	88.03%
		iter3	/	/	/	/	/	/									
	T	initial	0.40	0.10	0.70	-0.07	0.35	0.03	234	216	215	99.5%	1	0.5%	19	8.1%	91.49%
		iter1	0.38	0.07	0.75	-0.04	0.32	0.019	234	216	215	99.5%	1	0.5%	19	8.1%	91.49%
		iter2	0.38	0.07	0.75	-0.04	0.32	0.019	234	216	215	99.5%	1	0.5%	19	8.1%	91.49%
		iter3	/	/	/	/	/	/									
	O	initial	0.50	0.15	0.60	-0.12	0.60	0.05	234	217	216	99.5%	1	0.5%	18	7.7%	91.91%
		iter1	0.38	0.07	0.75	-0.04	0.34	0.019	234	216	215	99.5%	1	0.5%	19	8.1%	91.49%
		iter2	0.38	0.07	0.75	-0.04	0.32	0.018	234	216	215	99.5%	1	0.5%	19	8.1%	91.49%
		iter3	0.38	0.07	0.75	-0.04	0.32	0.018	234	216	215	99.5%	1	0.5%	19	8.1%	91.49%
	EO	initial	0.60	0.20	0.50	-0.15	0.70	0.05	234	217	216	99.5%	1	0.5%	18	7.7%	91.91%
		iter1	0.38	0.07	0.75	-0.04	0.34	0.019	234	216	215	99.5%	1	0.5%	19	8.1%	91.49%
		iter2	0.38	0.07	0.75	-0.04	0.32	0.018	234	216	215	99.5%	1	0.5%	19	8.1%	91.49%
		iter3	0.38	0.07	0.75	-0.04	0.32	0.018	234	216	215	99.5%	1	0.5%	19	8.1%	91.49%

Table 7.3. EM-based parameter estimation and tree detection results with different initial parameter settings for overlap plot.

Plots	Parameter Settings	Iterations	Parameter Estimation						Detection Results								
			Asymmetric		Area Ratio		Overlap		Reference	Detected	Correct		Commission		Omission		Overall
			μ'	λ'	μ'	λ'	μ'	λ'	Trees	Trees	No.	%	No.	%	No.	%	Accuracy
Overlap	ES	initial	0.20	0.03	0.90	-0.02	0.05	0.01	261	132	130	98.5%	2	1.5%	131	50.2%	49.43%
		iter1	0.48	0.13	0.66	-0.10	0.07	0.009	261	132	130	98.5%	2	1.5%	131	50.2%	49.43%
		iter2	0.48	0.13	0.66	-0.10	0.07	0.009	261	132	130	98.5%	2	1.5%	131	50.2%	49.43%
		iter3	/	/	/	/	/	/									
	S	initial	0.30	0.05	0.80	-0.03	0.10	0.01	261	188	185	98.4%	3	1.6%	76	29.1%	70.08%
		iter1	0.42	0.09	0.71	-0.07	0.19	0.028	261	185	183	98.9%	2	1.1%	78	29.9%	69.58%
		iter2	0.41	0.09	0.71	-0.06	0.71	0.209	261	182	180	98.9%	2	1.1%	81	31.0%	68.44%
		iter3	0.41	0.09	0.72	-0.06	0.71	0.211	261	182	180	98.9%	2	1.1%	81	31.0%	68.44%
	T	initial	0.40	0.10	0.70	-0.07	0.35	0.03	261	221	221	100.0%	0	0.0%	40	15.3%	84.67%
		iter1	0.41	0.09	0.72	-0.06	0.38	0.043	261	221	221	100.0%	0	0.0%	40	15.3%	84.67%
		iter2	0.41	0.09	0.72	-0.06	0.38	0.043	261	221	221	100.0%	0	0.0%	40	15.3%	84.67%
		iter3	/	/	/	/	/	/									
	O	initial	0.50	0.15	0.60	-0.12	0.60	0.05	261	224	224	100.0%	0	0.0%	37	14.2%	85.82%
		iter1	0.41	0.10	0.72	-0.06	0.45	0.035	261	222	222	100.0%	0	0.0%	39	14.9%	85.06%
		iter2	0.41	0.09	0.72	-0.06	0.37	0.044	261	221	221	100.0%	0	0.0%	40	15.3%	84.67%
		iter3	0.41	0.09	0.72	-0.06	0.37	0.043	261	221	221	100.0%	0	0.0%	40	15.3%	84.67%
	EO	initial	0.60	0.20	0.50	-0.15	0.70	0.05	261	224	224	100.0%	0	0.0%	37	14.2%	85.82%
		iter1	0.41	0.10	0.72	-0.06	0.45	0.035	261	222	222	100.0%	0	0.0%	39	14.9%	85.06%
		iter2	0.41	0.09	0.72	-0.06	0.37	0.044	261	221	221	100.0%	0	0.0%	40	15.3%	84.67%
		iter3	0.41	0.09	0.72	-0.06	0.37	0.043	261	221	221	100.0%	0	0.0%	40	15.3%	84.67%

Table 7.4. EM-based parameter estimation and tree detection results with different initial parameter settings for plot 1.

Plots	Parameter Settings	Iterations	Parameter Estimation						Detection Results								
			Asymmetric		Area Ratio		Overlap		Reference	Detected	Correct		Commission		Omission		Overall
			μ'	λ'	μ'	λ'	μ'	λ'	Trees	Trees	No.	%	No.	%	No.	%	Accuracy
Plot 1	ES	initial	0.20	0.03	0.90	-0.02	0.05	0.01	120	87	75	86.2%	12	13.8%	45	37.5%	56.82%
		iter1	0.50	0.16	0.62	-0.11	0.13	0.015	120	86	74	86.0%	12	14.0%	46	38.3%	56.06%
		iter2	0.49	0.16	0.61	-0.10	0.04	0.006	120	83	73	88.0%	10	12.0%	47	39.2%	56.15%
		iter3	0.49	0.15	0.61	-0.09	0.03	0.007	120	83	73	88.0%	10	12.0%	47	39.2%	56.15%
	S	initial	0.30	0.05	0.80	-0.03	0.10	0.01	120	109	97	89.0%	12	11.0%	23	19.2%	73.48%
		iter1	0.45	0.11	0.65	-0.06	0.57	0.007	120	109	97	89.0%	12	11.0%	23	19.2%	73.48%
		iter2	0.45	0.11	0.65	-0.06	0.58	0.008	120	109	97	89.0%	12	11.0%	23	19.2%	73.48%
		iter3	/	/	/	/	/	/									
	T	initial	0.40	0.10	0.70	-0.07	0.35	0.03	120	127	114	89.8%	13	10.2%	6	5.0%	85.71%
		iter1	0.44	0.11	0.67	-0.06	0.47	0.012	120	126	113	89.7%	13	10.3%	7	5.8%	84.96%
		iter2	0.44	0.11	0.67	-0.06	0.36	0.020	120	126	113	89.7%	13	10.3%	7	5.8%	84.96%
		iter3	0.44	0.11	0.67	-0.06	0.35	0.022	120	126	113	89.7%	13	10.3%	7	5.8%	84.96%
	O	initial	0.50	0.15	0.60	-0.12	0.60	0.05	120	129	116	89.9%	13	10.1%	4	3.3%	87.22%
		iter1	0.44	0.12	0.67	-0.07	0.31	0.037	120	124	112	90.3%	12	9.7%	8	6.7%	84.85%
		iter2	0.43	0.11	0.68	-0.06	0.36	0.022	120	124	112	90.3%	12	9.7%	8	6.7%	84.85%
		iter3	0.43	0.11	0.68	-0.06	0.35	0.023	120	124	112	90.3%	12	9.7%	8	6.7%	84.85%
	EO	initial	0.60	0.20	0.50	-0.15	0.70	0.05	120	129	116	89.9%	13	10.1%	4	3.3%	87.22%
		iter1	0.46	0.13	0.67	-0.07	0.47	0.023	120	124	112	90.3%	12	9.7%	8	6.7%	84.85%
		iter2	0.44	0.12	0.67	-0.07	0.43	0.020	120	124	112	90.3%	12	9.7%	8	6.7%	84.85%
		iter3	0.44	0.12	0.68	-0.07	0.43	0.020	120	124	112	90.3%	12	9.7%	8	6.7%	84.85%

Table 7.5 EM-based parameter estimation and tree detection results with different initial parameter settings for plot 2.

Plots	Parameter Settings	Iterations	Parameter Estimation						Detection Results								
			Asymmetric		Area Ratio		Overlap		Reference	Detected	Correct		Commission		Omission		Overall
			μ'	λ'	μ'	λ'	μ'	λ'	Trees	Trees	No.	%	No.	%	No.	%	Accuracy
Plot 2	ES	initial	0.20	0.03	0.90	-0.02	0.05	0.01	40	28	28	100.0%	0	0.0%	12	30.0%	70.00%
		iter1	0.55	0.17	0.59	-0.08	0.02	0.002	40	28	28	100.0%	0	0.0%	12	30.0%	70.00%
		iter2	0.55	0.17	0.59	-0.08	0.02	0.002	40	28	28	100.0%	0	0.0%	12	30.0%	70.00%
		iter3	/	/	/	/	/	/									
	S	initial	0.30	0.05	0.80	-0.03	0.10	0.01	40	38	35	92.1%	3	7.9%	5	12.5%	81.40%
		iter1	0.41	0.11	0.70	-0.05	0.20	0.050	40	37	35	94.6%	2	5.4%	5	12.5%	83.33%
		iter2	0.40	0.10	0.70	-0.05	0.20	0.043	40	37	35	94.6%	2	5.4%	5	12.5%	83.33%
		iter3	/	/	/	/	/	/									
	T	initial	0.40	0.10	0.70	-0.07	0.35	0.03	40	41	37	90.2%	4	9.8%	3	7.5%	84.09%
		iter1	0.42	0.13	0.69	-0.07	0.25	0.077	40	40	37	92.5%	3	7.5%	3	7.5%	86.05%
		iter2	0.41	0.11	0.69	-0.06	0.21	0.017	40	40	37	92.5%	3	7.5%	3	7.5%	86.05%
		iter3	0.41	0.11	0.69	-0.06	0.21	0.02	40	40	37	92.5%	3	7.5%	3	7.5%	86.05%
	O	initial	0.50	0.15	0.60	-0.12	0.60	0.05	40	41	37	90.2%	4	9.8%	3	7.5%	84.09%
		iter1	0.42	0.13	0.69	-0.07	0.25	0.077	40	40	37	92.5%	3	7.5%	3	7.5%	86.05%
		iter2	0.41	0.11	0.69	-0.06	0.21	0.017	40	40	37	92.5%	3	7.5%	3	7.5%	86.05%
		iter3	0.41	0.11	0.69	-0.06	0.21	0.020	40	40	37	92.5%	3	7.5%	3	7.5%	86.05%
	EO	initial	0.60	0.20	0.50	-0.15	0.70	0.05	40	41	36	87.8%	5	12.2%	4	10.0%	80.00%
		iter1	0.41	0.12	0.69	-0.07	0.25	0.121	40	38	35	92.1%	3	7.9%	5	12.5%	81.40%
		iter2	0.40	0.10	0.71	-0.05	0.22	0.017	40	38	35	92.1%	3	7.9%	5	12.5%	81.40%
		iter3	0.40	0.10	0.70	-0.05	0.22	0.020	40	38	35	92.1%	3	7.9%	5	12.5%	81.40%

Table 7.6. EM-based parameter estimation and tree detection results with different initial parameter settings for plot 3.

Plots	Parameter Settings	Iterations	Parameter Estimation						Detection Results								
			Asymmetric		Area Ratio		Overlap		Reference	Detected	Correct		Commission		Omission		Overall
			μ'	λ'	μ'	λ'	μ'	λ'	Trees	Trees	No.	%	No.	%	No.	%	Accuracy
Plot 3	ES	initial	0.20	0.03	0.90	-0.02	0.05	0.01	122	82	73	89.0%	9	11.0%	49	40.2%	55.73%
		iter1	0.56	0.19	0.58	-0.11	0.06	0.005	122	81	72	88.9%	9	11.1%	50	41.0%	54.96%
		iter2	0.56	0.19	0.58	-0.10	0.06	0.006	122	81	72	88.9%	9	11.1%	50	41.0%	54.96%
		iter3	0.56	0.18	0.58	-0.10	0.06	0.01	122	81	72	88.9%	9	11.1%	50	41.0%	54.96%
	S	initial	0.30	0.05	0.80	-0.03	0.10	0.01	122	108	97	89.8%	11	10.2%	25	20.5%	72.93%
		iter1	0.51	0.13	0.63	-0.08	0.62	0.005	122	108	97	89.8%	11	10.2%	25	20.5%	72.93%
		iter2	0.51	0.13	0.63	-0.08	0.62	0.003	122	108	97	89.8%	11	10.2%	25	20.5%	72.93%
		iter3	/	/	/	/	/	/									
	T	initial	0.40	0.10	0.70	-0.07	0.35	0.03	122	122	111	91.0%	11	9.0%	11	9.0%	83.46%
		iter1	0.49	0.13	0.66	-0.08	0.34	0.100	122	119	108	90.8%	11	9.2%	14	11.5%	81.20%
		iter2	0.49	0.12	0.66	-0.07	0.31	0.034	122	119	108	90.8%	11	9.2%	14	11.5%	81.20%
		iter3	0.49	0.12	0.66	-0.07	0.32	0.032	122	119	108	90.8%	11	9.2%	14	11.5%	81.20%
	O	initial	0.50	0.15	0.60	-0.12	0.60	0.05	122	123	112	91.1%	11	8.9%	10	8.2%	84.21%
		iter1	0.49	0.13	0.66	-0.08	0.30	0.029	122	121	110	90.9%	11	9.1%	12	9.8%	82.71%
		iter2	0.49	0.13	0.66	-0.08	0.30	0.043	122	120	109	90.8%	11	9.2%	13	10.7%	81.95%
		iter3	0.49	0.12	0.66	-0.08	0.29	0.040	122	120	109	90.8%	11	9.2%	13	10.7%	81.95%
	EO	initial	0.60	0.20	0.50	-0.15	0.70	0.05	122	124	113	91.1%	11	8.9%	9	7.4%	84.96%
		iter1	0.49	0.14	0.66	-0.09	0.34	0.048	122	121	110	90.9%	11	9.1%	12	9.8%	82.71%
		iter2	0.49	0.13	0.66	-0.08	0.3	0.042	122	120	109	90.8%	11	9.2%	13	10.7%	81.95%
		iter3	0.49	0.13	0.66	-0.08	0.29	0.040	122	120	109	90.8%	11	9.2%	13	10.7%	81.95%