3D SEMANTIC LABELLING OF URBAN LIDAR POINT CLOUD AND MULTISPECTRAL DATA

A thesis submitted
in partial fulfillment for the degree of

Doctor of Philosophy

by

A. M. RAMIYA

Department of Earth and Space Sciences

INDIAN INSTITUTE OF SPACE SCIENCE AND TECHNOLOGY
Thiruvananthapuram - 695547

December 2016

CERTIFICATE

This is to certify that the thesis titled **3D SEMANTIC LABELLING OF URBAN LIDAR POINT CLOUD AND MULTISPECTRAL DATA** submitted by **Ms. A. M. Ramiya** to the Indian Institute of Space Science and Technology, Thiruvananthapuram, for the award of the degree of **Doctor of Philosophy**, is a bona fide record of the research work done by her under our supervision. The contents of this thesis, in full, or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. Rama Rao Nidamanuri
Supervisor

Dr. R. Krishnan
Co-Supervisor

Place: Thiruvananthapuram
December 2016

Counter signature of HoD with seal.

DECLARATION

I declare that this thesis titled **3D SEMANTIC LABELLING OF URBAN LIDAR POINT CLOUD AND MULTISPECTRAL DATA** submitted in fulfillment of the Degree of Doctor of Philosophy is a record of original work carried out by me under the supervision of **Dr. Rama Rao Nidamanuri** and **Dr. R. Krishnan**, and has not formed the basis for the award of any degree, diploma, associateship, fellowship or other titles in this or any other Institution or University of higher learning. In keeping with the ethical practice in reporting scientific information, due acknowledgements have been made wherever the findings of others have been cited.

Ms. A. M. Ramiya
SC11D030

Place: Thiruvananthapuram
December 2016

ACKNOWLEDGEMENT

My deep gratitude and thanks to my supervisor, Dr. Rama Rao Nidamanuri, for his continuous support throughout my PhD research. He has patiently helped me in steering the research to the end with useful discussions, practical working, and continuous feedback. He has taught me both consciously and unconsciously how to be an independent researcher.

My sincere thanks to my co-supervisor, Dr. R. Krishnan, for his support during my entire research period. His insightful discussions was really useful in shaping up this thesis. I thank all the members of my doctoral committee - Dr. A. Senthilkumar (Director, IIRS Dehradun), Dr. K. R. Ramakrishnan (Professor, IISc Bangalore), Dr. S. Ramakrishnan (Scientist, SAC, Ahmedabad), Dr. B. Krishnamohan (Professor, IIT Bombay), Dr. N. Padhmanabhan (Scientist, SAC, Ahmedabad), Dr. Deepak Mishra(Assoc. Professor, IIST), Dr. P. S. Roy (Scientist, NRSC, Hyderabad), for their constructive suggestions which made this thesis better. They generously gave valuable comments which helped in structuring and improving my work.

My sincere thanks to Dr. K. S. Dasgupta, Former Director, IIST, for his immense support for this research. My sincere thanks to Dr. A. Chandrasekar, Registrar and Dean(Academics), IIST, for his support and motivation throughout my professional career. I also thank Dr. Anandmayee Tej, HoD, Earth and Space Sciences, for her support and encouragement during my research period. My heartfelt thanks to Dr. V.K. Dadhwal, Director, IIST, for his support.

My heartfelt thanks to Dr. L. Gnanappazham (Assoc. Professor, IIST), for the discussions and encouragement throughout this research period. I am indebted to many of my colleagues and friends in the institute for their encouragement during the entire course of my research work.

I also thank Dr. Bharat Lohani (Professor, IIT Kanpur) for his support and motivation during my visit to IIT Kanpur at the initial stages of my research.

I also thank Prof. Arthur P. Cracknell (Professor, University of Dundee, UK) and Prof

Manoj K Arora(Director, PEC University of Technology, Chandigarh) for acting as examiners for my thesis and for their constructive comments which helped improve my thesis.

My special thanks goes to all the research scholars in the remote sensing lab especially Ms. Dhanya, Mr. Arun, Mr. Gopakumar, Mr. Bharat, for the discussions and memorable times in the lab. I also thank Ms. Divya for her support in the remote sensing lab.

A special thanks to all my family members for always being a pillar of support without whom I could not have reached so far. I always consider myself lucky to have wonderful parents and role-models, my appa and amma (Dr. C. R. Anandakumar, and Mrs. Anu Anand), who shaped me up both in my professional and personal life. I am always indebted to them for their constant motivation and unconditional support in all possible ways. I am so grateful to my inlaws, my mama and athai (Mr. Sukumara Pillai and Mrs. K. Meena), for their immense support and encouragement in all stages of my life. And a big thanks to my sister, Ms. A. Dhanya, for cheering me up in all situations. I also want to thank my grandparents (Mr. Baghavathi Perumal, Mrs. Kamala and Mrs. Pankajam Raman) for their love and encouragement. All of their prayers are what sustained me thus far. I also thank my extended family members, my co-sisters, brothers-in-law, and the kids for their support and love.

This thesis would not have been possible without the encouragement, support, and love from my dear husband Mr. S. Subash, who has always been there with me during this bumpy research journey. A big thanks to my daughter Ms. Advika Subash for her little prayers and for cheering me up by being such a good girl.

ABSTRACT

LiDAR (Light Detection And Ranging) is one of the most unprecedented geospatial technologies introduced in the last decade for direct and accurate measurement of elevation of a point on the earth surface. The geocoded LiDAR data enable extraction of various 3D urban objects true to geometrical measurements which can be directly used for creating 3D models. However, due to the unstructured nature of the data and its high volume, localizing and extracting various urban objects in 3D perspective from the LiDAR data, generally known as point cloud, remains a challenge. Semantic labelling establishes geometrical-structural relationship in the point cloud and is very critical for exploiting the rich point cloud for various applications such as infrastructure mapping, disaster management, virtual reality, utility management.

The overall aim of this thesis is to develop an efficient and reliable algorithmic framework for semantically labelling 3D coloured LiDAR point cloud (point cloud with spectral data integrated) acquired over an urban environment using computer vision techniques in an open source prototype system.

Within this overall aim, the objective of this thesis is two fold. The first objective is to develop a novel 3D object-based framework for semantically labelling the 3D coloured LiDAR point cloud obtained by integrating LiDAR point cloud and multispectral imagery. Segmentation is an important stage in object-based labelling framework. In this thesis, initially color-based region growing segmentation algorithm has been used to create 3D segments. Further, to improve the efficiency of the algorithm while processing highly dense point cloud, a computationally efficient supervoxels-based LCCP (Local Cloud Connectivity Patches) segmentation approach has been adapted and extended for creating meaningful segments from the point cloud. The segments are classified using various machine learning techniques into multiple urban classes based on the spectral and geometric features extracted from the segments.

The methodological framework developed has been implemented on different airborne Li-

DAR and multispectral images captured at multiple sites assessing the generalization capability of the methodological framework developed. The validity and the accuracy of the labelled point cloud has been independently validated against the ground truth by International Society of Photogrammetry and Remote Sensing (ISPRS). Results appear promising and the labelled 3D points can be directly used for 3D surface reconstruction of various man-made and natural urban objects.

The second objective is to critically assess the role of spectral and geometrical information in the various stages of object-based point cloud labelling, namely, segmentation, feature extraction, and classification. Results indicate that both the spectral, and geometrical information is critical for semantic labelling of low point density LiDAR point cloud (point density < 5 points/m^2), whereas the geometrical information alone is sufficient for LiDAR point cloud with higher point density. It is also evident that the supervoxels-based LCCP segmentation is computationally efficient and offer superior labelling accuracy while processing high point density LiDAR point cloud.

The methodological studies carried out in this thesis make a significant contribution to the current suite of approaches used for 3D semantic labelling of LiDAR point cloud. As the methodologies presented in this work is an object-based approach with further flexibility of its implementation on an open source platform, it has promising application in labelling high density point cloud. The labelled point cloud will be an asset to the data users and decision makers who can easily segregate and visualize objects of interest and for further reconstruction of measurable 3D models.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

ALS	Airborne Laser Scanner
ASCII	American Standard Code for Information Interchange
ASPRS	American Society for Photogrammetry and Remote Sensing
DBH	Diameter at Breast Height
DEM	Digital Elevation Model
DGPS	Differential Global Positioning System
DSM	Digital Surface Model
DTM	Digital Terrain Model
EC	Euclidean Clustering
FPFH	Fast Point Feature Histogram
FP	False Positive
FN	False Negative
GEOBIA	Geographic Object Based Image Analysis
GPS	Global Positioning system
IMU	Inertial Measurement Unit
ISPRS	International Society of Photogrammetry and Remote Sensing
LAI	Leaf Area Index
LAS	LASer file format
LCCP	Local Cloud Connectivity Patches
LiDAR	Light Detection and Ranging
NB	Naive Bayes
NDEP	National Digital Elevation Programme
NDVI	Normalised Difference Vegetation Index
OBIA	Object Based Image Analysis
OBPA	Object Based Point Cloud Labelling Approach
PA	Producer Accuracy

RIFT	Rotation Invariant Feature Transform
RMS	Root Mean Square
SHOT	Signature of Histogram of Orientation
SVM	Support Vector Machine
TIN	Triangulated Irregular Network
TP	True Positive
TLS	Terrestrial Laser Scanner
TN	True Negative
TOF	Time of Flight
UA	User Accuracy
UN	United Nations
WGS	World Geodetic System
k-NN	*k*-Nearest Neighbour
3D	Third dimension

CHAPTER 1

INTRODUCTION

"The 19^{th} century was a century of empires, 20^{th} century was a century of nations and 21^{st} century will be a century of cities."

- Wellington E. Webb, the former mayor of Denver.

1.1 Urbanisation

According to the United Nations (UN) report, approximately 54% of the world's population lives in the urban region as of 2014 and the trend is expected to rise to 66% by 2050. India along with China and Nigeria are expected to contribute to 37% of growth of world's urban population. UN reports that managing the rapid urbanization for sustainable development is one of the key challenges of the 21^{st} century (United Nations, 2014).

Globally, governments of both developed and developing nations are embracing the concept of smart cities for improving the quality of life of urban population through a sustainable development (Caragliu et al., 2011; Batty et al., 2012). Smart city is an integration of information and communication technologies to manage the various assets of urban infrastructure. This will lead to improved governance leading to sustainable economic development.

Precise mapping of urban infrastructure is one of the key requirements for an integrated virtual urban utility platform. Over the years, geospatial technologies have proved to be valuable for mapping both human-made and natural resources (Paparoditis et al., 1998; Mayer, 1999; Wulder and Franklin, 2006). Exhaustive 3D (three dimensional) mapping of urban environment has been made possible using very high resolution satellite images, aerial photographs, maps, as well as data from traditional topographic surveys (Suveg and Vosselman, 2004; Jin and Davis, 2005; Luhmann et al., 2014; Haala et al., 2015).

1.2 Light Detection And Ranging

LiDAR (Light Detection And Ranging), an unparalleled geospatial technology introduced in the last decade, has proved to be one of the reliable and efficient sources of 3D data. Based on the time of flight principle, LiDAR is the culmination of three technologies, a compact and robust laser range finder, an inertial measurement unit (IMU), and a positioning system (GPS: Global Positioning System). By integrating these three sub-systems within a single instrument, it is possible to rapidly produce accurate three-dimensional digital topographic maps of the terrain, structures and objects on the surface.

LiDAR employs electromagnetic waves in the visible or in near-infrared region to record elevation of objects on the earth surface. The LiDAR ranger operates in the near-infrared region in case of topographic applications and in the blue/green region for bathymetric applications (Baltsavias, 1999; Wehr and Lohr, 1999; Burtch, 2002).

Typically, LiDAR sensor operates on three platforms : terrestrial-based, airborne or space-borne. The LiDAR system operated from the ground is commonly referred to as terrestrial laser scanner or TLS system. When the laser ranger is mounted on an aircraft, it is referred to as the airborne laser scanner or ALS system. Both these system operate as scanners or profilers and are now in widespread use for topographic mapping. Given the great distance and very high speed of the platform, there has been very few spaceborne LiDAR scanners (Petrie and Toth, 2008a). In this thesis, all the LiDAR datasets used are acquired by an ALS system.

LiDAR sensors are capable of recording the elevation of the earth surface by the principle of ranging. A schematic overview of an ALS system is presented in Figure 1.1. Presently, there are two types of ranging measurements used in the LiDAR systems for topographic applications (Petrie and Toth, 2008b): (*a*) timed pulse method or pulse echo method, and (*b*) phase comparison method. The timed pulse method is based on the time of flight (ToF) principle wherein a discrete laser pulse is used. In this method, the time when the laser pulse is sent from the ranger to the time the pulse is received after reflection from the target is precisely recorded. For targets that have surfaces at different ranges when illuminated by a single laser pulse, more than one backscattered pulse may be detected. Most of the ALS / TLS systems are capable of capturing multiple backscattered pulse responses per emitted pulse. Recently,

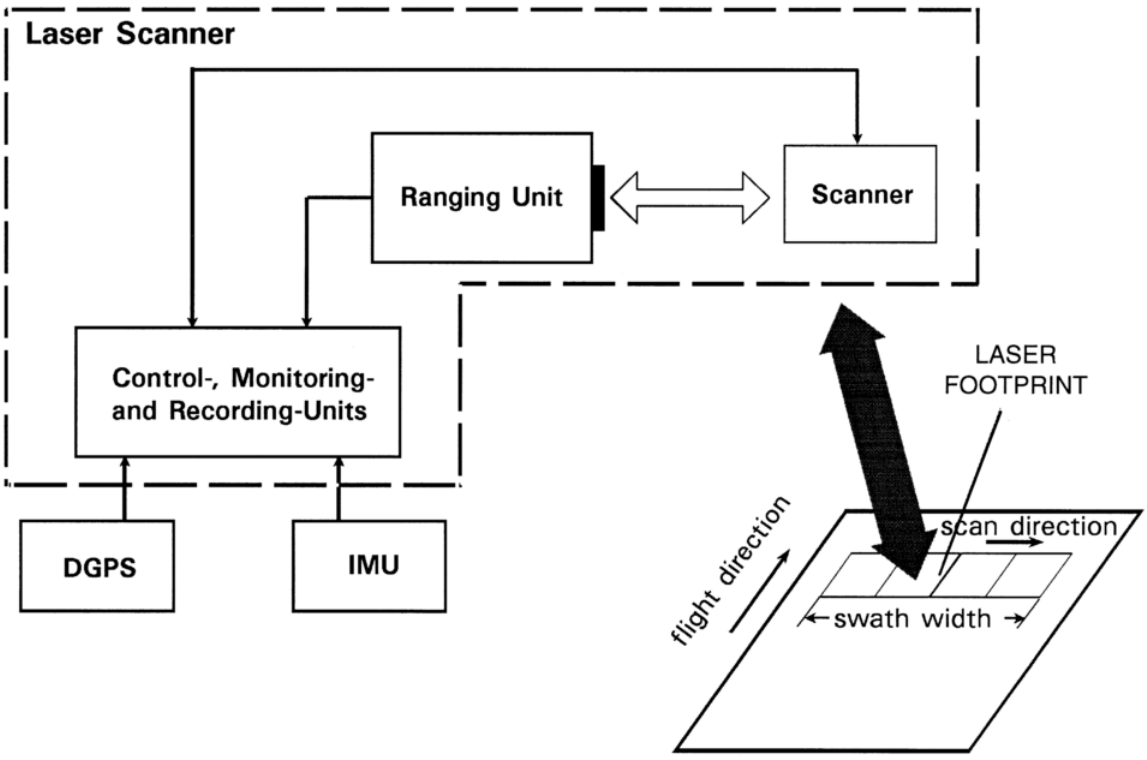


Figure 1.1: Schematic diagram of ALS system. (Source: Wehr and Lohr (1999))

commercial ALS / TLS systems are available which are capable of digitizing and recording the entire received signal of the reflected laser energy, thus resulting in full-waveform data (Mallet and Bretar, 2009). The phase comparison method employs a continuous wave laser ranging unit wherein laser light is emitted as a continuous beam instead of discrete pulses. The phase difference between the transmitted and the reflected signal is used in determining the distance to the target. These type of ranging units are popular in ground based systems and are rarely used in airborne and spaceborne platforms owing to the high power requirement of the laser.

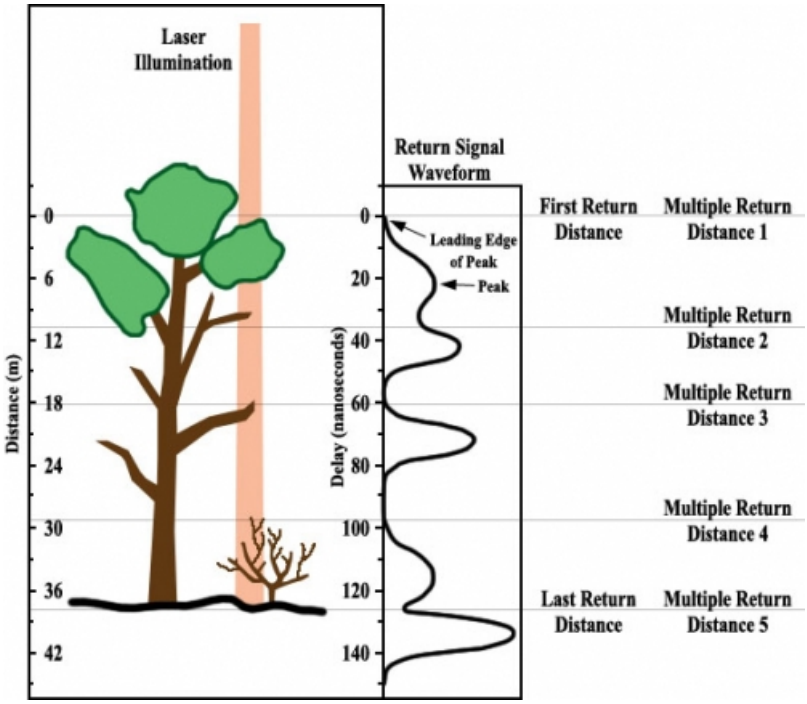


Figure 1.2: Schematic representation showing the return pulse concept. (Source: Harding et al. (2001))

The ranging distance measured by laser is integrated with positional and orientation unit of the laser scanner which comprises of DGPS (Differential Global Positioning system) and IMU (Inertial Mapping Unit). DGPS provides precise location of the target using two GPS receivers - a reference/ base GPS at the reference station and rover GPS mounted on the aircraft. IMU, a triad of accelerometers, and gyroscopes, records acceleration, rotation and velocity of the sensor (Woodman, 2007). The orientation data from IMU is integrated with positional information from the GPS resulting in an integrated geocoded 3D coordinates (X, Y, Z) of an objects. The mathematical equation to determine the geocoded location of a target is (Skaloud

and Lichti, 2006),

$$\mathbf{R} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_G = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{IMU} + \mathbf{R}_b^m \left[\left(I + \Omega_b^b\right) * T_s^b * \left(\begin{pmatrix} \rho sin(\theta) \\ 0 \\ \rho cos(\theta) \end{pmatrix} + \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix} \right) \right] \quad (1.1)$$

In the equation 1.1, $\mathbf{R} = [XYZ]_G$ refers to the coordinates of the target, $[XYZ]_{IMU}$ refers to IMU coordinates, $\mathbf{R}_b^m$ is the orientation matrix between IMU coordinate system and the mapping coordinate system (usually WGS-84 coordinate system) determined by three rotation angles ω, ϕ, κ. Ω_b^b is the rotation boresight matrix with small angles α, β, γ. These angles represent error in aircraft's flying pattern and hence are integrated with mirror scan angle measurements to determine the 3D coordinates of the point; T_s^b is the priori known rotation matrix from LiDAR frame to IMU body frame; ρ is the LiDAR range at time t and θ is the corresponding LiDAR's encoder angular value; $\mathbf{a} = [a_x a_y a_z]$ is the lever-arm offset between IMU and LiDAR measurement centers expressed in IMU body frame.

Characteristics of LiDAR Data The two important characteristics that define LiDAR data are point density, and point spacing (Baltsavias, 1999; Naus, 2008). Point density refers to the number of points per square meter on the ground, akin to the spatial resolution in remote sensing images. As defined by National Digital Elevation Program(NDEP) (Program, 2004), nominal point spacing, "the smallest distance between two points that can be explicitly represented in a gridded elevation dataset", corresponds to horizontal resolution. Typically LiDAR point density from ALS can be as low as 1 point / m^2 to as high as 100 points / m^2. The choice of point density and point spacing depends on the application for which data are intended.

A few studies suggest that the positional accuracy of LiDAR points is equivalent to that of traditional field surveying methods, with acquisition and processing time much faster than conventional aerial methods (Means et al., 2000; Burtch, 2002; Meng et al., 2010).

The scanner unit in ALS enables the laser ranging over a swath rather than in the nadir region. ALS units have typical scanning patterns to ensure maximum resolution or density

compensating the velocity of aircraft (Figure 1.3). The common scanning pattern used in the two large system of commercial suppliers of airborne laser scanners - Optech and Leica Geosystems, is the zigzag pattern produced by oscillating mirrors.

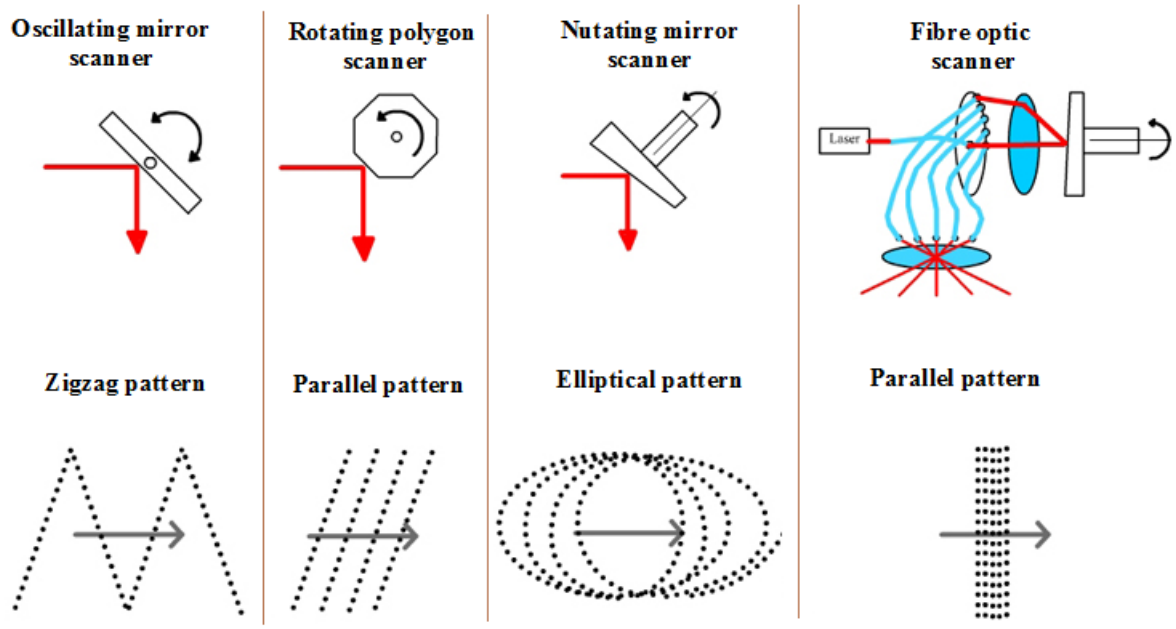


Figure 1.3: Schematic diagram showing the different scanning patterns of ALS. (Source:Burai (2012))

1.2.1 LiDAR data storage

The exploitation and transportation data format for a typical LiDAR data is ASCII (American Standard Code for Information Interchange) or binary data format. The emergent nature of data processing algorithms for LiDAR processing demands sensor characteristics to be recorded apart from the X, Y, Z information for every point (Graham, 2008). The size of data becomes cumbersome to handle when data are stored in ASCII format with all the additional sensor information. Hence in 2003, the American Society of Photogrammetry and Remote Sensing (ASPRS) introduced a standard for LiDAR exchange known as LAS (LASer) data format. This is a binary data format and hence is more convenient for data exchange as well as for software exploitation. A sample point record format in the current LAS 1.4 version is given in Table 1.1. The LAS 1.4 can store both discrete as well as full waveform information pertaining to every

point in the cloud. In 2012, a lossless compression data format, LASzip was introduced, which markedly reduces size of LiDAR point cloud. The different LiDAR characteristics based on the size of data in various data formats are compared in Table 1.2.

Table 1.1: Point record format of LAS 1.4 (Source:ASPRS (2013)).

Item	**Format**	**Size (bytes)**
X	long	4
Y	long	4
Z	long	4
Intensity	unsigned short	2
Return number	4 bits (0-3)	0.5
Number of returns	4 bits (4-7)	0.5
Classification flag	4 bits (0-3)	0.5
Scanner channel	2 bits (4-5)	0.25
Scan direction flag	1 bit (bit 6)	0.125
Edge of flight line	1 bit (bit 7)	0.125
Classification	unsigned char	1
User data	unsigned char	1
Scan angle	short	2
Point source ID	unsigned short	2
GPS time	double	8
Wave packet descriptor index	unsigned char	1
Byte offset to waveform data	unsignend long	8
Waveform packet size	unsigned long	4
Return point waveform location	float	4
X(t)	float	4
Y(t)	float	4
Z(t)	float	4

Table 1.2: Comparison of different LiDAR data characteristics and size of various data formats between two sample datasets

Parameters	**Sample dataset A**	**Sample dataset B**
Point spacing (in meters)	0.42	0.19
Point density (per square meters)	5.72	28.39
Number of points	3776182	1158349
Size of the data (ASCII format)	31.7 Mb	46 Mb
Size of the data (LAS format)	18.7 Mb	30.9 Mb
Size of the data (LAZ format)	2.64 Mb	5.9 Mb

1.3 Applications

The unprecedented accuracy and the ability to directly record elevation of terrain makes LiDAR a unique geospatial technology for mapping and monitoring of earth surface. Some of the applications of the LiDAR data are listed below.

- Digital elevation model (DEM) : One of the most popular applications of LiDAR is to generate an accurate and dense digital terrain at very high resolution. Studies have reported that the DEM generated from LiDAR is more accurate compared with the DEM generated from other geospatial sources such as photogrammetry, SAR (Hodgson et al., 2003; Gil et al., 2013).

- Urban studies: LiDAR point cloud are extensively used in urban studies for creating accurate 3D models. The 3D urban models are required in a range of applications such as urban planning, disaster management, flood mapping, virtual reality, insurance monitoring, energy management. The application of LiDAR for urban mapping is discussed in detail in section 1.4.

- Forest studies: The ability of LiDAR to penetrate through the gaps of tree canopy and hence capturing the true geometry of trees makes this technology find varied applications

in forest management (Lefsky et al., 2002; Reutebuch et al., 2005). Various structural attributes of trees including canopy height, crown diameter, diameter at breast height (dbh) can be directly be measured from LiDAR point cloud. These attributes can be modelled to estimate complex factors such as LAI (Leaf Area Index), biomass. Digital terrain models (DTM) of terrain in a dense forest environment, where other optical remote sensors fail, can be generated.

- Corridor mapping: LiDAR systems have also been extensively used for corridor mapping of linear corridors such as power utility, gas pipelines, highways, railways etc (Bethel et al., 2006). Particularly, in mapping power line corridors, the applicability of LiDAR varies from modelling sag of powerlines, ground clearance, encroachment, to accurate determination of tower locations.

- Mining: LiDAR data have been exploited for surveying large construction sites, and open-pit mines which demand timely availability of digital geo-referenced elevation data at very high accuracy.

- Disaster management: Natural disasters such as floods, earthquake, hurricane demand timely and accurate survey data to facilitate efforts towards post-disaster activities such as damage assessment, rescue services. Major topographic changes resulting from a disaster can be accurately mapped using LiDAR systems (Dash et al., 2004; Fritz et al., 2012).

1.4 LiDAR for Urban Mapping

With the recent availability of very high resolution 3D data from airborne LiDAR sensors, there is a growing interest in scientific community to develop fully automatic methods for extracting various objects of interest at high resolution for urban mapping. The level of automation in the processing of terrestrial laser scanning data is shown in Figure 1.4. The same can be extended for airborne laser scanning data as illustrated in Figure 1.5. The figure shows that the level of automation is comparatively low at labelling and reconstruction stages. This might be attributed to the basic nature of geocoded point cloud data which are just group of points fixed to a real-

world coordinate system and do not represent any defined structure with reference to ground level structure (Tomljenovic et al., 2015).

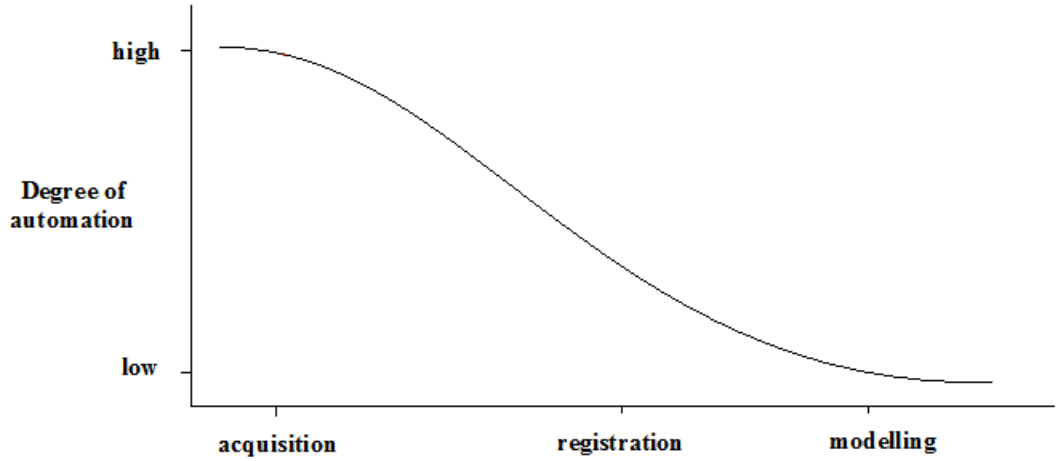


Figure 1.4: Automatic workflow for processing TLS data (Source: Gruen et al. (2009))

Classification methods are applied usually on the raster image produced by converting discrete LiDAR point cloud into a continuous two-dimensional raster image called digital surface model (DSM).

Numerous studies have been carried out using LiDAR point cloud for urban land cover mapping and buildings detection. High resolution surface height information from LiDAR in the form of a DSM has been used as ancillary data (Zhou et al., 2004; Matikainen et al., 2007; Sohn and Dowman, 2007; Lee et al., 2008; Demir and Baltsavias, 2012; Parmehr et al., 2013) or as the primary data for classification (Ma, 2005; Madhavan et al., 2006; Liu, 2008; Chen et al., 2012).

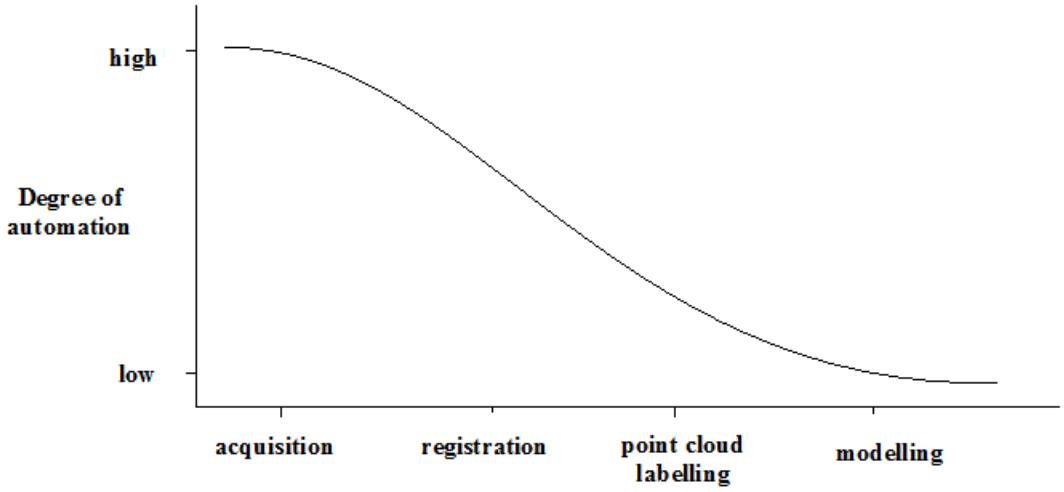


Figure 1.5: Automatic workflow for processing airborne laser data

1.5 Challenges & Motivation

Globally, LiDAR point cloud market is expected to reach US $625 million by 2020 (Transparency Market Research, 2014) with an increase in demand for fine-grade 3D mapping. Aerial LiDAR system (ALS) is going to dominate the market for its applications in forestry, transportation, infrastructure, flood mapping, transmission lines, defence and aerospace, and other sub-surface applications such as mining and hydrography.

Semantic labelling of point cloud into various landuse / landcover categories is one of the pivotal steps to utilize LiDAR point cloud for creating measurable 3D virtual models. The computationally complex unstructured LiDAR point cloud makes this a daunting task. As with any high resolution dataset, heterogeneous appearance of the same object makes it even more complex. Hence, automatic extraction of various objects from these complex datasets demands advanced 3D processing techniques and has remained an open research problem.

To extract various 3D objects from urban point cloud, it is imperative to assign class labels

to discrete points in the point cloud. This process, popularly known as point cloud labelling, can be either point-based or object-based (Zhang et al., 2013). Point-based methods, wherein local/global geometric description of points is used for labelling, have been widely reported in literature (Lalonde et al., 2006; Behley et al., 2012).

Given the promising results achieved in the analysis of high resolution multispectral images using object-based image analysis (OBIA) approach, recent developments in the methodological studies of LiDAR point cloud indicate a focused interest in object-based analysis of LiDAR point cloud. In contrast to the 2D objects in multispectral images, LiDAR point cloud, in principle, permits generation of 3D objects thereby enabling computation of 3D features for improved labelling.

Recent studies suggest that combined use of LiDAR point cloud and multispectral images improves land cover classification accuracy (Demir and Baltsavias, 2012; Li et al., 2013). Given the state-of-the-art airborne sensing technologies in which simultaneous acquisition of LiDAR and multispectral images is often the case, coloured point cloud (3D vector of point cloud with co-registered multispectral data) (Gerke and Xiao, 2014) is going to be the standard data product in near future. The rich spectral information from very high resolution multispectral image complements the inherent geometrical information from LiDAR aiding in improved object recognition.

The importance of 3D semantic labelling is evident from the international 3D semantic labelling contest initiated by the International Society of Photogrammetry and Remote Sensing(ISPRS) (ISPRS, 2016) and IQumulus (A high-volume fusion and analysis platform for geospatial point clouds, coverages and volumetric data sets) (IQumulus, 2015). While the former contains data collected by airborne platform, the latter contains data from ground based mobile laser scanner. Though both the contests have opened in the early 2015, there are very few participations indicating the complexities involved in 3D labelling. As indicated by the ISPRS work-group, though 3D LiDAR point cloud have been in the market for almost a decade now, there is no fully automatic or substantially automatic methods available for semantically labelling of point cloud.

1.6 Research Objectives

With this background, the main research goal of this thesis is to develop a novel computationally efficient 3D object-based point cloud labelling approach for semantic labelling of point cloud, which has been enriched by the spectral data from multispectral images. The methodological framework developed has been implemented by developing an appropriate open source prototype system.

Specific research questions addressed in this thesis are:

- Given the massive and computationally demanding LiDAR dataset over a city, is there a computationally efficient algorithm that can be utilized to semantically label multiple urban classes in a 3D dataset?
- Does spectral information from multispectral images complement the geometrical information from LiDAR in semantic labelling of 3D LiDAR point cloud?
- How does the combination of various computational parameters such as data fusion, point density, segmentation approaches, feature extraction, and classification approaches affect the 3D semantic labelling of LiDAR point cloud?
- How efficient is the open source 3D object-based LiDAR point cloud labelling approach compared to point-based labelling approaches adopted in commercial software packages?

1.7 Structure of the Thesis

The remainder of the thesis is organised into five chapters presenting the work related to the objectives of the research work. This section briefly discusses the content of each chapter. For the ease of readability and ready reference, the chapters are organised self-contained with reference to the corresponding objectives of the work.

Chapter 2 presents an overall methodological framework for 3D semantic labelling adopted in this work. This chapter also assesses the feasibility of segmentation-based approach for

building detection from urban LiDAR point cloud using open source library.

Chapter 3 explores the potential of integrating spectral information from multispectral images with the geometrical information from LiDAR point cloud for object-based semantic labelling. Colour-based region growing algorithm which integrates spectral and geometrical information has been adopted for creating 3D segments. Further, in order to reduce subjectivity involved in the selection of classifier and for improving the precision of results, multiple classifiers system is adopted to classify a scene into five land cover classes. The proposed methodology produces two outputs: (i) urban land cover classes, and (ii) buildings masks which are further reconstructed and vectorized into three-dimensional buildings footprints. Experiments were carried out on three subsets of airborne LiDAR datasets provided by ISPRS as part of the ISPRS Test Project on Urban Classification and 3D Building Reconstruction (ISPRS, 2016).

Chapter 4 proposes a novel computationally efficient supervoxels-based segmentation approach for multiple objects extraction from point cloud. Supervoxels are generated using voxel cloud connectivity algorithm in which voxels having colour, spatial, and geometric similarities are combined to produce supervoxels. The supervoxels are then merged based on the local convexity measures to create meaningful segments. Features are extracted from the clusters and are classified using machine learning techniques. The effect of feature selection (geometrical and spectral) and classifier selection on the accuracy of labelling is also assessed.

Chapter 5 critically assesses the different approaches used for 3D semantic labelling of urban point cloud. The aim of this study is to rigorously assess the degree of accuracy in 3D semantic labelling of LiDAR urban point cloud into multiple classes (9 in this case) through key parameters namely data fusion, segmentation approaches, classification algorithms and computational complexity. The labelling outputs of different methods are assessed in detail by ISPRS as part of 3D semantic labelling contest with the ground truth.

Chapter 6 summarizes the important findings and contributions of this thesis and elaborates directions for further research.

CHAPTER 2

OBJECT-BASED POINT CLOUD LABELLING

Prelude: In this chapter, the background of 3D semantic labelling and the different approaches used in semantic labelling of LiDAR point cloud is presented. Specifically, the methodological workflow in object-based semantic labelling of LiDAR used in this thesis is elaborated. Further, theoretical background of the different stages is presented. Finally, an empirical study for building detection is presented.

2.1 3D Semantic Labelling

Recent studies attempt classifying LiDAR point cloud without converting the point cloud into a raster DSM (Digital Surface Model) form (Rottensteiner, 2003; Sampath and Shan, 2010; Chen et al., 2012). This preserves the inherent geometric characteristics of the LiDAR point cloud and also reduces error in final classification accuracy (Golovinskiy et al., 2009; Zhang et al., 2013).

Semantic labelling or point cloud labelling is an imperative step to assign meaningful label (e.g. car, building, etc) to each point in the point cloud for exploiting the data for various applications. For example, to automatically count the number of cars in the parking lot from a set of millions of points, one of the first steps will be to semantically label each point and localise the points belonging to cars within the point cloud. Thus, semantic labelling tries to mimic how humans perceive point in the point cloud to be (Osgood, 2013). This research area is gaining popularity owing to the increasing demand for robotic applications in which field robots and automated vehicles navigate though urban environment in systems such as ADAS (Advanced Driver Assistance Systems). Semantically labelled 3D maps are sought for by the governments for improved resource management and governance. The rich geometrical information from the LiDAR point cloud is fully exploited for semantic labelling of various objects in a scene.

2.2 Methods of Semantic Labelling

Semantic labelling of point cloud can be broadly classified into two categories : (*a*) point-based labelling, and (*b*) object-based labelling, similar to raster image based classification.

2.2.1 Point-based labelling

Similar to pixel-based classification approaches in 2D image processing, point-based labelling treats each LiDAR point individually. Local geometric features describing each point in the point cloud is used for labelling. Most of the existing studies adopt point based labelling techniques for assigning class labels to points in the point cloud. Similarly, most of the LiDAR filtering algorithms, which separate ground points from non-ground points, are based on point-based labelling technique. Elevation is one of the important features used in point-based labelling. One of the early studies (Axelsson, 2000a) used slope and elevation at the point level to extract ground points from the entire point cloud.

Point-based labelling can be broadly categorised into: signature-based or histogram-based, depending on the method by which 3D descriptors are defined for each point in the point cloud (Tombari et al., 2010). In the signature-based method, a local invariant reference frame is defined according to the local coordinates and one or more geometric descriptors are computed individually on each point. In the histogram-based method, the local geometric / topological measures are accumulated into histogram according to specific quantized domain. A signature-based approach using local 3D point cloud statistics has been explored by Lalonde et al. (2006). The point cloud statistics were based on the distribution of 3D points over a defined neighbourhood. Histogram-based point descriptors have been widely used for object recognition in the computer vision community. Rusu et al. (2008) have used multivalue histogram that characterises local geometry around a query point for identifying objects. A new covariance-based descriptor has been used by Fehr et al. (2012) to define an object, while Tombari et al. (2010) have combined signature-based histograms to build descriptors of a point. A comparison of the various point descriptors such as histogram of normal orientation, spin images, distribution histogram, Signature of Histogram of Orientation (SHOT), spectral histogram has been presented

by Behley et al. (2012) for classifying 3D point cloud in an urban environment.

2.2.2 Object-based labelling

The provocative question raised by Blaschke and Strobl (2001), 'what's wrong with pixel' opened a new paradigm in the classification techniques used for processing of high spatial resolution remotely sensed images. The authors observed that while processing very high resolution images, statistical patterns formed from a group of pixels define land cover objects that are better than a single pixel. This led to the object-based image analysis (OBIA) for applications looking 'beyond pixels'. To emphasize its importance in GIScience, Hay and Castilla (2008) coined the term, GEOBIA, acronym for Geographic Object Based Image Analysis. Segmentation is considered to be the key stage in the OBIA analysis. Segments, the building blocks of OBIA, are based on the homogeneity criteria defined in one or more dimensions of the feature space. The segments are used to extract additional information such as shape, texture, area which describe an object. Many studies in the past decade (Blaschke, 2010) have proved the capability of object based image analysis for image classification.

Object-based point cloud labelling approach (OBPA) in the 3D domain is analogous to OBIA methods applied to raster images. It has been established that the rich spatial and geometric information associated with high resolution LiDAR data is valuable for accurate urban scene interpretation. Recent studies indicate that object-based point cloud labelling approaches have the capability of exploiting spatial dependencies of LiDAR point cloud and offer improved results (Golovinskiy et al., 2009; Zhang et al., 2013). Object-based point cloud labelling approach has been used in various applications. One of the first studies adopting the object-based point cloud labelling has been carried out by Höfle et al. (2007) for glacier surface classification. Rutzinger et al. (2008) employed a similar approach on full-waveform ALS point cloud to extract tall vegetation (shrubs ad trees) from urban data. Zhang et al. (2013) and Golovinskiy et al. (2009) have used object-based methods to label point cloud in an urban scene.

2.3 Object-Based Point Cloud Labelling

Based on the existing literature, the important stages in methodological approach of object-based point labelling can be defined as: segmentation, feature extraction, and classification (Figure 2.1). Object primitives, popularly known as segments, obtained from the raw point cloud are semantically labelled based on the features extracted. A brief theoretical background on each stage of OBPA approach is discussed in the following sections.

Figure 2.1: Methodological approach of object based point cloud labelling

2.3.1 Segmentation

Segmentation is one of the key steps in object-based point cloud analysis. Segmentation of point cloud in the 3D domain aims at extracting higher level information from a set of points in the cloud. The points belonging to a segment are homogeneous and can be assigned a single class label by classification techniques. In the case of LiDAR, homogeneity criteria are defined based on the geometrical relationship between points in the point cloud. In other words, homogeneity of a group of points refer to how best the points fit a mathematical surface. Ideally, each object should be represented by a single segment. But in real scenario, due to variation in geometry of an object and approximations on 'mathematical fitting ', a single object is segmented into more than one segment. For example, in the case of building with roof planes at different elevation, there is a greater probability that each plane is segmented into an individual segment. This is an example of over-segmentation. Another scenario is when a tree appears very close to a building, there is a probability that both the objects are clustered into a single segment.

This is a case of under-segmentation. Therefore, segmentation algorithms should be chosen such that there is a balance between over-segmentation and under-segmentation. However for feature extraction, over-segmentation is preferred to under-segmentation, as the segments can be merged during post processing and refinement stages to retain true geometry.

Definition

Let $\mathbf{R} = (x_i, y_i, z_i)$, $i = 1, 2, 3, ...n$ represent the point cloud. Segmentation is defined as the process that partitions **R** into *s* sub-regions such that

$$\Theta\mathbf{R} = \{\theta r \quad | \quad \forall r \quad \varepsilon \quad R\} \tag{2.1}$$

where $\Theta\mathbf{R}$ is the segmentation operation on the point cloud **R** and θr is the segment / cluster assignment for a single point, r, in R. The resultant segments, s, will have the following properties (Sithole, 2005)

- $S \quad = \quad \{s \quad | \quad s \subset \quad R\}$
- $\Theta \quad \Rightarrow \quad S$
- $\cup s_i \quad = \quad R \quad | \, s_i \, | \quad > \quad 0$
- $s_i \quad \cap \quad s_j \quad = \quad 0$ for all i and j where $i \quad \neq \quad j$.

From the above properties, it is evident that each segment is a closed subset of the point cloud, **R**. Each segment contains homogeneous points and there are no common points between any two segments.

Types of 3D segmentation algorithms

Segmentation algorithms can be categorised broadly into five categories:

- Feature-based techniques

- Surface growing techniques
- Model-based detection techniques
- Graph-based techniques
- Scanline-based methods

Feature-based techniques In this technique, n geometrical or radiometrical features are extracted for each point p_i such that it uniquely represents the points in the point cloud. The features represented in an n-dimensional feature space are then clustered based on similarity measures. Some of the features that can be extracted from the LiDAR are normal, curvature, and absolute height. The popular learning techniques such as *k*-means, *k*-Nearest Neighbourhood algorithms are used to demarcate the feature space. Some of the studies based on feature-based segmentation are briefly described in the following paragraphs.

One of the early studies based on feature-based segmentation for ALS point cloud is reported by Filin and Pfeifer (2006). The authors have adopted slope-adaptive method to characterise neighbourhood of a point from which several features are extracted. The features are clustered based on unsupervised learning technique. The authors conclude that the size and shape of neighbourhood greatly influences the features extracted. Lari et al. (2011) introduced a new approach based on an adaptive cylindrical neighbourhood for extracting features which are further used in clustering process. Nizar et al. (2006) demonstrated an approach using tangential plane and height difference measure as features for clustering based on spatial proximity. Sampath and Shan (2010) used fuzzy *k*-means algorithm to cluster the feature space comprising of the surface normals of points, while Ghosh and Lohani (2013) used density-based clustering techniques to establish a pattern within point cloud.

The main limitation of cluster-based methods is that it requires an initial estimate of number of clusters in the given feature space, which is difficult to obtain. The noise in data also affects the clusters created. The neighbourhood definition, and the distance similarity measure used substantially influence the features extracted. The features used in the clustering algorithm greatly influences the final clusters created.

Surface growing segmentation algorithms Surface growing algorithms are analogous to region growing methods in 2D image processing. The algorithms start by choosing seed points (patches) from the point cloud. Seed points (patches) represent planar surfaces in point cloud. The seed patches are grown based on similarity correspondence with the neighbouring points / patches.

Surface growing methods are initially adopted for segmenting point clouds obtained from an industrial scene. Rabbani et al. (2006) used surface growing methods for segmenting point cloud of industrial scenes. Initially normals are estimated for all the points within a fixed k-NN neighbourhood. A plane is fitted to the points and the residual of the each point within the neighbourhood is computed. This is an indicator of the curvature and the point with minimum curvature is the seed point. The surface is grown based on the similarity between seed point and the neighbouring point. A similar algorithm has been adopted by Pu et al. (2006) to extract building planes from terrestrial laser scanned point cloud. An arbitrary planar patch is chosen as seed patch. The patch is grown based on the criteria of proximity and global planarity of neighbouring points.

Wang and Tseng (2011) proposed a split and merge octree-based technique for segmenting point cloud. In this algorithm, the points are initially organised into an octree based voxel structure. Points belonging to a voxel are regarded as neighbouring points. A connected component labelling approach is used to merge neighbouring voxels in octree to form a larger group of coplanar points.

The initial seed points chosen by the surface growing algorithms greatly influence the clusters created and the conditions for region growing influence the segments created.

Model-based segmentation algorithm Most of the man-made structures on the earth surface can be decomposed into geometric primitives such as plane, cylinder, cone. Model-based algorithms are based on fitting geometric primitives in the point cloud. A group of points which conform to a geometric shape is grouped into one segment (Schnabel et al., 2007). 3D Hough Transform and RANSAC (Random Sample Consensus) are two popular model fitting algorithms which can be directly applied on the point cloud.

Hough transform is a technique which tries to detect primitive objects in parameter space. Each point in the point cloud is mapped to the parameter space or the Hough space. This space is then discretised into accumulator bins of dimension equal to the number of unknowns in the parameter space. The accumulator bins are populated based on a value by which points can be represented in the parameter space. The bins with maximum votes, i.e., the local maxima, correspond to the location where the geometric structure is present. Various studies have attempted detecting planar structures such as roof from the point cloud (Maas and Vosselman, 1999; Vosselman et al., 2004). Rabbani and Van Den Heuvel (2005) have reported automatically detecting more complex geometric primitives such as cylinders using these algorithms.

Another popular model fitting algorithm is RANSAC which selects a set of random points which is necessary to estimate the parameters of geometric primitive from a given set of points. A basic transformation equation is established with minimum required points for model fitting. The remaining points are tested to the candidate geometric model for best fit. The points which are within the set threshold value are considered as inliers and the remaining points are considered as outliers. A score is given to the basic model created based upon the number of points that are within the threshold distance. The model equation with the highest score is considered to be the best detected geometric primitive model (Schnabel et al., 2007). The number of trials required to find the minimum number of points and the cost of computing threshold of every point with respect to the geometric model are two challenges with RANSAC algorithm (Schnabel et al., 2007). Tarsha-Kurdi et al. (2007) have used RANSAC algorithm to successfully detect roof planes from LiDAR point cloud of varying complexity.

Model-based techniques are purely mathematical based. They are robust and fast with noisy point cloud. However, their performance declines while dealing with point cloud of complex geometries.

Scanline-based methods Scan-line based segmentation algorithms are used for structured point cloud where the scan geometry is retained. The points in a scan line is compared with other points within the scan line and the points in neighbouring scan line. If the points satisfy a defined geometric criterion, the points are clustered into the same segment. This algorithm works well for 2.5 D data though rarely used for 3D datasets (Jiang and Bunke, 1994; Han

et al., 2007).

Graph-based techniques The basic assumption of graph-based techniques is that the points within a segment are close to each other than the points belonging to a different segment. A proximity graph $G(V, E)$ is established for all the points in the point cloud. Here, a relation is defined for each point or a small group of points in set V. Set E defines the connections / edges between pair of points. The weights based on similarity between pair of points is given at each edge E.

Golovinskiy et al. (2009) have adopted a *k*-nearest neighbourhood (*k*-NN) method to establish a 3D graph. A graph cut algorithm is then used to cluster objects into foreground and background based on a penalty function where foreground is weakly connected with the background. Contextual and shape features are extracted from the clusters which are finally classified using various machine learning techniques. In this paper, the author have demonstrated the potential of object-based labelling approach for the point cloud. The study has been carried out using mobile laser scanning dataset. However, this approach requires prior knowledge of the objects to be segmented. Another study by Ural and Shan (2012) used min-cut algorithm on an airborne dataset. Initially a local neighbourhood is established. Point features are extracted for points within the neighbourhood. A min-cut algorithm is used for labelling the points.

Graph-based methods are robust and efficient for segmenting complex scenes inspite of the noise in point cloud. However, it is computationally demanding compared to the other methods.

2.3.2 Feature extraction

The resulting segments from segmentation stage contains set of points which are closely related to each other. Point cloud descriptors or features, that characterize the set of points within each segment are computed for object recognition. Let $S = \{S_1, S_2\}$ represent the segments resulting from the segmentation of the entire dataset. Let $S_1 = \{p_1^k, p_2^k....\}$, represent a set of points inside a segment S_1. A feature descriptor representation F can be described for the set of points inside a segment which captures the local geometrical / spectral information of the set of points. Let $\{F_1, F_2...\}$ represent the feature vector for segments $\{S_1, S_2.....\}$. Let Γ represent

the similarity measure between two features $\{F_1, F_2...\}$.

$$\Gamma = d(F_1, F_2) \tag{2.2}$$

Two segments are considered to be similar if $d \to 0$. If d is large, the segments are considered to represent different surface geometries.

Some of the important properties of a good descriptor are:

- rigid transformation : feature estimation should not be influenced by 3D rotation and 3D translation,
- varying sample density : similar geometric objects represented by different point density should have the same features, and
- noise: the segment should retain the same feature vector inspite of the noise in data(Rusu, 2009).

Feature extraction can be broadly categorized as geometrical features, radiometrical features, topological features, and echo features.

Geometrical features describe the shape and geometry of points inside each segment. Some of the popular geometrical features include planarity, convexity, compactness, linearity. Recently, 3D histogram-based geometric feature descriptors are also gaining popularity. Some of the popular histogram-based descriptors include Fast Point Feature Histogram (FPFH) (Rusu, 2009), Rotation Invariant Feature Transform (RIFT) (Tombari et al., 2010), Signature of Histogram of Orientation (SHOT) (Alexandre, 2012).

Radiometrical features capture the variations in reflectance / intensity of points within a segment. Most of the ALS systems operate in NIR region. The instruments are capable of recording back scattered energy of emitted laser pulse. Few studies have successfully attempted using intensity value for object discrimination (Jutzi and Gross, 2009). If spectral information is available for each point within a point cloud from multispectral images, radiometrical features such as mean intensity, NDVI (Normalised Difference Vegetation Index) can be computed.

Topological features compare the topological relation between neighbouring segments. The average elevation difference, slope between two segments are some of the topological features that can be extracted and is useful for discriminating various land cover classes (Zhang et al., 2013).

The state-of-the-art LiDAR sensors are capable of recording upto six returns if the object on ground can produce multiple reflections such as a tree does. This unique property of LiDAR aids in discriminating various land cover objects (Darmawati, 2008). For example, a building or a road surface produce single return for a given laser pulse whereas vegetation classes produce multiple returns. Based on this principle, echo-based features such as average elevation between first and last echo, average number of echo per pulse for a particular segment, and proportions of multiple echoes can be extracted.

2.3.3 Classification

Advanced machine learning techniques are used to classify the feature vectors into various classes. Either of supervised or unsupervised learning techniques can be employed for classification. Some of the popular feature-based classification algorithms found in literature are discussed below.

***k*-nearest neighbour classifier** k-Nearest Neighbour (k-NN) is a non-parametric lazy learning supervised algorithm (Altman, 1992). The algorithm classifies an unknown sample x_i based on majority vote of the class that is present in the k-nearest neighbourhood obtained from the training samples. The k-Nearest Neighbour to a point is computed based on a distance metric measuring the distance between query point and training samples. Some of the popular distance

metrics are

$$\begin{aligned} &\text{Euclidean} && \sqrt{\sum_{i=1}^{k}(x_i - y_i)^2} \\ &\text{Manhattan} && \sum_{i=1}^{k}|x_i - y_i| \\ &\text{Minkowski} && \sum_{i=1}^{k}(|x_i - y_i|)^q \end{aligned} \tag{2.3}$$

Though *k*-NN is a simple classification algorithm, it gives highly accurate results(Wu et al., 2008).

Naive Bayes classifier Based on the Bayes rule and a set of conditional independent assumptions, the Naive Bayes algorithm is a fast and space efficient probabilistic classification algorithm (Lewis, 1998). The classifier assumes that the effect of value of predictor *x* on a given class *c* is independent of other predictors. The classifier estimates the probability of class c_j generating instance *x* given by

$$p(x|c_j) = p(x_1|c_j) * p(x_2|c_j) * .... * p(x_n|c_j) \tag{2.4}$$

One of the main advantages of this classifier is it requires very less number of training samples to estimate the parameters required to create a model.

Support vector machines Support Vector Machine (SVM) is a binary classification tool developed by Vapnik (2013) based on statistical learning theory. The algorithm tries to minimize the bounds of generalisation error (learning error rate on unknown data) by maximising the margin of separation between classes.

Let a set of feature vectors, x_i be training dataset with class labels y_i, where $x_i \, \epsilon \, \Re_n$ and **y** ϵ

$\{1, -1\}$. SVM tries to find an objective function, the standard form of it can be represented as

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2} \parallel w \parallel^2 + C \sum_{i=1}^{m} \xi_i \\ \text{subject to} \quad & y_i(w^T \phi x_i + b) \geq 1 - \xi_i, i = 1, ..., m \\ & \xi_i \geq 0, i = 1, ..., m \end{aligned} \tag{2.5}$$

where w is a set of normals from each sample to the hyperplane, b is bias term, ξ_i is positive slack variables which allow the data to violate the margin constraint in primal problems and C is penalty parameter (Fan et al., 2008). The function ϕ in the objective function maps the vector x_i into a high dimensional space. If the classes are not linearly separable in *n* dimensional features space, then SVM uses kernel trick to map the feature to high dimensional feature space. The mapping is done using kernel function represented as

$$\kappa(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j) \tag{2.6}$$

Some of the popular kernels that are used for mapping the high dimensional feature space are

- linear kernel: $\kappa(x_i, x_j) = x_i^T x_j$
- polynomial kernel: $\kappa(x_i, x_j) = (\gamma x_i^T x_j + r)^d$
- Radial Basis Function (RBF) kernel: $\kappa(x_i, x_j) = exp(-\gamma \|x_i - x_j\|^2$
- sigmoid kernel: $\kappa(x_i, x_j) = tanh(\gamma x_i^T x_j + r)$

  where γ, r, d are the kernel tuning parameters.

Support Vector Machines can be also used for multiple class problem by treating sets of pair of classes to train a classifier. If the dataset contains M classes, it results in ${}^M C_2$ binary classifiers.

***k*-means clustering** A popular unsupervised algorithm that aims to find the positions μ_i, $i = 1, 2, 3...k$ of the clusters that minimise the distance from the data points to cluster center (MacQueen et al., 1967). The *k*-means clustering algorithm solves the following equation

$$\arg\min_{c} \sum_{i=1}^{k} \sum_{x \varepsilon c_i} d(x, \mu_i) = \arg\min_{c} \sum_{i=1}^{k} \sum_{x \varepsilon c_i} \|x - \mu_i\|_2^2 \tag{2.7}$$

where c_i is the set of points that belong to cluster i.

Though this algorithm is fast and robust, it requires initialisation of the number of clusters ie., *k*.

Random forest algorithm An ensemble classifier, random forest algorithm is based on the principle of bagging approach on decision tree classifiers. Introduced by Breiman (2001), the classifier consists of tree structured classifiers $\{h(\mathbf{x}, \Theta_k), k = 1\}$, where the $\{\Theta_k\}$ are independent and identically distributed random vectors. Each tree structured classifier casts an independent vote for the most popular class at input **x**. As in the concept of bagging, successive trees are independently constructed based on bootstrap sample of the dataset and does not depend on the earlier trees.

Decision trees are predictive models used widely in machine learning and are built based on training features. The feature with maximum information amongst a given set of features is chosen as the root node. The entropy or the degree of uncertainty of each feature in predicting output is computed.

2.3.4 Related studies

A few recent studies have employed object-based workflow for point cloud labelling. Zhang et al. (2013) have used SVM (Support Vector Machine) to classify thirteen features extracted from airborne LiDAR point cloud. Rutzinger et al. (2008) used a rule based classification scheme based on statistical classification tree for classifying urban vegetation classes.

2.4 LiDAR Point Cloud Pre-Processing

Unlike the conventional two-dimensional image data and the three dimensional (3D) gridded elevation data, the geocoded LiDAR points are recorded as discrete points in an irregular distribution form. These LiDAR points are commonly referred to as 'point cloud' because of its irregular distribution and ill-defined boundaries (Graham, 2008). The pattern in which the first 1000 points from two sample LiDAR data are stored is shown in Figures 2.2 and 2.3. As can be seen, the distribution of the LiDAR points is a combined effect of various sensor characteristics such as the scanning pattern, pulse repetition frequency, the altitude of the sensor. This results in directionally dependent spacing between points (Baltsavias, 1999). Thus point cloud organization poses challenge in data processing as simple data processing algorithms such as establishing neighbourhood, interpolation are more complicated compared to the conventional two-dimensional image processing operable on regular grids. The following section discusses the important data pre-processing steps adopted in LiDAR point cloud processing.

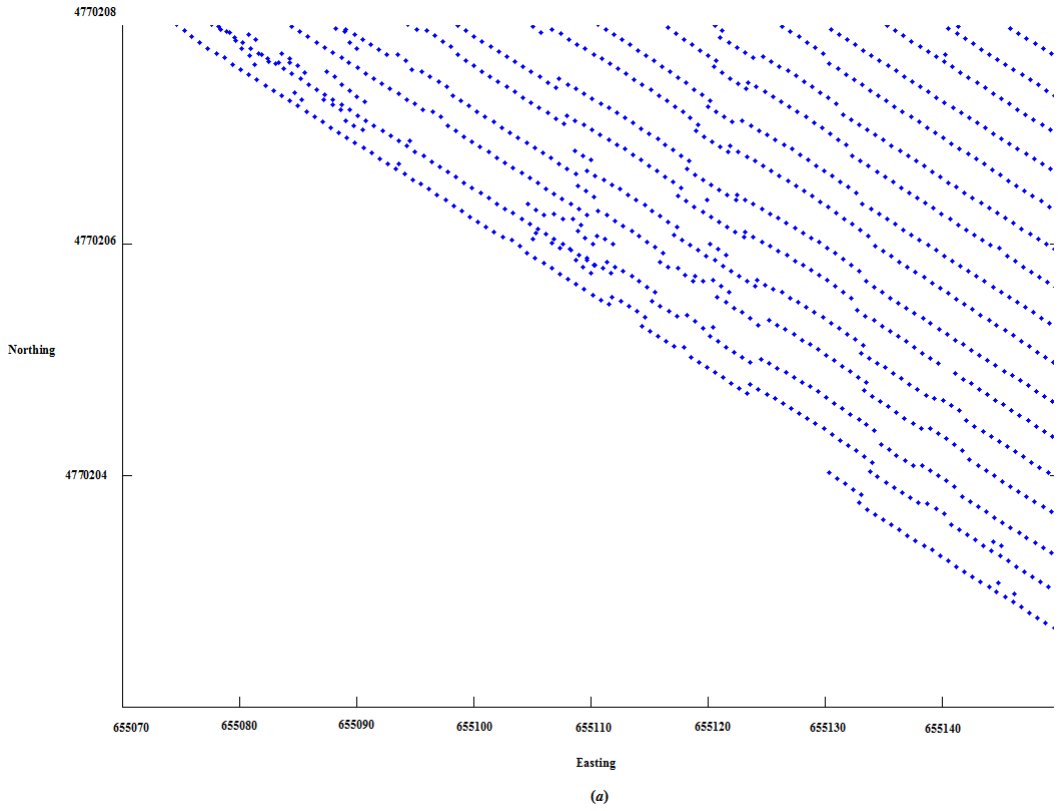


Figure 2.2: Sample data distribution from LiDAR point cloud: sample dataset-A

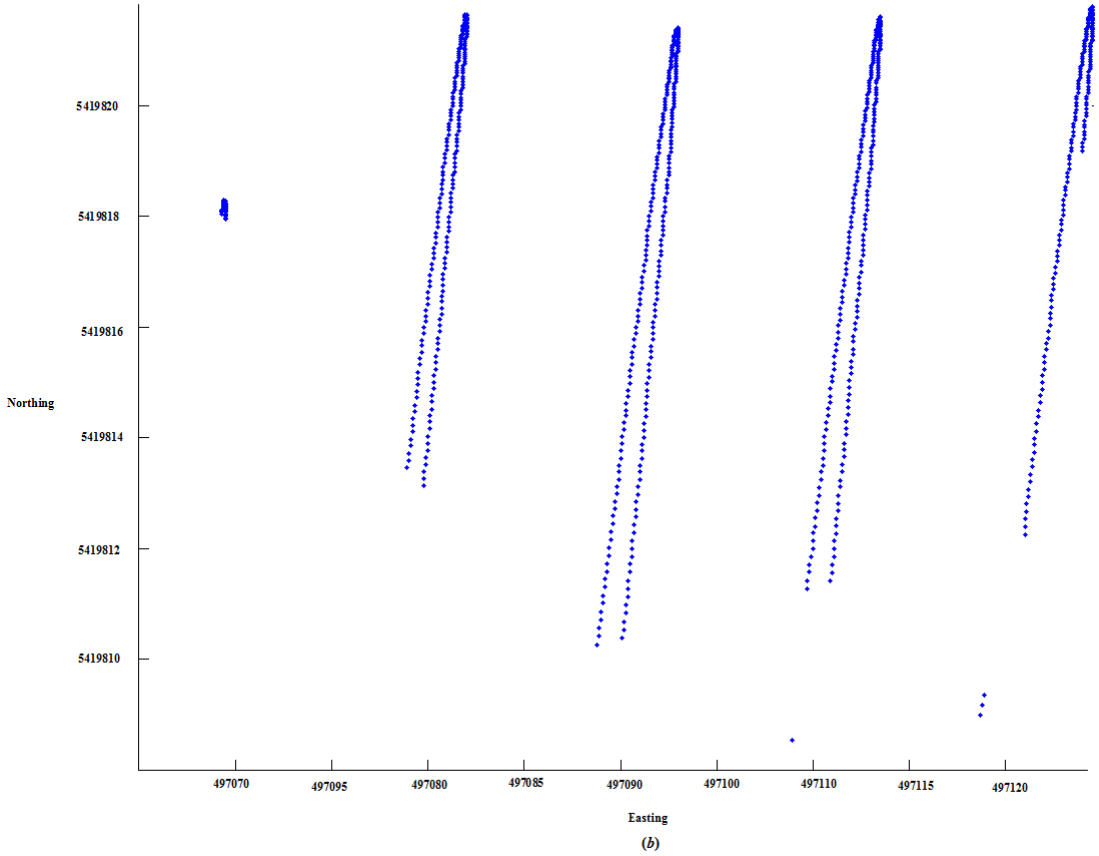


Figure 2.3: Sample data distribution from LiDAR point cloud: sample dataset-B

2.4.1 Data structuring algorithm

Data structuring is one of the important pre-processing stages in point cloud processing that is used to establish neighbourhood between points in point cloud. Two popular data structuring algorithms used for LiDAR point cloud, *k*-d tree structuring, and octree structuring, are explained in the following section.

The *k*-d tree structuring

The *k*-d tree is a binary space partitioning technique which organises the points in *k*-dimension (Bentley, 1980; Moore, 1990). The *k*-d tree formed with LiDAR point X,Y, Z is three dimensional. At each level, the data are split along a specific dimension based on a hyperplane perpendicular to a particular plane (YZ, XY, XZ). At the root of the *k*-d tree, all the data point are split based on the first dimension (say, X direction). All the points in the sub-tree with values less than the root node will appear in the left sub-tree and the points with greater value than the root node appears in the right sub-tree. At the next level, the split happens along the second direction (say, Y) and then in the Z direction. After this, the split starts again in the *X* direction. This recursive splitting happens until there are less than two points in the leaf node (Figure 2.4).

Octree structuring

Octree data structuring is an hierarchical data structuring algorithm for partitioning three dimensional space in which the root octant is recursively divided into eight octants (Samet, 1988). This structuring is an analogue to quadtree in two dimensional space. The entire set of points is initially assumed to be enclosed in a cuboid and this is recursively partitioned into eight octants at each level. Thus each root node will have eight children and each non-terminal node in an octree will have eight children. Also, each of the octant except the root node will have only one parent (Figure 2.5). The partitioning happens until the remaining number of points inside an octant are below threshold or maximum tree-depth is reached.

Both the data structuring algorithms are widely used in LiDAR point cloud for establishing

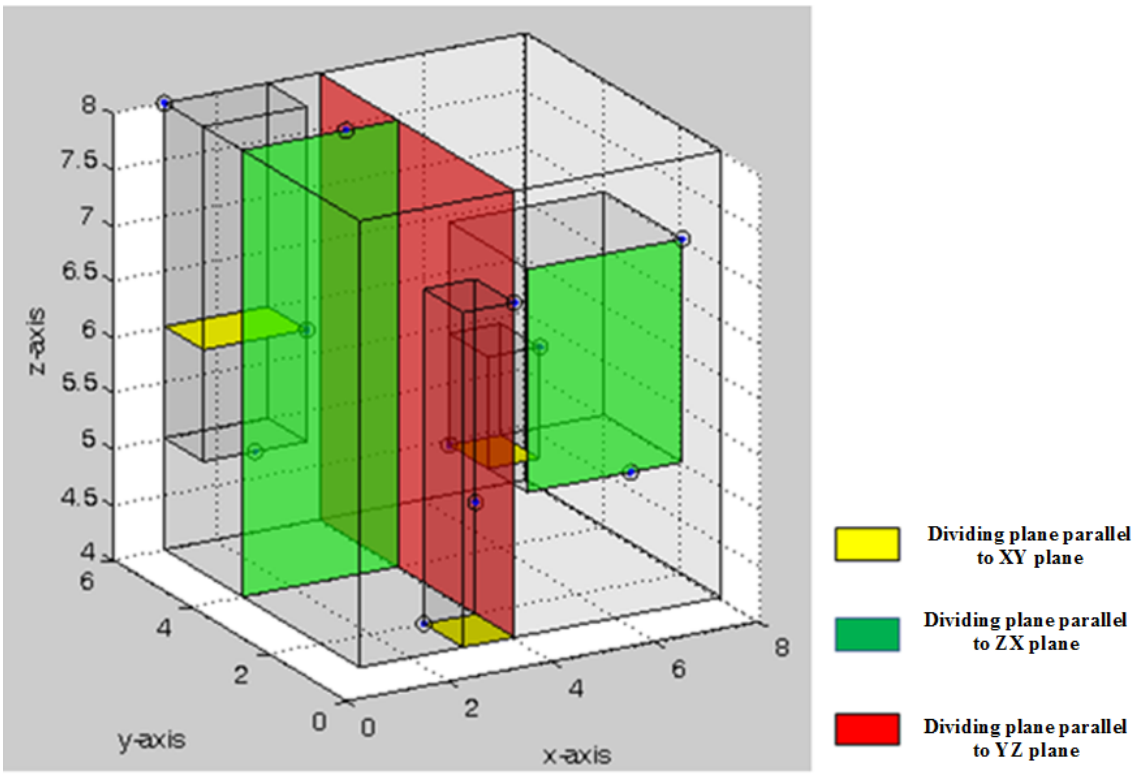


Figure 2.4: *k*-d tree representation of 3D dataset (Source: Panhalkar et al. (2014))

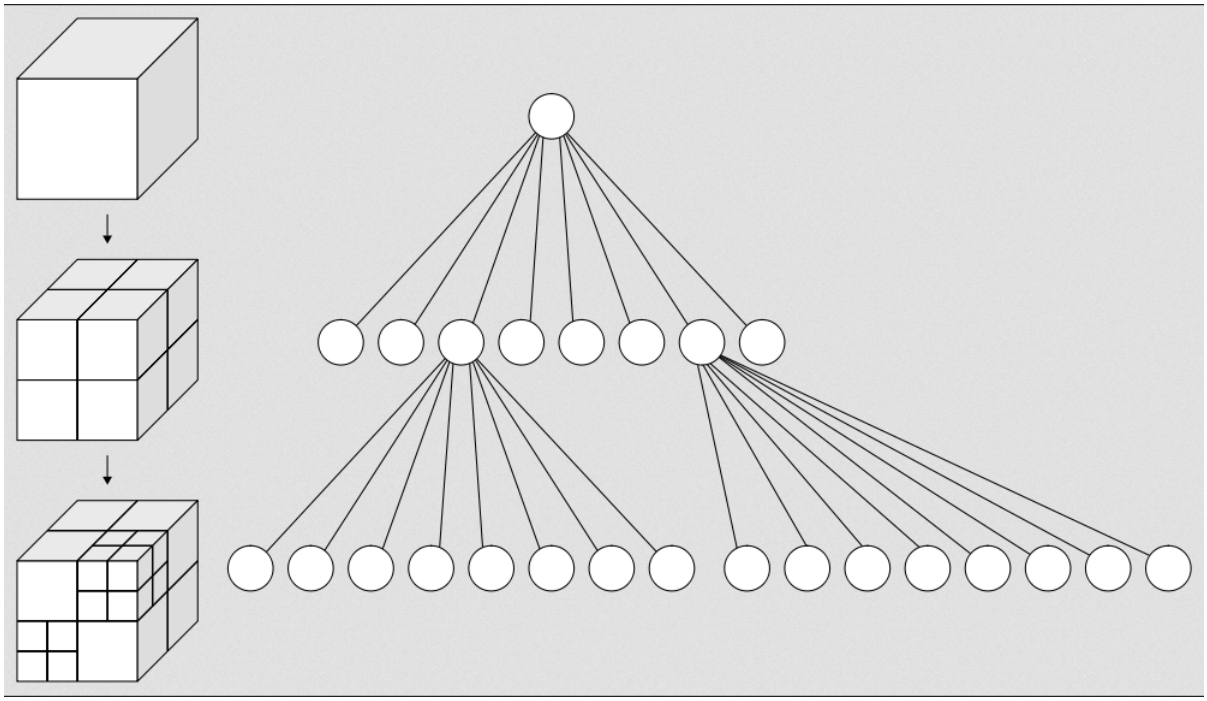


Figure 2.5: Octree representation of the 3D dataset (Source: Coeurjolly (2015))

neighbourhood, nearest neighbour search, 3D reconstruction, visualisation, and storage (Liu et al., 2008; Wang and Tseng, 2011).

2.4.2 Data filtering

Data filtering is another important pre-processing step wherein the ground points are separated from non-ground points. Filtering plays an important role in semantic labelling of LiDAR point cloud as it reduces the computational complexity of classifiers in further stages of labelling. For example, if the aim is building modelling or tree modelling, instead of working on the entire dataset, it is more meaningful to work with only the non-ground points. This reduces the computational complexity of dataset in further stages of processing. Some of the popular filtering techniques are morphological operators based filtering (Vosselman, 2000), progressive densification based filtering (Axelsson, 2000a), surface-based filtering (Pfeifer, 2005), and segmentation-based filtering (Filin and Pfeifer, 2006). The following section briefly describes each of the filtering algorithms.

Morphological operators based filtering

Based on the concepts of mathematical morphology, these set of filters utilise structural elements $\Delta h_{max}(d)$ which describe the admissible height difference as a function of the horizontal distance d in the erosion process. The main assumption is that the points which are in the neighbourhood of ground points have lesser height difference accepted between them. Based on the structuring element used, a point is marked as a non-ground point if the height difference to its neighbour is greater than the admissible height difference.

As defined by (Vosselman, 2000), ground point P_T is defined by the following equation

$$P_T = \{p_i \, \varepsilon \, P \mid \forall \, p_k \, \varepsilon \, P \; : z_i - z_k \leq \Delta h_{max}(d_{ik})\}, d_{ik} = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2} \quad (2.8)$$

where

$$h_{max}(d) = \sqrt{2}\sigma_z + dtan(\gamma), d \leq d_{max} \tag{2.9}$$

where σ_z denotes the vertical measurement precision, and tan(γ) denotes the maximum slope.

Progressive densification based filtering

This group of filters densify the ground points progressively by adding more points to the set of initially selected ground points. The sub-randomly selected set of ground points are triangulated using TIN network. The offset angles made between each unclassified point and the vertices of underlying triangle are determined. If the values are less than the predetermined threshold, the point is classified as ground point. The set of ground points is progressively densified until all the unclassified points are labelled.

Surface-based filtering

Contrary to progressive densification based filters wherein the terrain is built from a group of points, the surface-based filters iteratively remove the points that do not fit a surface from the initially built surface from the entire point set. A robust interpolation is used to describe the surface from a set of points. Weights are determined iteratively for the set of points based on the residual from the best fit plane. An asymmetric weight function is used by Kraus and Pfeifer (1998) to assign weights (*w*) to the set of points as given in

$$w(r) = \begin{cases} r < g & : 1; \\ g \leq r \leq g + h & : \dfrac{1}{1 + (a(r-g))^b}; \\ r > g + h & : 0; \end{cases} \tag{2.10}$$

where *r* is the residual of the point, parameters a & b determine how fast the weights drop, *g* determines how fast the drop starts, and *h* is a parameter controlling the exclusion of definite vegetation points from the system.

Segmentation-based filtering

Another group of filters which analyse the segments or groups of points rather than individual points fall under segmentation-based filters (Filin and Pfeifer, 2006). Points in the neighbourhood are grouped together to form segments based on homogeneity criteria. The segments are further classified as terrain or non-terrain points based on certain geometrical features extracted from the segments.

A review of the ground filtering algorithms by Liu (2008) and Meng et al. (2010) summarize that the current ground filtering algorithms use one of the following characteristics to separate ground points from non ground points: lowest feature in specific area, ground slope threshold, ground surface elevation difference threshold, and smoothness. The study concludes that most of the filtering algorithms work well for a flat terrain dataset but have considerable limitations over an undulating terrain.

2.5 Feasibility Study on Segmentation-based Building Detection

We carried out a feasibility study to evaluate the potential of object-based semantic labelling approach to extract building points from the entire LiDAR point cloud over an urban environment. The following sections describe the study carried out and evaluation of its performance with point-based labelling approach [1]. For the ease of reading and ready reference, the study has been described as a set of self-contained text.

The airborne LiDAR point cloud used over an urban setting consist of returns from both the natural (trees, bare earth) and man-made features (buildings, cars, roads etc.). Building features identification and extraction is a key step in urban modelling. Over the past decade, several algorithms for the extraction of buildings from point cloud have been widely reported

[1]This part of the chapter is published in *The Egyptian Journal of Remote Sensing and Space Sciences* 2016 Publisher: Elsevier. *http://dx.doi.org/10.1016/j.ejrs.2016.04.001* Authors: Anandakumar M Ramiya, Rama Rao Nidamanuri, Ramakrishnan Krishnan.

(Lari et al., 2011; Wang and Tseng, 2011; Lafarge and Mallet, 2012; Zhang and Lin, 2012), which are either point-based or object-based.

Apart from the key algorithms necessary for processing and analysis, implementation platform such as software platform is a key resource for the effective and affordable analysis of LiDAR point cloud. Commercial LiDAR processing software such as Terrasolid (Terrasolid, 2016) offers routines to automatically detect building points from the input LiDAR point cloud. However, there is no defined method for automatically extracting buildings from a point cloud besides the lower affordability of commercial software for many researchers. Recently many open source libraries are available to process 3D data. However there is no literature available showcasing the utility of the open source library for building detection. This part of work presents a building extraction methodology using object-based approaches in an open source platform. An open source point cloud library (PCL) was adapted and modified to cluster the entire point cloud into segments. A novel building detection algorithm was then employed to separate the building clusters from the non-building clusters. The buildings extracted using the proposed method are compared with the buildings extracted from a popular commercial LiDAR processing software (which adapts the point-based labelling approach).

2.5.1 Data pre-processing

Filtering is an important pre-processing step which separates ground points from non-ground points thereby reducing the data size and helping in identifying building points. The filtering algorithm adopted belongs to the category of surface based filtering. Each point was given a weight based upon the distance to the mean interpolated surface. A threshold was determined based on the distance of each point from the mean surface. Based on the threshold value, the points were classified either ground or non-ground point. The points were organised in a hierarchical data structuring using the *k*-d tree method.

2.5.2 Segmentation

3D segmentation of point cloud enables partitioning spatially isolated regions. We implemented the Euclidean distance based segmentation algorithm using PCL (Point Cloud Library) which is an open source VC++ library (Rusu, 2009). This segmentation method creates clusters based upon the Euclidean distance between a point and every other point. If the distance is within a particular threshold, the point is placed in a new cluster; otherwise the point is placed in a queue. The process continues until all the points are processed and no point remains in a queue. The pseudocode of the algorithm is given below.

- Arrange the input point cloud $\boldsymbol{P}$ using a *k*-d tree data structuring algorithm
- Initialise two arrays : empty lists of cluster $\boldsymbol{C}$ and a queue $\boldsymbol{Q}$ for points that needs to be checked
- Add the points of each nodes of the data structure to temporary array $\boldsymbol{P_1}$
- For every point $\boldsymbol{p_i}\ \varepsilon\ \boldsymbol{P_1}$, perform the following steps
  - Add $\boldsymbol{p_i}$ to the current queue $\boldsymbol{Q}$
  - For every point $\boldsymbol{p_i}\ \varepsilon\ \boldsymbol{Q}$, do,
    - Search for the set $\boldsymbol{p_k^i}$ of point neighbours of $\boldsymbol{p_i}$ in a radius $r < d_{th}$
    - Set flag for the processed points
    - For every neighbour $\boldsymbol{p_k^i} \varepsilon \boldsymbol{P_k^i}$, check if the point has been processed, if not move it to $\boldsymbol{Q}$
    - Move all the flagged points to cluster $\boldsymbol{C}$ with labels
- When all the points in $\boldsymbol{Q}$ has been processed reset $\boldsymbol{Q}$ to empty list
- Algorithm terminates when all points are processed and part of cluster $\boldsymbol{C}$

Minimum and maximum number of points in a cluster is user defined and hence the number of clusters varies for a particular dataset. Euclidean distance based segmentation can detect both natural and man-made object which makes it applicable across many types of landscapes.

2.5.3 Building cluster detection

To recreate buildings in the scene, the building clusters have to be separated from the clusters of other natural features such trees. One simple method is to open each of the clusters separately in LiDAR visualization software and separate the building and the non-building cluster. However, this process is cumbersome and time consuming. Our attempt to the automation of this process led to a novel building detection algorithm based on the histogram of the local normal of each point in the clusters. In this method, the local surface normal was computed for each of the points in the cluster. The direction cosines of the normal were found out and histogram was generated. The statistical parameters of the histogram such as mean, range and standard deviation were computed. These parameters vary significantly for a tree cluster from that of a building cluster. This methodology can be adopted to separate building cluster from non-building cluster.

2.5.4 Building detection using commercial software

Terrasolid is a commercial software capable of end to end processing of LiDAR point cloud. This software uses the progressive densification algorithm for filtering, i.e., separating ground points from non-ground points. The ground points which are identified from the filtering process are used to classify the rest of the points. The non-ground points are labelled into various land cover categories such as low vegetation, high vegetation, buildings based on the distance of the point with respect to the ground points. To detect the building points from the non-ground points, the algorithm starts by identifying the holes in the ground points. The non-ground points which are in the location of the holes are checked for planarity conditions. If it satisfies, the points are classified as building points (Soininen, 2004).

2.5.5 Experiment on LiDAR datasets

The proposed methodology was implemented using the open source point cloud library in Visual C++ on two different airborne LiDAR datasets which differ by point density and return numbers. The first dataset (dataset 1 in Table 2.1) was collected in 2004 over the Niagara falls

neighbourhood using the Airborne Laser Terrain Mapping (ALTM) 3100 sensor (Optech Inc., Concord, Canada) at a flying height of 1190 m. The density of the point cloud is 2 points $/m^2$. The second dataset (dataset 2 in Table 1) was part of the point cloud data collected in 2005 over the Yakima county of southern Washington using the Terrapoint-s40 ALTMS flying at a height of 1060 m. The density of the point cloud is 5.5 points $/m^2$.

Table 2.1: Airborne LiDAR datasets used for the study

Parameters / dataset	**Dataset 1**	**Dataset 2**
Point density (points /m^2)	2	5.5
Number of 1^{st} return	31033	108198
Number of 2^{nd} return	2594	0
Number of 3^{rd} return	76	0
Total number of points	33703	108198
x range (m)	112	103
y range (m)	139	201

2.5.6 Results and discussion

The filtering output of the both LiDAR datasets is shown in Figures 2.6, 2.7 and Table 2.2. The ground points are effectively removed from the point cloud. The filtered dataset was arranged using *k*-d tree structuring. The points were then segmented into different clusters by 3D Euclidean distance based segmentation. From the clusters obtained, building and non-building clusters were separated using the histogram of the local surface normal of each cluster.

Each of the clusters obtained for the dataset was categorized into building and non-building cluster based on the peaks of the respective histogram. The histogram of the building and non-building cluster is given in Figures 2.8 and 2.9. The range and mean of the values of the elements in the histogram can be used to separate building cluster from non-building cluster. As seen in Figures 2.8 and 2.9, the histograms derived from the building clusters exhibit few distinct peaks whereas the histogram of the non-building clusters spreads wide apart. For

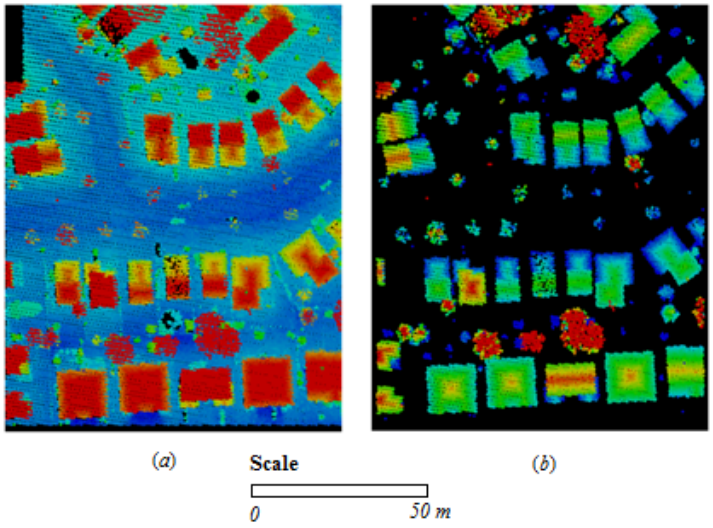


Figure 2.6: Before and after filtering dataset 1: (*a*) original dataset, (*b*) non-ground points.

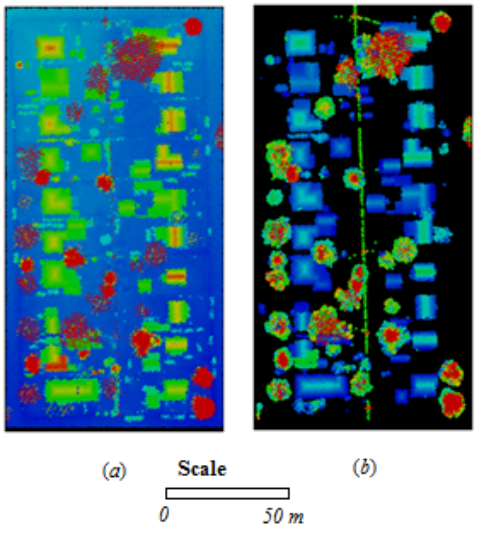


Figure 2.7: Before and after filtering dataset 2: (*a*) original dataset, (*b*) non-ground points.

Table 2.2: Number of points before and after filtering.

Parameters / dataset	**Dataset 1**	**Dataset 2**
Total number of points	33703	108198
Ground points	22645	72478
Non-ground points	11058	35720

dataset 1, 35 clusters are identified. Some of the user defined parameters include minimum and maximum number of points in the clusters. In this case, the minimum cluster size was 50 and maximum cluster size was 5000. Complete enumeration suggest that there are 25 buildings and 3 trees in dataset 1. Figure 2.10 presents the result of the proposed algorithm for building detection in dataset 1. It can be seen that some of the buildings identified are over segmented as multiple cluster as indicated by MBC. This can be attributed to the presence of multiple roofs for the building at different heights. These partial buildings must be merged together while creating the final output for 3D model of the building. The algorithm has successfully identified all the tree clusters in the dataset, though some of them are partial tree clusters. Buildings which are classified under multiple clusters are marked using MBC1, MBC2, and MBC3. In total, 25 building clusters including the multiple building clusters are identified in dataset 1.

For dataset 2, 24 clusters are identified. The minimum and maximum number of clusters set chosen is 200 and 1200. For this dataset, there are no multiple clusters or the presence of tree clusters. Manual inspection indicates the presence of 28 buildings in dataset 2, of which 23 building clusters are identified (Figure 2.11). Table 2.3 summarizes the results of Euclidean distance based clustering on dataset 1 and dataset 2. The choice of minimum and maximum number of points in the cluster is to minimize the over-segmentation and under-segmentation of the cluster.

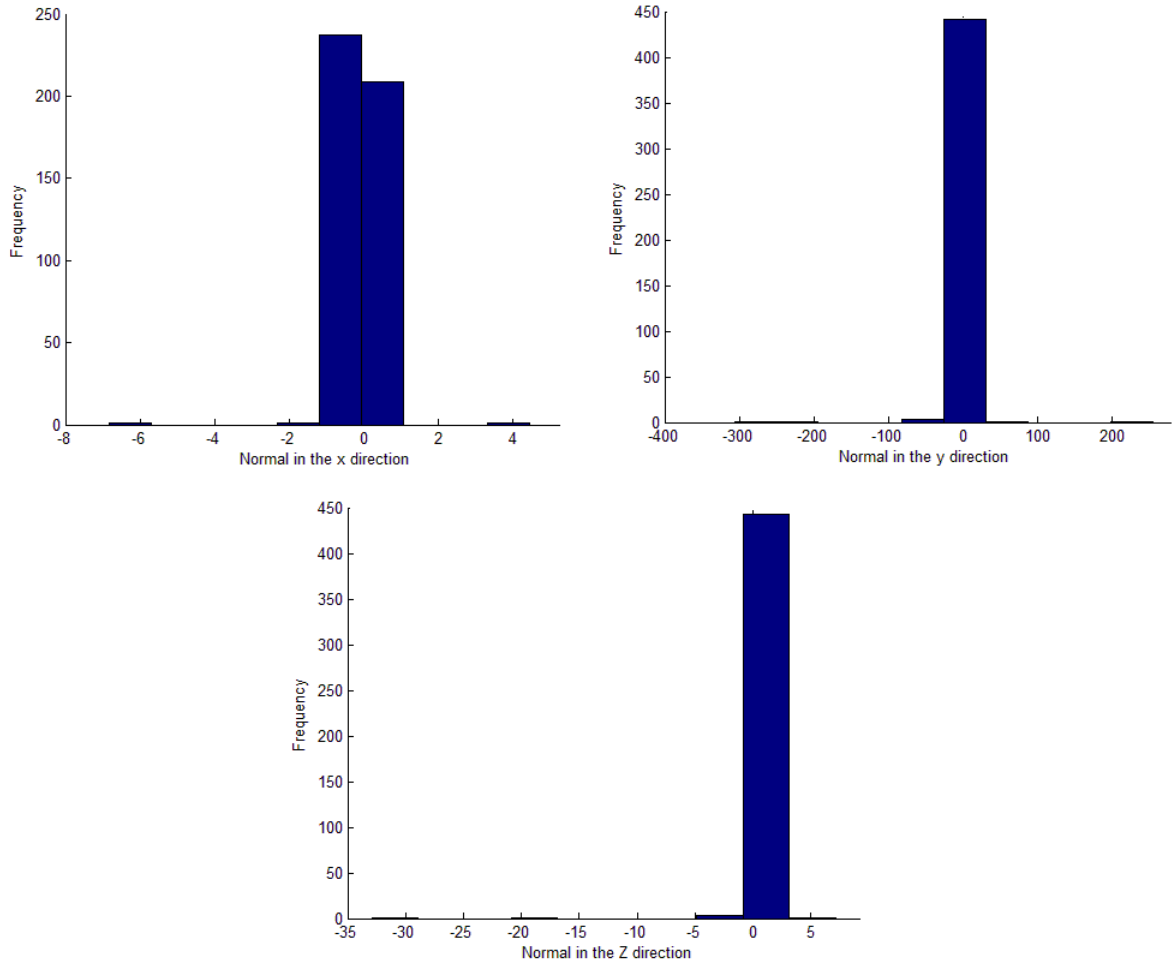


Figure 2.8: Histogram of the surface normal of a building cluster(*a*) histogram of surface normal of building in the x direction; (*b*) histogram of surface normal of building in the y direction, and (*c*) histogram of surface normal of building in the z direction.

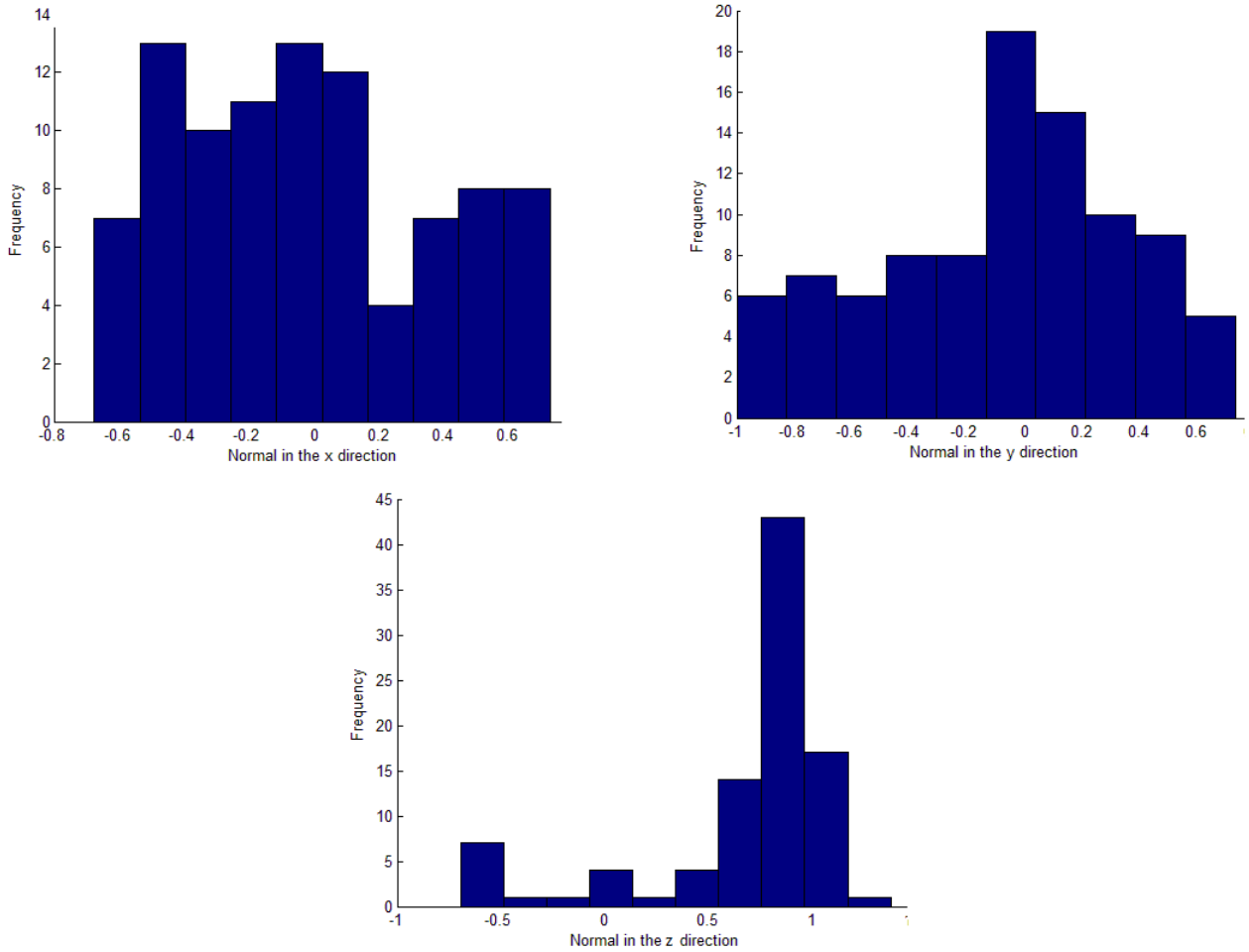


Figure 2.9: Histogram of the surface normal of a tree cluster (*a*) histogram of surface normal of tree in the x direction, (*b*) histogram of surface normal of tree in the y direction, and (*c*) histogram of surface normal of tree in the z direction.

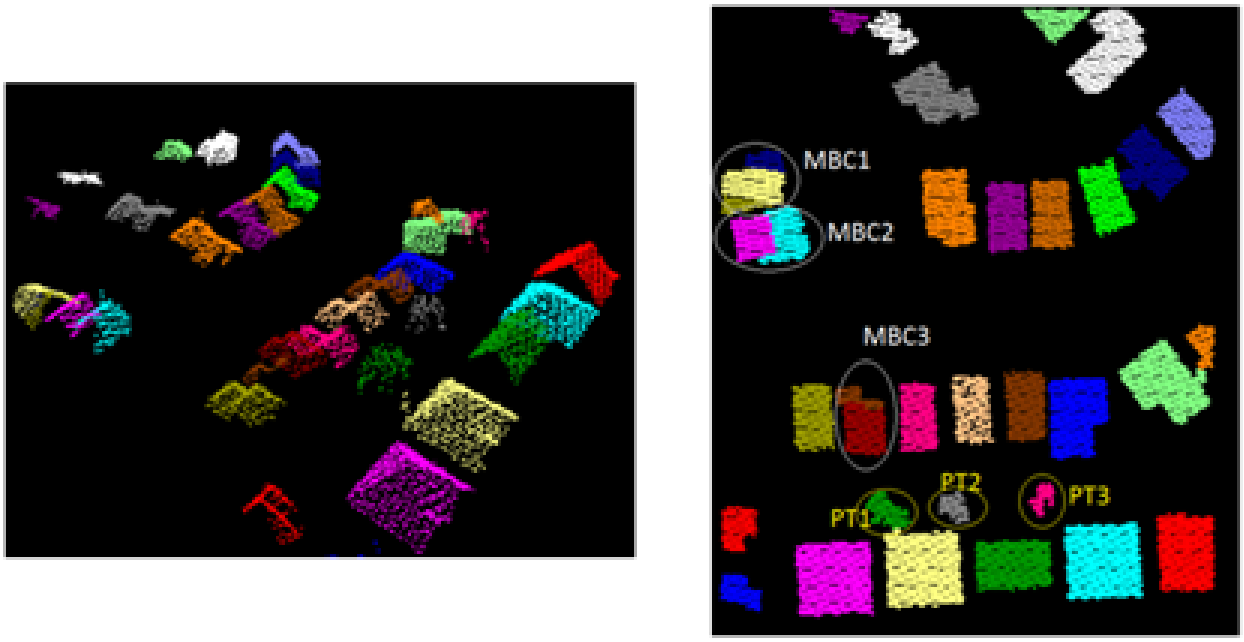


Figure 2.10: Result of Euclidean distance based segmentation: (*a*) top view of building clusters for dataset 1, (*b*) building clusters for dataset 1.

Table 2.3: Results of Euclidean distance based segmentation.

Parameters / dataset	**Dataset 1**	**Dataset 2**
Cluster tolerance (d_{th})	1 m	0.75 m
Min/max number of points in the cluster	50 - 5000	200 - 1200
Non-ground points	11058	35720
No. of clusters	35	24
No. of buildings in the survey area	25	25
No. of buildings identified in the survey area	25	24

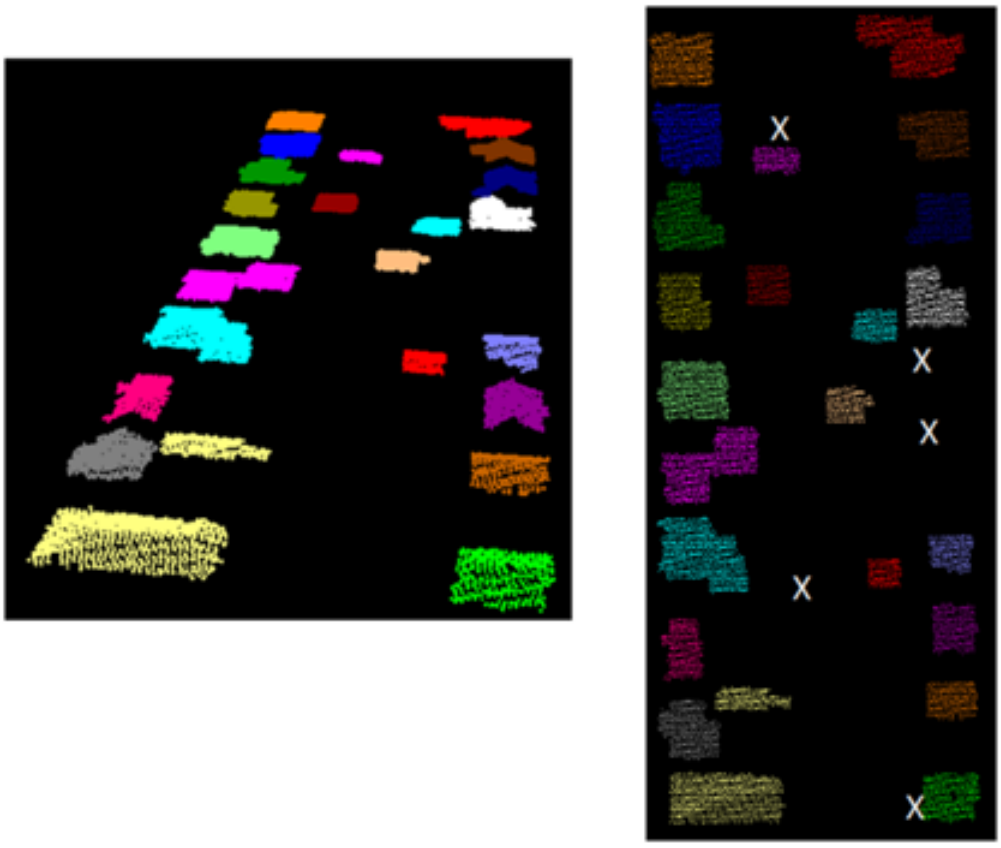


Figure 2.11: Result of Euclidean distance based segmentation: (*a*) top view of building clusters for dataset 2, (*b*) building clusters for dataset 2.

Table 2.4: Comparison of the buildings extracted by the proposed Euclidean distance based segmentation based methodology with the Terrasolid based results for dataset 1.

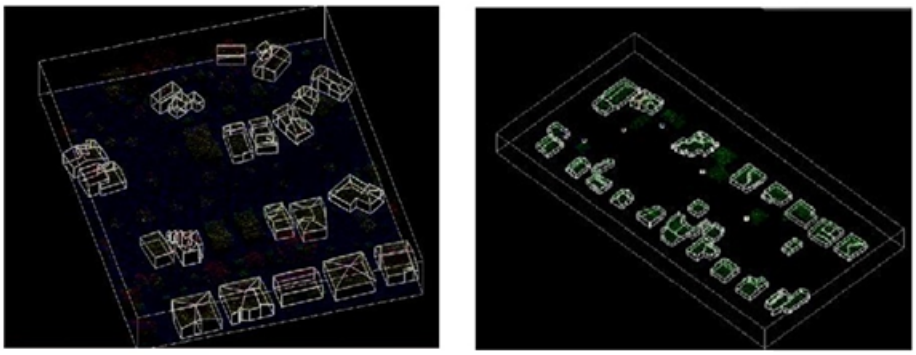
Parameters	**By Euclidean cluster based segmentation**	**Using commercial software**
Non-ground points	11016	11460
Building points	6433	5191
No. of buildings in the surveyed area	25	25
No. of buildings identified	25	19
No. of buildings modelled	24	25
False positives	0	0
True negatives	0	0
Modelling accuracy (%)	100	86.3

Table 2.5: Comparison of the buildings extracted by the proposed Euclidean distance based segmentation based methodology with the Terrasolid based results for dataset 2.

Parameters/ Method	**By Euclidean cluster based segmentation**	**Using commercial software**
Non-ground points	35720	38206
Building points	13676	15966
No. of buildings in the surveyed area	28	28
No. of buildings identified	23	25
No. of buildings modelled	23	20
False positives	0	0
True negatives	5	5
Modelling accuracy (%)	82	89.2

2.5.7 Comparison with the buildings identified using a commercial software

Using Terrasolid, for the dataset 1, out of the total 33703 points, 22243 points are classified as non-ground points. For dataset 2, out of the total 108198 points, 69992 points are classified as ground points. Building detection algorithm in the Terrasolid software is based on the fitting plane to a set of points which are initially identified based on the holes created identified in the ground points as discussed in the section 2.5.4. User defined parameters such as the smallest size of the building footprint, tolerance value, i.e., the minimum elevation difference of a point from the plane fitted, determine the building points. In this study, the parameters were set based on heuristic method. The results of building modelling using Terrasolid are given in Figure 2.12.



(*a*) (*b*)

Figure 2.12: Result of buildings detection using Terrasolid: (*a*) dataset 1, (*b*) dataset 2.

A comparison of the results of the buildings modelled using Euclidean distance based segmentation algorithm in the open source library PCL and Terrascan software algorithm is given in Tables 2.4 and 2.5. As evident from these Tables, the performance of buildings extraction is comparable. In the dataset 1, all the 25 buildings are identified by the Euclidean distance based segmentation algorithm whereas the commercial software is able to identify only 19 buildings. This can be attributed to the failure in the detection of planar patches amongst the detected building points. In the dataset 2, the commercial software outperformed the Euclidean distance based segmentation in identifying the buildings. Five building clusters marked as *X* are not identified as building points. The missing building patches are those which have less number

of points than the minimum threshold value in the Euclidean distance based segmentation. The minimum cluster value was set to avoid over segmentation of the building objects. From the experiments, it is evident that the choice of the user defined parameters influences substantially the building clusters detected.

In spite of these limitations, it is evident that the buildings clusters detected using the proposed segmentation based method in open source implementation is comparable with the commercial software. This observation also signifies the functional utility of the open source point cloud library to effectively implement buildings detection and modelling using LiDAR point data, besides the affordability. More studies are required for automatically determining the user defined parameters based on point density of the point cloud for improving the building detection.

2.5.8 Conclusion

This study presents a novel data driven methodology for buildings detection using the open source point cloud library. The point cloud was segmented in 3D using Euclidean distance based segmentation methods. From the segments created, building clusters are separated from non-building clusters by a novel histogram based method. The performance of the proposed methodology has been compared with the automatic building detection routine available in a popular commercial LiDAR data processing software (Terrasolid). The performance of the proposed methodology is comparable that of the commercial version. Overall, their accuracies are 100%, 82% for dataset 1 and 86.3%, 89.2% for dataset 2 respectively from the two different implementations (segmentation-based approach and automatic building detection routine in Terrasolid). A major limitation of the study is that analyst needs to specify the minimum and maximum number of points in a cluster. This makes the procedure a semi automatic process.

2.6 Chapter Summary

Object-based 3D semantic labelling is an emerging methodological approach for assigning meaningful label to each and every point in the massive LiDAR point cloud data. This chapter

has presented a background on the object-based labelling approach and a brief mathematical description of the different stages of object-based labelling approach. Also the LiDAR point cloud pre-processing concepts which are used in the rest of this thesis are discussed. Finally an experimental study assessing the advantage of object-based approach for building detection over point-based labelling approach is presented.

CHAPTER 3

INTEGRATION OF SPECTRAL AND GEOMETRICAL INFORMATION FOR OBJECT-BASED SEMANTIC LABELLING

Prelude: This chapter proposes an object-based point cloud labelling technique for semantic labelling of LiDAR point cloud integrating the geometrical information from LiDAR and spectral information from multispectral images captured over an urban scene. Initial object primitives are created using a colour-based region growing technique. Multiple classifier system is then applied on the features extracted from the segments for classification and also for reducing the subjectivity involved in the selection of classifier and improving the precision of the results. Experiments carried out on three airborne LiDAR datasets show that the proposed technique successfully discriminates urban land covers and identify urban buildings.[1]

3.1 Introduction

Multi-sensor fusion technique has been a very active research area in recent years wherein the complementary characteristics of various sensors are exploited for improved object detection. Owing to the diverse applications it can cater to, the LiDAR and multispectral images are potential pairs of sensors for multi-sensor fusion. LiDAR which captures rich geometrical information from the earth surface is suitable for generating three-dimensional surface models. Additionally, the sensor also records the the backscattered intensity value (usually in the near-infrared region for topographic applications). However, since it operates in a monochromatic wavelength, it cannot record the spectral reflectance in multiple wavelengths useful for discriminating object types as in the case of the multispectral sensors. It is a well established fact

[1]The major contents of this chapter is published in *Geocarto International* Vol: 31(2), 2016 Publisher: Taylor & Francis. Authors: Anandakumar M Ramiya, Rama Rao Nidamanuri, Ramakrishnan Krishnan.

that the spectrally rich data available in multispectral images are very useful in discriminating various earth surface objects. The distinct characteristic of LiDAR and multispectral images make them preferable for many applications and studies have revealed that the fusion of these two sensors improves the object detection than what can be achieved with a single sensor alone (Sohn and Dowman, 2007; Demir and Baltsavias, 2012). Recent advances in sensor/surveying technology allow simultaneous acquisition of LiDAR point cloud, and multispectral images enabling generation of a co-registered dataset.

Numerous studies have been carried out using LiDAR point cloud for urban land cover mapping and buildings detection. High resolution surface height information from the LiDAR in the form of a digital surface model (DSM) derived from LiDAR point cloud has been used as ancillary data (Zhou et al., 2004; Matikainen et al., 2007; Sohn and Dowman, 2007; Lee et al., 2008; Demir and Baltsavias, 2012) or as the primary data for classification (Ma, 2005; Madhavan et al., 2006; Liu, 2008; Chen et al., 2012).

Recent studies attempt classifying the LiDAR point cloud without converting the point cloud into a raster DSM (Rottensteiner, 2003; Sampath and Shan, 2010; Chen et al., 2012). This preserves the inherent geometric characteristics of the LiDAR point cloud and also reduces the error in the final classification accuracy (Golovinskiy et al., 2009; Zhang et al., 2013). The LiDAR points are treated as un-ordered listings of measurements without any particular spatial arrangement. Similar to raster-based image classification, there are two ways by which LiDAR point cloud can be classified - pixel(point) -based and object-based labelling. In point-based classification, points are classified by using the few features inherent to the point cloud such as distance from the ground, planarity. Recently, object-based point cloud analysis is gaining momentum as it produces more realistic results. It has been established that the rich spatial and geometric information associated with high resolution LiDAR point cloud is valuable for accurate urban scene interpretation. Recent studies indicate that object-based point cloud (OBPC) labelling approaches have the capability to exploit spatial dependencies of LiDAR point cloud and offer improved results (Zhang et al., 2013; Golovinskiy et al., 2009). In object-based point cloud labelling, object primitives are obtained from the raw point cloud which are later semantically labelled based on the extracted features from the objects. The methods are discussed in section 2.2.

Most of the existing studies rely only the range information from the LiDAR point cloud for the segmentation process. However, as stated above, it is well known that multispectral data can discriminate various materials based on its spectral reflectance characteristics. Thus, combining geometric properties from LiDAR based range information and spectral properties from multispectral images could complement each other, thereby offering enhanced performance in the detection and identification of urban land features such as buildings, shrubs, trees (Zhang et al., 2013).

Apart from the quality of objects from the segmentation, the classification algorithm (classifier) used for labelling of the objects plays a key role in the accuracy of urban objects detection. Our literature survey indicates that the studies which use LiDAR point cloud for urban buildings detection have used a single classifier (Sun and Salvaggio, 2013; Moussa and El-Sheimy, 2012; Zhang et al., 2013). As the performance of a classifier is data and site specific, it has been a recurrent task to identify beforehand the classifier appropriate for the task. Developments in the pattern recognition theory offer a solution 'multiple classifier system (or ensemble of classifiers)' in which several classifiers are applied on the dataset simultaneously and the intermediate outputs of the classifiers are combined mathematically to produce a single output, which is better than any of the individual outputs (Fauvel et al., 2006; Benediktsson et al., 2007; Du et al., 2012; Damodaran and Nidamanuri, 2014). Recent developments in image classification show the potential of multiple classifier system for remote sensing image classification.

This part of the work proposes a novel object-based methodology for urban buildings detection and reconstruction by combining LiDAR and multispectral images. The proposed methodology produces near true geometry output containing buildings as three dimensional polygons, akin to buildings' footprints. The colour-based region growing segmentation algorithm was used for segmenting the coloured point cloud. The coloured point cloud contains spectral details from multispectral images acquired over the same area along with the LiDAR point cloud. Several features were extracted from the three dimensional segments created from the coloured point cloud. The extracted features were then classified using multiple classifier system (MCS). The proposed methodology has been implemented in C++ with open source point cloud library for ease of sharing and compatibility.

3.2 Land Cover Complexity Index

Despite the availability of very high resolution multispectral images and high density point cloud, the spatial setting, heterogeneity, and morphological profiles of landscape have significant bearing on the accuracy of remote sensing based mapping and modelling. The occurrence, distribution, and areal extent of several different classes in a relatively small area (compared to spatial resolution of remote sensing data) increase the within-class heterogeneity resulting in spatial clutter and hence leading to lower classification accuracy. Several indicators of landscape dynamics available, which quantify the patchiness of land covers (Honnay et al., 2003), are not directly applicable for pixel / point level diversity assessment in a digital raster classified image. Therefore, a new indicator for quantifying the spatial complexity at pixel level is proposed for assessment of the role of spatial complexity in the classification performance. Let **I** be a classified image with dimensions $m \times n$ and let L denote the class number (numeric equivalent of class label) and *N*, the total number of classes. A local neighbourhood template $k(i, j)$ operating over the neighbourhood of each classified pixel is defined as:

$$k(i,j) = \frac{(L_{i+1} - L_{i-1}) + (L_{j+1} - L_{j-1})}{4} \tag{3.1}$$

The pixel value at the centre of the template indicates the heterogeneity of the land cover classes. The value of the template will be zero for homogeneous landscape and non-zero for non-homogeneous landscape. However, the values of the template $k(i, j)$ in the heterogeneous region are influenced by the class numbers assigned. To remove this ambiguity, another function $f(i, j)$ defined below is operated upon the output generated from the above procedure.

$$f(i,j) = \begin{cases} 0, & k(i,j) = 0 \\ 1, & k(i,j) \neq 0 \end{cases} \tag{3.2}$$

The frequency of non-zero elements in the resultant matrix indicates the spatial land cover

complexity. We name this index as land cover complexity index (LCCI) and expressed as

$$\mathrm{LCCI} = \frac{\sum_m^x \sum_n^y f(i,j)}{mn} \tag{3.3}$$

For a homogeneous landscape with a single class, the value of LCCI is zero. A region would be at its lowest spatial complexity if it is composed of only a single land cover type, thus leading to the corresponding remote sensing datasets at the lowest spatial complexity. The spatial complexity increases if the number of different types of classes increases in a region. Hence the value of LCCI increases proportionately. While the computation of LCCI measure does not form part of the quality evaluation metrics of image or point cloud classification per se, it is useful in explaining the site-specific performance of classification methodologies.

3.3 Dataset Used

The dataset used for the study was provided by the ISPRS as a benchmark dataset as part of the ISPRS Test Project on Urban classification and 3D Building Reconstruction. [2]

Vaihingen dataset The dataset was collected over the city of Vaihingen, Germany during August 2008 and was provided to us by the ISPRS as part of the ISPRS test project on urban classification and 3D building reconstruction (Cramer, 2010). The LiDAR data was collected using Leica ALS50 from a flying altitude of 500 m above the ground. There was a simultaneous capture of the spectral information using Integraph / ZI DMC. The multispectral images, captured in the green, red and near infrared regions parts of the electromagnetic spectrum, have a spatial resolution of 5 cm and radiometric resolution of 11 bits.

Three subsets of LiDAR point cloud covering about 170 m^2 area each with different urban complexities were extracted from the dataset (Fig. 3.1). Study area 1 and study area 2 have

[2]The Vaihingen data set was provided by the German Society for Photogrammetry, Remote Sensing and Geoinformation (DGPF) (Cramer, 2010):http://www.ifp.uni-stuttgart.de /dgpf/DKEP-Allg.html. The data was provided as part of ISPRS WG/III urban classification and 3D Building Reconstruction and we thank the group for evaluating our results.

Figure 3.1: Aerial photographs of the study area with NIR band mapped to red, red band mapped to green and green band mapped to blue to create a false colour composite (*a*) study area 1 : inner city, (*b*) study area 2 : high riser, (*c*) study area 3: residential area

a point density of 0.5 points/m^2. Study area 1 primarily covers the city center consisting of dense complex buildings with some trees, while study area 2 is a mixed urban area comprising of few high rising residential buildings surrounded by trees.

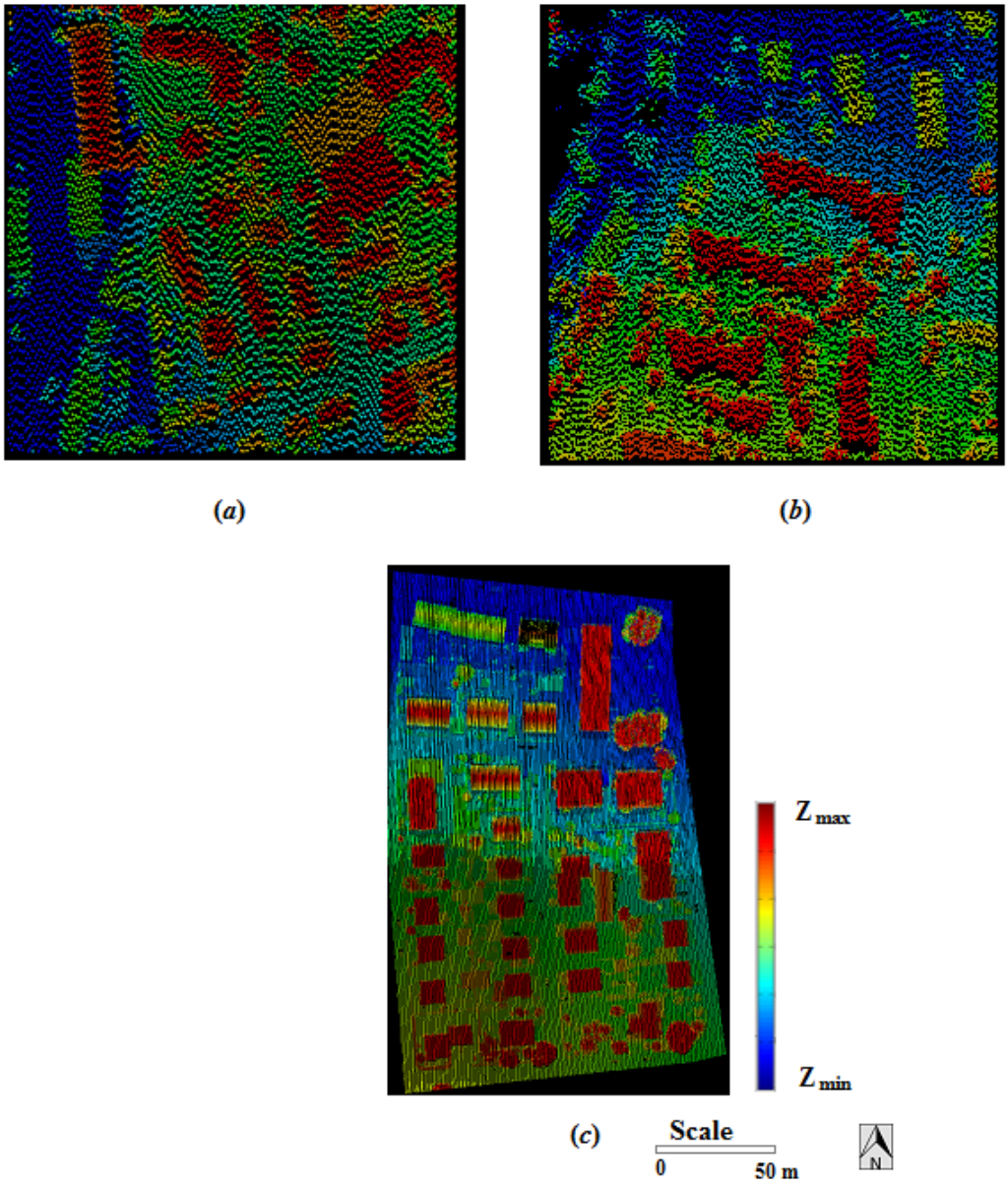


Figure 3.2: LiDAR point cloud coloured based on elevation (Z) (a) study area 1 : inner city, (b) study area 2 : high riser, (c) study area 3: residential area.

Study area 3 with a higher point density of around 5.5 points/m^2 is purely a residential area with detached houses. ISPRS reference results indicate that the number of large buildings (ie.,

buildings greater than 2.5 m^2) in these three areas is 38, 15, and 57 respectively.

Table 3.1: Details of study areas from Vaihingen dataset

Parameters / dataset	**Study area 1**	**Study area 2**	**Study area 3**
Point density (points /m^2)	0.5	0.5	5.5
Number of 1st return	10084	14779	230171
Number of 2nd return	156	483	0
Number of 3rd return	2	20	0
Total number of points	10242	15282	230171
x range (m)	170	200	151
y range (m)	170	200	233
Land cover complexity index	0.16	0.14	0.09

The land cover complexity index was calculated for the three datasets based on the procedure discussed in the section 3.2. The LCCI index for the study areas are 0.16, 0.14, and 0.09 respectively. The index correlates with the complexity of the dataset discussed earlier in this section. The details of the Vaihingen dataset is given in Table 3.1.

3.4 Methodology

The goal of this part of the research work is to automatically assign a label c_i to each of the 3D points i. The methodology adopted in this research is shown in Figure 3.3. The point clouds are assigned to one of the five classes: pavement/road, lawn, flat roof building, gable roof building, and shrubs/trees. The detailed steps are explained in the following sections.

3.4.1 Coloured LiDAR point cloud

Combined use of spatial information from LiDAR point cloud with spectral information from multispectral images can improve the accuracy of point cloud labelling. An orthoimage was

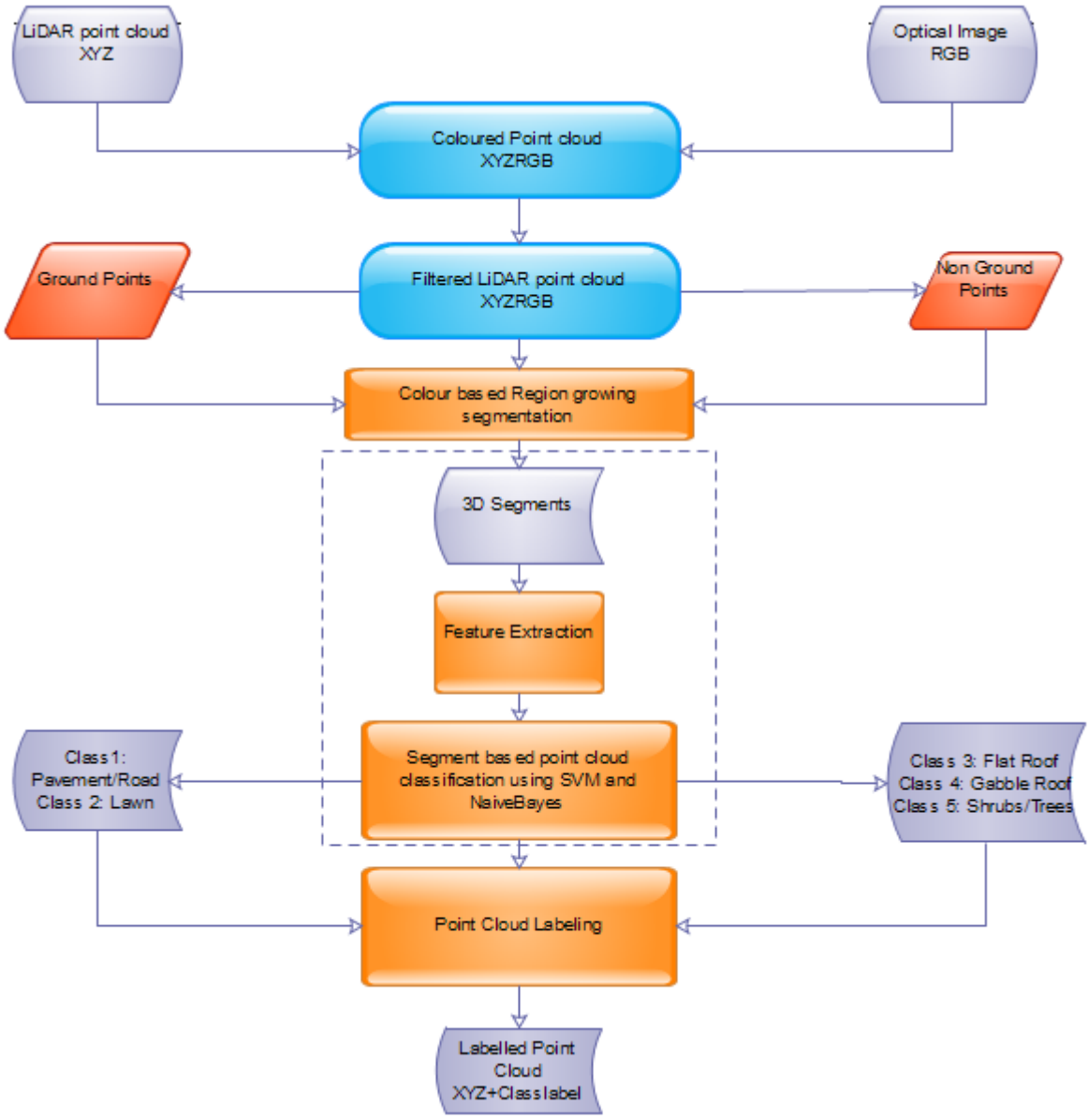


Figure 3.3: Methodology depicting the object-based point cloud labelling in the study

generated from the aerial multispectral images using the interior and exterior orientation parameters of the images. To produce a coloured point cloud data, the colour information (spectral values) from the orthoimage was extracted and embedded with the X, Y, Z information from the LiDAR point cloud to form an XYZRGB dataset[3]. The datasets were referenced to the same coordinate system and of comparable resolutions and hence was used to produce coloured point cloud. The coloured point cloud was produced using Lastools, a highly-efficient scriptable tools for LiDAR processing (Isenburg, 2012).

3.4.2 Filtering

To reduce the complexity of classification, the LiDAR points were separated into ground and non-ground points before further processing. This step is essential as the first two classes ie., pavement / road, and lawn belong to ground class, whereas the remaining three classes belong to the non-ground class.

In this study, the progressive densification algorithm (Axelsson, 2000a) was used. An initial TIN (Triangulated Irregular Network) surface was generated from a set of seed points chosen based on the statistical information derived from the LiDAR point cloud. The initial TIN surface was densified iteratively using the threshold that was obtained based on two parameters: angle to the nodes and the distance from the nodes.

3.4.3 Segmentation

The coloured point cloud was segmented using colour-based region growing segmentation method (Zhana et al., 2009). The segmentation algorithm has two stages: region growing, and region merging and refinement.

Since the LiDAR point cloud captured is stored in an unorganized / random fashion, it is required to establish neighbourhood relation amongst the points in the LiDAR point cloud. The

[3]Throughout the thesis, XYZRGB is used to denote the coloured point cloud data. Here, RGB corresponds to the false colour composite created using reflectance in the near-infrared, red and green bands captured by the aerial multispectral camera.

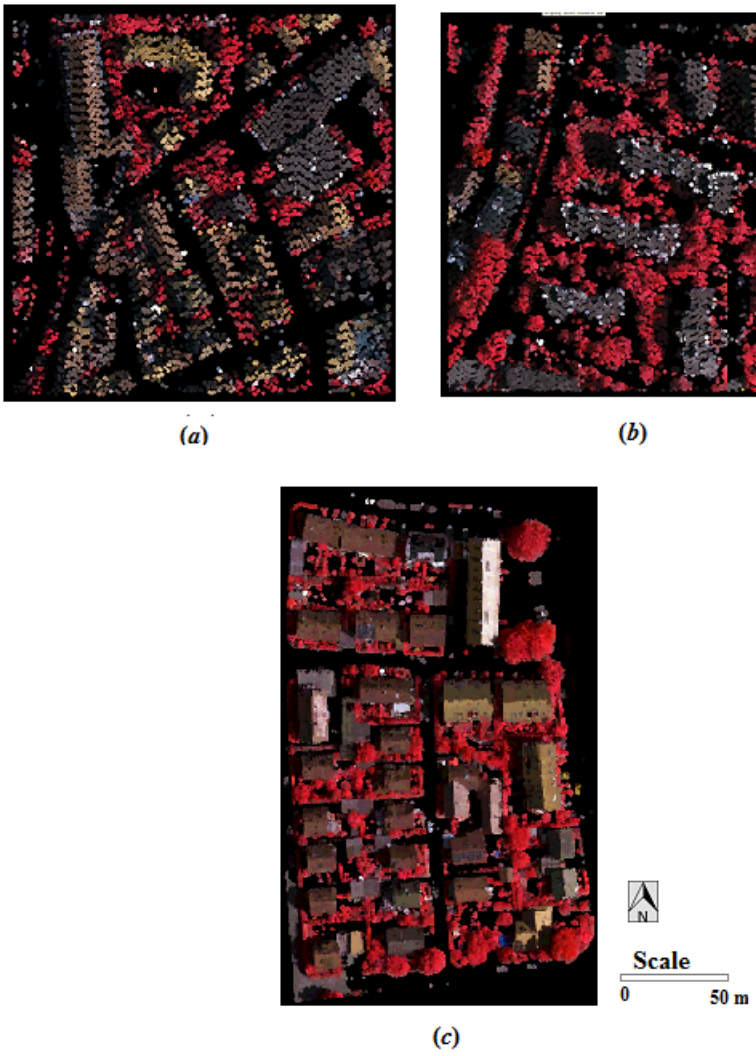


Figure 3.4: LiDAR point cloud coloured based on spectral values derived from multi-spectral image with NIR band mapped to red, red band mapped to green and green band mapped to blue to create a false colour composite (*a*) study area 1 : inner city, (*b*) study area 2 : high riser, and (*c*) study area 3: residential area

k-d (*k*-dimensional) tree algorithm (Arya et al., 1998), a space partitioning algorithm, was used to establish neighbourhood in the LiDAR point cloud.

Region growing

Initially, seed points were selected from the available points, *P*, by fitting a normal to the set of neighbouring points determined by *k*-NN (*k*-Nearest Neighbours). *k*-NN neighbours is preferred to fixed distance neighbours (FDN) based on fixed AOI (area of interest), since it is an adaptive method and neighbourhood is adjusted based on local point density. Hence, it is suitable for datasets with varying point density. The method also avoids degenerate cases (e.g. points having no neighbours). The residual of the point to the normal plane was calculated which indicate the curvature of the points. Points with minimum curvature were chosen as the seed points.

The algorithm then selects the points which are within the *k*-NN region of the seed point. This ensures local connectivity between the points. The points in the neighbourhood were also checked for local smoothness, where the residual of the plane of best fit of the points is within a defined threshold. The colorimetric distance (CD) defined by Equation 3.4 was computed between the seed points and the neighbouring points.

$$CD(C_1, C_2) = \sqrt{(R_1 - R_2)^2 + (B_1 - B_2)^2 + (G_1 - G_2)^2} \tag{3.4}$$

where R,G,B denote the spectral values obtained from the multispectral image.

If the distance computed is within a threshold, it was added to region *R*. The process continues until all the points in the set *P* are labelled to a region. This resulted in regions R_i, where i = 1,2,...n.

Region merging and refinement

In the region merging step, the roughly segmented regions $\boldsymbol{R_i}$ from the region growing process were checked for colorimetric similarity using the equation 3.4. The regions were merged if

the colorimetric distance is less and if it satisfies the local connectivity based on *k*-Nearest Neighbourhood.

Finally, a refinement step was carried out to merge the regions having less number of points than acceptable threshold values within its neighbours. The threshold values used in this algorithm depend on the point density of the dataset. In this study, a distance threshold of 10 m, point colour threshold of 6, and region colour threshold of 5 were used. This results in homogeneous 3D segments which were used in the further stages of object-based point cloud labelling.

3.4.4 Feature extraction

Since the coloured point cloud used in this work are rich both spectrally and geometrically, features were selected such that both the domains are well represented. The spectral features used in our study include mean spectral values of the different bands corresponding to the study area. In addition to this, for the study areas $1, 2,$ and 3, for which the spectral reflectance in the near infrared region is also available, NDVI (Normalised Difference Vegetation Index), an important index useful in discrimination of vegetation class from the other classes, was also computed.

Local saliency features based on the distribution of points within a cluster form a set of important geometrical features (Lalonde et al., 2006). Mathematically defining the distribution of points within a cluster was used to define the geometrical features. In this study, the eigen values and eigen vectors are used to define the geometry. The covariance matrix was computed for each cluster to compute the principal components. The principal components were decomposed to derive the eigen values $e_0, e_1,$ and e_2 and their corresponding eigen vector $\boldsymbol{\lambda}_0, \boldsymbol{\lambda}_1,$ and $\boldsymbol{\lambda}_2$ (in the decreasing order). The physical representation of the values is illustrated in Figure 3.5.

These parameters were used to compute three crucial geometric features - pointedness,

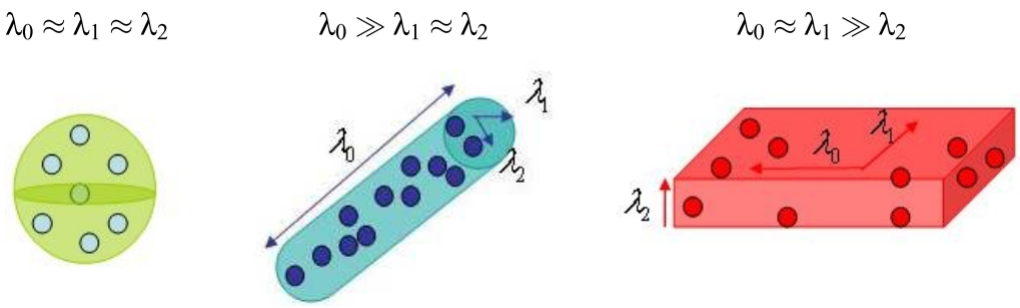


Figure 3.5: Illustration of local saliency features (Lalonde et al., 2006)

curveness, and surfaceness as given in equation 3.5.

$$\text{Pointedness} = \frac{\boldsymbol{\lambda}_2}{\boldsymbol{\lambda}_0} \tag{3.5a}$$

$$\text{Curveness} = \frac{\boldsymbol{\lambda}_0 - \boldsymbol{\lambda}_1}{\boldsymbol{\lambda}_0} \tag{3.5b}$$

$$\text{Surfaceness} = \frac{\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2}{\boldsymbol{\lambda}_0} \tag{3.5c}$$

Apart from this, the range of *Z*, which primarily indicates absolute elevation of objects within a cluster, the area and volume of the hull enclosed by the set of points were also used as the geometrical features.

3.4.5 Classification

The 3D segments created in the segmentation process are non-overlapping three-dimensional objects mathematically and the points in each 3D segment are not linked together. It is assumed that the segments contain homogeneous classes and all points within the segments belong to the same class. A supervised classification method was used for identifying and assigning the labels for each of the 3D segments. To reduce the complex heuristics involved in the selection of appropriate classifier and for improving the accuracy and the reliability of classification, a non-parametric multiple classification system (MCS) was developed for classification

of the points within the 3D segments. Three popular machine learning algorithms techniques: *k*-Nearest Neighbours classifier (*k*-NN), Naive Bayes (NB) classifier, and Support Vector Machines (SVM) were used as base classifiers. These classification algorithms are discussed in the section 2.3.3.

MCS combines the decision function values of many base classifiers to arrive at a decision label of higher classification accuracy. Let $C = C_m \forall$ m, where m = 1, 2...m represent the set of classifiers. Each classifier in the set C will produce labelling output l_m. A combination function, mathematical function/scheme, which combines the intermediate decision is used to get the final classified result. In this study, we have used the 'maximum voting approach', a non-trainable classification function to label the classes (Benediktsson et al., 2007; Du et al., 2012). Each 3D segment s_k was initially assigned a class label based on the base classifiers. The resultant labelling outputs of each segment were combined to arrive at the final decision. This method ensures more reliable classification labelling. The points in the dataset were classified into one of the following classes: 1. pavement / road, 2. lawn, 3. flat roof, 4. gabled roof, and 5. shrubs / trees.

3.4.6 Accuracy assessment

The performance of the methodology developed was critically examined and validated at two different stages: semantic labelling of the points, and using the 2D building mask. Further, the results were validated independently by the International Society for Photogrammetry and Remote Sensing (ISPRS) (ISPRS WG III/4) as part of the 'ISPRS Test Project on Urban Classification and 3D Building Reconstruction'.

In addition, the accuracy of the results obtained from the methodology was assessed by computing a confusion matrix $\boldsymbol{p}$ with $\boldsymbol{k}$ classes, based on the ground truth map prepared by complete enumeration. To assess whether the classification was due to chance, the popular statistical measure Kappa coefficient (κ) was used. Kappa coefficient is calculated using the equation

$$\kappa = \frac{p_0 - p_e}{1 - p_e}, \tag{3.6}$$

where p_0 is the sum of the relative frequency in the diagonal of confusion matrix (proportion of true agreement) and p_e indicates proportion of agreement expected by chance. The value of κ varies between 0 and 1. Value of κ between 0.6 and 0.8 indicates substantial agreement between the classification and ground truth data whereas κ above 0.8 and 1 indicates almost perfect agreement.

Statistical significance

Further, the statistical significance of the classifiers can be assessed using z-test (Congalton et al., 1983). The variance for the kappa statistic was computed using the formula given below

$$var(\kappa) = \frac{1}{N(1-p_c)^4}\{\sum_{i=1}^{k} p_{ii}[(1-p_o) - (p_{.i}+p_{i.})(1-p_o)]^2 + (1-p_o)^2 \sum_{\substack{i=1 \\ i\neq j}}^{k}\sum_{j=1}^{k} p_{ij}(p_{.i}+p_{j.})^2 - (p_o p_c - 2p_c + p_o)^2\} \tag{3.7}$$

The statistical significance of the kappa coefficient is then computed to assess whether the classification result is significantly better than the random result. The test statistic for a testing a single confusion matrix is given by z-score(z) given by the following equation

$$z = \frac{\kappa}{\sqrt{var(\kappa)}} \tag{3.8}$$

.

Based on the z-score, the hypothesis was laid out.

Null hypothesis : H_0 the observation results are purely by chance
Alternative hypothesis : H_1 the observation results are by non-random causes

The standardised and normally-distributed z indicates how the variance are distributed from the normal. At 95% confidence level, the standard normal critical value is given by z_c= 1.96, which is the number of standard distributions from the mean of z-distribution. When $z > z_c$,

we reject null hypothesis H_0 and accept H_1. When $z < z_c$, we accept null hypothesis H_0.

Validation of building footprint

From the labelled point cloud, the building points corresponding to flat, and gabled roof surface were extracted. The extracted building points were then projected onto the 2D surface to create a 2D building mask (also known as building footprint). A grid with pixel size of 10 cm was generated. Due to the discrete sampling pattern of the LiDAR scanning and the possibility of no returns, there would be voids (holes) in the LiDAR point cloud for surface features such as buildings making gaps in the building mask. In order to make the building mask complete and fill the gaps, the 'dilation' morphological operator was applied on the 2D building mask (Serra, 1986).

The accuracy of building mask was independently evaluated by the International Society of Photogrammetry and Remote Sensing (ISPRS WG III/4) using the ISPRS benchmark datasets for study areas 1, 2, and 3. As buildings are the prominent features in the study areas, building detection based evaluation was carried out. From the labelled point cloud dataset, the points corresponding to flat and gabled roof were extracted. For evaluation, a 2D raster grid of 10 cm resolution was generated by projecting the building points.

The evaluation procedure described by Rutzinger et al. (2009) was used to assess the building detection at three levels: (a) pixel-level, (b) object-level, and (c) objects greater than 50 m^2 - large objects. At each level, the correctness, and completeness were computed based on the equations,

$$\text{Correctness/Precision} = \frac{(\text{TP})}{(\text{TP}) + (\text{FP})} \tag{3.9a}$$

$$\text{Completeness/Recall} = \frac{(\text{TP})}{(\text{TP}) + (\text{FN})} \tag{3.9b}$$

where TP(true positive) corresponds to the points which were correctly labelled with reference to the ground truth, FP (false positives) corresponds to the points which are incorrectly labelled as belonging to a particular class. FN (false negative) corresponds to the points which do

not belong to the positive class but should have been labelled. Pixel based evaluation was performed on the raster image. A point-in-polygon test based on the vector representation of the buildings was carried out for the object-based evaluation. The geometrical accuracy of the boundary polygons was also computed. Root mean square error was used as a measure to assess the geometrical accuracy.

3.5 Results and Discussion

Results from the application of the developed prototype on the three different datasets are presented in this section following the sequence of the processes involved.

3.5.1 Filtering

Study area 1 contained a total of 10242 points. After applying the filtering algorithm there are 3806 ground points and 6436 non-ground points. Study area 2 has 15282 points of which are were 5887 ground, and 9395 non-ground points. Study area 3 has 230171 points, of which 122937 are non-ground points and the rest are ground points. The results of filtering are shown in Figure 3.6.

3.5.2 Segmentation

The results of segmentation of the filtered LiDAR point cloud, integrated with spectral data, are shown in Figure 3.7. The clusters are displayed in separate colours for ease of visualization. From the entire point cloud of study area 1, the segmentation algorithm generated 32 clusters, of which, 11 belong to ground points and the rest to the non-ground points. For the study area 2, the segmentation algorithm generated 56 clusters, of which 19 belong to ground points, and the rest to the non-ground points. Similarly, for the study area 3, there are 85 clusters from ground points, and 125 clusters from non-ground points. A visual inspection of the results of segmentation indicates that most of the urban features are clustered distinctively.

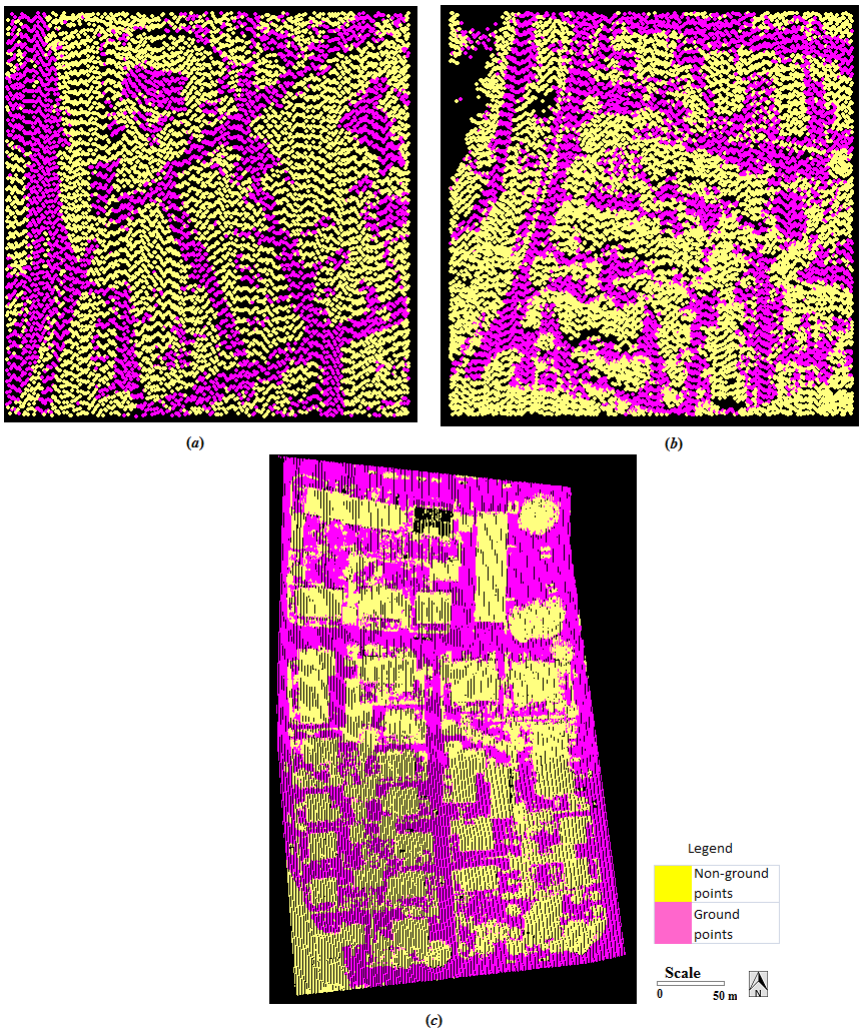


Figure 3.6: LiDAR point cloud after filtering: (*a*) study area 1 : inner city, (*b*) study area 2 : high riser, and (*c*) study area 3: residential area.

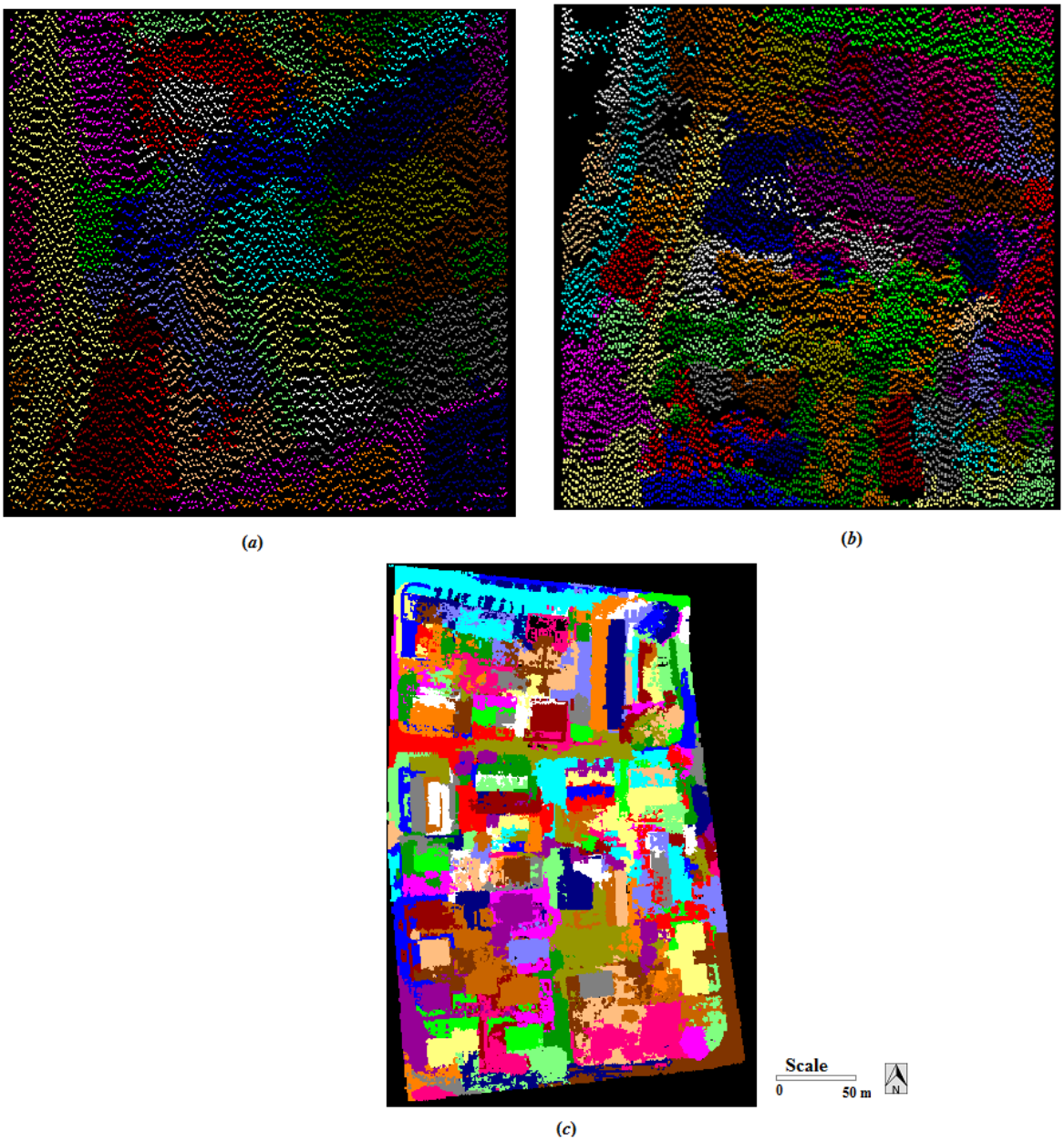


Figure 3.7: Segmented point cloud (different colours represents different segments): (*a*) study area 1 : inner city, (*b*) study area 2 : high riser, and (*c*) study area 3: residential area

3.5.3 Classification

Table 3.2: Performance evaluation of different classifiers. The cases which are statistically significant, based on the z score, are highlighted in bold

Classes / Classifiers		**Naive Bayes**	***k*-NN**	**SVM**	**MCS**
Ground Points	Overall Accuracy (%)	93	97	93	97
	Kappa Coefficient (κ)	0.86	0.92	0.84	0.92
	z-score (z)	**2.82**	**2.82**	**4.42**	**2.83**
Non-Ground Points	Overall Accuracy (%)	79	84	81	84
	Kappa Coefficient (κ)	0.73	0.77	0.71	0.77
	z-score (z)	**3.56**	**3.74**	**4.15**	**3.74**

Each point in the segmented point cloud was assigned a class label based on the labels generated by the different classifiers. The possible urban land covers in all the three datasets are: 1. pavement / road, 2. lawn, 3. flat roof, 4. gabled roof, and 5. shrubs / trees.

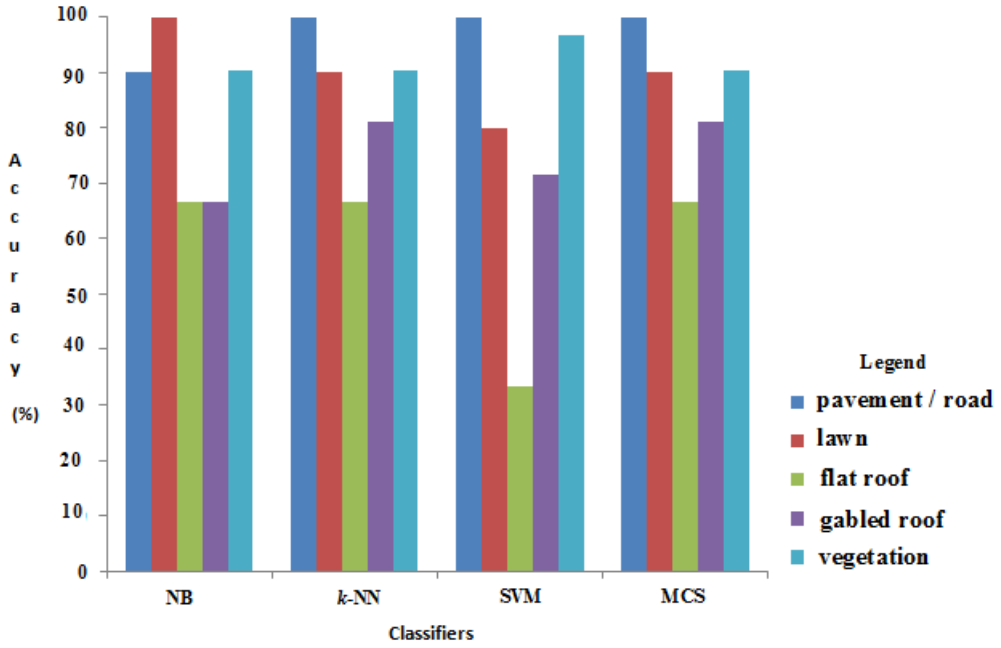


Figure 3.8: Producers accuracy for different urban features

Results of the accuracy assessment of the classified point cloud for the various classifiers are shown in Table 3.2 and Figure 3.8. The corresponding classified images are shown in Figure 3.10.

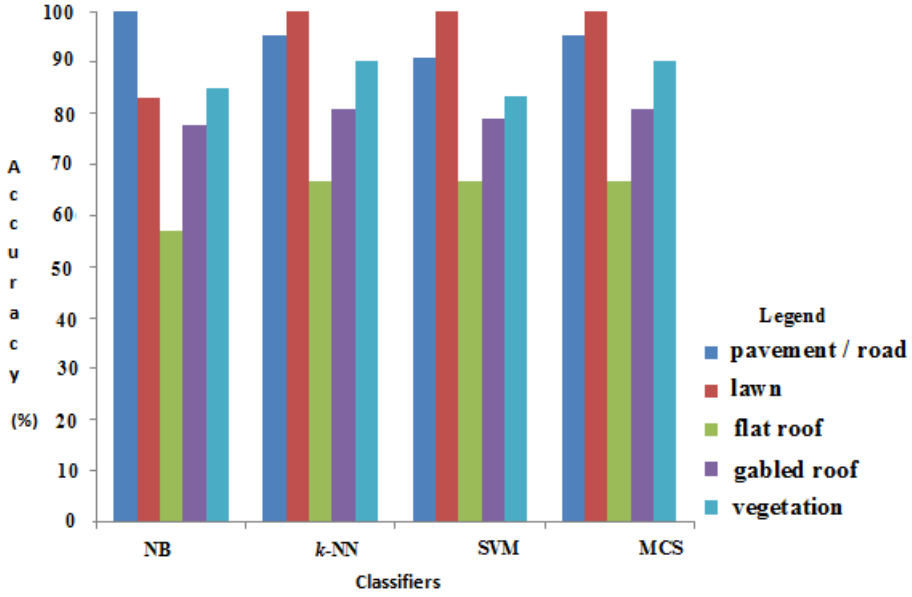


Figure 3.9: Users accuracy for different urban features

From Table 3.2, it can be seen that all the four classifiers perform relatively well in discriminating the ground classes from the non-ground classes. Amongst the three independent classifiers, *k*-NN performed well for both the ground and non-ground classes with 97%, and 84% accuracy respectively. The mean overall accuracy for all the three independent classifiers is approximately 90%, and 80% for the ground and non-ground classes. The ground classes (lawn and the pavement / road) are classified with fairly high accuracy. Amongst the non-ground classes, the vegetation class is discriminated with fairly higher accuracy from the building classes (flat roof buildings and gabled roof buildings). However, there is a relatively lower accuracy for the non-ground classes due to the apparent confusion between the flat roof and gabled roof buildings, as evident from the corresponding producer and user accuracies in Figures 3.8 and 3.9). Results from the multiple classifier system (MCS) indicate marginally higher accuracy when compared to the best performing independent classifier (i.e. *k*-NN). Further, the values of κ exhibit uniformity in the accuracy of the different urban land cover classes indicating a good agreement between classified results and the ground truth data. Also, the z-

score indicates that the values statistically significant for all the classifiers under the condition $z > 1.96$ at 95% confidence. Hence, the prototype system developed for the detection of urban buildings has been made to use MCS as the default classifier.

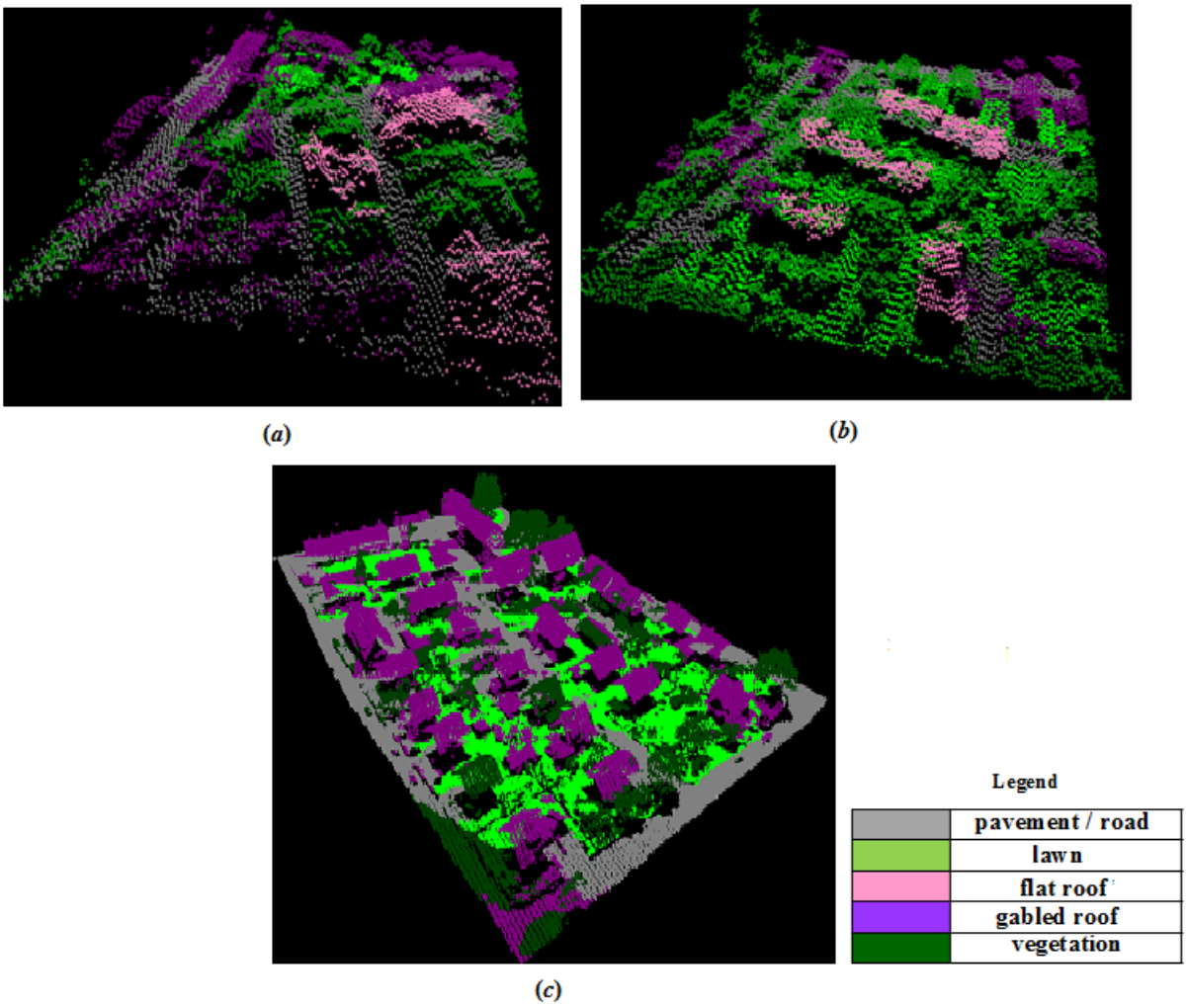


Figure 3.10: Labelled point cloud data representing various urban features shown in 3D view (*a*) study area 1 : inner city, (*b*) study area 2 : high riser, and (*c*) study area 3: residential area

3.5.4 Detection of building

The LiDAR points labelled as buildings were extracted from the classified dataset and building masks were generated on a 10-cm grid for all the three datasets. The resulting buildings masks

were independently evaluated by the ISPRS by computing various evaluation metrics at both point and object-levels. The corresponding results of the evaluation, quantified in terms of accuracy and completeness, are shown in Tables 3.3 and 3.4 respectively for the point-level evaluation and object-level evaluation.

Table 3.3: Pixel-based evaluation

Dataset	**Completeness (%)**	**Correctness (%)**	**RMS extracted (m)**	**RMS reference (m)**
Study area 1	73	77	1.3	1.38
Study area 2	81	94	0.98	0.86
Study area 3	85	80	0.75	1.05

Table 3.4: Object-based evaluation: Completeness, and Correctness indicate the metrics for objects balanced by area. Completeness and Correctness for large objects indicate the evaluation metrics for objects balanced by area which are larger than 50 m^2.

Dataset	**Completeness balanced by area (%)**	**Correctness balanced by area (%)**	**Completeness for large objects (%)**	**Correctness for large objects (%)**
Study area 1	87	98	79	100
Study area 2	99	98	100	91
Study area 3	90	92	87	94

As evident from the Table 3.3, an average completeness of 80%, and an average correctness of 83% is achieved at the point level. The overall quality is better for the study area 2 when compared to that of study area 1, and study area 3.

From the Figure 3.11, it can be seen that few small buildings are not detected by the proposed algorithm. These false-negative (FN) points (represented in blue colour in Figure 3.11) correspond to an object in reference which has been erroneously classified as background. This can be attributed to the labelling error. False-positive (FP) points (the points erroneously la-

belled as building points) are found surrounding most of the buildings. This may be due to the application of the morphological operators 'dilate' and 'close' on the buildings' masks to fill the inherent data missing caused by non-returning LiDAR pulses.

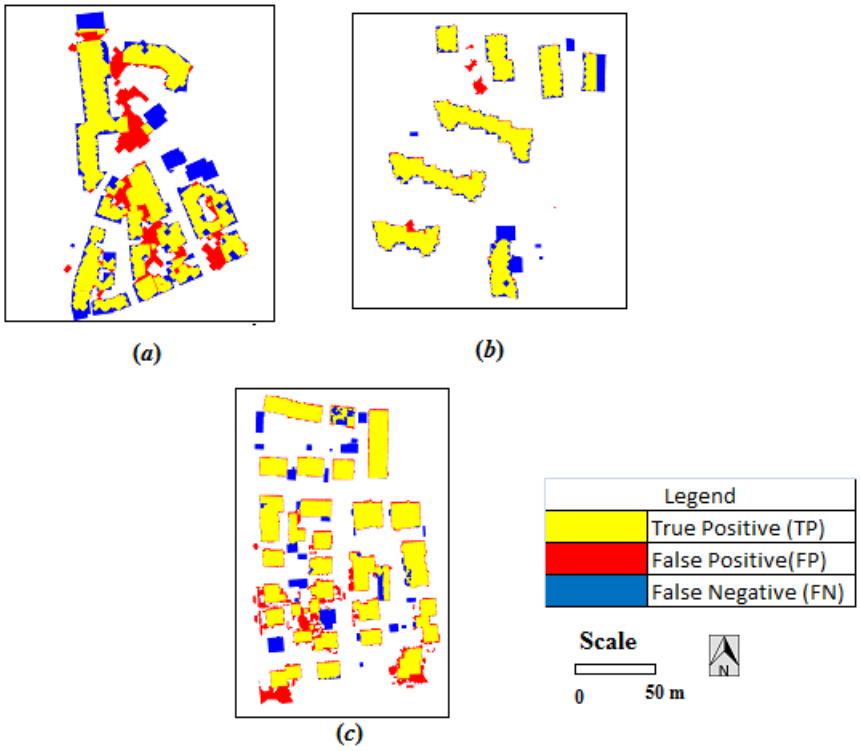


Figure 3.11: Results of evaluation: (*a*) study area 1 : inner city, (*b*) study area 2 : high riser, and (*c*) study area 3: residential area.

Discussion

The potential of LiDAR point cloud in urban environment can be better exploited by processing and labelling the LiDAR point cloud in the 3D environment and integrating with high resolution multispectral image. This part of the study proposed a novel object-oriented methodology for semantic labelling of LiDAR point cloud for urban land cover mapping and buildings detection and footprints masking. A colour-based region growing segmentation has been implemented to create 3D segments in the LiDAR point cloud. The resultant 3D segments are further classified by three different classifiers independently and also by constructing a multiple classifier system. Use of multiple classifier system improves the confidence of classification as evident from the

results (see Table 3.2). The labels generated for 3D segments in the classification process are assigned to all the points in the corresponding 3D segments thus completing the semantic labelling of each and every LiDAR point.

The following are important points that emerge from the work presented in this chapter.

(a) The proposed methodology labels urban LiDAR point cloud into multiple classes by preserving the geometric nature of the earth surface features with the complementary information from multispectral images.

(b) The comprehensive validation of the results indicates an overall accuracy of about 97% for the ground classes and about 85% for the non-ground classes. The relatively lower accuracy for the discrimination of the non-ground classes can be attributed to inability of the classifiers to distinguish between flat roof buildings and gabled roof buildings. This is evident from the users and producers accuracy as seen in Figures 3.8 and 3.9. The class 'flat roof' exhibits a relatively low accuracy compared to any other class. This may be attributed to the apparent lack of significant difference in the curvature values of the roof class and vegetation class. This can be circumvented by introducing additional geometric features in the methodology. Ground class has a higher overall accuracy when classified by the single best classifier as well as by the multiple classifier system. The spectral values play a key role attributing to the high classification accuracy of the ground classes.

(c) *k*-NN classifier proves to be the single best classifier for both the ground and non-ground classes. The results of classification indicate that the accuracy produced by the multiple classifier system is at par with the single best classifier for all the classes. This indicates the significance of multiple classifier system as it reduces the complex heuristics involved in the selection of appropriate classifier and also has the potential to improve the reliability and accuracy of classification.

(d) The labelled point cloud, with further processing, can be used for delineation and modelling of various objects of interest in the urban environment. In this study, the labelled point cloud has been further processed for buildings detection and modelling. At the pixel level, validation of the results by ISPRS gives Correctness scores of 96%, 100% and 92%, and Completeness scores of 79%, 100%, and 86% respectively for the LiDAR datasets of study area 1,

study area 2, and study area 3. Completeness corresponds to the detection rate and Correctness corresponds to how well the detected entries match with the reference. The pixel-level score for the three study sites indicate there is a significant correspondence of the detected pixel to the reference pixels. The shape, and the complexity of the building structure evidently have an influence on the accuracy of the methodology. This is conspicuous from the result which shows that study area 1 with complex building structure produces comparatively low detection rate at the pixel level. Study area 3 which consists of low residential buildings are well detected at the pixel and object-level. This is reflected in the relatively low root mean square (RMS) error in study area 3 compared to study area 1 and study area 2. An object-level evaluation balanced by area and object-level evaluation for large object indicate that the large buildings are detected clearly. Evaluation metrics at the object-level weighted by the area produced an average Completeness of 92% and an average Correctness of 96%. The resulting correctness of 96% is significant given the complexities in the semantic labelling of the LiDAR point cloud.

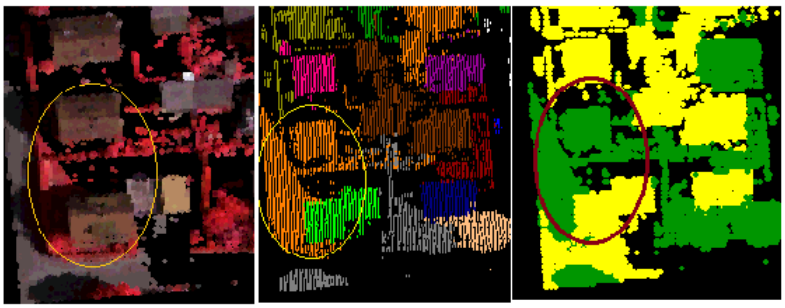


Figure 3.12: Example of misclassification of a portion in study area 3: (a) coloured point cloud, (b) segmented point cloud, and (c) classified point cloud

(e) Also it can be attributed that the final results obtained at the labelling stage are very much influenced by the previous stages. Segmentation is an important step in object-based classification and the quality of segments produced influences the final results of classification. Figure 3.12 shows a portion of a building which has been misclassified. It is apparent that the error has propagated from the segmentation stage. Further, the possibility of trees hovering over some portions of buildings may lead to under-segmentation. As evident from the Figure 3.12, the building and the surrounding trees are classified as a single segment thereby reducing the purity of the features extracted. Urban environment is very complex and the co-occurrence

of tall trees and buildings, and the possibility of casting shadows on the buildings pose fundamental limitations on the performance of the segmentation and classification algorithms.

3.6 Conclusion

A novel object-based LiDAR point cloud labelling method for discrimination of various urban land cover classes has been developed and tested on three LiDAR datasets. The methodology integrates geometrical information from LiDAR point cloud data and spectral information from multispectral images for labelling of LiDAR point cloud. The semantically labelled point cloud has been further used for detection and reconstruction of urban builds in three-dimensional environment. An overall accuracy of about 97%, and 85% has been achieved, respectively, for the discrimination of ground and non-ground urban land covers. Further, the methodology has been used for the detection and reconstruction of urban buildings. The results indicate accurate buildings detection and modelling with about 97% correctness, and 95% completeness.

3.7 Chapter Summary

In this chapter we have proposed an object-based framework which exploits the complementary information from two different sensors, LiDAR and multispectral imaging, for semantic labelling of urban point cloud datasets. The results appear encouraging for automatically detecting and visualising various urban objects from the point cloud.

LiDAR technology is a boon to the remote sensing community enabling direct measurement of the elevations of points in the earth surface. However, the technology adept in capturing elevation with its multiple returns, produces point clouds with high point densities. This makes the dataset complex and computationally demanding. Moreover, the spectral information from multispectral images, though complementing the geometrical information from LiDAR datasets, markedly increases the computational complexity of the algorithm. This demands computationally efficient solutions in all the processing stages of object-based approach while processing the massive LiDAR datasets. The next chapter is focused on improving the

segmentation algorithm, which is one of the the key stages in the proposed framework.

CHAPTER 4

A SUPERVOXELS-BASED SPECTRO-SPATIAL APPROACH FOR 3D URBAN POINT CLOUD LABELLING

Prelude: This chapter proposes a novel 3D object-based classification framework for labelling the urban LiDAR point cloud using a computer vision technique, supervoxels. The supervoxels approach is promising for representing dense LiDAR point cloud in a compact manner for 3D segmentation and for improving the computational efficiency. Starting with a crisp literature review and briefly explaining the concept of supervoxels, this chapter presents the detailed experiments carried out for the proposed supervoxels-based approach for object-based point cloud labelling. The results are discussed in detail before drawing the main conclusion of the work presented in this chapter.[1]

4.1 Introduction

Segmentation is a critical step in object-based point cloud analysis. A marginal error in the segmentation propagates further to the classification stage affecting the accuracy of classification. Several algorithms have been used in recent studies for producing meaningful segments as discussed in chapter 2. However as the point density increases, the computational cost of segmentation algorithms increases tremendously as it requires to handle millions of points in each stage of the labelling. Moreover, when spectral information from the optical images is used along with the geometrical information from LiDAR data in semantic labelling, the complexity further increases as discussed in chapter 3. This demands computationally efficient algorithms for handling the massive point cloud data.

[1]This chapter is published in *International Journal of Remote Sensing* Vol: 37(17), 2016. Publisher: Taylor & Francis. Authors: Anandakumar M Ramiya, Rama Rao Nidamanuri, Ramakrishnan Krishnan.

Developments in the theory of computer vision suggest that, instead of using pixel grid as the underlying representation for segmentation or scene understanding, it is more meaningful to have a low-level grouping of pixels of similar nature. These groups of pixels, commonly known as superpixels, capture the redundancy in the image. The superpixels form the basic units for further processing in lieu of the individual pixels, thereby decreasing the computational load (Achanta et al., 2010).

Supervoxels, analogue of superpixels, in the 3D domain is a promising alternative by which redundancy in 3D data can be tremendously reduced, enabling computational efficiency for venturing to fully automatic operations. The supervoxels are used for further analysis instead of the entire point cloud. Recent studies on processing of 3D medical images indicate that supervoxels-based segmentation significantly improves the computational complexity and performance time. Lucchi et al. (2012) have used supervoxels-based segmentation on mitochondria obtained by stacking images from electron microscopy and proved that the supervoxels-based approach is good for 3D segmentation. Ortiz et al. (2014) have used supervoxel segmentation for 3D dendrite spine detection. Supervoxel based approach has also been considered for processing 3D video images (Xu and Corso, 2012; Weikersdorfer et al., 2013).

Our literature review indicates that there are very few studies which evaluated the potential of supervoxels approach for segmentation of LiDAR point cloud. Lim and Suter (2009) have labelled terrestrial laser scanner point cloud captured on a built environment by creating supervoxels based on colour and geometrical similarity within a neighbourhood. The supervoxels are combined by multiscale conditional random fields to form segments for classification. However, this method produces overlapping supervoxels. Moreover, to create final segments, they have used a supervised learning technique by which labelled supervoxels are used as training data for further analysis. This process is cumbersome when working with LiDAR point cloud which has multiple classes. Similarly, Aijazi et al. (2013) have labelled a terrestrial laser scanner data by creating voxels based on a nearest neighbourhood algorithm. The voxels are then transformed into supervoxels by assigning the spectral and geometrical properties of points within the voxels. The transformed features in the 3D space are perceived as supervoxels without regard for the spatial distribution of voxels. A recent study by Yang et al. (2015) proposes a multi-scale approach to generate supervoxels for mobile laser scanning data. In

this paper, the authors have adapted a multiscale approach in which supervoxels of different sizes are integrated using a set of optimization strategy. This was necessary to ensure that the size of supervoxels best depicts the real world scenario of the complex urban object as it is the key factor for the geometric information extraction. An hierarchical classification scheme was adopted to finally classify the objects. All the three studies emphasise the importance of supervoxelisation.

In object-based point cloud analysis, supervoxelisation is an important pre-segmentation stage. The size of the supervoxels and further the segments, is a key factor for computing the geometric and spectral feature which is used for classification. The complex urban objects demand computationally efficient methods, preserving the geometry and the spatial connectivity of the objects, for generating meaningful segments which can be directly used for geometric information calculation.

Due to the scanning pattern and the nature of the data, processing airborne LiDAR point cloud is different from the mobile laser data, and terrestrial laser data. Moreover, to the best of our literature review, the applicability of supervoxels-based segmentation on airborne LiDAR point cloud for urban environment is not yet reported. The availability of computationally less demanding segmentation approach will be very useful for segmentation when dealing with LiDAR point cloud of high spatial complexities such as airborne urban datasets.

Therefore, the part of this research carried out in this thesis proposes a spectro-spatial approach for labelling urban airborne LiDAR point cloud using a 3D object-based point cloud labelling method. The main contribution of this work includes extension and adaptation of a supervoxels-based segmentation approach for objects creation using airborne datasets having different point density and different spatial complexity. The generated supervoxels are then combined using novel local convexity measures to create meaningful segments. Further, the effect of feature selection (geometrical and spectral), and classifier selection on the accuracy of labelling has been assessed.

4.2 Methodology

A brief outline of the methodological framework is given in Figure 4.1. The proposed methodology contains four main stages: data preprocessing, segmentation, feature extraction and classification.

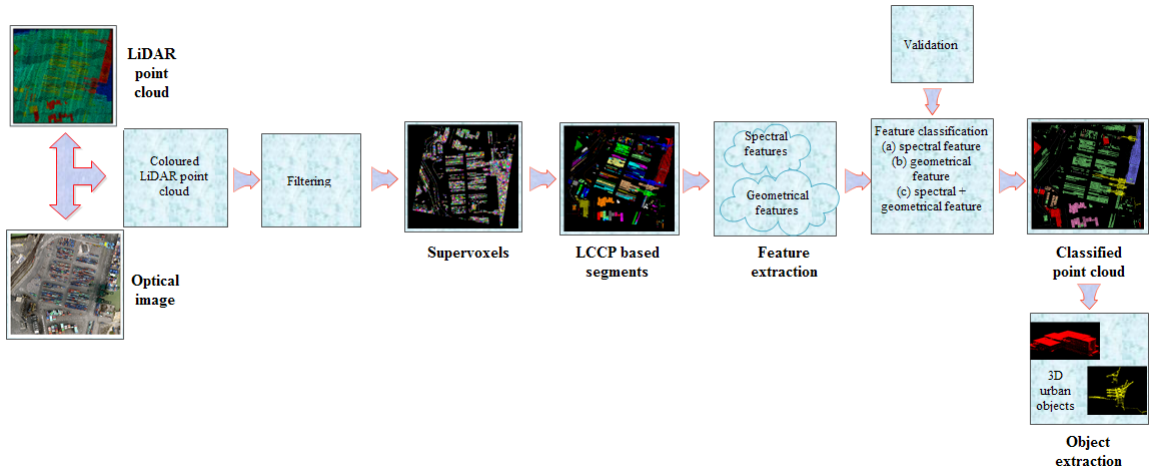


Figure 4.1: Outline of the approach used in the study (pictorial representations is given for the input and output stages of the workflow).

4.2.1 Data preprocessing

The spectral information from multispectral images is integrated with the geometrical information from LiDAR point cloud as discussed in the section 3.4.1.

This study focusses on extracting geometrically complex urban structures. Hence to reduce the computational complexity of the entire process, we removed the returns from the ground by progressive densification based filter technique (Axelsson, 2000b) described in the section 2.4.2. The algorithm progressively adds points to the TIN surface generated on a set of seed points chosen amongst the ground points based on the distance and angle of the points to the underlying triangles.

4.2.2 Supervoxels

Processing of LiDAR point cloud for urban studies is highly complex and computationally demanding. Assigning a label to each point in the point cloud data is hence a challenging task. This necessitates methods which can be adapted to improve the computational load. Similar to the superpixel methods in computer vision, recent developments in 3D domain compute supervoxels from a set of voxels. The continuous 3D space containing the point cloud can be discretised by the volume based on the density and distance of the point in the LiDAR point cloud. Each discrete unit is called a volumetric element (voxel). Voxel represents 3D pixel in the 3D space. Voxels are generated from the 3D points based on structuring algorithm such as octree or *k*-d tree methods, which group points based on neighbourhood and spatial property. The voxels are further grouped based on geometrical and spectral relationships between them to create supervoxels (Fig. 4.2). The supervoxels thus generated are used in the further computations in lieu of the actual point cloud. This will tremendously simplify the complexity involved in further stages of point cloud segmentation and classification thereby reducing the computational load.

Voxel cloud connectivity segmentation

We implemented the recently proposed Local Cloud Connectivity Patches (LCCP) based segmentation (Christoph Stein et al., 2014) for creating meaningful clusters from the LiDAR point cloud coupled with spectral data. The LCCP algorithm was originally developed for processing RGB-D data from 3D depth sensors such as Microsoft Kinect. This segmentation approach relies on local connectivity, i.e., the convexity/concavity between two neighbouring patches to create realistic segments (objects).

Supervoxels were created using the voxel cloud connectivity segmentation (VCCS) (Papon et al., 2013). In our implementation of VCCS for supervoxels creation, voxels were created using the octree data structuring algorithm. Spatial relationship between voxels was established by generating an adjacency graph based on 26 neighbourhood connectivity of the voxel. From the voxelised space, seed voxels were selected, based on the chosen seed resolution, which are grown to form supervoxels. These seed voxels are evenly distributed in space on the voxel

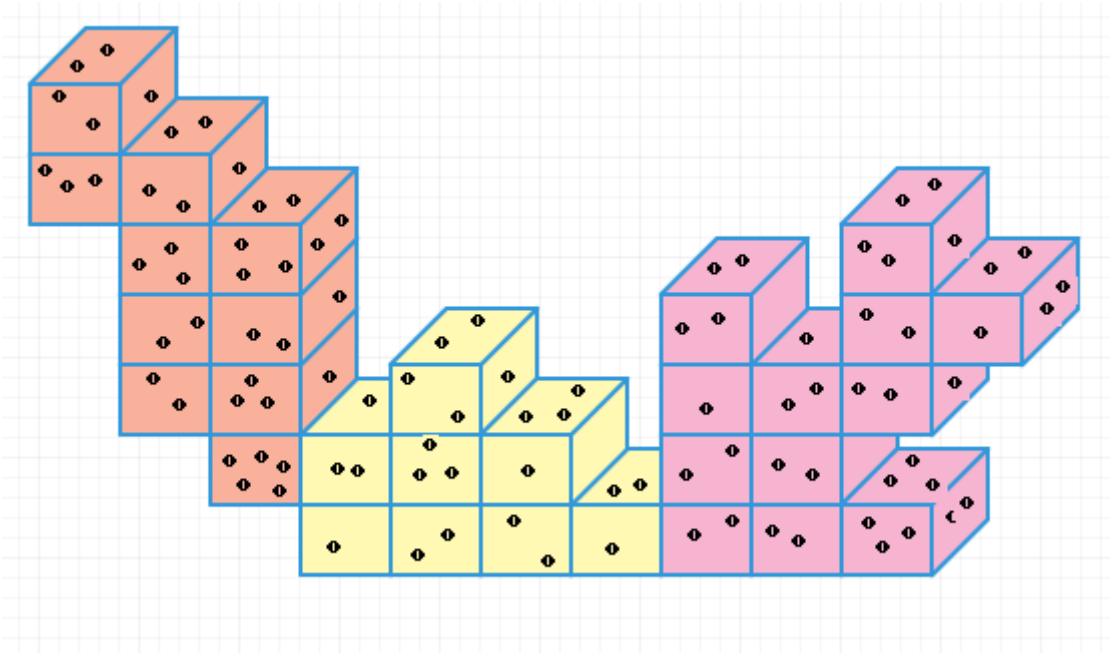


Figure 4.2: Graphical illustration of voxels and supervoxels. Voxels are discrete units containing many points (Each cube represents a voxel). Group of voxels forms supervoxels (Different color represents different supervoxels).

grid. For each seed voxel, the supervoxel feature vector consisting of spatial, color and normal estimates of points within the voxel was initialised. The distance was computed from the supervoxel centres to the adjacent voxels within a limited search space using the equation (Papon et al., 2013)

$$D = \sqrt{\frac{\lambda D_{\mathrm{c}}^2}{m^2} + \frac{\mu D_{\mathrm{s}}^2}{3R_{\mathrm{seed}}^2} + \epsilon D_{H_{ik}}^2} \tag{4.1}$$

where D_c is the color distance in the CIELab space (Connolly and Fleiss, 1997) normalised by constant *m*, D_{s} is the spatial distance normalised by the seed resolution R_{seed}, $D_{H_{ik}}$ is the distance in the FPFH (Fast Point Feature Histogram) space using the Histogram Intersection Kernel (H_{ik})(Rusu, 2009) and λ, μ, ϵ are the parameters controlling the color, spatial distance and geometric similarity between the voxels. CIELab color space is a perceptually uniform color space with a luminance (lightness) channel defined by L and two other color channels (a and b). The lightness axis values varies from 0(black) to 100(white). The a axis extends

from red (positive) to green (negative) while the b axis extends from blue (positive) to yellow (negative). We gave equal weight to spatial distance and color, with distance in the FPFH space having a slightly higher weight. The voxels are assigned to their neighbouring supervoxel centers based on the distance measure. Once all the voxels were assigned to the any of the supervoxel centres, their centroids were updated. The process is carried out iteratively until the centroids stabilize. This is very similar to the local region growing variant of *k*-means clustering in 2D space (Achanta et al., 2010). The supervoxels thus created are geometrically constrained and disjoint in the 3D space.

3D segmentation using local cloud connectivity patches

Supervoxels are oversegmented objects and hence will not represent the true geometric nature of the objects. Meaningful 3D segments are produced by merging appropriate supervoxels. The geometrical connectivity between neighbouring supervoxels was established based on local connectivity between adjacent patches identified using the adjacency graph (Christoph Stein et al., 2014). Two neighbouring supervoxels can be merged into a segment if they satisfy the convexity criteria based on the centroid and the normals of the individual supervoxels. Consider two supervoxels s_1 and s_2, with centroids at positions $\boldsymbol{x}_1, \boldsymbol{x}_2$ and have normals $\boldsymbol{n}_1, \boldsymbol{n}_2$. Normal of a voxel is estimated by fitting a plane to all the points within the voxel. Let $\boldsymbol{d} = \boldsymbol{x}_1 - \boldsymbol{x}_2$ be the distance between the two neighbouring patches. For convex connections, $(\boldsymbol{n}_1 - \boldsymbol{n}_2) \cdot \boldsymbol{d} > 0$ and for concave connections, $(\boldsymbol{n}_1 - \boldsymbol{n}_2) \cdot \boldsymbol{d} < 0$. The supervoxels s_1 and s_2 are said to be connected if they satisfy the equation

$$\mathrm{CC_b}(\boldsymbol{s}_1, \boldsymbol{s}_2) = (\boldsymbol{n}_1 - \boldsymbol{n}_2) \cdot \boldsymbol{d} > 0 \vee (\beta < \beta_{\mathrm{thresh}}) \tag{4.2}$$

where $\mathrm{CC_b}$ defines the basic convexity criteria, β denotes the angle between the normals $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$. To reduce errors in segmentation, a threshold β_{thresh}, was used to merge concave surfaces with low curvature values. The segments thus created were object primitives which better describe the object geometry. To illustrate the effect of supervoxelisation and segmentation by local convexity on the objects in the LiDAR point cloud, an experiment was performed on a subset of the ISPRS Vaihingen benchmark dataset. Figure 4.3a shows the coloured point

cloud dataset comprising of buildings and vegetation. The point cloud after supervoxelisation is given in Figure 4.3b and the resultant segments after LCCP segmentation is given in Figure 4.3c.

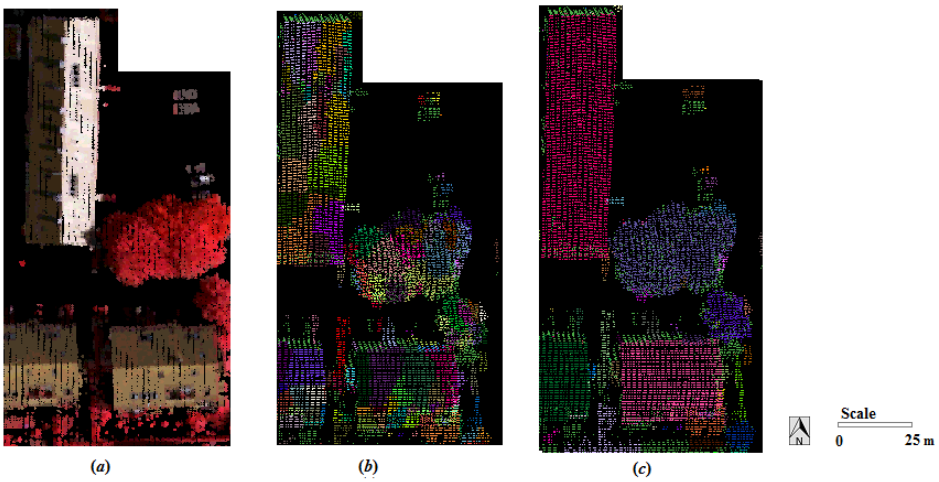


Figure 4.3: Different stages of segmentation: (*a*) coloured point cloud (*b*) supervoxels (*c*) segments.

4.2.3 Feature extraction

Features were extracted from the segments which were further used for labelling. The spectral and geometrical features described in the section 3.4.4 were extracted from the segments for all the study sites. Spectral features include mean spectral values and normalised difference vegetation index (NDVI). Geometrical features include pointedness, curveness, surfaceness, absolute elevation of objects within the cluster, the area and volume of the hull enclosed by the set of points within the cluster.

In order to study the influence of feature vectors in the classification performance, classification was performed in three test cases: 1) using geometrical features alone, 2) using spectral features alone, and 3) using a combination of spectral, and geometrical feature vectors.

4.2.4 Classification

The features extracted from the 3D segments of the point cloud were classified by various supervised classifiers: *k*-Nearest Neighbour classifier (*k*-NN), Naive Bayes (NB) classifier, and Support Vector Machine classifier (SVM), as described in the section 3.4.5. In addition to the set of classifiers, we used random forest classifier, a specialized case of multiple classifier. This is a generalised case of bagging (bootstrap aggregation) classifier for learning and classification developed by (Breiman, 2001). Random forest consists of an ensemble of simple decision tree classifiers formed based on random sub-space method. From the training dataset, a sub-sample was chosen independently (with-replacement) to create a decision tree. Given a training set $X = x_1, x_2, x_3, x_4....$ with response $Y = y_1, y_2, x_3, x_4....$, the samples were bagged repeatedly with random sub-space method, in which random features were selected in each case for creating the tree. This method of feature bagging reduces the correlation between the estimators and increase the diversity amongst the trees. The predictive models created by many such trees were 'ensembled ' to arrive at the final prediction based on posterior probability which is better than the individual model prediction.

4.2.5 Evaluation strategy

The reference data required for training of the algorithms and testing the performance were generated from the datasets by manual inspection and comparison with high resolution orthophotos. The results of classification were validated by holdout cross validation approach.

Study areas 1, 2, and 3 were classified into three classes namely, flat roof, gabled roof, and vegetation. Study area 4 was classified into eight classes namely, buildings, linear marine vessels, rectangular marine vessels, big container cranes, small container cranes, ship, cars and road. Various evaluation metrics such as overall accuracy, kappa coefficient (κ), producers accuracy, users accuracy were computed for the datasets. Accuracy of the classification was assessed using the error matrix generated comparing the predicted class and the actual class and their statistical significance were tested as described in section 3.4.6. Additionally the statistical significance of the performance of the different classifiers were also computed based on the error matrix. It was used to assess whether the methodologies produce significantly

different results. Let κ_A and κ_B denote the kappa coefficients of the two classifiers and var_A and $varB$ denote the variance from the error matrix.

The test statistic $\boldsymbol{z}_{AB}$ for statistically testing two confusion matrix is given by the equation 4.3 which assesses the statistical significance difference at 95 % confidence interval.

$$\boldsymbol{z}_{AB} = \frac{\mid \kappa_A - \kappa_B \mid}{\sqrt{var_A + var_B}} \tag{4.3}$$

In addition to this, the results were independently evaluated by the International Society of Photogrammetry and Remote Sensing (ISPRS WG III/4) using the ISPRS benchmark datasets for study areas 1, 2, and 3. As buildings are the prominent features in the study areas, building detection based evaluation was carried out. From the labelled point cloud dataset, the points corresponding to flat and gabled roof were extracted and building footprints were generated. The evaluation procedure similar to method described in 3.4.6 was carried out at(a) pixel-level, (b) object level and (c) objects greater than 50 m^2. At each level, the correctness, and completeness were computed based on the equations 3.9.

4.3 Dataset used

Experiments were performed on four different study areas obtained from two different LiDAR datasets of varying point density and landcover complexity and their corresponding orthophotos.

The first three study areas are part of Vaihingen benchmark dataset provided by the ISPRS as part of the ISPRS test project on urban classification and 3D building reconstruction as described in the section 3.3. The fourth study area is the Belgium benchmark dataset described below.

Belgium Dataset [2] The dataset grss_dfc_2015 was provided by the IEEE GRSS (The Institute of Electrical and Electronics Engineers Geoscience and Remote Sensing Society) as part of the IEEE GRSS Data Fusion Contest 2015 (2015 IEEE GRSS Data Fusion Contest). The data were collected over Zeebrugge, Belgium covering urban and harbour area on March 2011 from an airborne platform flying at an altitude of 300 m. The data consist of very dense LiDAR point cloud with the corresponding aerial image at 5 cm resolution. Both the datasets were captured simultaneously and were georeferenced to WGS-84 (World Geodetic System-84). A subset of around 500 m^2 area covering the urban and harbour areas of Zeebrugge comprising eight classes such as buildings, marine vessels, container cranes etc. was extracted. The point density of the study area 4 is about 55 points/m^2. The mean land cover complexity index computed based on the procedure described in section 3.2 is around 0.42. The details of the dataset is summarized in Table 4.1.

Table 4.1: Details of study area from Belgium dataset

Parameters / dataset	**Study area 4**
Point density (points /m^2)	55
Number of 1st return	20,022,938
Number of 2nd return	0
Number of 3rd return	0
Total number of points	20,022,938
x range (m)	501
y range (m)	501
Land cover complexity index	0.42

[2]The data was provided as part of the IEEE GRSS Image Analysis and Data Fusion contest organised by the IEEE GRSS Image Analysis and Data Fusion Technical Committee. It was acquired by the Belgian Royal Military Academy.

4.4 Results and Analysis

4.4.1 Data pre-processing

Because of the fact that we are interested in detecting various urban objects, LiDAR point cloud from the various study areas were pre-processed to filter out the ground points from the entire point cloud. Initially, study area 1 consisted of $10,242$ points, study area 2 consisted of $15,282$ points and study area 3 consisted of $230,171$ points. After filtering, the total number of non-ground points for study area 1 was $6,436$. For study areas 2 and 3 the total non-ground points were $9,395$, and $122,937$ respectively. The Belgium harbour dataset covering the study area 4 has $20,022,938$ points. After filtering, there are $5,434,862$ points. Fig. 4.4 shows visual display of the coloured point cloud.

4.4.2 Segmentation

The choice of voxel resolution and seed resolution depends upon the density of the point cloud. The effect of varying the seed resolution and voxel resolution is well studied in Papon et al. (2013). As suggested in Papon et al. (2013), the seed resolution must be almost ten times the voxel resolution. For datasets with less point density (< 2 points/m^2), the seed resolution was selected so as sufficient number of voxels are available to form supervoxels. For example, we used 1 m as the voxel resolution and 14 m as seed resolution for the study area 1 and study area 2. However, for datasets with point density > 5 points/m^2, the seed resolution can be less as sufficient number of points will be available in the neighbourhood. For the study areas 3 and 4, with higher point density, voxel resolution of 0.5 m and seed resolution of 7 m was used. The supervoxel patches were combined to produce meaningful segments based on convexity criteria. As indicated in the section 4.2.2, the concave surfaces with low curvature value are treated as convex and are used to produce segment patches. From Figure 4.6, the effect of concavity of the segmentation results is evident. For airborne urban datasets, concavity threshold of 20° is ideal for producing meaningful segments.

On applying the VCCS, study area 1 produced 5728 supervoxels. Study areas 2, 3 produced

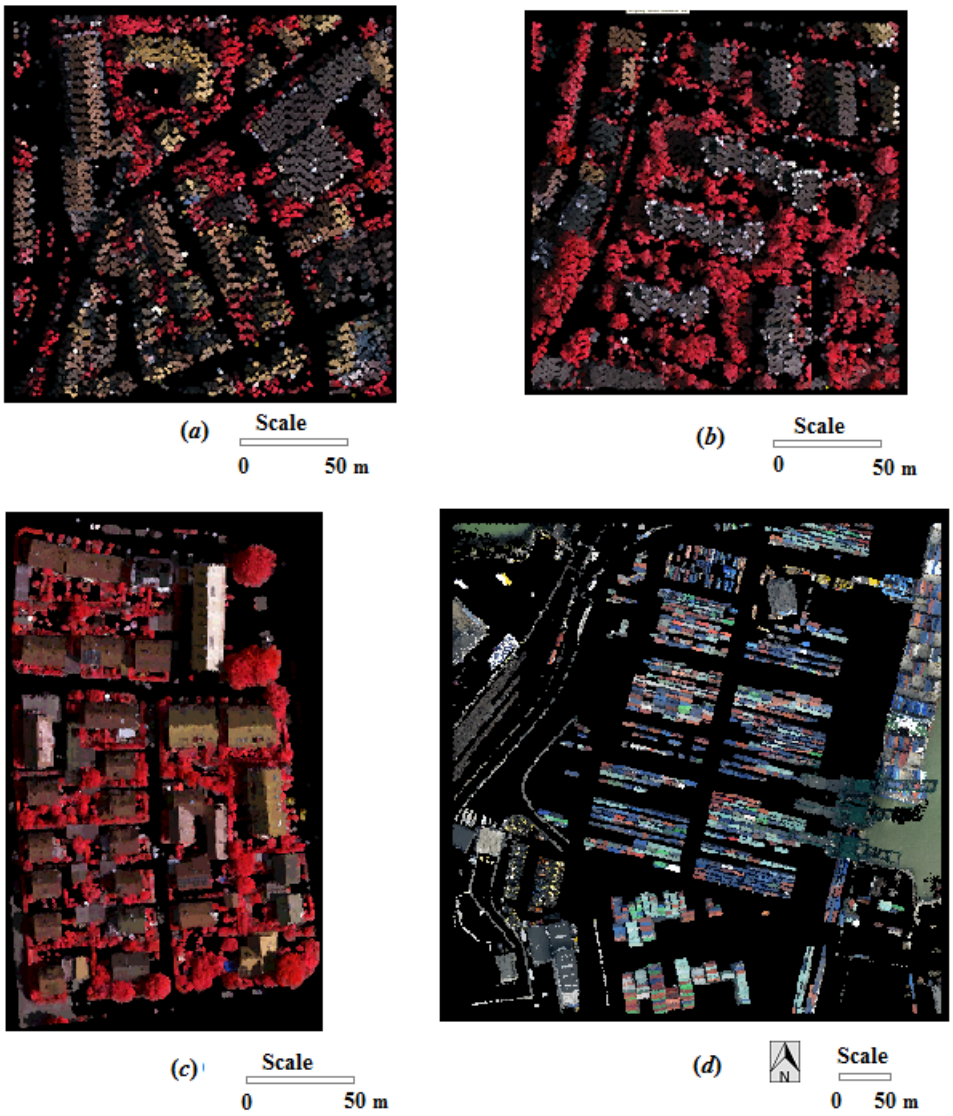


Figure 4.4: LiDAR point cloud coloured based on spectral values derived from optical image (*a*) study area 1 : inner city, (*b*) study area 2 : high riser, (*c*) study area 3: residential area, and (*d*) study area 4 : Belgium harbour dataset.

7022 supervoxels and 9297 supervoxels respectively. Studyarea 4 produced 5670 supervoxels. The results of segmentation are presented in Figures 4.8 through 4.11, and Table 4.3. To assess the effect of different parameters on the segmentation, a small test region comprising of rectangular iso-containers was selected from the high point density dataset ie., study area 4. The test region consisted of 222133 points. One of the important parameters for VCCS segmentation, the seed resolution parameter was varied for observing the variation in the supervoxels created. The results are presented in Figure 4.5. As seen from the Figure 4.5, as the seed resolution increases, the size of supervoxels also increases, thus reducing the number of supervoxels created. When the seed resolution was set to 1 m there are 3448 supervoxels. This number reduced to 1171 supervoxels when seed resolution was increased to 3 m. The number of supervoxels reduced to 440 and 218 when the seed resolution was set to 5 m and 7 m respectively. The relatively small number of superpixels compared to original point cloud indicate close to 99% reduction in the number of points used for further classification as evident from Table 4.2. This is an important observation as it indicates the advantage of using supervoxels based method for processing high density point cloud.

Table 4.2: Effect of seed resolution on data reduction

Seed resolution (m)	**Number of supervoxels**	**Percentage reduction in the data points(%)**
1.0	3448	98.65
3.0	1171	99.54
5.0	440	99.83
7.0	218	99.91

The computational load of the algorithm on the different datasets is given in Table 4.3. From the Tables 2 and 3, it can be noticed that the number of supervoxels created for lower point density dataset is high. This can be attributed to the seed resolution and voxel resolution initialised. However, the number of segments formed after merging the supervoxels appears meaningful as evident from Figures 4.8 through 4.11. After merging the supervoxels based on connectivity, there are 42 objects in study areas 1 and 2, whereas study area 3 and study area 4 have 222 and 413 objects respectively. As expected, the computational load increases

with the size of the data. Figure 4.6 shows the impact of concavity threshold on the number of supervoxels created and the objects generated. As indicated in section 4.2.2, concavity threshold aims at merging concave surfaces with low curvature values, treating it as a convex surface. As evident from Figure 4.6, when the concavity threshold value is set to a higher value, more meaningful segments are created and over-segmentation is minimised.

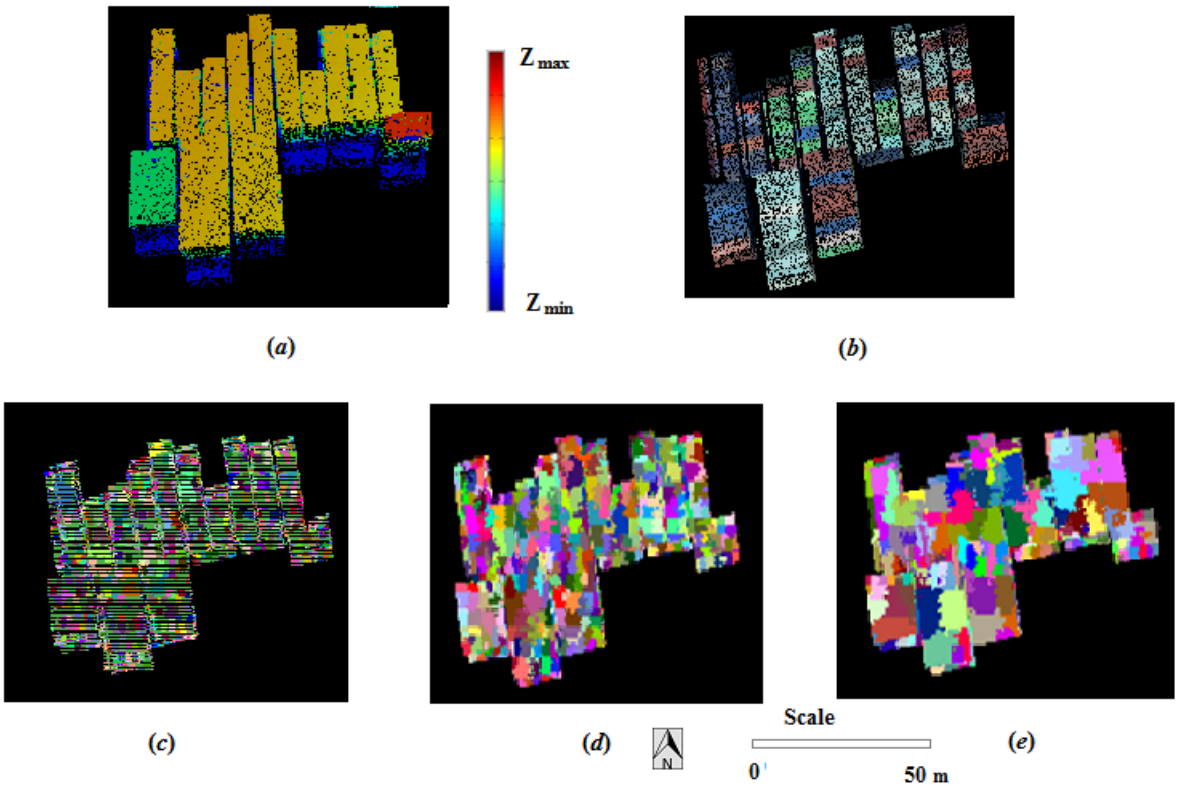


Figure 4.5: Effect of seed resolution on the supervoxels created: (*a*) point cloud coloured by elevation, (*b*) coloured point cloud, (*c*) seed resolution 1 m, (*d*) seed resolution 3 m, and (*e*) seed resolution 5 m (legend: different colour represents different supervoxels).

4.4.3 Classification

Tables 4.4 and 4.5 give the distribution of training and testing samples used for the study areas.

The overall classification accuracy estimates of the proposed method are presented in Tables

Table 4.3: Results of segmentation on different datasets

Study area	**Point density (points/m^2)**	**Total number of points**	***Z* range (m)**	**Size of the data**	**Total no of supervoxels**	**Total number of segments**	**Total run time (s)**
1	0.5	6436	28.3	202 kB	5728	42	4.90
2	0.5	14755	52.9	294 kB	7022	42	5.14
3	5.5	122937	35.9	4961 kB	9297	222	75.54
4	55.0	5434862	149.6	204 MB	5670	413	2302.22

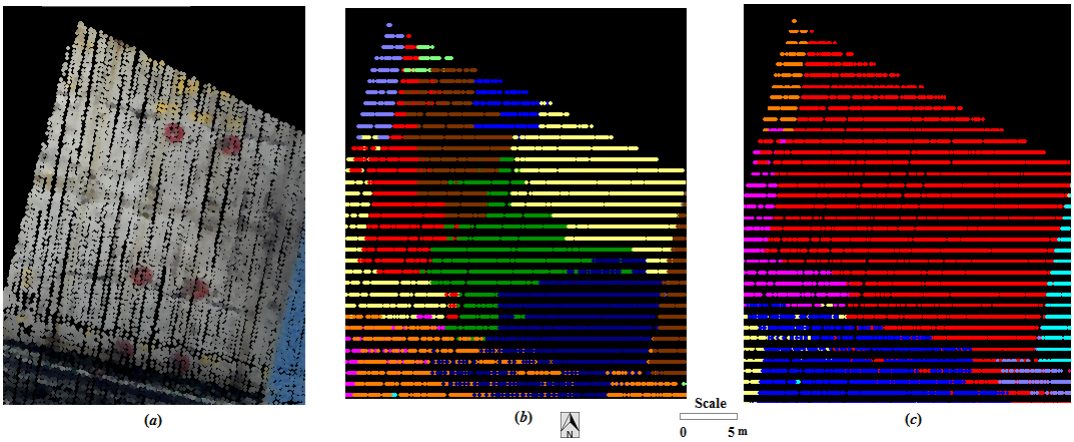


Figure 4.6: Effect of concavity threshold on the segments created (*a*) coloured point cloud, (*b*) concavity of 0°, and (*c*) concavity of 20° (legend: different color represents different segments).

Table 4.4: Training samples distribution for study area 1, study area 2, and study area 3

	Study area 1 and Study area 2		**Study area 3**	
Class	**Total number of training samples**	**Percentage distribution (%)**	**Total number of training samples**	**Percentage distribution (%)**
Flat roof	31	37.8	6	14.0
Gabled roof	17	20.7	28	65.1
Vegetation	34	41.5	9	20.9

Table 4.5: Training sample distribution in study area 4

Class	Total number of training samples	Percentage distribution (%)
Buildings	7	2.3
Linear iso-containers	170	55.4
Rectangular iso-containers	16	5.2
Shipyard cranes	4	1.3
Forklift	21	6.8
Ship	3	1.0
Cars	23	7.5
Road	63	20.5

4.6 through 4.10.

Validation of the results

Datasets of the four study sites contained different information classes and are captured at different point density. We present the results of study areas 1 and 2 as case 1 (both have similar point density), study area 3 as case 2, and study area 4 as case 3.

As Figure 4.7 indicates, at the pixel level, an overall accuracy of 89% was obtained for study area 1 and 2 with best per-class accuracy of 97%. For study area 3, the overall accuracy increased marginally to 91% with the best per class accuracy of 100%. Figure 4.7 indicates that despite containing same type of information classes, there is a significant difference between the per-class accuracy of flat roof, and gabled roof in case 1 and case 2. This can be attributed to the relatively higher abundance of the gabled roof in study area 3 (case 2) when compared to study areas 1 and 2.

For study area 4, while the overall accuracy (90%) is similar to that of case 1 and case 2, there are substantially more number of information classes belonging to geometrically important classes such as ship, car, container etc. The presence of the geometrically important

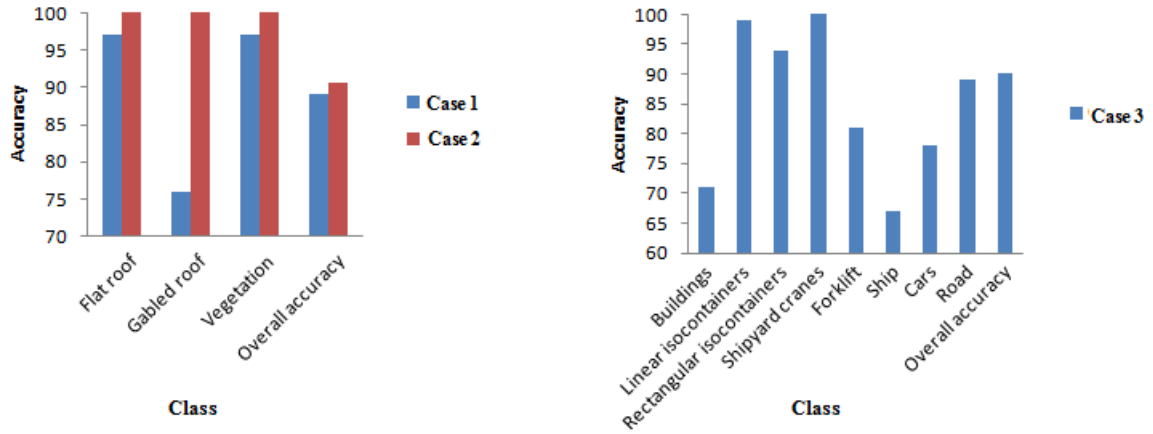


Figure 4.7: Best per-class accuracy and overall accuracy: (*a*) case 1, (study area 1, 2) and case 2 (study area 3) and (*b*) case 3 (study area 4).

classes compared to the features of buildings in the point cloud led to relatively per-class low accuracy (71%) for buildings. Comparison of the per-class user accuracy estimates amongst the three cases explain the relatively moderate overall accuracy when compared to the per-class producer accuracy. The apparently large difference in the user accuracies of classes such as shipyards, cranes and cars indicate that at the pixel level, the substantial geometrical extents of the classes impact the classification results. At the object level, the classification results are stable across the four study sites with a correctness of 90% and completeness of 95%. From the results, it can be inferred that the *k*-NN classifier performs consistently well in all the three cases inspite of balanced or unbalanced dataset. NB classifier shows a comparatively poor performance in all the three cases. In study area 3, the reason for poor performance for most of the classifiers can be attributed to the unbalanced distribution of classes in the datasets.

Impact of number of features

Features play a vital role in determining the classification accuracy. Complementary spectral and geometrical information available in the coloured point cloud segments created, was exploited to improve the classification accuracy. The inherent geometrical information available in the LiDAR point cloud alone may not be sufficient to discriminate between two geomet-

rically similar clusters. For example, in order to classify two clusters of similar geometrical nature such as concrete pavement and lawn, spectral features may be used to improve the separability.

We have manipulated the features to suit the test cases required for assessing the significance of spectral and geometrical features on classification accuracies with, (a) only spectral features, (b) only geometrical features, and (c) both spectral and geometrical features. The feature vectors were added progressively to the classification model and model accuracy was computed for each case. As indicated in section 4.2.3, for study areas $1, 2,$ and 3, the spectral features include NDVI in addition to the mean spectral reflectance values. For study area 4, only mean spectral value was used due to the lack of reflectance value in the near-infrared band. The geometric features for all the study areas include elevation range, pointedness, curveness, surfaceness, convex hull area, and convex hull volume. Hence for the Vaihingen dataset, the spectral features used are mean spectral value and NDVI. For Belgium dataset, only mean spectral value was used. For both the datasets, all the geometrical features mentioned above were used.

Figure 4.12 indicate that, for the same set of training data, the accuracy increases when both spectral and geometrical features are used. For the study areas 1 and 2, the overall accuracy when spectral features alone was considered is 88%. It decreases to 76% when the geometrical features alone are used for classification. However, when both the spectral and geometrical features are used, the overall accuracy has increased to 89%. The same trend follows with the other classifiers as can be seen in Fig. 4.12. For study area 3, using geometrical features alone, there has been a better classification accuracy compared to using spectral features alone. This can be attributed to the fact that the study area consists of well defined low roof buildings. Also, the point density of this dataset is higher, containing rich geometric information. The study area 4 with a highly dense point cloud data and a high resolution image has been classified with a higher overall accuracy with *k*-NN classifier. Using spectral features alone, the overall accuracy obtained is 88%. This decreases marginally when geometrical features alone are used. The overall accuracy has increased by 2% when both spectral and geometrical features are used. The result of classification is given in Figures 4.8 through 4.11. The results of statistical significance between two independent classifier on the classification accuracy is given in Tables

4.6, 4.8, and 4.10.

Table 4.6: Producer Accuracy (PA) and User Accuracy (UA) for study area 1, and study area 2.The cases which are statistically significant, based on the z-score, are highlighted in bold.

Study area 1, Study area 2								
	***k*-NN**		**SVM**		**NB**		**Random forest**	
	PA (%)	UA (%)	PA (%)	UA (%)	PA (%)	UA (%)	PA (%)	UA (%)
Flat roof	87	84	97	83	84	68	97	88
Gabled roof	76	87	59	100	41	64	76	87
Vegetation	97	94	97	92	91	94	94	97
Overall accuracy (%)	89		89		78		92	
Kappa coefficient (κ)	0.86		0.89		0.78		0.92	
z-score (z)	**7.82**		**8.05**		**6.91**		**8.32**	

Table 4.7: Statistical significance test between pair of classifiers for study areas 1 and 2. The cases which are statistically significant,based on the z-score, are highlighted in bold

Study area 1 and Study area 2				
	***k*-NN**	**SVM**	**NB**	**Random forest**
k-NN	0	0.04	**2.00**	0.54
SVM	0.04	0	**1.97**	0.58
NB	**2.00**	**1.97**	0	**2.55**
Random forest	0.54	0.58	2.55	0

For study areas 1 and 2, with the single best classifier, maximum accuracy was obtained for flat roof, and vegetation using both spectral and geometrical features. However, there are some cases in which there is a misclassification between flat roof and gabled roof. If the point density of the dataset increases, this error rate can be reduced.

Table 4.8: Producer Accuracy (PA) and User Accuracy (UA) for study area 3. The cases which are statistically significant, based on the z-score, are highlighted in bold.

Study area 3								
	***k*-NN**		**SVM**		**NB**		**Random forest**	
	PA (%)	UA (%)	PA (%)	UA (%)	PA (%)	UA (%)	PA (%)	UA (%)
Flat roof	67	100	50	100	50	43	50	75
Gabled roof	100	90	100	83	79	88	89	89
Vegetation	78	88	67	100	100	82	100	82
Overall accuracy (%)	91		86		79		88	
Kappa coefficient (κ)	0.74		0.75		0.72		0.84	
z-score (z)	**6.28**		**5.06**		**4.72**		**4.60**	

Table 4.9: Statistical significance test between pair of classifiers for study area 3. The cases which are statistically significant, based on the z-score, are highlighted in bold.

Study area 3				
	***k*-NN**	**SVM**	**NB**	**Random forest**
k-NN	0	**2.12**	**2.77**	**2.02**
SVM	0.04	0	0.44	0.25
NB	**2.77**	0.44	0	0.72
Random forest	**2.02**	0.25	0.72	0

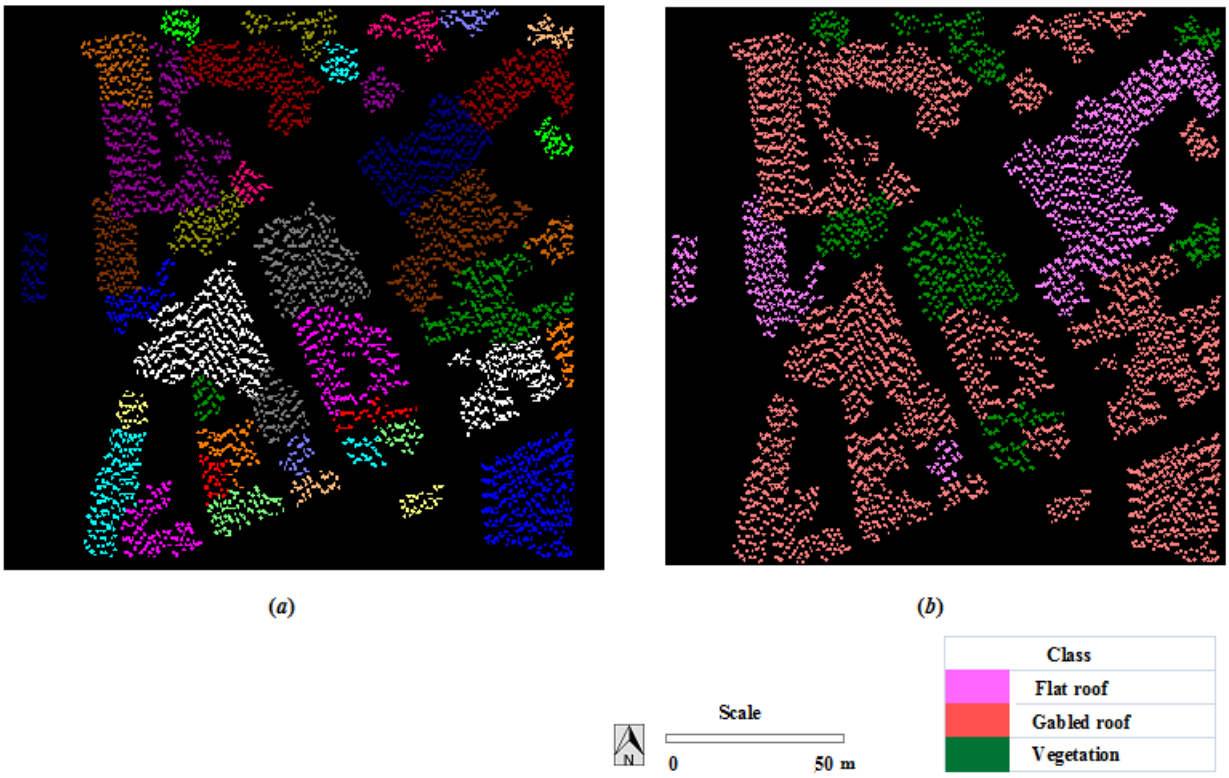


Figure 4.8: Results of (*a*) segmentation, and (*b*) classification of study area 1.

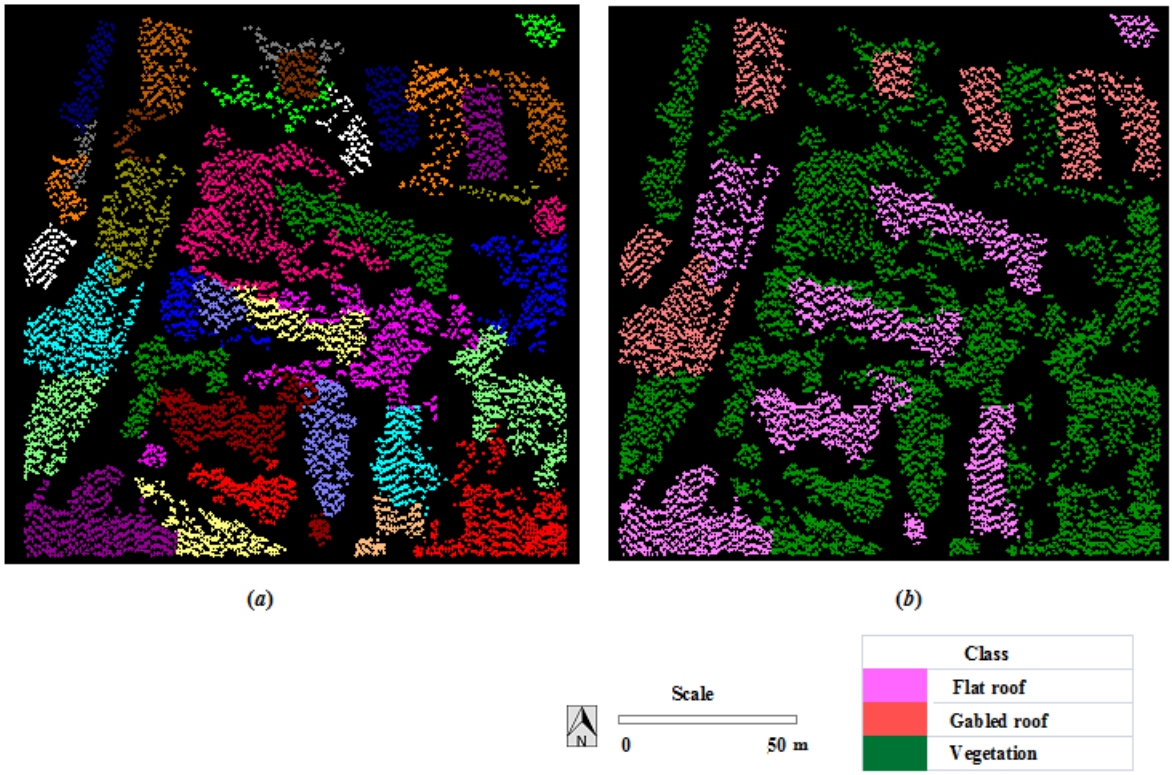


Figure 4.9: Results of (*a*) segmentation, and (*b*) classification of study area 2.

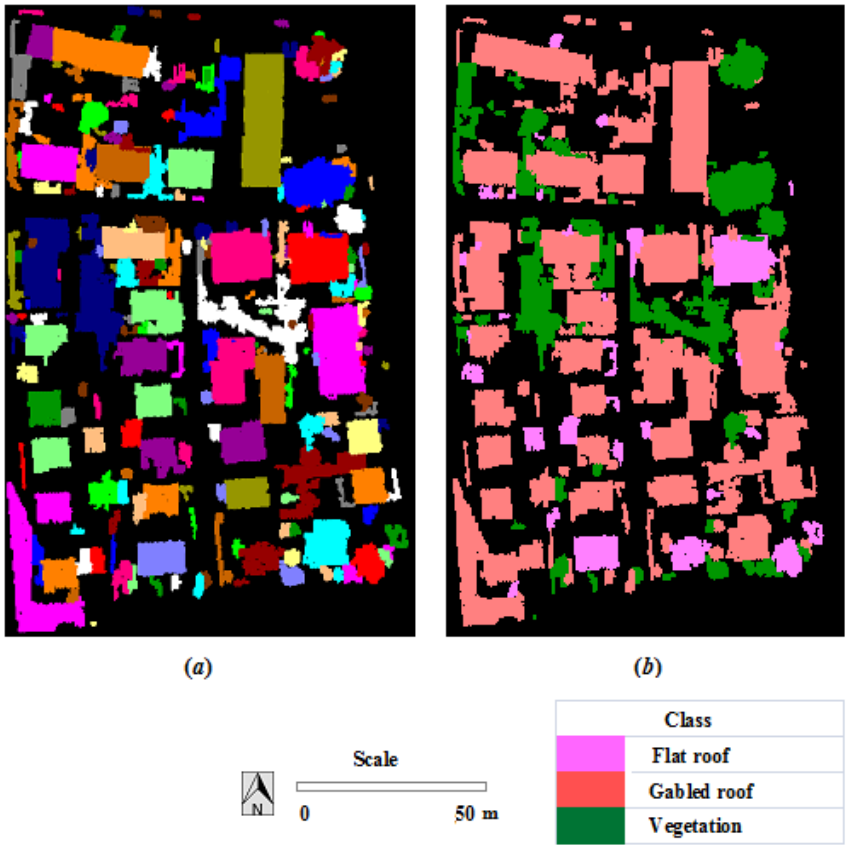


Figure 4.10: Results of (*a*) segmentation, and (*b*) classification of study area 3.

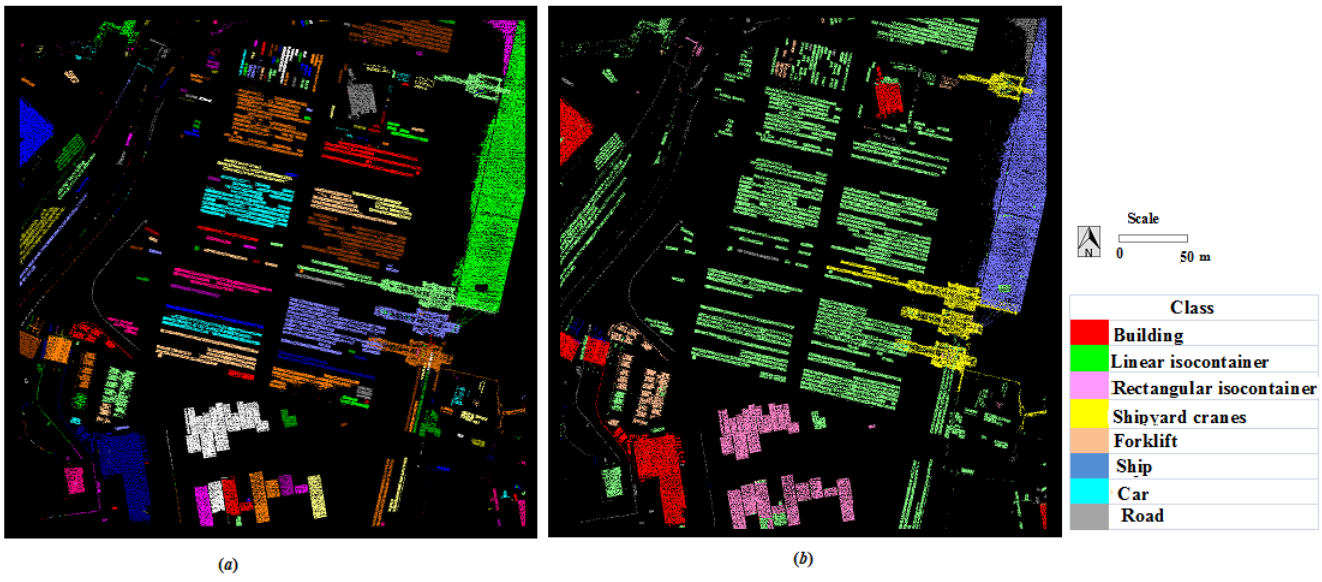


Figure 4.11: Results of (*a*) segmentation, and (*b*) classification of study area 4.

Table 4.10: Producer Accuracy (PA) and User Accuracy (UA) for study area 4. The cases which are statistically significant, based on the z-score, are highlighted in bold. (*Random forest classifier failed to detect all the classes other than linear isocontainers and road leading to zero producer / user accuracy for the respective entries in the table).

Study area 4								
	***k*-NN**		**SVM**		**NB**		**Random forest**	
	PA (%)	UA (%)	PA (%)	UA (%)	PA (%)	UA (%)	PA (%)	UA (%)
Buildings	71	100	0	0	29	11	0	0
Linear isocontainers	94	92	99	61	1	100	98	60
Rectangular isocontainers	94	79	6	100	69	48	0	0
Shipyard cranes	100	100	75	1	1	1	0	0
Forklift	81	74	5	100	33	26	0	0
Ship	67	100	0	0	33	33	0	0
Cars	78	90	9	100	91	11	0	0
Road	89	92	33	75	8	11	43	79
Overall accuracy (%)	90		68		17		62	
Kappa coefficient (κ)	0.90		0.46		0.12		0.36	
z-score (z)	**15.83**		**4.94**		**0.91**		**3.31**	

Table 4.11: Statistical significance test between pair of classifiers for study area 4. The cases which are statistically significant, based on the z-score, are highlighted in bold.

	Study area 4			
	***k*-NN**	**SVM**	**NB**	**Random forest**
k-NN	0	**8.30**	**14.02**	**9.93**
SVM	**8.30**	0	**4.42**	**2.17**
NB	**14.02**	**4.42**	0	**1.88**
Random forest	**9.93**	**2.17**	**1.88**	0

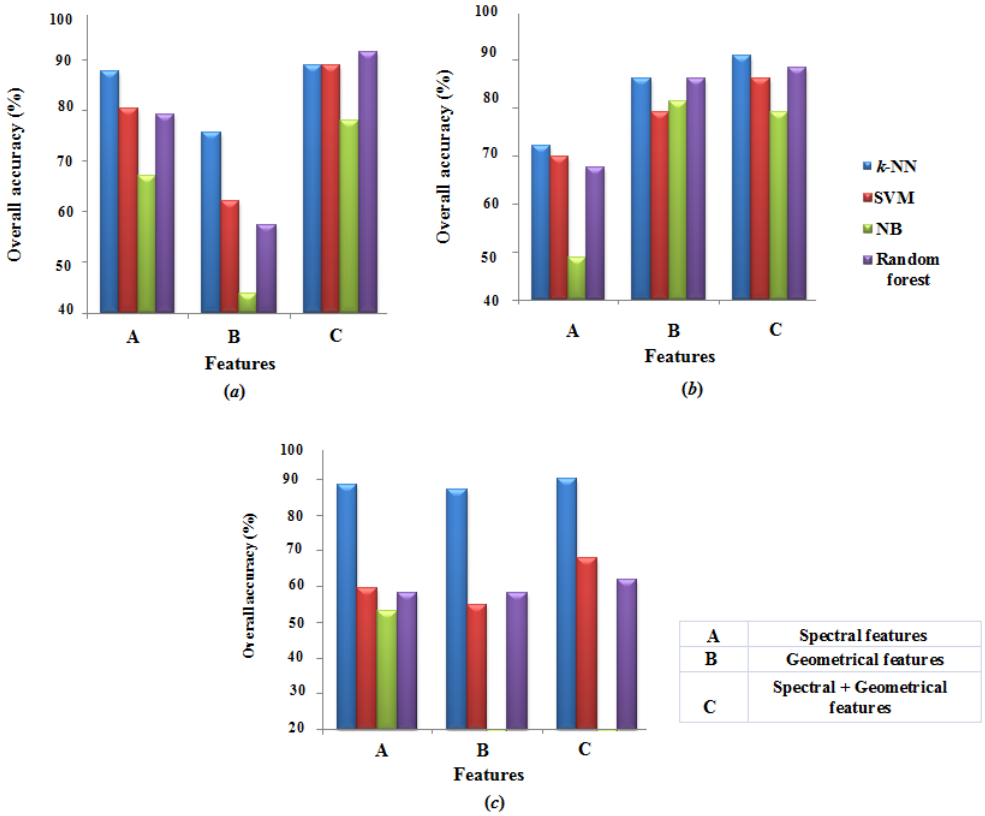


Figure 4.12: Influence of features on the classification performance for: (*a*) study area 1 : inner city and study area 2 : high riser, (*b*) study area 3: residential area, and (*c*) study area 4 : Belgium harbour dataset.

In addition to the inherent limitations of classifiers, in the object-based point cloud labelling approach, there is a possibility of misclassification due to the error in segmentation. This is evident from Fig. 4.13. The linear marine vessels are not properly segmented. The cars parked very close to the buildings are put in the building class. This error in segmentation was carried over to the classification stage as well.

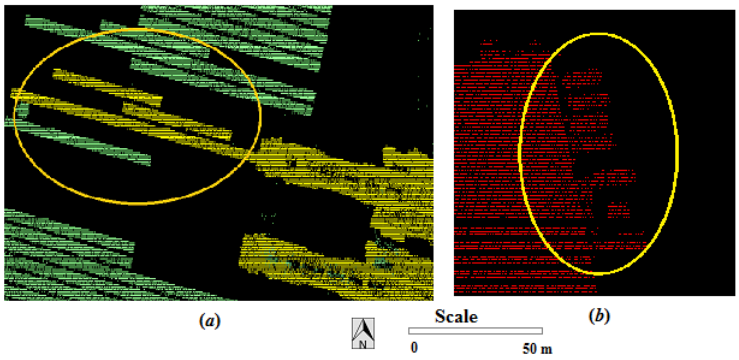


Figure 4.13: Mislabelled point cloud: (*a*) linear marine vessels mislabelled as crane, and (*b*) cars mislabelled as buildings.

Results of the independent evaluation by the ISPRS on study areas $1, 2$ and 3 are shown in Table 4.12. Evaluation results indicates that this methodology is able to extract the buildings with high accuracy, as evident from the high Completeness and Correctness value. In study area 3, pixel level accuracy is greater than 90%. The object level accuracy for all the test cases exceeds 95%. RMS error was calculated to understand the geometric accuracy of the building polygons with respect to the ground reference polygon. For study area 3 with low roof residential building, the RMS error is less than a meter. Amongst the three study areas evaluated, for the study area 1 with complex buildings, the rms error is relatively higher (1.72 m). This might be attributed to the complex building structures. However, results indicate that for building detection the proposed method is highly reliable offering high accuracy estimates. This method thus can be applied for any land cover object detection.

Table 4.12: Evaluation based on ISPRS benchmark for study areas 1, 2 and 3

		Pixel level		**Objects balanced by area**		**Objects larger than 50** m^2		
	Dataset	**Completeness (%)**	**Correctness (%)**	**Completeness (%)**	**Correctness (%)**	**Completeness (%)**	**Correctness (%)**	**RMS (m)**
1	Study area 1	81	82	94	93	89	96	1.26
2	Study area 2	86	90	99	99	100	91	1.00
3	Study area 3	91	76	95	86	95	82	0.72

4.5 Discussion

With the increasing demand for extracting 3D objects true to geometry from the modern high density LiDAR point cloud, semantic labelling has been gaining importance amongst the researchers. Object-based point cloud labelling is an efficient way to label point cloud data. Most of the past studies in this arena have focussed on adapting / proposing new 3D segmentation methods with achieving higher classification accuracy as the criterion of success of the method(s). A critical literature survey indicates that classification accuracy of about 90% is generally obtained from most of the methodologies available. A few studies report further higher classification accuracies of 98% to 100%. However, apart from achieving the highest possible classification accuracy, processing complexity and computational load is an important issue in the operational usage of LiDAR point cloud. LiDAR datasets with moderate point density for several km^2 of land surface area contain several millions of points and demands great computational resources, not available with desktop computers. The methodology framework proposed method in this work is computationally efficient for 3D segmentation and classification of point cloud. In addition, there are relatively fewer number of user dependent tuning parameters making it suitable for point cloud with different land cover geometries.

The classification performance of the proposed approach has been compared with several studies which used the benchmark dataset and deal broadly with 'buildings'detection. Table 4.13 compares the performance of various algorithms for building detection. It should be noted that the assessment has been done using the 2D building boundary. Hence, in our case, we extracted the only building points from the semantically labelled point cloud and projected it onto a raster surface. A morphological operations was used for smoothing the obtained image.

From the Table 4.13, it can be seen that the first three studies indicate highest classification accuracy for both the pixel level and object level validations. However, these studies have used only point cloud and have performed classification at the point level. The next four studies have classified the point cloud at the object level using segmentation method. Of these, two studies have used only the LiDAR point cloud while the other two studies have used both LiDAR point cloud and multispectral images. A comparison of classification accuracy obtained from our method with these studies indicates a numerically lower accuracy estimates from

Table 4.13: Summary of classification accuracies using ISPRS benchmark for study areas 1, 2 and 3

S.No	Author	Pixel-level		Object-level		Objects balanced by area		RMS error (m)	Method	Remarks	Type	Classes
		Comp (%)	Corr (%)	Comp (%)	Corr (%)	Comp (%)	Corr (%)					
1	J. Niemeyer (Niemeyer et al., 2014)	90	93	85	70	100	100	0.83	CRF + Random forest	ALS point cloud	Point based method	Buildings only
2	L. Feng (Feng, 2013)	92	88	80	99	99	100	0.87	Morphological operators + Gradient density	ALS point cloud	Point based method	Buildings only
3	B. Yang (Yang et al., 2014)	90	91	83	98	99	98	0.90	Reversible-jump Markov-chain Monte Carlo	ALS point cloud	Point based method	Buildings only
4	M. Awrangjeb (Awrangjeb et al., 2014)	93	89	83	93	99	100	0.91	Planar Segmentation	ALS point cloud	Segmentation based method	Buildings only
5	P. Dorninger (Dorninger and Pfeifer, 2008)	86	98	80	100	98	100	0.87	Planar Segmentation	ALS point cloud	Segmentation based method method	Buildings only
6	M.Gerke (Gerke and Xiao, 2014)	93	81	85	51	99	89	1.13	Region Growing Segmentation	ALS point cloud & Image	Segmentation based method	Buildings, trees vegetation,ground
7	M.Gerke (Gerke and Xiao, 2014)	91	91	80	71	98	100	0.93	Region Growing Segmentation	ALS point cloud & Image	Segmentation based method	Buildings, trees vegetation,ground
8	Our Approach	86	83	80	77	95	90	0.99	Supervoxel based LCCP segmentation	ALS point cloud & Image	Segmentation based method	Lawn, concrete, flat roof gabled roof, vegetation

our method. It should be noted that the first five studies in the table which report highest accuracy are single-class classification experiments; classified only buildings from the point cloud and filtered out other land cover categories. Whereas, our method has been applied from the 'multi-class'classification perspective as apparent from the eight types of landcover classes considered in different study sites. In principle, multi-class pattern recognition is very complicated compared to binary-class pattern recognition. Confusion in labelling of pixels (points) increases if some of the classes are similar functionally or structurally (e.g. different vegetation types, buildings of different shapes and construction materials etc.) leading to lower classification accuracy (Duda et al., 2012).

Apart from this, relative differences in area extents of different classes have significant bearing on the quality of training phase in classification. As can be seen from the Table 4.4, there is a balance in the dataset (with almost equal distribution of classes) for study area 1 and study area 2 with three classes (flat roof, gabled roof and vegetation). However, study area 3 and study area 4 have an unequal distribution of classes. For instance, study area 3, which mainly comprises of low lying residential buildings, almost 65% of the dataset is dominated by gabled roof. Similarly, in the study area 4 with eight classes, 'linear iso-containers' dominates the scene followed by 'road'. This relatively unbalanced distribution of classes negatively impacts the classification process.

Spatial complexity of the data is another important factor that influences segmentation and classification results. The LCCI values for the different study sites are presented in section 3.3 and 4.3. The values of LCCI supports the apparent complexity of the point cloud pertaining to the presence of several types of land surface objects and different geometrical shapes. The data complexity reduces if the point cloud is filtered out for a single-class mapping, as has been done in the first five studies summarized in Table 4.13. The proposed methodology indicates superior computational performance and fairly good accuracy estimates even when the point cloud has been processed and analysed from a multi-class perspective.

The computational performance of the proposed methodology with the similar studies (Table 4.13) could not be compared as these studies have not reported the computational aspects of the methods used.

The following are the important points that emerge from this part of the work.

(1) Supervoxels-based segmentation is a computationally effective process for segmenting the point cloud. Supervoxels, analogous to superpixels, reduce the computational complexity of the further stages as only the supervoxel vectors are considered rather than the entire point cloud. This increases the segmentation speed. As evident from the results, even for the highly dense dataset (study area 4 with almost 5 million points) the entire segmentation procedure took only several minutes. This is an advantage when processing airborne LiDAR point cloud covering a wide area and with high point density.

(2) When the point density is less (as in study areas 1 and 2), more number of supervoxels get created close to the number of points we actually used. This might be due to the lack of neighbours within the seed resolution. This indicates that for lesser point density and lesser number of points, supervoxels are not necessary and entire dataset can be used as such. In such cases, simple segmentation algorithms such as colour-based region growing algorithm (described in the previous chapter) which demand less computational complexity can be used. However, when the point density increases voxel based cloud connectivity algorithm is very efficient as the number of supervoxels is drastically reduced. For example, in the study area 4, there has been almost 99% reduction in the number of points used for further processing. Moreover, supervoxels are formed taking in to account, the spectral, spatial and geometrical distribution of points.

(3) Local cloud connectivity based algorithm efficiently merges the supervoxels based on convexity criteria. This segmentation approach needs very minimal parameters for merging the supervoxels. The segments produced are meaningful and are close to the real objects.

(4) To achieve a better classification accuracy and thereby a better point cloud labelling, it is important to choose the best features well representing the segments. We have extracted both spectral and geometrical features from the segments created. The results indicate that for datasets with less point density (< 5 points/m^2), complementary information from multi-spectral image is necessary for point cloud labelling. This is an important observation as it indicates that if LiDAR point cloud alone is used with low point density, the labelling might not be reliable. Also, when only spectral features are used, there might be again uncertainty in

the labelling due to the shadows that affect the actual spectral values. Other local descriptors such as 3D texture has the potential to improve the classification accuracy. However, for point cloud dataset with higher point density, as in case of the study area 4, geometric features from the LiDAR point cloud alone may be sufficient to produce the expected labelling accuracy.

(5) Selection of proper classifier is also an important aspect to achieve the desired accuracy. Amongst the different supervised learning techniques used, *k*-NN and Random forest classifiers perform well in most of the cases. Computationally, *k*-NN is more efficient than the other classifiers. However, Random forest classifier gives comparable or better accuracy than the other supervised learning techniques though computationally demanding. The performance of multiclass SVM classifier depends heavily on the kernel used and is comparable with the other classifiers. However, the training class distribution impacts the performance of these classifiers as evident from the results. Both Random forest classifier and SVM shows a slump in the performance when unbalanced dataset is used as in the case of study area 4. From the study, it is evident that the simple *k*-NN classifier performs better in most of the cases irrespective of the balanced or unbalanced dataset.

(6) Having the point cloud labelled to the representative classes is an asset to the data users and decision makers, who can easily visualise their object of interest in 3D. For example, as seen in Fig.5.5, the point cloud representing a particular object such as buildings, shipyard cranes, and ship, alone can be extracted. The extracted point cloud can be reconstructed to produce realistic 3D models.

As future LiDAR systems offer point cloud of extremely high point density, our approach uses supervoxels for processing the dense point cloud at a reasonable amount of time while offering accuracy estimates at par with contemporary methods.

However the methods used for the extraction of spectral information from multispectral images and integrating it with point cloud significantly influences the results. A recent study by Gerke and Xiao (2014) discusses the importance of integrating the colour information with the point cloud and present a method for visibility analysis of point cloud and pixels from orthophoto. This seems useful particularly when the optical images are captured at a comparatively different resolution and viewing geometry. The method used for extraction of multi-

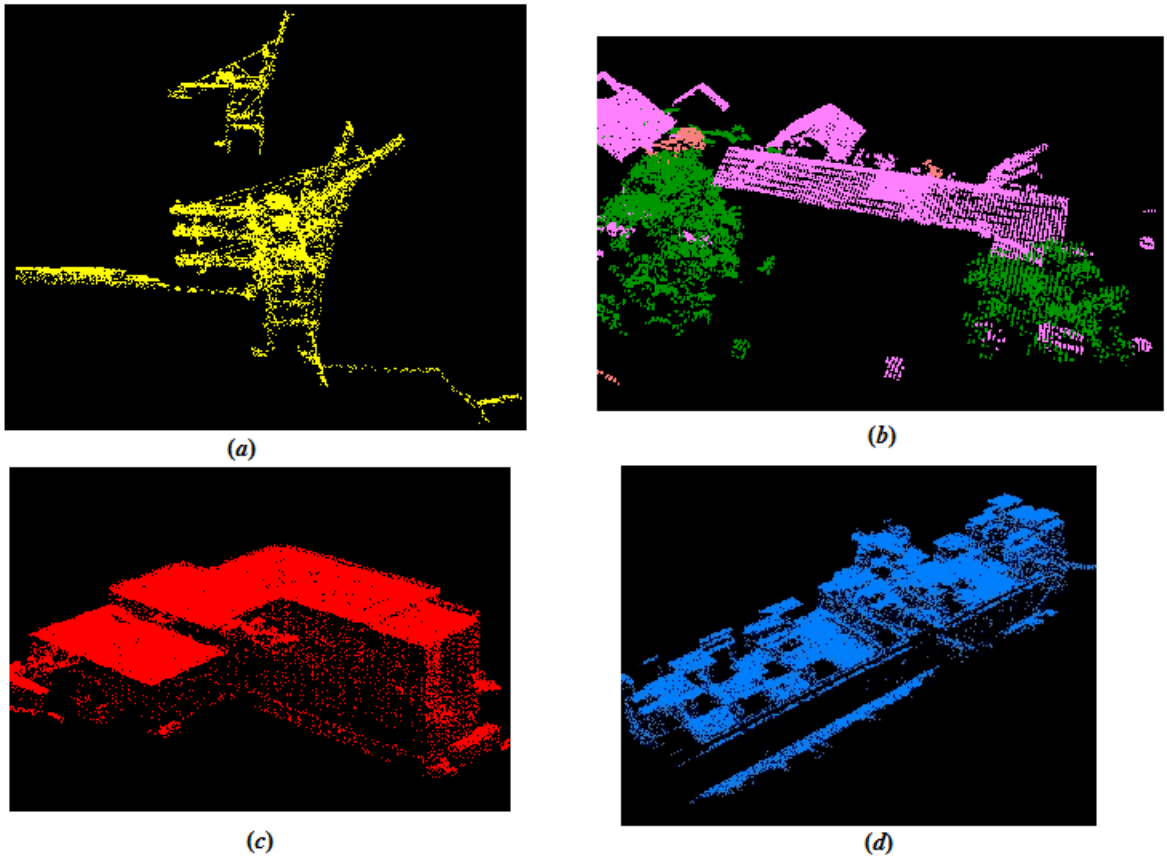


Figure 4.14: Labelled point cloud representing various objects: (*a*) cranes, (*b*) tree and gabled roof, (*c*) building, and (*d*) ship.

spectral values for integrating with point cloud affects the utility of multispectral features in the point cloud classification. The interpolation based method used in this study might not be optimal for point cloud capturing multiple levels of surface objects which would be obscured in multispectral images. Methods which extract spectral values based on visibility analysis (e.g. Gerke and Xiao (2014)) of point cloud may be useful.

4.6 Conclusions

A novel supervoxels-based 3D object-based approach has been developed for semantic labelling of the coloured airborne urban point cloud data. We have used voxel cloud connectivity for creating supervoxels. The supervoxels created using the cloud connectivity algorithm are combined to form segments. The segments are classified with different machine learning algorithms based on various geometrical and spectral features. The proposed methodology has been tested on three different datasets with varying point density. The results indicate that the supervoxels based 3D segmentation approach yields fairly higher classification accuracy with less computational demand. Especially, when the point density is high, the proposed algorithm gives accurate classification results with low computational demand. 3D solid models for the objects of interest can be directly created from the semantically labelled point cloud.

4.7 Chapter Summary

Computational efficiency and accuracy significantly depends upon the methods used for 3D segmentation of the point cloud. In this chapter, we have proposed a novel 3D object-based framework using a supervoxel-based LCCP segmentation approach for direct labelling of the coloured point cloud. Spectral and geometric feature vectors are extracted from the segments and used for classification using various machine learning algorithms. The effect of point density and the number of features on the classification accuracy has also been studied. The results reveal that supervoxels-based approach significantly reduces the computational complexity in the segmentation stage when used for LiDAR datasets with high point density (> 5 points/m^2).

Another important result that emerge from this study is that for LiDAR datasets with low point density, spectral information from optical images complements the geometric information inherent to the LiDAR, and thus enhancing the semantic labelling accuracy.

CHAPTER 5

CRITICAL ASSESSMENT OF SEMANTIC OBJECT-BASED POINT CLOUD LABELLING ON URBAN LIDAR DATASET

Prelude: This chapter critically assess the role of different parameters such as segmentation, dataset used, computational complexity adopted for object-based semantic labelling of urban point cloud data. The designed experiments were tested on the datasets provided by the ISPRS as a part of 3D semantic labelling challenge. The results of labelling at individual point level were independently evaluated by the ISPRS. This chapter also has been organized as self contained for ease of readability and ready reference.

5.1 Introduction

Advancements in the processing power of computers have paved way for the creation of 3D maps which offer realistic details compared to the traditional 2D maps. This is more evident from the launch of 3D maps from corporations such as Google and Apple. Geospatial Technology is considered to be the main source of data for producing 3D maps ranging from ground survey to stereo images. Tremendous progress made during the past decade indicates that LiDAR remote sensing is emerging as the main source of 3D spatial data because of its very dense and discrete point cloud which are ideal for creating accurate 3D models.

Automated extraction of various urban objects true to geometry with high precision from very high resolution images and LiDAR is one of the challenging tasks that is yet to be solved by the remote sensing community (Mayer, 2008; Niemeyer et al., 2014). The heterogeneous appearance of various urban objects in very high resolution images makes the intra-class variance high, though the inter-class variance remains low. Though there is advancement in processing techniques with mature machine learning algorithms, recently the International Society for

Photogrammetry and Remote Sensing (ISPRS, 2016) has reported that there is no fully automated technique available for labelling - either in 2D (very high resolution images) or 3D domain (LiDAR point cloud). A semantic labelling contest has been initiated to address this issue by the 'ISPRS WG III/4 : 3D scene Analysis ' by providing standard benchmark datasets. While this contest focusses on airborne LiDAR data set, a similar initative by the IQumuls (a high-volume fusion and analysis platform for geospatial point clouds, coverages and volumetric data sets) focuses on mobile datasets. The results submitted to these contests by researchers in the international arena are being actively validated against the ground truth.

Semantic labelling of the urban point cloud into various urban objects is very much essential for technology based urban planning and development. Geometrically similar classes can be discriminated with better accuracy when complementary spectral information from multispectral images is used as reported in the previous chapters. As discussed in the chapter 2, besides the range information, LiDAR sensors are capable of recording the amplitude of the backscattered laser pulse or the laser intensity. The intensity data corresponds to the reflectance of object at the near-infrared wavelength (for topographic applications). There are few studies reported mostly in robotics for navigation purpose, which has used intensity information for distinguishing road patch from grass surface (Laible et al., 2013; Wurm et al., 2009). However, there are very limited studies which explore the potential of the backscattered intensity on the urban object classification using airborne laser scanner data (Liu, 2009). The study by Jutzi and Gross (2009) reports that the intensity value recorded by the sensor is severely affected by parameters such as atmospheric transmission, local incidence angle, and the distance between the sensor and the target. This limits its utilization while processing airborne LiDAR point cloud. A recent study by Aijazi et al. (2013) has reported that while processing complex urban point cloud with multiple classes, colour and intensity plays an important role in discriminating geometrically related classes. The study was carried out using ground-based LiDAR point cloud. However, the potential of the complementary information on the 3D labelling of airborne LiDAR datasets is still unknown.

In this thesis, two different segmentation approaches - colour-based region growing segmentation and a computationally efficient supervoxels-based local convexity connectivity patches have been used to create 3D segments as part of the object-based point cloud labelling frame-

work. This chapter studies the effect of these segmentation algorithms on the labelling accuracy.

The aim of this part of the work is to rigorously assess the degree of accuracy in 3D semantic labelling of urban LiDAR point cloud due to the effect of various computational parameters such as data fusion, segmentation approach, and computational complexity. The study was carried out using the benchmark airborne LiDAR dataset provided by the ISPRS. The results were independently evaluated by ISPRS. Though the bench mark dataset for 3D semantic labelling was initiated by the ISPRS during the later half of 2015, it should be noted that, given the recent emphasis on this topic and the challenges involved, to our knowledge, there is no participation in the '3D semantic labelling contest' from researchers except this study until now (September 2016). However, there has been a strong response with several participants in the '2D labelling contest' where 2D outputs are generated.

5.2 Methodology

A brief methodological framework is given in Figure 5.1. The study was carried out using three types of datasets : (*a*) LiDAR point cloud data with only the geometrical information (XYZ), (*b*) LiDAR point cloud with the backscattered intensity (XYZI), and (*c*) LiDAR point cloud combined with spectral information from the optical images(XYZRGB)

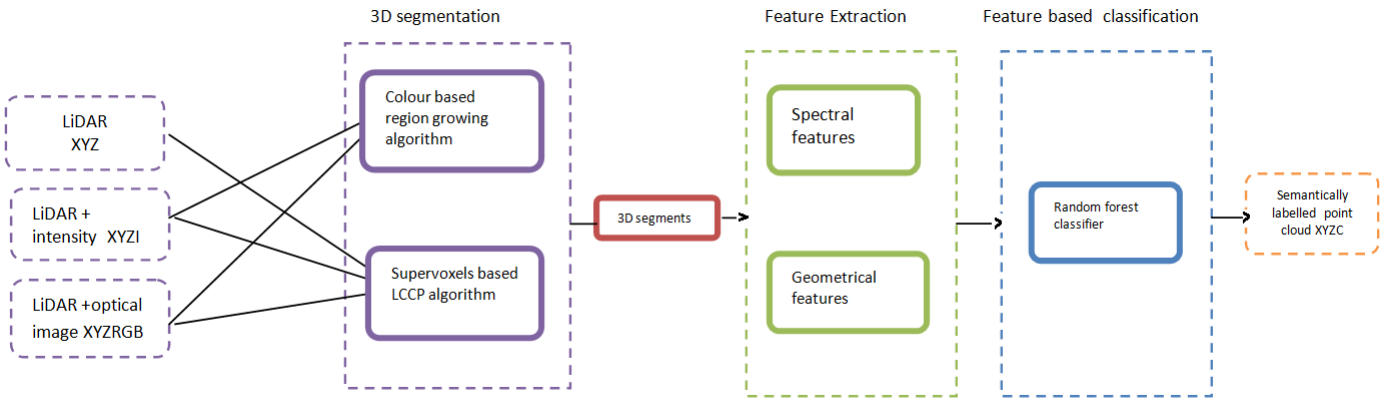


Figure 5.1: Methodological workflow.

In order to critically evaluate the influence of segmentation algorithms (colour-based region growing algorithm and supervoxels-based local cloud connectivity algorithm) on the different datasets, five test cases were developed as given in Table 5.12.

Table 5.1: Test cases evaluating different approaches

Test Case	**Segmentation approaches**	**Dataset**
1	Colour-based region growing	XYZI
2	Colour-based region growing	XYZRGB
3	LCCP-based region growing	XYZ
4	LCCP-based region growing	XYZI
5	LCCP-based region growing	XYZRGB

The 3D segments were generated was used to extract spectral and geometrical features which was further used for classification as given in section 3.4.4. A feature-based supervised learning using random forest classifier was adopted as described in the section 4.2.4.

5.2.1 Evaluation strategy

Results of semantic labelling were independently evaluated by the International Society of Photogrammetry and Remote Sensing (ISPRS) with the ground truth data as part of '3D semantic challenge'. The evaluation was based on computation of point-level confusion matrix and relative confusion matrix for all the nine classes. Based on the matrices the accuracy assessment measures described in equation 3.9 are computed. In addition, harmonic mean of precision and recall popularly known as F1 score, is also computed.

Based on the confusion matrix, the statistical significance between the different test cases were computed based on the methodology described in section 4.2.5.

5.3 Dataset used

Experiments were performed on LiDAR datasets and their corresponding orthophotos provided as part of 'ISPRS Test Project on Urban Classification, 3D Building Reconstruction and Semantic Labelling' by the ISPRS working group III/4 '3D scene analysis'.

The dataset was captured over Vaihingen/Enz as part of German Camera Evaluation Project initiated by the German Society of Photogrammetry, Remote Sensing and Geoinformation (DGPF) in summer 2008. Leica ALS50 system flying at an altitude of 500 m was used to capture dense LiDAR point cloud with point density of around 8 points $/m^2$. Corresponding orthophoto was acquired using Intergraph - DMC camera with a 8 cm ground resolution.

As part of 3D semantic labelling challenge, two subsets of the data were provided - a training dataset and a test dataset.

The training dataset, with 753876 points, comprises both complex buildings as well as residential buildings. Labels of nine urban classes namely powerline, low vegetation, impervious surfaces, car, fence / hedge, roof, facade, shrub, tree were provided as part of the training dataset. The test dataset is from the city center which includes densely developed areas consisting of historic buildings with complicated shapes and few trees. Another part of the test data have small detached houses surrounded by trees. Test dataset consists of 411722 points. The details of the study area is summarised in Table 5.2

Table 5.2: Details of ISPRS 3D semantic labelling challenge dataset

Parameters / dataset	**Training dataset**	**Testing dataset**
Point density (points /m^2)	8	8
Number of 1st return	753876	411722
Number of 2nd return	0	0
Total number of points	753876	411722
x range (m)	170	200
y range (m)	170	200
Land cover complexity index	0.47	0.48

Figure 5.2: Training dataset with nine training classes.

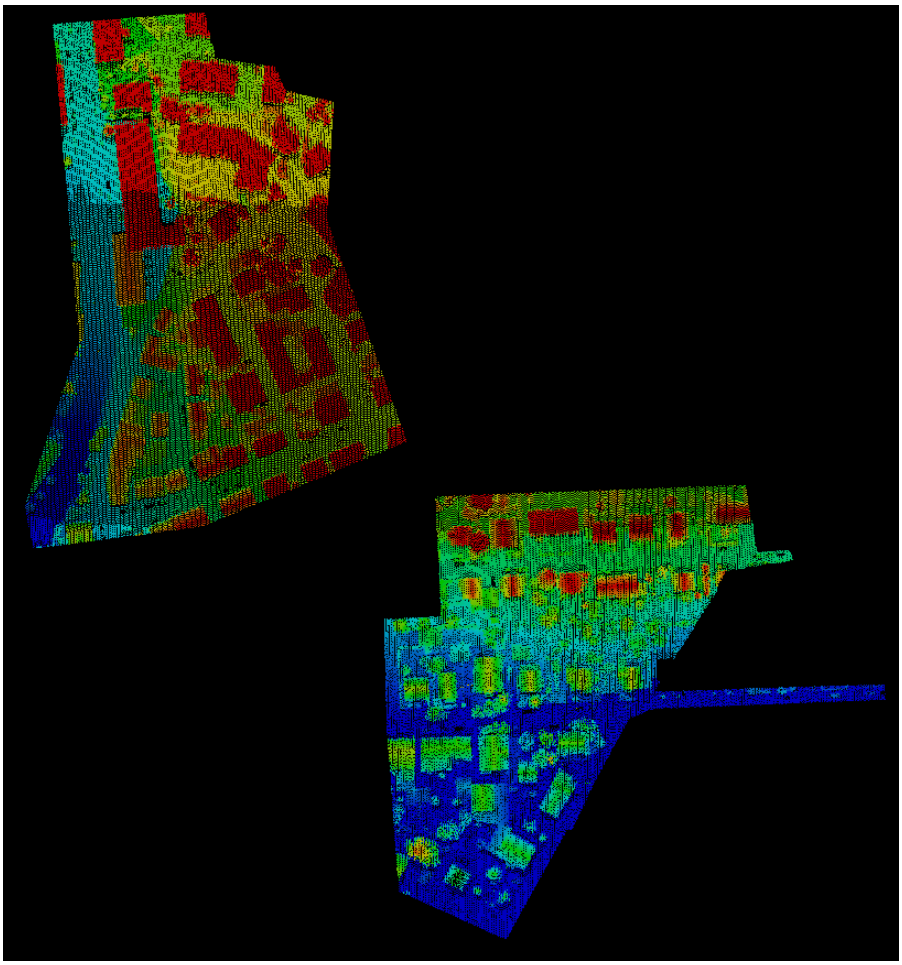


Figure 5.3: Test dataset

Figure 5.4: Orthophoto (multispectral image)

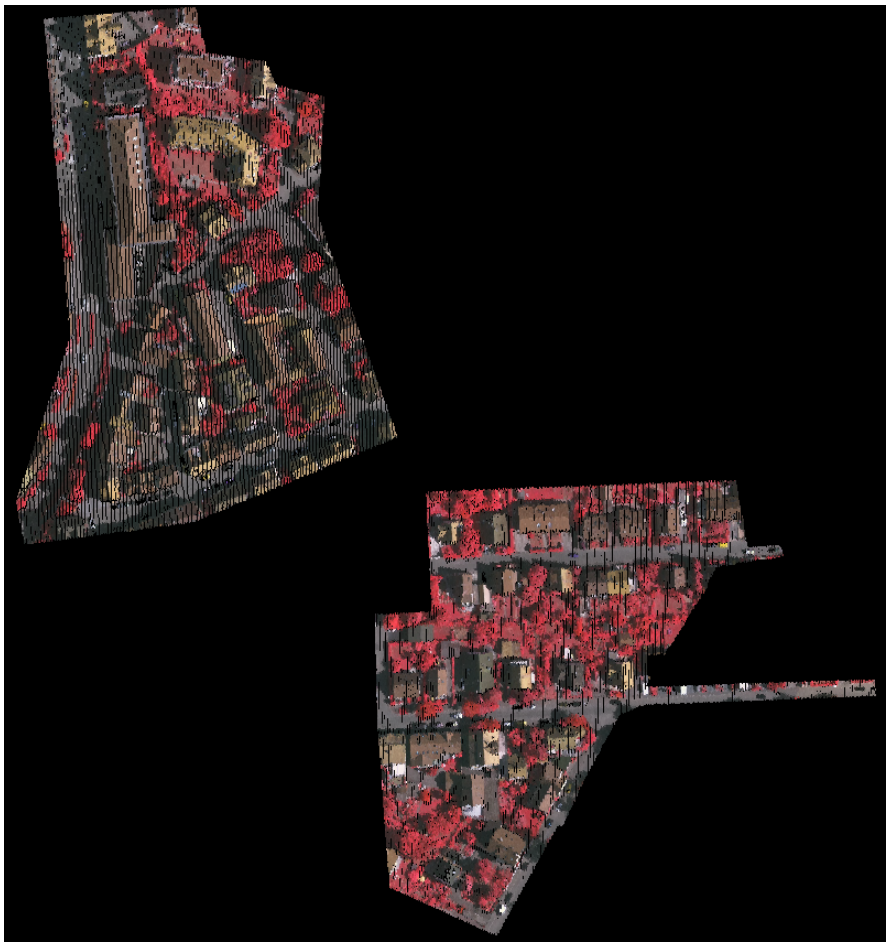


Figure 5.5: Coloured LiDAR point representing the test dataset.

5.4 Results and Analysis

5.4.1 Segmentation

The results of segmentation on the various test cases are given in Table 5.3. As it can be seen from the table, for the same test dataset, when colour-based region growing algorithm is used, more segments are produced. For test cases 1 and 2, there were 4117 and 6919 segments created using colour-based region growing segmentation algorithm. However, the number of segments created have drastically reduced to 1553, 1516 and 1515 respectively for test cases 3, 4, and 5 segments when supervoxels-based LCCP algorithm was used. A close examination indicates that the colour-based region growing segmentation results in oversegmentation as evident from Figure 5.6. The size of segments is a key factor for geometric feature extraction. Most of the segments resulting from LCCP based segmentation are based on the geometrical distribution of the point cloud and hence extracted geometric features are meaningful.

Table 5.3: Result of segmentation for various test cases.

Test case	**Total number of supervoxels**	**Total number of segments**	**Total time (seconds)**
1	—	4117	563.38
2	—	6919	888.14
3	11861	1553	214.62
4	11677	1516	215.06
5	11726	1515	213.45

The computational time is given in Table 5.3. As it can be seen, for test case 1, where colour-based region growing algorithm was applied on the LiDAR dataset with intensity, the total time taken was 563.38 seconds. The computational time increased to 888.14 seconds when the spectral information was used along with the LiDAR point cloud as in test case 2. However, the computational time reduced by almost four times when the supervoxels based LCCP algorithm was used. For test cases 3, 4 and 5, the time taken for creating 3D segments

are 214.62, 215.06 and 213.45 seconds respectively.

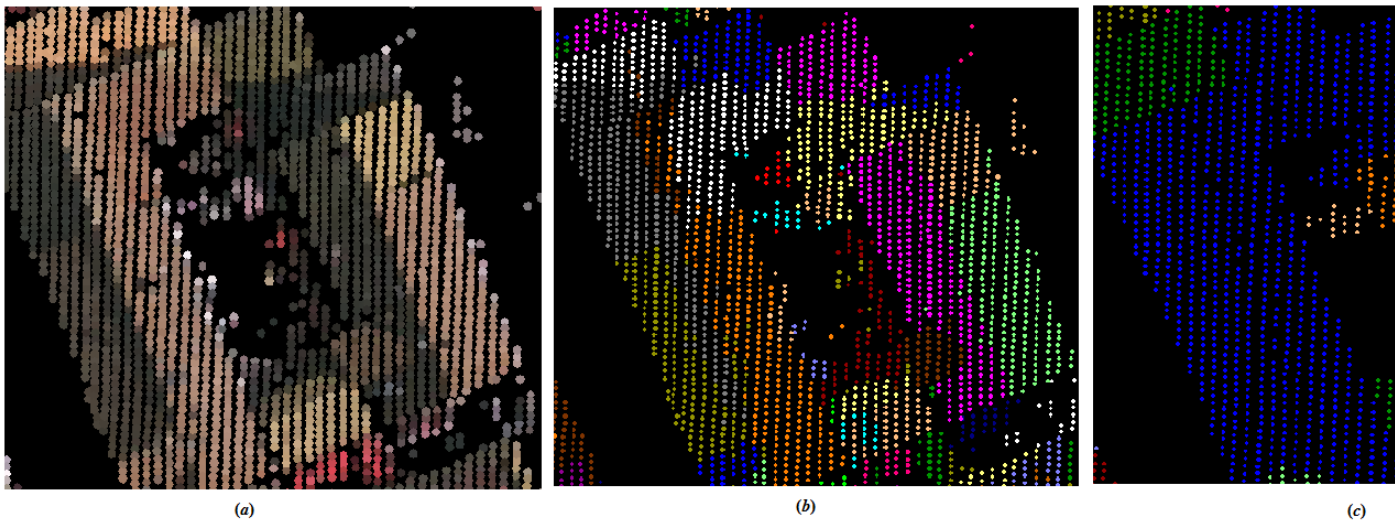


Figure 5.6: Results of segmentation of a building: (*a*) coloured point cloud data, (*b*) colour based region growing segmentation, and (*c*) supervoxels-based LCCP segmentation.

5.4.2 Classification

The results of feature based classification were independently evaluated for all the test cases by the International Society of Photogrammetry and Remote Sensing (ISPRS) against the ground truth as part of the 3D semantic labelling contest. The results of classification of the test dataset into nine classes are given in Figures 5.7 and 5.8. Tables 5.4 through 5.13 give the absolute and normalised confusion matrices which indicates the accuracy of labelling of individual classes. The values in the absolute confusion matrices correspond to the actual number of points that are classified to a particular class. In the normalised confusion matrix, the cells are normalised with respect to the reference class.

From the results, it is evident that the overall accuracy is maximum (76 %) for test case 5 in which supervoxels-based LCCP segmentation was applied to coloured point cloud dataset. The individual class accuracies for the different classes by various methods are shown in Figure 5.9. The performance of various algorithms for different classes is summarized in Table 5.14. The results indicate that most of the classes are best detected in test case 5 (supervoxels-based LCCP

Table 5.4: Confusion matrix with absolute values - Test Case 1

	reference class →										
predicted class ↓	**power**	**low vegetation**	**impervious surface**	**car**	**fence hedge**	**roof**	**facade**	**shrub**	**tree**	**missing points**	**FN**
power	0	0	0	23	56	472	0	13	36	0	600
low vegetation	0	54757	23303	1625	2366	11747	0	4076	300	501	43918
impervious surface	0	10305	90143	59	166	979	0	270	29	35	11843
car	0	183	26	253	856	1390	0	908	87	5	3455
fence hedge	0	127	176	563	402	3138	0	2828	79	9	7020
roof	0	595	93	2072	2092	85702	0	17313	1065	106	23336
facade	0	363	153	923	345	7345	0	1616	432	47	11224
shrub	0	1343	455	2831	2337	11158	0	6262	346	76	18546
tree	0	566	59	5628	1368	35535	0	8698	2265	95	51949

Table 5.5: Confusion matrix with normalised values - Test Case 1

	reference class →									
predicted class ↓	**power**	**low vegetation**	**impervious surface**	**car**	**fence hedge**	**roof**	**facade**	**shrub**	**tree**	**missing points**
power	—	0	0	0.038	0.093	0.787	—	0.022	0.06	0
low vegetation	—	0.555	0.236	0.016	0.024	0.119	—	0.041	0.003	501
impervious surface	—	0.101	0.884	0.001	0.002	0.01	—	0.003	0	35
car	—	0.049	0.007	0.068	0.231	0.375	—	0.245	0.023	5
fence hedge	—	0.017	0.024	0.076	0.054	0.423	—	0.395	0.011	9
roof	—	0.005	0.001	0.019	0.019	0.786	—	0.159	0.01	106
facade	—	0.032	0.014	0.082	0.031	0.654	—	0.144	0.038	47
shrub	—	0.054	0.018	0.114	0.094	0.45	—	0.252	0.014	76
tree	—	0.01	0.001	0.104	0.025	0.655	—	0.16	0.042	95
Precision/ Correctness	—	0.802	0.788	0.018	0.0.04	0.544	—	0.149	0.488	
Recall/ Completeness	—	0.555	0.884	0.068	0.054	0.786	—	0.252	0.042	
F1 score	—	0.656	0.833	0.029	0.046	0.643	—	0.187	0.077	
Overall Accuracy (%)	58									

Table 5.6: Confusion matrix with absolute values - Test Case 2

	reference class →										
predicted class ↓	**power**	**low vegetation**	**impervious surface**	**car**	**fence hedge**	**roof**	**facade**	**shrub**	**tree**	**missing points**	**FN**
power	0	0	0	0	40	520	0	24	16	0	600
low vegetation	0	67781	10279	219	3994	1900	0	13522	479	501	30894
impervious surface	0	16196	84252	65	305	222	0	907	4	35	17734
car	0	188	21	1393	759	532	0	810	0	5	2315
fence hedge	0	212	91	138	3093	359	0	3443	77	9	4329
roof	0	239	449	755	1020	99036	0	7266	167	106	10002
facade	0	332	184	193	799	7809	0	1850	10	47	11224
shrub	0	1632	166	402	4362	2105	0	15679	386	76	9129
tree	0	493	132	59	5291	8776	0	17211	22091	95	32123

Table 5.7: Confusion matrix with normalised values - Test Case 2

	reference class →									
predicted class ↓	**power**	**low vegetation**	**impervious surface**	**car**	**fence hedge**	**roof**	**facade**	**shrub**	**tree**	**missing points**
power	—	0	0	0	0.067	0.867	—	0.04	0.027	0
low vegetation	—	0.687	0.104	0.002	0.04	0.019	—	0.137	0.005	501
impervious surface	—	0.159	0.826	0.001	0.003	0.002	—	0.009	0	35
car	—	0.051	0.006	0.376	0.205	0.143	—	0.218	0	5
fence hedge	—	0.029	0.012	0.019	0.417	0.048	0	0.464	0.01	9
roof	—	0.002	0.004	0.007	0.009	0.908	—	0.067	0.002	106
facade	—-	0.03	0.016	0.017	0.071	0.696	—	0.165	0.001	47
shrub	0	0.066	0.012	0.019	0.176	0.085	—	0.632	0.016	76
tree	0	0.009	0.002	0.001	0.098	0.162	—	0.319		95
Precision/ Correctness	—	0.778	0.882	0.432	0.157	0.817	—	0.258	0.951	
Recall/ Completeness	—	0.687	0.826	0.376	0.417	0.908	—	0.632	0.407	
F1 score	—	0.73	0.853	0.402	0.228	0.86	—	0.366	0.571	
Overall Accuracy (%)	71									

Table 5.8: Confusion matrix with absolute values - Test Case 3

	reference class →										
predicted class ↓	**power**	**low vegetation**	**impervious surface**	**car**	**fence hedge**	**roof**	**facade**	**shrub**	**tree**	**missing points**	**FN**
power	245	0	0	0	0	186	0	6	5	158	355
low vegetation	0	38386	39120	1113	986	2820	0	14048	42	2151	60280
impervious surface	0	1375	99027	206	66	415	0	560	0	331	2953
car	0	2	207	2458	139	120	0	485	0	297	1250
fence hedge	0	24	262	397	1071	632	0	4873	19	144	6351
roof	19	13	653	572	0	103364	0	3394	433	591	5675
facade	17	214	234	363	17	7168	0	1515	24	1667	11219
shrub	1	623	1141	725	298	3681	0	17108	518	703	7690
tree	19	420	153	189	260	5198	0	22405	23959	1611	30255

Table 5.9: Confusion matrix with normalised values - Test Case 3

	reference class →									
predicted class ↓	**power**	**low vegetation**	**impervious surface**	**car**	**fence hedge**	**roof**	**facade**	**shrub**	**tree**	**missing points**
power	0.408	0	0	0	0	0.31	—	0.01	0.008	158
low vegetation	0	0.389	0.396	0.011	0.01	0.029	—	0.142	0	2151
impervious surface	0	0.013	0.971	0.002	0.001	0.004	—	0.005	0	331
car	0	0.001	0.056	0.663	0.037	0.032	—	0.131	0	297
fence hedge	0	0.003	0.035	0.053	0.144	0.085	—	0.657	0.003	144
roof	0	0	0.006	0.005	0	0.948	—	0.031	0.004	591
facade	0.002	0.019	0.021	0.032	0.002	0.639	—	0.135	0.002	1667
shrub	0	0.025	0.046	0.029	0.012	0.148	—	0.69	0.021	703
tree	0	0.008	0.003	0.003	0.005	0.096	—	0.413	0.442	1611
Precision/ Correctness	0.814	0.935	0.703	0.408	0.378	0.836	—	0.266	0.958	
Recall/ Completeness	0.408	0.389	0.971	0.663	0.144	0.948	—	0.69	0.442	
F1 score	0.544	0.549	0.816	0.505	0.209	0.889	—	0.384	0.605	
Overall Accuracy (%)	71									

Table 5.10: Confusion matrix with absolute values - Test Case 4

	reference class →										
predicted class ↓	**power**	**low vegetation**	**impervious surface**	**car**	**fence hedge**	**roof**	**facade**	**shrub**	**tree**	**missing points**	**FN**
power	245	0	0	21	0	134	0	0	43	157	355
low vegetation	0	43637	33825	1838	2823	226	0	14090	28	2199	55029
impervious surface	0	2349	99038	307	234	28	0	665	10	349	3942
car	0	3	206	2445	490	1	0	266	0	297	1263
fence hedge	0	30	262	328	2188	41	0	4343	92	138	5234
roof	19	493	173	1320	1720	83290	0	8046	13368	610	25749
facade	17	233	215	1526	885	2647	0	2784	1230	1682	11219
shrub	1	547	1221	1476	1875	665	0	17700	612	701	7098
tree	19	236	337	1607	962	3562	0	18694	27142	1653	27070

Table 5.11: Confusion matrix with relative values - Test Case 4

	reference class →									
predicted class ↓	**power**	**low vegetation**	**impervious surface**	**car**	**fence hedge**	**roof**	**facade**	**shrub**	**tree**	**missing points**
power	0.408	0	0	0.035	0	0.223	—	0	0.072	157
low vegetation	0	0.442	0.343	0.019	0.029	0.002	—	0.143	0	2199
impervious surface	0	0.023	0.961	0.003	0.002	0	—	0.007	0	349
car	0	0.001	0.056	0.659	0.132	0	—	0.072	0	297
fence hedge	0	0.004	0.035	0.044	0.295	0.006	—	0.585	0.012	138
roof	0	0.005	0.002	0.012	0.016	0.764	—	0.074	0.123	610
facade	0.002	0.021	0.019	0.136	0.079	0.236	—	0.248	0.11	1682
shrub	0	0.022	0.049	0.006	0.076	0.027	—	0.714	0.025	701
tree	0	0.004	0.006	0.003	0.018	0.066	—	0.345	0.501	1653
Precision/ Correctness	0.814	0.918	0.73	0.225	0.196	0.919	—	0.266	0.638	
Recall/ Completeness	0.408	0.442	0.961	0.659	0.295	0.764	—	0.714	0.501	
F1 score	0.544	0.597	0.83	0.335	0.235	0.834	—	0.387	0.561	
Overall Accuracy (%)	71									

Table 5.12: Confusion matrix with absolute values - Test Case 5

	reference class →										
predicted class ↓	**power**	**low vegetation**	**impervious surface**	**car**	**fence hedge**	**roof**	**facade**	**shrub**	**tree**	**missing points**	**FN**
power	245	0	0	0	11	177	0	0	9	158	355
low vegetation	0	49267	28223	25	3346	1278	0	12326	2033	2168	49399
impervious surface	0	1979	98408	27	212	85	0	915	5	349	3572
car	0	2	207	1730	650	56	0	663	103	297	1978
fence hedge	0	24	268	0	2931	312	0	2785	972	130	4491
roof	19	5	661	60	240	104868	0	2127	464	595	4171
facade	17	220	225	369	638	5802	0	1897	375	1676	11219
shrub	1	695	1073	41	3216	3071	0	12896	3159	700	11902
tree	19	156	417	16	1647	6146	0	6879	37297	1635	16915

Table 5.13: Confusion matrix with relative values - Test Case 5

	reference class →									
predicted class ↓	**power**	**low vegetation**	**impervious surface**	**car**	**fence hedge**	**roof**	**facade**	**shrub**	**tree**	**missing points**
power	0.408	0	0	0	0.018	0.295	—	0	0.015	158
low vegetation	0	0.499	0.286	0	0.034	0.013	—	0.125	0.021	2168
impervious surface	0	0.019	0.965	0	0.002	0.001	—	0.009	0	349
car	0	0.001	0.056	0.467	0.175	0.015	—	0.179	0.028	297
fence hedge	0	0.003	0.036	0	0.395	0.042	—	0.375	0.131	130
roof	0	0	0.006	0.001	0.002	0.962	—	0.02	0.004	595
facade	0.002	0.02	0.02	0.033	0.057	0.517	—	0.169	0.033	1676
shrub	0	0.028	0.043	0.002	0.13	0.122	—	0.52	0.127	700
tree	0	0.003	0.008	0	0.03	0.113	—	0.127	0.688	1635
Precision/ Correctness	0.814	0.941	0.76	0.763	0.227	0.861	—	0.319	0.84	
Recall/ Completeness	0.408	0.499	0.965	0.467	0.395	0.962	—	0.52	0.688	
F1 score	0.544	0.652	0.85	0.579	0.289	0.909	—	0.395	0.756	
Overall accuracy (%)	76									

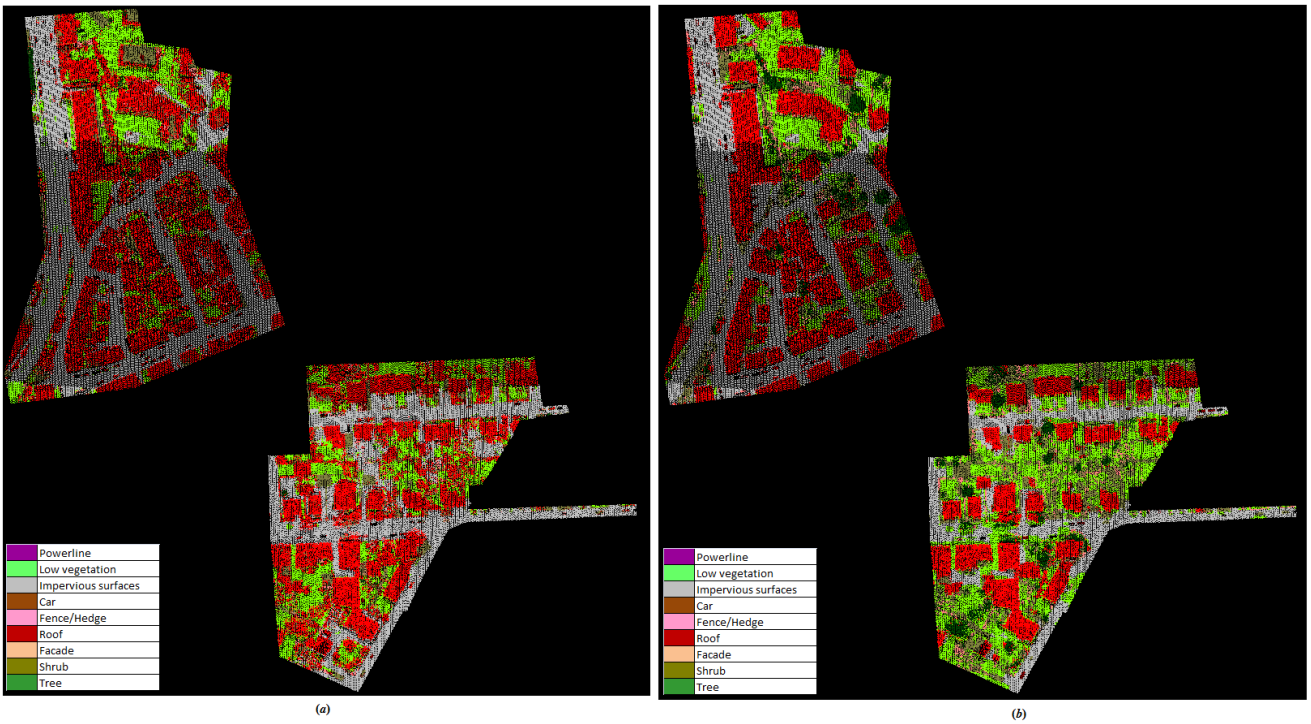


Figure 5.7: Results of classification: (*a*) test case 1, and (*b*) test case 2.

segmentation on the coloured LiDAR point cloud). The results show that the man-made classes are better distinguished than natural classes. Amongst the different classes present, buildings and impervious surfaces are detected with better accuracy compared to the other classes with almost 96% recall rate. Cars and powerlines are detected with precision rate of 76%, and 81% respectively. Amongst the vegetation classes, trees are classified with almost 96% precision and 76% class accuracy. The discrimination rates are lower in the case of shrubs and fence/hedge. Table 5.15 shows the statistical difference in the classification results based on the confusion matrices arrived from the two different test cases. The result shows that at confidence interval 95%, all the test cases are statistically different.

5.5 Discussion

With the advancement of remote sensing technologies capable of capturing the earth surface at very fine resolution, there is a demand for fully automatic algorithms for extracting various earth surface features. Despite the developments in image processing and computer vision,

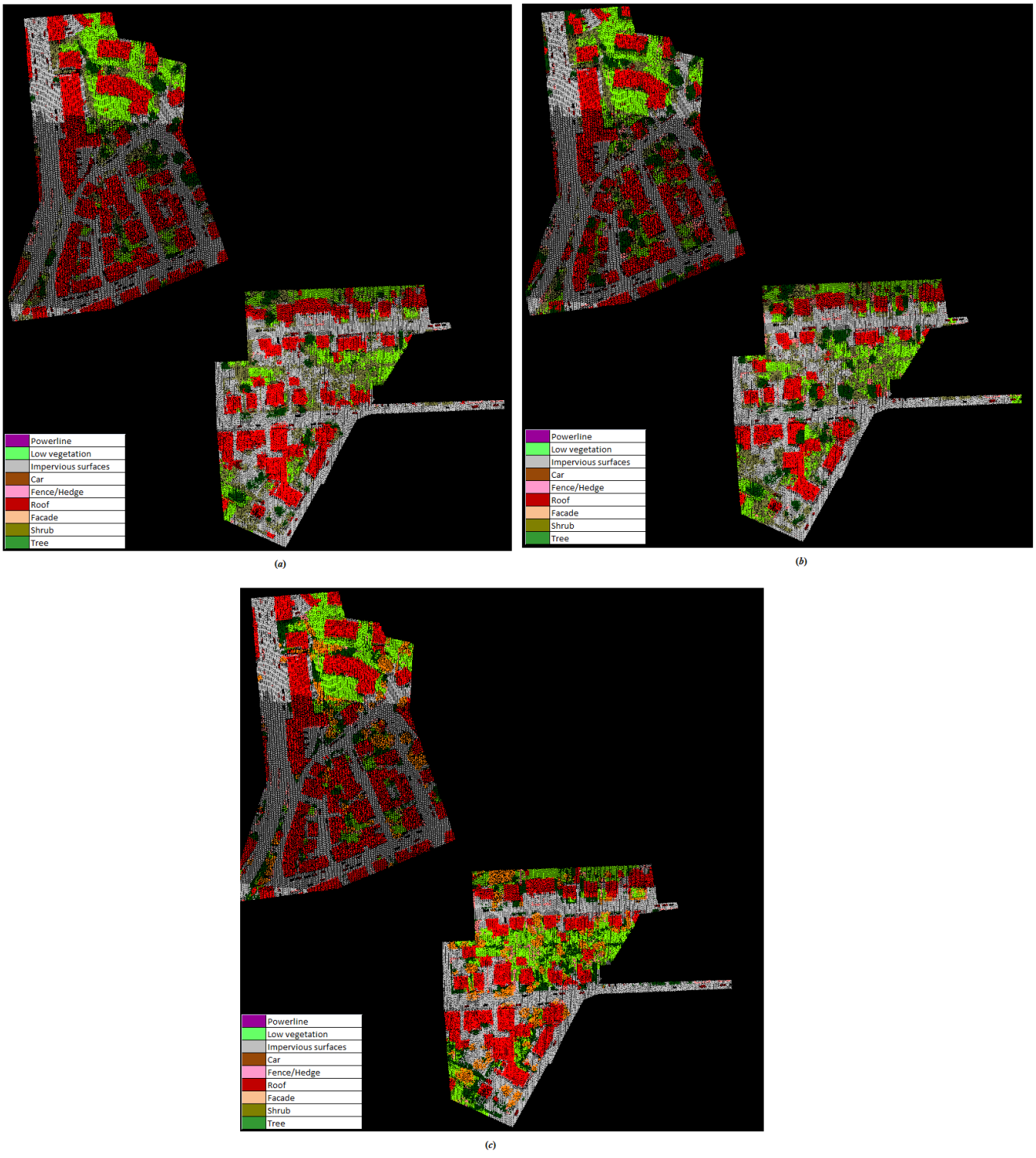


Figure 5.8: Results of classification: (*a*) test case 3, (*b*) test case 4, and (*c*) test case 5.

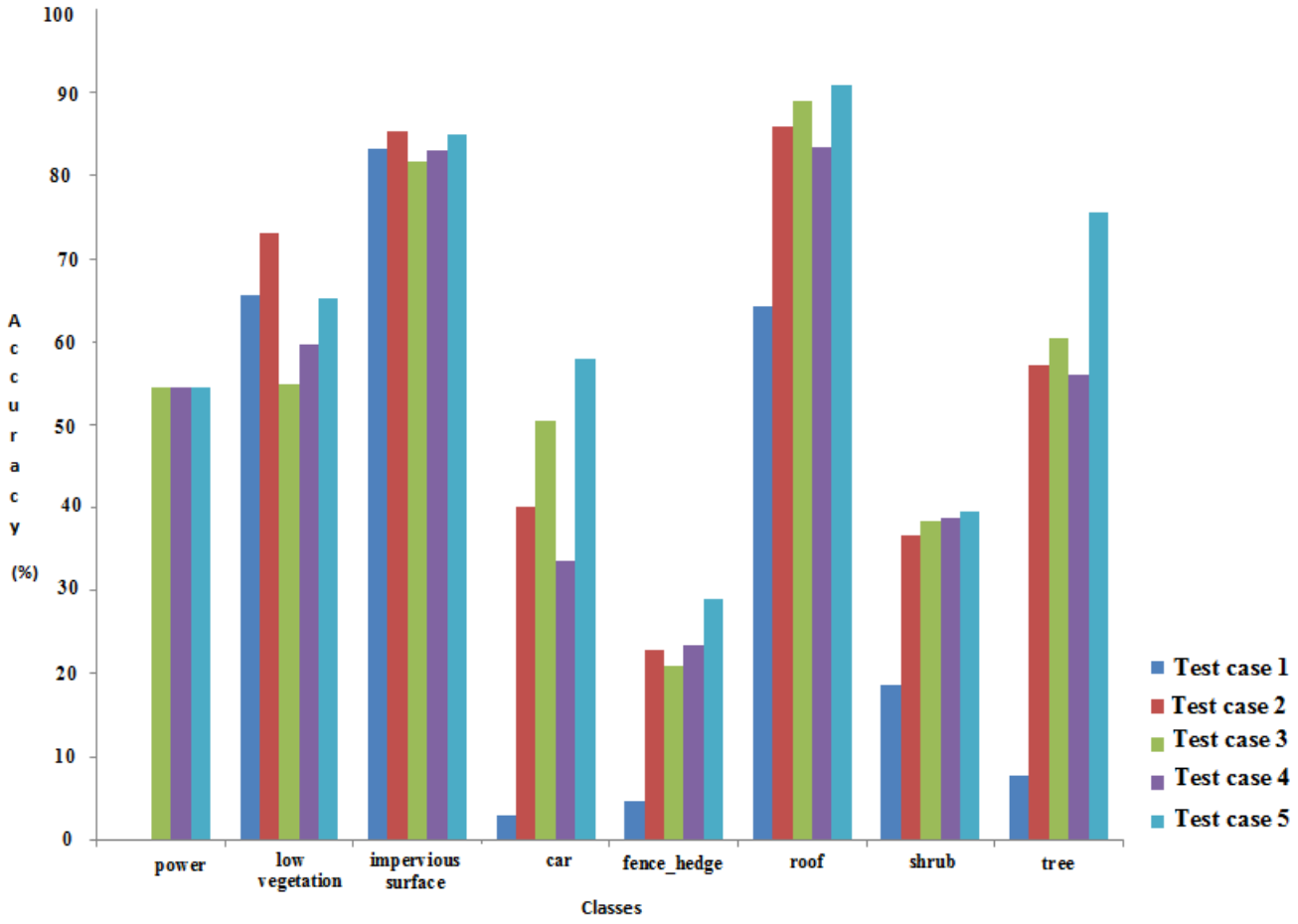


Figure 5.9: Per-class accuracy of classes in different test cases

Table 5.14: Minimum and maximum per-class accuracy, precision, and recall

Classes	Accuracy		Precision / Correctness		Recall / Completeness	
	max (%)	min (%)	max (%)	min (%)	max (%)	min (%)
low vegetation	Test case 2 (73)	Test case 3 (60)	Test case 5 (94)	Test case 2 (78)	Test case 2 (69)	Test case 2 (39)
impervious surface	Test case 5 (85)	Test case 3 (82)	Test case 2 (88)	Test case 3 (73)	Test case 5 (97)	Test case 2 (38)
car	Test case 5 (58)	Test case 1 (3)	Test case 5 (76)	Test case 1 (2)	Test case 3 (70)	Test case 2 (38)
fence hedge	Test case 5 (29)	Test case 1 (5)	Test case 3 (38)	Test case 1 (4)	Test case 2 (48)	Test case 1 (5)
roof	Test case 5 (91)	Test case 1 (64)	Test case 4 (92)	Test case 1 (54)	Test case 5 (96)	Test case 4 (76)
shrub	Test case 5 (40)	Test case 1(19)	Test case 5 (32)	Test case 1 (15)	Test case 4 (71)	Test case 1 (24)
tree	Test case 5 (76)	Test case 1 (8)	Test case 3 (96)	Test case 1 (72)	Test case 5 (69)	Test case 1 (50)

Table 5.15: Statistical significance test (z_{AB})between different test cases. The test cases which are statistically significant are highlighted in bold

	Test case 1	**Test case 2**	**Test case 3**	**Test case 4**	**Test case 5**
Test case 1	0	**141.33**	**130.33**	**108.5**	**189.83**
Test case 2	**141.33**	0	**10.53**	**32.25**	**32.25**
Test case 3	**130.33**	**10.53**	0	**21.65**	**58.54**
Test case 4	**108.5**	**32.25**	**21.65**	0	**80.28**
Test case 5	**189.83**	**32.25**	**58.46**	**80.28**	0

fully or semi-automatic semantic labelling of the various earth features is considered an unsolved problem in the photogrammetry and the remote sensing community. This is due to the heterogeneity and the intra-class variance of the various earth surface objects.

The focus of this part of study is to semantically label each and every point in the test LiDAR point cloud based on the training dataset consisting of nine urban classes. An object-based point cloud labelling approach was adopted to label the various urban classes. The study has critically examined the various parameters on the 3D object-based semantic labelling.

Some of the important results that emerge from this part of study are summarized below.

1) Object-based point cloud labelling is a promising approach for semantically labelling airborne urban point cloud with multiple classes. In this study, an overall labelling accuracy of 76% was achieved with nine urban classes.

2) Segmentation is an important stage in object-based point cloud analysis. The quality of segments created affects the final classification accuracy. Amongst the two segmentation algorithms tested, colour-based region growing algorithm, (test cases 1 and 2) and supervoxels-based LCCP algorithm (Test cases 3, 4, and 5), the latter produces a more geometrically defined segments. This might be attributed to the parameters used for defining the segments. In the case of colour-based region growing segmentation algorithm, the normals and spectral values are used to create the segments. Whereas, in the case of supervoxels-based cloud connectivity algorithm, the supervoxels are initially created based on the colour, spatial distance and geo-

metrical similarity between voxels. These supervoxels are merged into meaningful geometrical segments based on the local connectivity between adjacent supervoxels. This improves the feature extraction and thus leads to better point cloud labelling.

3) As the point cloud captured through LiDAR technology is massive, it is imperative to have computationally efficient point cloud processing algorithms which can handle such huge data. Supervoxels-based approach is a promising alternative to normal point based processing approaches which markedly reduces the processing time of subsequent processing stages in the point cloud labelling. With the supervoxels-based LCCP segmentation approach, there is a significant decrease in the computational processing time (almost four times) for segmentation. This is an important result when handling large datasets covering wider geographical area.

4) The man-made urban objects with defined geometry are detected with more precision than the natural urban classes. The per-class accuracies of different classes indicate that buildings and impervious surface have maximum detection rate of around 96%. However, there is a low detection rate for the natural urban classes such as shrubs, and fence / hedges. This might be due to the similar geometry and spectral characterisation of these two classes. Trees are detected with maximum precision (correctness) of 96% amongst the natural vegetation. This might be attributed to the elevation information, a geometrical feature used in the classification.

5) From the labelling results, it can be comprehended that the spectral information complements the geometrical information. The accuracy of labelling is maximum for the test cases 2 and 5 in which both spectral and geometrical details are used for segmentation and classification. There is about 6 % reduction in accuracy when geometrical information alone is used for labelling (test case 3). However, when intensity data are used with the geometrical information from LiDAR, the classification accuracy has reduced further. This might be attributed the noise in the backscattered intensity value as discussed in Jutzi and Gross (2009). As discussed in Langford et al. (2006) the intensity value is not only influenced by the reflectivity, moisture content , roughness and other target properties but also by the dynamic relationship between the sensor and the target. As it can be seen in Test cases 1 and 4 (Figure: 5.9, per-class accuracy of most the classes is comparatively low when intensity value is used as a complementary information. This can be correlated to the fact that the same material (e.g. gabled roof) will yield different values of intensity due to the variation in the incidence angle as reported in lit-

eratures (Gross et al., 2008; Jutzi and Gross, 2009). The intensity values must be normalised considering the sensor-target geometry for improved results. Further studies are recommended in investigating the influence of normalised intensity value on the labelling accuracy.

5.6 Conclusion

In this part of the work, we rigorously assessed the effect of various parameters on the object-based semantic labelling of urban point cloud. Experiments were carried out on benchmark datasets provided by the ISPRS. A comprehensive validation against the ground truth was done by the ISPRS. The results show that supervoxels-based LCCP segmentation is a computationally efficient approach for creating 3D segments. The segments represent the real world objects better by geometry and hence improve the classification accuracy based on feature based classification. Further, the study emphasized the importance of both spectral and geometrical details for improved labelling accuracy.

5.7 Chapter Summary

3D semantic labelling accuracy for multiple classes significantly depends on various parameters used as part of the proposed object-based point cloud labelling framework. This chapter quantitatively evaluates the effect of those parameters on the labelling accuracy. This chapter also has analysed the role of additional information such as intensity, spectral information from multispectral images in the labelling accuracy of LiDAR data. Also, the role of segmentation algorithms (presented as part of this thesis) in the labelling accuracy has been explored. This chapter has also compared the computational complexity of the algorithms which plays a significant role in the selection of approaches to be used in operational remote sensing.

CHAPTER 6

SUMMARY, CONCLUSIONS AND FUTURE DIRECTIONS

Prelude: This chapter summarizes the overall observations of the studies carried out as part of the doctoral research work. Also, the important contributions of this thesis to the state-of-the art in LiDAR point cloud processing is emphasised. Further, recommendations and directions for further research relevant to the challenges observed in this work are presented.

6.1 Summary of the Chapters

This thesis has investigated an important research problem of three dimensional land cover objects extraction of an urban setting from the data obtained from the culmination of two different sensors - LiDAR point cloud, and multispectral data. In pursuance of the objectives, we have developed and implemented a novel object-based point cloud labelling approach for semantic labelling of the urban LiDAR point cloud and multispectral data. The complementary spectral information from the multipectral data has been used along with the geometrical information from the LiDAR in all the stages of object-based labelling (segmentation, feature extraction, and classification) to improve the labelling accuracy. In order to cater to the computational demands of high density LiDAR dataset, we have proposed a supervoxels-based labelling approach for producing geometrically well defined segments. Here the emphasis has been on creating segments based on over-segmented point cloud (super-voxels) instead of individual points. This approach markedly reduces the computational time of the algorithm and hence improves the efficiency and reliability of the algorithm for 3D semantic labelling.

The summary of the thesis is presented below chapter wise.

Chapter 2 : A generic methodological framework for object-based semantic labelling of 3D point cloud is discussed. The chapter has elaborated the theoretical background of the vari-

ous stages in the semantic labelling. The feasibility of developing the prototype for semantic labelling based on open source tools has also been explored. Further, to evaluate the performance, a feasibility study has been carried out to automatically extract buildings, one of the major urban land cover classes, from the LiDAR point. The results substantiate the potential of the proposed method for LiDAR point cloud labelling.

Chapter 3 : Advancements in sensor technologies enable the simultaneous acquisition of LiDAR and multispectral data. In this chapter, we have assessed the potential of utilising the complementary information from the multiple sensors for improved urban point cloud labelling. We adopted and extended a modified colour-based region growing algorithm to create the initial object primitives. Further, in this chapter we tested the potential of a multiple classifier system to improve the labelling accuracy. The methodology was tested on benchmark dataset provided by ISPRS and the results are independently validated against the ground truth. The promising results emphasize the importance of combining data from multiple sensors for semantic labelling.

Chapter 4 : In this chapter, we have proposed a supervoxels-based local cloud convexity method for creating segments. The supervoxels, analogous to super-pixels in 2D domain, produces oversegmented point cloud based on colour, spatial distance and geometry between the points in the point cloud. The supervoxels are used in further stages of processing instead of the individual points thereby decreasing the computational demand. The supervoxels are combined to produce geometrically true segments based on local cloud connectivity algorithm. Spectral and geometric features extracted from these segments are used for semantically labelling the point cloud. The effect of features on classification has also been studied. Experiments are carried out on LiDAR dataset of varying point density. The results indicate a high classification performance with added advantage of low computational demand.

Chapter 5 : Experimental results on the impact of various parameters on the object-based point cloud labelling for multiple classes (9 classes) has been presented in this chapter. The study has rigorously compared the segmentation methods - colour-based region growing algorithm, and supervoxels-based local cloud connectivity algorithm, for producing 3D segments. The computational performance of the two algorithms are compared and analysed. Results of assessment on the importance of additional information - back-scattered intensity, colour on

semantic labelling are also presented. Per-class accuracies of various classes are analysed in detail to arrive at the optimal methodology for semantic labelling.

6.2 Answers to the Research Questions

(1) *Given the massive and computationally demanding LiDAR dataset over a city, is there a computationally efficient algorithm that can be utilized to semantically label multiple urban classes in a 3D dataset ?*

Yes. In this thesis, we have proposed an object-based framework for semantically labelling the LiDAR point cloud. This is more efficient than point-based labelling methods. Segmentation is the major step in the object-based labelling. We have proposed a computationally efficient supervoxels-based local cloud connectivity method for creating 3D segments. The main advantage of this approach is that initially the points in the point clouds are grouped to form supervoxels based on spatial distance, spectral and geometrical similarity. Further processing relies on the supervoxels rather than individual points. Our study has revealed that for highly dense point cloud data (>5 points/m^2) there is close to 99% reduction in the number of points used for processing. This drastically reduces the processing time of the segmentation algorithm. Compared to the colour-based region growing algorithm, the computation time is reduced by almost 4 times, which is a significant achievement while processing LiDAR point cloud. Now, with increasing availability of highly dense LiDAR point cloud, supervoxels-based approach offers a promising solution for handling computationally demanding datasets even with high land cover complexity.

(2) *Does spectral information from multispectral images complement the geometrical information from LiDAR in semantic labelling of 3D LiDAR point cloud ?*

Yes. Our results reveal that the spectral information complements the geometrical information for semantically labelling LiDAR point cloud data with less point density (< 5 points/m^2). When the point density is high (eg: as in case of the Belgium dataset with 55 points/m^2), the geometrical information alone is sufficient for discriminating various urban objects. However when the point density is low, the labelling accuracy might not be reliable if only the LiDAR

point cloud is used. Spectral information complements LiDAR point cloud when discriminating geometrically similar classes such as impervious surface and lawn.

(3) *How does the combination of various computational parameters such as data fusion, point density, segmentation approaches, feature extraction, and classification approaches affect the 3D semantic labelling of LiDAR point cloud ?*

Different computational parameters impact the accuracy of 3D semantic labelling. Our studies reveal that data fusion improves the semantic labelling. The complementary information from LiDAR and the optical sensor increases the separability of spectrally and geometrically similar classes, especially when using low point density dataset. This improves the segments created as well as the features extracted for classification. However, it is also evident that while processing highly dense LiDAR point cloud, the geometrical information alone is sufficient. Computationally, the proposed supervoxels-based LCCP segmentation offers superior labelling accuracy owing to the geometrically well defined segments created. Features extracted from the segments affect the labelling accuracy. Spectral and geometrical features are important for improving the labelling accuracy for low point density dataset. Regarding the classification algorithms, multiple classifier system, and ensemble classifiers such as Random forest offer a better solution for feature based labelling compared to a single best classifier.

(4) *How efficient is the open source 3D object-based LiDAR point cloud labelling approach compared to point-based labelling approaches adopted in commercial software packages ?*

We have developed the entire methodology on open source framework assessing the potential of the same for LiDAR point cloud labelling. We tested our methodology on dataset with multiple classes provided by the ISPRS as part of semantic labelling challenge. The results are independently validated against the ground truth by the ISPRS. We have obtained a detection accuracy of > 96% for man-made classes. This is at par or better than the labelling accuracy achieved using commercial software.

6.3 Major Contributions

The major contributions of this thesis to the existing literature on LiDAR data processing are summarized below.

- Adapting and improving the object-based approach for 3D point cloud labelling.
- Integration of spectral and geometrical information in the segmentation, feature extraction and classification stages.
- Extension and adaptation of a supervoxel-based segmentation approach for object creation using airborne LiDAR datasets having different point density and different land surface complexity.
- Combining the supervoxels patches using convexity criteria (local cloud connectivity) to arrive at meaningful segments.
- Assessment of feature selection (geometrical and spectral) and classifier selection on the accuracy of point cloud labelling.
- Critical assessment of different methodologies for 3D semantic labelling of multiclass urban LiDAR data.

6.4 Recommendation for Future Research

The following are some of the recommendations for further research in this high impact research area.

- The method for integrating LiDAR point cloud with optical sensor can be further explored taking into account the visibility analysis as suggested by Gerke and Xiao (2014).
- Further work can be done on feature extraction enabling improved demarcation between natural vegetation classes such as hedge and shrubs. Features such as texture, histogram based features can be explored.

- Another interesting area is the extension of this proposed technique on integrating spectrally richer hyperspectral images with LiDAR point cloud.
- The proposed methodology can be tested on datasets from complex terrains with various type of urban coverage distribution (e.g. urban-rural interface) offering complex scenarios of validation.
- The extension and investigation of this proposed method for forest species labelling with better spectral data, e.g. hyperspectral data. This would enable species level 3D modelling as well as biophysical parameter extraction.
- We recommend studies for assessing the computational performance of the existing methods for various types of landscapes and LiDAR point densities.

REFERENCES

1. (2009). Classification of Airborne LIDAR Intensity Data Using Statistical Analysis and Hough Transform with Application to Power Line Corridors. *Digital Image Computing : Techniques and Applications Conference (DICTA)*, (pp. 462–467).

2. 2015 IEEE GRSS Data Fusion Contest (2015). Online: http://www.grss-ieee.org/community/technical-committees/data-fusion.

3. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. (2010). *SLIC superpixels*. Technical Report.

4. Aijazi, A. K., Checchin, P., and Trassoudaine, L. (2013). Segmentation based classification of 3d urban point clouds: A super-voxel based approach with evaluation. *Remote Sensing*, *5*, 1624–1650.

5. Alexandre, L. A. (2012). 3d descriptors for object and category recognition: a comparative evaluation. In *Workshop on Color-Depth Camera Fusion in Robotics at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vilamoura, Portugal* (p. 7). Citeseer volume 1.

6. Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, *46*, 175.

7. Arya, S., Mount, D. M., Netanyahu, N. S., Silverman, R., and Wu, A. Y. (1998). An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *Journal of the ACM (JACM)*, *45*, 891–923.

8. ASPRS (2013). *Las Specification Version 1.4*. Technical Report.

9. Awrangjeb, M., Lu, G., and Fraser, C. (2014). Automatic building extraction from lidar data covering complex urban scenes. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *40*, 25.

10. Axelsson, P. (2000a). Dem generation from laser scanner data using adaptive tin models. *International Archives of Photogrammetry and Remote Sensing*, *33*, 111–118.

11. Axelsson, P. (2000b). Dem generation from laser scanner data using adaptive tin models. *International Archives of Photogrammetry and Remote Sensing*, *33*, 111–118.

12. Baltsavias, E. P. (1999). Airborne laser scanning: basic relations and formulas. *ISPRS Journal of Photogrammetry and Remote Sensing*, *54*, 199–214.

13. Batty, M., Axhausen, K. W., Giannotti, F., Pozdnoukhov, A., Bazzani, A., Wachowicz, M., Ouzounis, G., and Portugali, Y. (2012). Smart cities of the future. *The European Physical Journal Special Topics*, *214*, 481–518.

14. Behley, J., Steinhage, V., and Cremers, A. B. (2012). Performance of histogram descriptors for the classification of 3d laser range data in urban environments. In *IEEE International Conference on Robotics and Automation (ICRA)* (pp. 4391–4398).

15. Benediktsson, J. A., Chanussot, J., and Fauvel, M. (2007). Multiple classifier systems in remote sensing: from basics to recent developments. In *International Workshop on Multiple Classifier Systems* (pp. 501–512). Springer.

16. Bentley, J. L. (1980). Multidimensional divide-and-conquer. *Communications of the ACM*, *23*, 214–229.

17. Bethel, J. S., Van Gelder, B. H., Cetin, A. F., and Sampath, A. (2006). *Corridor mapping using aerial technique*. Technical Report Joint Transportation Research Program Report.

18. Blaschke, T. (2010). Object based image analysis for remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, *65*, 2–16.

19. Blaschke, T., and Strobl, J. (2001). What 's wrong with pixels? some recent developments interfacing remote sensing and gis. *GeoBIT/GIS*, *6*, 12–17.

20. Breiman, L. (2001). Random forests. *Machine Learning*, *45*, 5–32.

21. Burai, P. (2012). Applied remote sensing: Aerial lidar technology.

22. Burtch, R. (2002). Lidar principles and applications. In *IMAGIN Conference, Traverse City, MI* (pp. 1–13).

23. Caragliu, A., Del Bo, C., and Nijkamp, P. (2011). Smart cities in europe. *Journal of Urban Technology*, *18*, 65–82.

24. Chen, D., Zhang, L., Li, J., and Liu, R. (2012). Urban building roof segmentation from airborne lidar point clouds. *International Journal of Remote Sensing*, *33*, 6497–6515.

25. Christoph Stein, S., Schoeler, M., Papon, J., and Worgotter, F. (2014). Object partitioning using local convexity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 304–311).

26. Coeurjolly, D. (2015). Computational geometry: spatial data structures, voronoi diagram and delaunay triangulation.

27. Congalton, R., Oderwald, R. G., and Mead, R. A. (1983). Assessing Landsat classification accuracy using discrete multivariate analysis statistical techniques. *Photogrammetric Engineering and Remote Sensing*, *49*, 1671–1678.

28. Connolly, C., and Fleiss, T. (1997). A study of efficiency and accuracy in the transformation from rgb to cielab color space. *IEEE Transactions on Image Processing*, *6*, 1046–1048.

29. Cramer, M. (2010). The dgpf-test on digital airborne camera evaluation–overview and test design. *Photogrammetrie-Fernerkundung-Geoinformation*, *2010*, 73–82.

30. Damodaran, B. B., and Nidamanuri, R. R. (2014). Dynamic linear classifier system for hyperspectral image classification for land cover mapping. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *7*, 2080–2093.

31. Darmawati, A. T. (2008). Utilization of multiple echo information for classification of airborne laser scanning data. Enchede, ITC.

32. Dash, J., Steinle, E., Singh, R., and Bähr, H. (2004). Automatic building extraction from laser scanning data: an input tool for disaster management. *Advances in Space Research*, *33*, 317–322.

33. Demir, N., and Baltsavias, E. (2012). Automated modeling of 3d building roofs using image and lidar data. In *Proceedings of the XXII Congress of the International Society for Photogrammetry, Remote Sensing, Melbourne, Australia*. volume 25.

34. Dorninger, P., and Pfeifer, N. (2008). A comprehensive automated 3d approach for building extraction, reconstruction, and regularization from airborne laser scanning point clouds. *Sensors*, *8*, 7323–7343.

35. Du, P., Xia, J., Zhang, W., Tan, K., Liu, Y., and Liu, S. (2012). Multiple classifier system for remote sensing image classification: A review. *Sensors*, *12*, 4764–4792.

36. Duda, R. O., Hart, P. E., and Stork, D. G. (2012). *Pattern Classification*. John Wiley & Sons.

37. Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. (2008). Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, *9*, 1871–1874.

38. Fauvel, M., Chanussot, J., and Benediktsson, J. A. (2006). Decision fusion for the classification of urban remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, *44*, 2828–2838.

39. Fehr, D., Cherian, A., Sivalingam, R., Nickolay, S., Morellas, V., and Papanikolopoulos, N. (2012). Compact covariance descriptors in 3d point clouds for object recognition. In *IEEE International Conference on Robotics and Automation (ICRA* (pp. 1793–1798).

40. Feng, L. (2013). Detection of building point clouds from airborne lidar data using roof points attributes. ISPRS Test Project on Urban Classification and 3D Building Reconstruction: Results.

41. Filin, S., and Pfeifer, N. (2006). Segmentation of airborne laser scanning data using a slope adaptive neighborhood. *ISPRS Journal of Photogrammetry and Remote Sensing*, *60*, 71–80.

42. Fritz, H. M., Phillips, D. A., Okayasu, A., Shimozono, T., Liu, H., Mohammed, F., Skanavis, V., Synolakis, C. E., and Takahashi, T. (2012). The 2011 japan tsunami current velocity measurements from survivor videos at kesennuma bay using lidar. *Geophysical Research Letters*, *39*.

43. Gerke, M., and Xiao, J. (2014). Fusion of airborne laserscanning point clouds and images for supervised and unsupervised scene classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, *87*, 78–92.

44. Ghosh, S., and Lohani, B. (2013). Mining lidar data with spatial clustering algorithms. *International Journal of Remote Sensing*, *34*, 5119–5135.

45. Gil, A. L., Núñez-Casillas, L., Isenburg, M., Benito, A. A., Bello, J. J. R., and Arbelo, M. (2013). A comparison between lidar and photogrammetry digital terrain models in a forest area on tenerife island. *Canadian Journal of Remote Sensing*, *39*, 396–409.

46. Golovinskiy, A., Kim, V. G., and Funkhouser, T. (2009). Shape-based recognition of 3d point clouds in urban environments. In *2009 IEEE 12th International Conference on Computer Vision* (pp. 2154–2161).

47. Graham, L. (2008). Management of LiDAR Data. In *Topographic Laser Ranging and Scanning* (pp. 295–306). CRC Press.

48. Gross, H., Jutzi, B., and Thoennessen, U. (2008). Intensity Normalization By Incidence Angle and Range of Full-Waveform Lidar Data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XXXVII.*, 405–412.

49. Gruen, A., Behnisch, M., and Kohler, N. (2009). Perspectives in the reality-based generation, n d modelling, and operation of buildings and building stocks. *Building Research & Information*, *37*, 503–519.

50. Haala, N., Rothermel, M., and Cavegn, S. (2015). Extracting 3d urban models from oblique aerial images. In *2015 Joint Urban Remote Sensing Event (JURSE)* (pp. 1–4).

51. Han, S.-H., Lee, J.-H., and Yu, K.-Y. (2007). An approach for segmentation of airborne laser point clouds utilizing scan-line characteristics. *ETRI journal*, *29*, 641–648.

52. Harding, D., Lefsky, M., Parker, G., and Blair, J. (2001). Laser altimeter canopy height profiles: Methods and validation for closed-canopy, broadleaf forests. *Remote Sensing of Environment*, *76*, 283–297.

53. Hay, G. J., and Castilla, G. (2008). Geographic object-based image analysis (geobia): A new name for a new discipline. In *Object-based image analysis* (pp. 75–89). Springer.

54. Hodgson, M. E., Jensen, J. R., Schmidt, L., Schill, S., and Davis, B. (2003). An evaluation of lidar-and ifsar-derived digital elevation models in leaf-on conditions with usgs level 1 and level 2 dems. *Remote Sensing of Environment*, *84*, 295–308.

55. Höfle, B., Geist, T., Rutzinger, M., and Pfeifer, N. (2007). Glacier surface segmentation using airborne laser scanning point cloud and intensity data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *36*, W52.

56. Honnay, O., Piessens, K., Van Landuyt, W., Hermy, M., and Gulinck, H. (2003). Satellite based land use and landscape complexity indices as predictors for regional plant species diversity. *Landscape and urban planning*, *63*, 241–250.

57. IQumulus (2015). Iqmulus processing contest 2015. *Available at : http://iqmulus.eu/iqpc/iqmulus-processing-contest-2015. [Accessed June 1, 2016]*, .

58. Isenburg, M. (2012). Lastools – efficient tools for lidar processing. *Available at: http: http://www. cs. unc. edu/˜ isenburg/lastools/ [Accessed October 9, 2012]*, .

59. ISPRS (2016). Isprs 3d semantic labeling contest. *Available at : http://www2.isprs.org/commissions/comm3/wg4/3d-semantic-labeling.html [Accessed June 1, 2016]*, .

60. Jiang, X., and Bunke, H. (1994). Fast segmentation of range images into planar regions by scan line grouping. *Machine Vision and Applications*, *7*, 115–122.

61. Jin, X., and Davis, C. H. (2005). Automated building extraction from high-resolution satellite imagery in urban areas using structural, contextual, and spectral information. *EURASIP Journal on Advances in Signal Processing*, *2005*, 1–11.

62. Jutzi, B., and Gross, H. (2009). Normalization of lidar intensity data based on range and surface incidence angle. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *38*, 213–218.

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

64. Lafarge, F., and Mallet, C. (2012). Creating large-scale city models from 3d-point clouds: a robust approach with hybrid representation. *International Journal of Computer Vision*, *99*, 69–85.

65. Laible, S., Khan, Y. N., and Zell, A. (2013). Terrain classification with conditional random fields on fused 3d lidar and camera data. In *Mobile Robots (ECMR), 2013 European Conference on* (pp. 172–177).

66. Lalonde, J.-F., Vandapel, N., Huber, D. F., and Hebert, M. (2006). Natural terrain classification using three-dimensional ladar data for ground robot mobility. *Journal of Field Robotics*, *23*, 839–861.

67. Langford, J., Niemann, O., Frazer, G. W., Wulder, M. A., and Nelson, T. (2006). Exploring small footprint lidar intensity data in a forested environment. *International Geoscience and Remote Sensing Symposium (IGARSS)*, (pp. 2416–2419).

68. Lari, Z., Habib, A., and Kwak, E. (2011). An adaptive approach for segmentation of 3d laser point cloud. In *ISPRS Workshop Laser Scanning* (pp. 29–31).

69. Lee, D. H., Lee, K. M., and Lee, S. U. (2008). Fusion of lidar and imagery for reliable building extraction. *Photogrammetric Engineering & Remote Sensing*, *74*, 215–225.

70. Lefsky, M. A., Cohen, W. B., Parker, G. G., and Harding, D. J. (2002). Lidar remote sensing for ecosystem studies. *BioScience*, *52*, 19–30.

71. Lewis, D. D. (1998). Naive (bayes) at forty: The independence assumption in information retrieval. In *European conference on machine learning* (pp. 4–15). Springer.

72. Li, Y., Wu, H., An, R., Xu, H., He, Q., and Xu, J. (2013). An improved building boundary extraction algorithm based on fusion of optical imagery and lidar data. *Optik-International Journal for Light and Electron Optics*, *124*, 5357–5362.

73. Lim, E. H., and Suter, D. (2009). 3d terrestrial lidar classifications with super-voxels and multi-scale conditional random fields. *Computer-Aided Design*, *41*, 701–710.

74. Liu, H., Huang, Z., Zhan, Q., and Lin, P. (2008). A database approach to very large lidar data management. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *37*, 463–468.

75. Liu, X. (2008). Airborne lidar for dem generation: some critical issues. *Progress in Physical Geography*, *32*, 31–49.

76. Lorensen, W. E., and Cline, H. E. (1987). Marching cubes: A high resolution 3d surface construction algorithm. In *ACM Siggraph Computer Graphics* (pp. 163–169). ACM volume 21.

77. Lucchi, A., Smith, K., Achanta, R., Knott, G., and Fua, P. (2012). Supervoxel-based segmentation of mitochondria in em image stacks with learned shape features. *IEEE Transactions on Medical Imaging*, *31*, 474–486.

78. Luhmann, T., Robson, S., Kyle, S., and Boehm, J. (2014). *Close-range photogrammetry and 3D imaging*. Walter de Gruyter.

79. Ma, R. (2005). Dem generation and building detection from lidar data. *Photogrammetric Engineering & Remote Sensing*, *71*, 847–854.

80. Maas, H.-G., and Vosselman, G. (1999). Two algorithms for extracting building models from raw laser altimetry data. *ISPRS Journal of Photogrammetry and Remote Sensing*, *54*, 153–163.

81. MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* (pp. 281–297). Oakland, CA, USA. volume 1.

82. Madhavan, B. B., Wang, C., Tanahashi, H., Hirayu, H., Niwa, Y., Yamamoto, K., Tachibana, K., and Sasagawa, T. (2006). A computer vision based approach for 3d building modelling of airborne laser scanner dsm data. *Computers, Environment and Urban Systems*, *30*, 54–77.

83. Mallet, C., and Bretar, F. (2009). Full-waveform topographic lidar: State-of-the-art. *ISPRS Journal of photogrammetry and remote sensing*, *64*, 1–16.

84. Matikainen, L., Kaartinen, H., and Hyyppä, J. (2007). Classification tree based building detection from laser scanner and aerial image data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *36*, W52.

85. Mayer, H. (1999). Automatic object extraction from aerial imageryâĂŤa survey focusing on buildings. *Computer Vision and Image Understanding*, *74*, 138–149.

86. Mayer, H. (2008). Object extraction in photogrammetric computer vision. *ISPRS Journal of Photogrammetry and Remote Sensing*, *63*, 213–222.

87. Means, J. E., Acker, S. A., Fitt, B. J., Renslow, M., Emerson, L., Hendrix, C. J. et al. (2000). Predicting forest stand characteristics with airborne scanning lidar. *Photogrammetric Engineering and Remote Sensing*, *66*, 1367–1372.

88. Meng, X., Currit, N., and Zhao, K. (2010). Ground filtering algorithms for airborne lidar data: A review of critical issues. *Remote Sensing*, *2*, 833–860.

89. Moore, A. W. (1990). *Efficient Memory-based Learning for Robot Control*. Technical Report.

90. Moussa, A., and El-Sheimy, N. (2012). A new object based method for automated extraction of urban objects from airborne sensors data. In *Proceedings of: XXII ISPRS Congress, Melbourne, Australia*.

91. Naus, T. (2008). Unbiased lidar data measurement (draft). *ASPRS PAD LiDAR Committee*, .

92. Niemeyer, J., Rottensteiner, F., and Soergel, U. (2014). Contextual classification of lidar data and building object detection in urban areas. *ISPRS journal of photogrammetry and remote sensing*, *87*, 152–165.

93. Nizar, A. A., Filin, S., and Doytsher, Y. (2006). Reconstruction of buildings from airborne laser scanning data. In *ASPRS Annual Conference*.

94. Ortiz, C. A., Gonzalo-Martí, C., Peña, J. M., and Menasalvas, E. (2014). 3d dendrite spine detection - a supervoxel based approach. In *International Conference on Rough Sets and Intelligent Systems Paradigms* (pp. 359–366).

95. Osgood, T. J. (2013). *Semantic labelling of road scenes using supervised and unsupervised machine learning with lidar-stereo sensor fusion*. Ph.D. thesis University of Warwick.

96. Panhalkar, N., Paul, R., and Anand, S. (2014). Increasing part accuracy in additive manufacturing processes using a kd tree based clustered adaptive layering. *Journal of Manufacturing Science and Engineering*, *136*, 061017.

97. Paparoditis, N., Cord, M., Jordan, M., and Cocquerez, J.-P. (1998). Building detection and reconstruction from mid-and high-resolution aerial imagery. *Computer vision and image understanding*, *72*, 122–142.

98. Papon, J., Abramov, A., Schoeler, M., and Worgotter, F. (2013). Voxel cloud connectivity segmentation-supervoxels for point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2027–2034).

99. Parmehr, E. G., Fraser, C. S., Zhang, C., and Leach, J. (2013). Automatic registration of optical imagery with 3d lidar data using local combined mutual information. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *5*, W2.

100. Petrie, G., and Toth, C. K. (2008a). Airborne and spaceborne laser profilers and scanners. chapter Topographic Laser Ranging and Scanning. CRC Press.

101. Petrie, G., and Toth, C. K. (2008b). Introduction to laser ranging, profiling, and scanning. chapter Topographic Laser Ranging and Scanning. CRC Press.

102. Pfeifer, N. (2005). A subdivision algorithm for smooth 3d terrain models. *ISPRS Journal of Photogrammetry and Remote Sensing*, *59*, 115–127.

103. Program, N. D. E. (2004). Ndep guidelines for digital elevation data. *Available at :* $http : //www.ndep.gov/NDEP_Elevation_Guidelines_Ver1_10May2004.pdf$, *1*, 1–93.

104. Pu, S., Vosselman, G. et al. (2006). Automatic extraction of building features from terrestrial laser scanning. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *36*, 25–27.

105. Rabbani, T., and Van Den Heuvel, F. (2005). Efficient hough transform for automatic detection of cylinders in point clouds. *ISPRS WG III/3, III/4*, *3*, 60–65.

106. Rabbani, T., Van Den Heuvel, F., and Vosselmann, G. (2006). Segmentation of point clouds using smoothness constraint. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *36*, 248–253.

107. Reutebuch, S. E., Andersen, H.-E., and McGaughey, R. J. (2005). Light detection and ranging (lidar): an emerging tool for multiple resource inventory. *Journal of Forestry*, *103*, 286–292.

108. Rottensteiner, F. (2003). Automatic generation of high-quality building models from lidar data. *IEEE Computer Graphics and Applications*, *23*, 42–50.

109. Rusu, R. (2009). *Semantic 3D object maps for everyday manipulation in human living environments*. Ph.D. thesis Computer Science Dept, Technische Universit,Munchen,Germany.

110. Rusu, R. B., Marton, Z. C., Blodow, N., and Beetz, M. (2008). Learning informative point classes for the acquisition of object model maps. In *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on* (pp. 643–650). IEEE.

111. Rutzinger, M., Höfle, B., Hollaus, M., and Pfeifer, N. (2008). Object-based point cloud analysis of full-waveform airborne laser scanning data for urban vegetation classification. *Sensors*, *8*, 4505–4528.

112. Rutzinger, M., Rottensteiner, F., and Pfeifer, N. (2009). A comparison of evaluation techniques for building extraction from airborne laser scanning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *2*, 11–20.

113. Salolahti, M. (2010). *Advanced Use of Lidar Data - Automatic Building Vectorization and Contour Production*. Technical Report April 2010.

114. Samet, H. (1988). An overview of quadtrees, octrees, and related hierarchical data structures. In *Theoretical Foundations of Computer Graphics and CAD* (pp. 51–68). Springer.

115. Sampath, A., and Shan, J. (2010). Segmentation and reconstruction of polyhedral building roofs from aerial lidar point clouds. *IEEE Transactions on Geoscience Remote Sensing*, *48*, 1554–1567.

116. Schnabel, R., Wahl, R., and Klein, R. (2007). Efficient ransac for point-cloud shape detection. In *Computer Graphics Forum* (pp. 214–226). Wiley Online Library volume 26.

117. Serra, J. (1986). Introduction to mathematical morphology. *Computer Vision, Graphics, and Image Processing*, *35*, 283–305.

118. Sithole, G. (2005). *Segmentation and classification of airborne laser scanner data*. TU Delft, Delft University of Technology.

119. Skaloud, J., and Lichti, D. (2006). Rigorous approach to bore-sight self-calibration in airborne laser scanning. *ISPRS Journal of Photogrammetry and Remote Sensing*, *61*, 47–59.

120. Sohn, G., and Dowman, I. (2007). Data fusion of high-resolution satellite imagery and lidar data for automatic building extraction. *ISPRS Journal of Photogrammetry and Remote Sensing*, *62*, 43–63.

121. Soininen, A. (2004). Terrascan user's guide. *Available at: http*: $//www.terrasolid.com/download/userguides.php$[*Accessed June 1, 2016*], .

122. Sun, S., and Salvaggio, C. (2013). Aerial 3d building detection and modeling from airborne lidar point clouds. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *6*, 1440–1449.

123. Suveg, I., and Vosselman, G. (2004). Reconstruction of 3d building models from aerial images and maps. *ISPRS Journal of Photogrammetry and Remote Sensing*, *58*, 202–224.

124. Tarsha-Kurdi, F., Landes, T., Grussenmeyer, P. et al. (2007). Hough-transform and extended ransac algorithms for automatic detection of 3d building roof planes from lidar data. In *Proceedings of the ISPRS Workshop on Laser Scanning* (pp. 407–412). volume 36.

125. Terrasolid (2016). Terrasolid - point cloud intelligence.

126. Tombari, F., Salti, S., and Di Stefano, L. (2010). Unique signatures of histograms for local surface description. In *European Conference on Computer Vision* (pp. 356–369).

127. Tomljenovic, I., Höfle, B., Tiede, D., and Blaschke, T. (2015). Building extraction from airborne laser scanning data: An analysis of the state of the art. *Remote Sensing*, *7*, 3826–3862.

128. Transparency Market Research (2014). Lidar market - global industry analysis, size, share, growth, trends and forecast 2014 - 2020.

129. United Nations (2014). *World urbanization prospects*. Technical Report.

130. Ural, S., and Shan, J. (2012). Min-cut based segmentation of airborne lidar point clouds. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *1*, 167–172.

131. Vapnik, V. (2013). *The nature of statistical learning theory*. Springer Science & Business Media.

132. Vosselman, G. (2000). Slope based filtering of laser altimetry data. *International Archives of Photogrammetry and Remote Sensing*, *33*, 935–942.

133. Vosselman, G., Gorte, B. G., Sithole, G., and Rabbani, T. (2004). Recognising structure in laser scanner point clouds. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *46*, 33–38.

134. Wang, M., and Tseng, Y.-H. (2011). Incremental segmentation of lidar point clouds with an octree-structured voxel space. *The Photogrammetric Record*, *26*, 32–57.

135. Wehr, A., and Lohr, U. (1999). Airborne laser scanning − an introduction and overview. *ISPRS Journal of photogrammetry and remote sensing*, *54*, 68–82.

136. Weikersdorfer, D., Schick, A., and Cremers, D. (2013). Depth-adaptive supervoxels for rgb-d video segmentation. In *IEEE International Conference on Image Processing* (pp. 2708–2712).

137. Woodman, O. J. (2007). An introduction to inertial navigation. *University of Cambridge, Computer Laboratory, Tech. Rep. UCAMCL-TR-696*, *14*, 15.

138. Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G. J., Ng, A., Liu, B., Philip, S. Y. et al. (2008). Top 10 algorithms in data mining. *Knowledge and information systems*, *14*, 1–37.

139. Wulder, M. A., and Franklin, S. E. (2006). *Understanding forest disturbance and spatial pattern: remote sensing and GIS approaches*. CRC Press.

140. Wurm, K. M., Kümmerle, R., Stachniss, C., and Burgard, W. (2009). Improving robot navigation in structured outdoor environments by identifying vegetation from laser data. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 1217–1222).

141. Xu, C., and Corso, J. J. (2012). Evaluation of super-voxel methods for early video processing. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1202–1209).

142. Yang, B., Dong, Z., Zhao, G., and Dai, W. (2015). Hierarchical extraction of urban objects from mobile laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, *99*, 45–57.

143. Yang, B., Xu, W., and Yao, W. (2014). Extracting buildings from airborne laser scanning point clouds using a marked point process. *GIScience & Remote Sensing*, *51*, 555–574.

144. Zhana, Q., Liangb, Y., and Xiaoa, Y. (2009). Color-based segmentation of point clouds. *Int Arch Photogrammetry, Remote Sens Spat Inf Sci*, *38*, 248–252.

145. Zhang, J., and Lin, X. (2012). Object-based classification of urban airborne lidar point clouds with multiple echoes using svm. *ISPRS Annals of Photogrametry, Remote Sensing and Spatial Information Sciences*, *3*, 135–140.

146. Zhang, J., Lin, X., and Ning, X. (2013). Svm-based classification of segmented airborne lidar point clouds in urban areas. *Remote Sensing*, *5*, 3749–3775.

147. Zhou, G., Song, C., Simmers, J., and Cheng, P. (2004). Urban 3d gis from lidar and digital aerial images. *Computers & Geosciences*, *30*, 345–353.

APPENDIX A

APPENDIX : 3D RECONSTRUCTION

A.1 Introduction

With the emphasis on sustainable development for improved quality of life of urban population, policy makers depend on geospatial dataset for intelligent decision making. 3D city models, digital models of urban terrain and objects are key for managing the complex urban information spaces. LiDAR technology allows directly capturing the third dimension (i.e., elevation) precisely at relatively lesser time with unprecedented accuracy. Semantically labelling the LiDAR point cloud into various land cover categories by automatic methods is one of the challenges posed to remote sensing community. The semantically labelled LiDAR point cloud is an asset to the decision makers as they can visualise, identify and extract objects of their interest. The semantically labelled LiDAR dataset can be used to create 3D models. These models reconstructed from the labelled point cloud are crucial to obtain a holistic view for several applications not restricted to infrastructure management, urban planning, emergency management, augmented reality.

In this thesis, we have developed an object-based framework for semantically labelling the urban coloured LiDAR point cloud into multiple classes. In this appendix section, the labelled dataset has been further processed to demonstrate the potential of creating 3D models of objects such as buildings and cranes. Also, a fly-through model has been created with the reconstructed buildings for better visualisation.

A.2 Reconstructed Objects

A.2.1 Buildings

Apart from detecting buildings in the urban environment, the LiDAR point cloud, in principle, can be exploited to reconstruct realistic three dimensional models of buildings, thus paving the way for 3D modelling of the urban environments. This has been achieved in this work by separating the points belonging to buildings and vectorizing the points in 3D environment. Buildings are mostly planar and hence '3D building vector model' macro in Terrascan was used to create 3D models. Figures A.1 and A.2 show the 3D buildings models created. The algorithm implemented finds the roof of a building by fitting a plane to the points (Salolahti, 2010). The plane of the roof was extruded to the ground class so as to reconstruct the buildings. Various geometrical parameters such as planarity tolerance, minimum building area, maximum roof slope, maximum gap between the buildings were set based on heuristics to produce a complete building model.

In addition, a small experiment was carried out to study the effect of point density on building model reconstruction using the macro. It has been found that when the point density is high (> 5 points / m^2), a very detailed building model is created (A.1). However, when the point density is low, less accurate models are created. For such dataset, manual editing is required to further enhance the visualization. Also, it was seen that small buildings are not well modelled when the point density is low (Figure A.2).

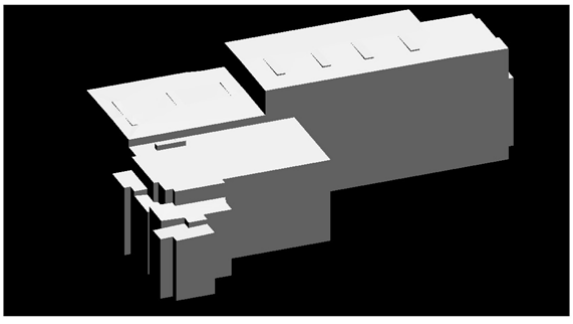


Figure A.1: Reconstructed building with high point density LiDAR dataset.

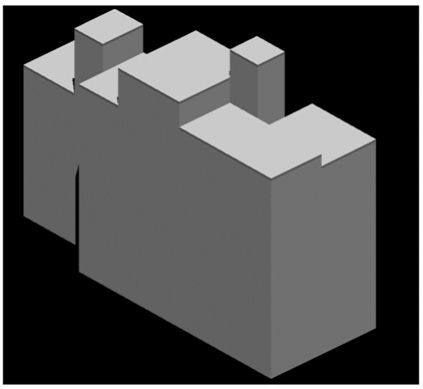


Figure A.2: Reconstructed building with low point density LiDAR dataset.

A.2.2 Cranes

We have also reconstructed a geometrically complex object - cranes from the Belgium dataset (study area 4) used in the chapter 4. The marching cube reconstruction algorithm (Lorensen and Cline, 1987) was used to reconstruct the object. In this algorithm, the data was divided into elementary cubes and a standard triangulation was followed inside each cube to determine the polygon for representing the part of isosurface that passes through each cube. The algorithm works by iterating over a uniform grid of cubes superimposed over a region of function. An index was created based on how the surface interacts with the cube. If the surface is completely above or below the cube, then all the 8 vertices of the cube are positive or negative and no surface would be created. There are technically 2^8 ways the surface can straddle the cube. Eliminating the rotation, translational and reflective symmetry, this can be reduced to 15 unique cases as shown in Figure A.3. Based on the way the cube straddles the function, triangles and vertices were generated. The final mesh is the union of these triangles. The smaller the cubes, well defined mesh triangles and a better approximation to the target function.

The cranes were reconstructed the marching cube reconstruction algorithm in Meshlab. The results are shown in Figure A.4.

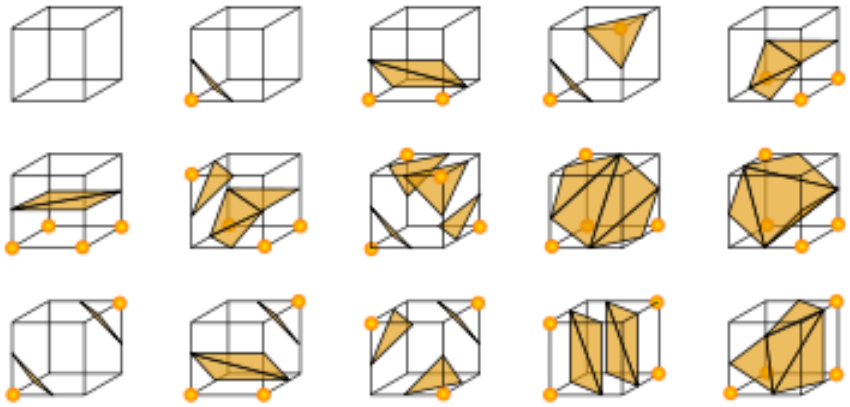


Figure A.3: Unique cases by which an iso-surface passes through a cube (Source: Lorensen and Cline (1987)).

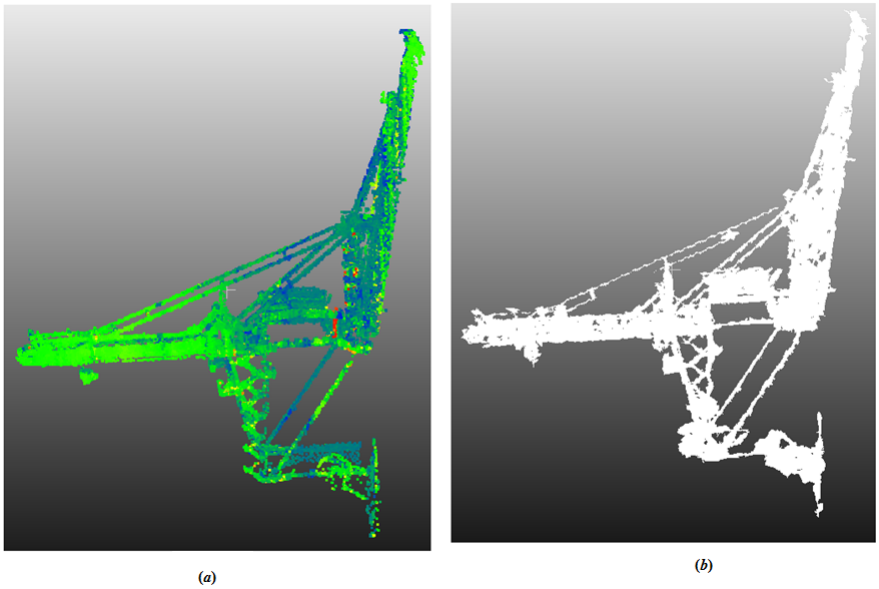


Figure A.4: Reconstructed cranes: (*a*) original point cloud, and (*b*) reconstructed model.

A.3 Flythrough Movie

A flythrough movie is created using terramodel software based on the semantic labelled point cloud and the building models. To have a realistic visualisation, the colour from the multispectral images is draped over the point cloud. The movie is included in the CD-ROM attached with this thesis.

LIST OF PAPERS BASED ON THESIS

Peer-Reviewed International Journals

1. Ramiya, Anandakumar M., Rama Rao Nidamanuri, and Ramakrishnan Krishnan. (2016). Supervoxels based spectro - spatial approach for 3D urban point cloud labelling, International Journal of Remote Sensing. Volume 37, Issue 17 **Publisher : Taylor** & **Francis**

2. Ramiya, Anandakumar M., Rama Rao Nidamanuri, and Ramakrishnan Krishnan. (2016). Object-oriented semantic labelling of spectral spatial LiDAR point cloud for urban land cover classification and buildings detection, Geocarto International. 31(2). **Publisher : Taylor** & **Francis**

3. Ramiya, Anandakumar M., Rama Rao Nidamanuri, and Ramakrishnan Krishnan. (2016). Segmentation based building detection approach from LiDAR point cloud, Egypt. J. Remote Sensing Space Sci. DOI: 10.1016/j.ejrs.2016.04.001. **Publisher : Elsevier**

4. Ramiya, Anandakumar M., Rama Rao Nidamanuri, and Ramakrishnan Krishnan. (2014). Semantic labelling of urban point cloud data, ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-8, 2014, pp.907-911. DOI : 10.5194/isprsarchives-XL-8-907-2014

5. Ramiya, Anandakumar M., Rama Rao Nidamanuri, and Ramakrishnan Krishnan. (2015). Supervoxel approach for 3D object-based point cloud labelling, Submitted as an entry to IEEE Data fusion 2015 contest (10 entries worldwide).

Conference Proceedings

1. Ramiya, Anandakumar M., Rama Rao Nidamanuri, and Ramakrishnan Krishnan. (2014). Comparative analysis of 3D segmentation algorithms for building detection from LiDAR point cloud data, Research Scholars day, Indian Institute of Space Science and Technology, India.

Manuscript under review

1. Ramiya, Anandakumar M., Rama Rao Nidamanuri, and Ramakrishnan Krishnan. (2016). Critical assessment of semantic object based point cloud labelling on urban LiDAR dataset, ISPRS Journal of Photogrammetry and Remote Sensing **Publisher : Elsevier**