CONTINUOUS MODELING OF 3D BUILDING ROOFTOPS FROM AIRBORNE LIDAR AND IMAGERY

JAEWOOK JUNG

A DISSERTATION SUBMITTED TO
THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

GRADUATE PROGRAM IN EARTH AND SPACE SCIENCE
YORK UNIVERSITY
TORONTO, ONTARIO

MAY 2016

© JAEWOOK JUNG, 2016

ABSTRACT

In recent years, a number of mega-cities have provided 3D photorealistic virtual models to support the decision making process for maintaining the cities' infrastructure and environment more effectively. 3D virtual city models are static snap-shots of the environment and represent the status quo at the time of their data acquisition. However, cities are dynamic systems that continuously change over time. Accordingly, their virtual representations need to be regularly updated in a timely manner to allow for accurate analysis and simulation results that decisions are based upon.

The concept of "continuous city modeling" is to progressively reconstruct city models by accommodating their changes recognized in the spatio-temporal domain, while preserving unchanged structures. As one of the most prominent objects comprising the virtual city model, automatic reconstruction of building rooftops have been targeted by many researchers over the last three decades. However, the goal of error-free rooftop reconstruction from remotely sensed data is still not achieved yet. Moreover, most of the existing research works have focused on the reconstruction of rooftops using a single source of data captured at one specific epoch. Not many research methods have been proposed for addressing the issues related to progressive reconstruction of rooftops using multi-sensor data.

This thesis proposes a novel research framework for continuously reconstructing 3D building rooftops using multi-sensor data, which are acquired at different epochs. For achieving this goal, we first propose a 3D building rooftop modeling method using a popular single data source (i.e., airborne LiDAR data). The main focus is on the

implementation of an implicit regularization method which imposes a data-driven building regularity to noisy boundaries of roof planes for reconstructing 3D building rooftop models. The "implicit regularity" is achieved by introducing flexible regularity constraints which can be adjusted to the given objects. The implicit regularization process is implemented in the framework of Minimum Description Length (MDL) combined with Hypothesize and Test (HAT). Secondly, we propose a context-based geometric hashing (CGH) method to align newly acquired image data with existing building models as a prerequisite process of the subsequent building refinement application. The novelty is the use of context features to achieve robust and accurate matching results. Thirdly, the existing building models are refined by a newly proposed sequential fusion method. The main advantage of the proposed method is its ability to progressively refine modeling errors frequently observed in LiDAR-driven building models. The refinement process is conducted in the framework of MDL combined with HAT. Markov Chain Monte Carlo (MCMC) coupled with Simulated Annealing (SA) is employed to perform a global optimization. Lastly, we propose an evaluation metric to robustly assess various quality aspects of reconstructed and refined 3D building models. The performance of the proposed methods have been evaluated using the International Society of Photogrammetry and Remote Sensing (ISPRS) benchmark datasets. The results demonstrate that the proposed continuous rooftop modeling methods show promising aspects to support various critical decisions by not only reconstructing 3D rooftop models accurately, but also by updating the models using multi-sensor data.

To God

And to my family

ACKNOWLEDGMENTS

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

It would be difficult to overstate my gratitude to my supervisor, Dr. Gunho Sohn, who provided me with his excellent guidance and great encouragement. His academic passion inspire me to grow into a better scientist. Without his valuable advice, I would never been able to accomplish my research and this dissertation. Meeting him was the luckiest thing that happened in my lifetime.

I would like to extend my gratitude to the committee members, Dr. Costas Armenakis, Dr. Regina Lee, Dr. Franz Newland, Dr. Yun Zhang, and Dr. Minas Spetakis, who offered many valuable advices and insightful discussions and suggestions on my research. Such great helps from them resulted in a considerable improvement over this dissertation. I especially thank Dr. Costas Armenakis who provided encouragement during my PhD. I also thank Dr. Woosug Cho for his continuous encouragement.

I would like to thank all GeoICT members, Kiin Bang, Yoonseok Jwa, Heungsik Kim, Junie Zhang, Langyue Wang, Solomon Chan, Connie Ko, Chao Lue, Mojgan Jadidi, Kivanc Babacan, Ali Baligh, Phillip Robbins, Eros Gulo, Leihan Chan and Ravi Persad. I thank Andreas Wichmann and Mozhdeh Shahbazi, who collaborated on many researches. Also, I especially thank my old friends, Kwangsik Choi and Juhyeong Hwangbo.

I wish to thank Ontario Centres of Excellence (OCE), Natural Science and Engineering Research Council of Canada (NSERC) Discovery, Mitacs, and Hyundai for arranging the financial support for this PhD research.

Above all, my heartfelt and everlasting gratitude goes to my family, especially my parents, Byeongsam Jung and Younghee Hwang, my sister, Seunghye Jung, my wife, Kyoungim Jung, and my most lovely son, Jaejin, for their patience, encouragement and unconditional love.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ALS	Airborne Laser Scanning
BSP	Binary Space Partitioning
CGH	Context-based Geometric Hashing
CRF	Conditional Random Field
DL	Description Length
DN	Digital Number
DP	Douglas Peucker
DPW	Digital Photogrammetric Workstation
DSM	Digital Surface Model
EOP	Exterior Orientation Parameters
GCP	Ground Control Point
GSD	Ground Sample Distance
HAT	Hypothesize And Test
I-Corner	Image Corner
I-Line	Image Line
IOP	Interior Orientation Parameter
ISPRS	International Society of Photogrammetry and Remote Sensing
L-Model	LiDAR-driven Model
LiDAR	Light Detection And Ranging
LOD	Level Of Detail
MAR	Mobile Augmented Reality
MCMC	Markov Chain Monte Carlo
MDL	Minimum Description Length
MLS	Mobile Laser Scanning
MRF	Markov Random Field
MS	Multi-Spectral
NDVI	Normalized Difference Vegetation Index
PAN	PANchromatic
RANSAC	RANdom SAmple Consensus

RJMCMC	Reversible Jump Markov Chain Monte Carlo
RMSE	Root Mean Square Error
RTG	Roof Topology Graph
SA	Simulated Annealing
SFM	Structure From Motion
TIN	Triangulated Irregular Network
TLS	Terrestrial Laser Scanning
UAV	Unmanned Aerial Vehicle

Chapter 1
Introduction

1.1 Motivation

Urbanization is an inevitable movement which is not merely a modern phenomenon, but a rapid and historic transformation of human social roots on a global scale. According to the United Nations (UN), half of the world's population lived in urban areas at the end of 2008 and the number will increase to about 70 percent by 2050 (International Herald Tribune, 2008). The rapid urbanization has led the dramatic change of city environments and has presented an urgent need to construct, synthesize and update environmental information for the purpose of planning, managing, and making various critical decisions that impact growing cities. To create useful and accurate representations of various dynamic city entities, researchers put forth numerous efforts in computer vision, photogrammetry and remote sensing fields in the last few decades. Particularly, a building, a structure very closely connected with human life, is recognized as the most important object in generating of 3D virtual models of city environment. A study by the European Organization for Experimental Photogrammetric Research (OEEPE) highlighted the demand for accurate 3D city models (Fuchs et al., 1998). Of note, 95% of the study's participants identified three-dimensional building data as the most interesting feature in digital city models, clearly emphasizing the importance of buildings in representing urban environments. Consequently, since initial efforts on automatic building extraction from

remotely sensed data in the early 1990s (Grün et al. 1995, 1997), a large number of research studies have been conducted to recognize, detect, reconstruct and represent building objects (Baltsavias, 2004; Brenner, 2005; Remondino and EL-Hakim, 2006; Mayer, 2008; Haala and Kada, 2010; Musialski et al., 2012; Wang, 2013; Tomljenovic et al., 2015). As a result, many applications for web mapping services and mobile use have been developed by major companies, including Google, Apple, HERE, and Uber, and are able to provide 3D building models for consumer use (Figure 1.1).

The 3D building models are used as base data for many geo-spatial information-based applications such as coordination, web mapping service, and navigation. Furthermore, recently emerging technologies such as Mobile Augmented Reality (MAR) allow the 3D building models to be used as an interactive tool on a computer or mobile device. Users enter queries utilizing the building models and the computer responds to the request by presenting relevant information of the building on a display. Thus, accurate and reliable 3D building models are an essential prerequisite to support these applications.

According to Skyscraperpage.com, in 2015, there were over 2,000 high-rise buildings in the city of Toronto and 139 high-rise and mid-rise buildings were under construction in January that year (Economic Dashboard-Annual Summary, 2015). With expansion of different types of building structures, even more changes are expected to take place in the cityscapes. A city is a dynamic entity as the environment continuously changes. Accordingly, its virtual models also need to be regularly updated. In order to address the continuous changes in the city environment, companies like HERE have been updating their maps on a bi or tri-monthly basis (HERE 360, 2015). However, for a large-scale area,

newly generating building models whenever new data is acquired is cost-inefficient and labor-intensive. Therefore, existing building models should be reused and appropriately updated in cost-effective and automatic manners to record changes. As such, continuous modeling of 3D cityscapes using remotely sensed multi-data taken at different epochs is expected to play an important role in generating timely and accurate building models.

Figure 1.1 Photorealistic 3D building models on Google Earth (Location: Toronto, Ontario, Canada)

In terms of data sources, the advance of data acquiring technologies has made it possible to reconstruct 3D building models. Aerial images have been one of the most common sources and are considered indispensable. At the early stage of building reconstruction, image data was manually digitized to depict building boundaries in a stereo view using a digital photogrammetric workstation (DPW). With the arrival of automatic computer techniques from computer vision, the photogrammetric approach enables

automatic extraction of modeling cues (collection of building evidences). More recently, structure from motion (SFM) techniques with the help of feature descriptors such as SIFT (Lowe, 2004) and SURF (Bay et al., 2008) provide methods to automatically register unordered multiple images and to recover sparse 3D point clouds whose density can be later increased by dense matching methods. However, the few main disadvantages of image data, such as the low level of automation due to incomplete modeling cue extraction and matching ambiguity caused by the effects of shadows, low contrast, and occlusion, still remains. Thus, it requires manual editing or human intervention to fully describe the buildings. On the other hand, the emergence of airborne LiDAR system (ALS) in the middle of 1990s has made significant changes in automatic building rooftop reconstruction. Airborne LiDAR, as an active sensor, directly provides 3D point clouds over a large scale scene with a high degree of accuracy. Its direct geo-referencing ability improves the level of automation in the building reconstruction process. Nowadays, with improvements of laser scanning techniques, obtaining accurate and dense points over a large-scale area has become feasible for building rooftop reconstruction. Furthermore, these sensors, mounted on various platforms such as ground, mobile, and unmanned aerial vehicle (UAV), have provided new types of data: Image sensors mounted on UAVs provide oblique images and video streams; and Terrestrial Laser Scanning (TLS) and Mobile Laser Scanning (MLS) systems produce very dense 3D points for building facades. These data taken from different time epochs and from different viewpoints encourage a full description of 3D building models. However, in-depth understanding of data characteristics and registration between data is required for accurate and reliable 3D building modeling.

In the perspective of building reconstruction, raw data acquired from remotely sensed data are converted into "building models". A large number of building reconstruction methods, which range widely in terms of levels of automation (automatic vs. semi-automatic), data sources (single data vs. multi-data), and data processing strategies (data-driven, model-driven, or hybrid), have been explored to effectively represent a full description of buildings. However, in spite of constant efforts, developing a "universal" intelligent machine enabling the massive generation of highly accurate rooftop models in a fully-automated manner still remains a challenging task. Many researchers (Ameri, 2000; Sohn and Dowman, 2007) pointed out several reasons for the problem as follows:

- ***Scene Complexity***: Remotely sensed data from the urban scene contain a large amount of information of non-building objects (e.g., ground, tree, car, and clutter) in addition to the building objects. Although some heuristic knowledge (e.g., building height, certain brightness, or nearby shadow) can be used to recognize building objects, detecting individual buildings is not easy because buildings are attached and form blocks. In terms of building interpretation, buildings in urban scenes have enormous variants in structure and shapes with multi-story planes, the landmark buildings of the city in particular. The variety of shapes cannot be described by common types of building structures. Thus, a method to simplify complex building scenes is required for effective interpretation.
- ***Incomplete cues***: There is always a significant loss of information in data. Occlusion of buildings or building parts by themselves or adjacent objects causes problems in data integrity. Also, shadow, noise, low contrast, and superstructures

on building roofs cause redundant or spurious cues, bringing about ambiguity and confusion to the building reconstruction process.

- ***Sensor dependency***: Sensors used for building modeling have unique characteristics related to the acquisition mechanism. This inherent property has a considerable influence over the reconstructed building models; for instance, LiDAR data provides accurate plane information, while the accuracy of building boundaries is less than that of image data due to its irregular point distribution. Thus, fully understanding sensor characteristics is one of the most important tasks in building reconstruction.

Even though many algorithms for reconstructing 3D building models using single data source have been introduced and can provide promising results (Rottensteiner, 2014), the methods still have some limitations due to inherent sensor dependent properties, levels of automation, model accuracies and missing data problems. One promising approach to address these problems is to combine multi-sensor data which have different characteristics. In this regard, combining LiDAR point clouds and optical imagery for building reconstruction have been exploited by many researchers (Haala and Kada, 2010). This is due to the fact that the characteristics of the modeling cues from the two data are complementary. Compared to LiDAR point clouds, the optical imagery better provides semantically rich information, geometrically accurate step and eave edges, while it has weakness in detecting roof edges and 3D information such as planar patches when single

imagery is used. However, LiDAR has somewhat opposite characteristics to optical imagery.

Generally, data fusion for building reconstruction can be divided into two approaches: parallel fusion and sequential fusion (Sohn et al., 2013). The parallel fusion approach allows each modeling cue to be extracted from two datasets in parallel. Then, a rooftop model is generated through various mechanisms recovering its spatial topology using the extracted modeling cues. In contrast, sequential fusion generates a building rooftop model relying on a single information source, which is later refined by the other data. Although the sequential fusion approach has not been studied as extensively as the parallel fusion approach, it is expected to play an important role in continuous modeling. In the sequential fusion framework, existing 3D building models can be updated using newly acquired data taken from different epochs.

Regardless of which fusion approach is applied, the registration between different sensor data is recognized as an essential and prerequisite process. The accuracy of registration has a substantial impact on the quality of results. The registration method should provide accurate and robust relations between datasets taken from different sensors or from different viewpoints at different epochs. In addition, a registration between existing models and newly taken sensor data should be addressed, particularly in continuous city modeling. However, while many registration methods that deal with correspondence problems between different sensor data have been studied, the registration between valuable 3D building models over a large-scale area and remotely sensed data has

been studied relatively less. Therefore, more research on development of registration methods using valuable 3D building models are required for continuous city modeling.

1.2 Research Objectives

As discussed in the previous section, it is obvious that reconstruction and update of city objects, particularly buildings, is essential to making various critical decisions impacting the city environment. The overall objective of this thesis is to address critical steps toward making available continuous city modeling, which includes 3D building rooftop reconstruction, model-to-image registration, update of building models and quality evaluation. In order to achieve the overall goal, several issues need to be addressed, as follows:

First, the proposed building reconstruction method should provide accurate and robust 3D building rooftop models. The accuracy of reconstructed building models should meet engineering level accuracy to support critical decisions in the city environment. Regardless of scene complexity and the configuration of buildings, the methods should produce geometrically and topologically correct 3D building rooftop models. Secondly, the proposed methods should produce regularized building models. A building is constructed with certain regularities such as orthogonality, parallelism, and symmetry. These regularities should be taken into account in the building reconstruction process so that the model represents the regular properties of real buildings. Thirdly, a reliable registration accuracy should be achieved when using multi-sensor data for continuous city modeling to be successful. A newly taken datum should be robustly and accurately aligned with the

existing 3D building models. Finally, the numerous valuable existing 3D building models should be able to be effectively updated and their modeling errors corrected. Also, automatic methods should be proposed to deal with large scale scene.

1.2.1 General Research Framework

Figure 1.2 represents the workflow of the subsequent processes and the interrelation between the major components of the continuous city modeling proposed in this thesis.

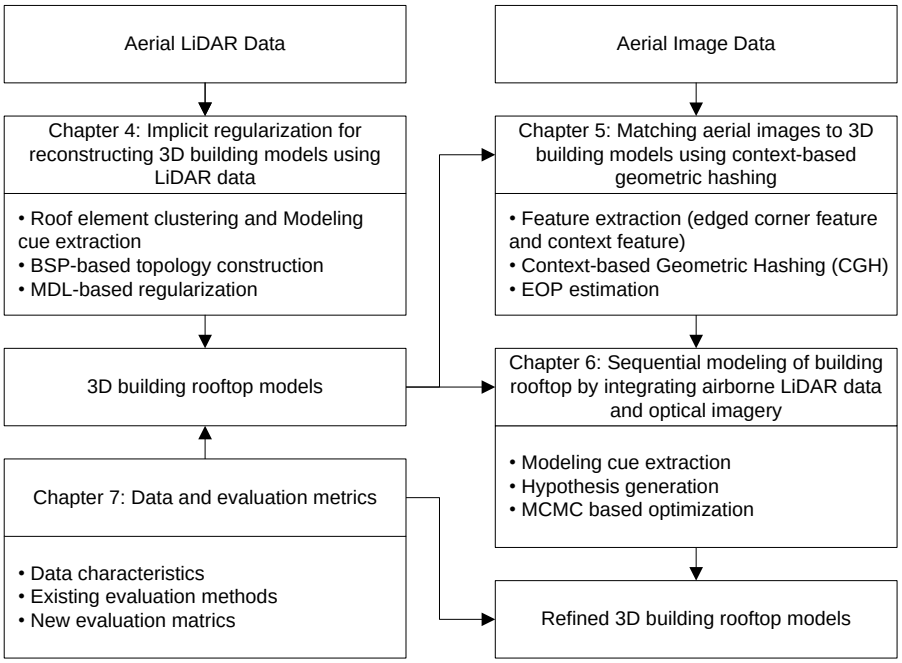


Figure 1.2 Proposed setup for continuous modeling

First, 3D building rooftop models are reconstructed using airborne LiDAR data (Chapter 4). The method presents a full chain of 3D rooftop modeling which cover from low level processing to more realistic models. The process consists of four main stages: 1)

element clustering, 2) modeling cue extraction, 3) topology construction, and 4) regularization. In element clustering and modeling cue extraction processes, this dissertation explains how the modeling evidence can be effectively gathered from complex building scenes. In topology construction, the Binary Space Partitioning (BSP) technique proposed by Sohn et al. (2007) is utilized to recover geometrically and topologically correct rooftop models from incomplete modeling cues. As a main part of the study, an implicit regularization method based on Minimum Description Length (MDL) is applied to produce regularized 3D building rooftop model. In the proposed MDL-based objective function, the weight parameters are automatically determined based on a Min-Max weighting method and Entropy-based weighting method.

Secondly, a model-to-image registration method using context-based geometric hashing aligns a single image with existing LiDAR-driven building rooftop models (Chapter 5). The method consists of three typical registration steps: 1) feature extraction, 2) similarity measure and matching, and 3) EOPs estimation. In the feature extraction step, two new features, the edged corner feature and the context feature, are introduced. For similarity measure and matching, the geometric hashing method is refined by introducing a newly designed score function which consists of a unary term and context term. EOPs of a single image are adjusted by the least square method based on collinearity equations.

Thirdly, this dissertation proposes a sequential fusion method to refine LiDAR-driven building models by incorporating image and airborne LiDAR data (Chapter 6). The sequential fusion method progressively rectifies geometrical and topological errors based on Hypothesize and Test (HAT) optimization using MDL. A new method to generate

hypotheses is designed by introducing topological lines connecting different data sources. Markov Chain Monte Carlo (MCMC) coupled with Simulated Annealing (SA) is employed to perform global optimization.

Lastly, an evaluation metric is proposed to assess the quality of reconstructed building models (Chapter 3). New evaluation methods, which can measure shape similarity and angle similarity, are proposed in order to compensate for limitations of existing evaluation methods.

1.2.2 Contributions

As mentioned before, major components, which need to be addressed in continuous modeling, are identified and then a solution for each component is provided. More specifically, the contributions of this study can be summarized as follows:

- Suggesting an evaluation metric to assess the quality of reconstructed building rooftop models: In order to complement the limitation of existing evaluation methods, which mainly focus on measuring a local similarity, shape-based and angle-based methods, which can measure a global similarity of building models, are added to existing evaluation methods. The added evaluation methods are used to evaluate different characteristics of building models and show the performance of our proposed algorithms.
- Proposing an implicit regularization method for reconstructing 3D building models using LiDAR: Building regularity is implicitly imposed by introducing flexible regularity constraints in a framework of MDL combined with HAT. In contrast to

explicit regularization, the implicit regularization method provides flexibility for describing more complex rooftop models while preserving building regularities. Also, the parameters governing the MDL optimization are estimated based on a Min-Max weighting method and Entropy-based weighting method. The proposed weighting methods provide appropriate weight parameters, which balance sub-terms in MDL, by considering the properties of individual buildings.

- Proposing a new model-to-image method to align a single image with existing building models: Edged corner feature, which provides local information of building structure, and context feature, which provides global information, are used as features of a subsequent matching process. In order to complement standard geometric hashing, context-based geometric hashing method is proposed by introducing a newly designed score function. The key aspect in CGH method is that context term in the score function, which represents relations between edged corner features, is used to reduce matching ambiguity and to achieve accurate and robust matching results.
- Introducing a new sequential fusion method to refine LiDAR-driven building models: Modeling errors observed in LiDAR-driven building models are progressively rectified by incorporating image information based on HAT optimization using MDL. A novel concept of topological line is proposed to integrate modeling cues extracted from different information sources. MCMC coupled with SA is adopted to generate model hypotheses and perform a global

optimization where three proposition kernels are proposed to deal with transitions from the current configuration to a new configuration in Markov chain.

- Conducting comprehensive experiments and analyses over the large-scale datasets to support the proposed methods.

1.3 Thesis Outline

This thesis is organized in seven chapters. An overview of the chapters follows:

Chapter 1 presents an introduction to the motivation of this thesis, and the proposed methods and strategy for solving research questions.

Chapter 2 gives background information that aids in understanding of this thesis, and comprehensive literature reviews concerning building reconstruction, regularization, data fusion and registration.

Chapter 3 introduces the study area, data characteristics and evaluation methods. The existing evaluation methods are categorized according to their properties and a new evaluation metric is proposed to effectively assess the quality of reconstructed building models.

Chapter 4 presents a method to reconstruct 3D building rooftop models using LiDAR data. MDL-based regularization method is used to impose geometrical regularity on 3D building models. Weight parameters in the MDL-based objective function are automatically determined based on a Min-Max criterion and Entropy-based weighting method.

Chapter 5 proposes a new model-to-image registration method to register a single image with large-scale LiDAR-driven building models. Newly developed context-based geometric hashing is applied to estimate accurate EOPs of a single image.

Chapter 6 introduces a sequential fusion method to refine LiDAR-driven building models by integrating image information. A new method to generate hypotheses is designed by topological lines connecting two different data. A MCMC coupled with SA is employed to perform global optimization.

Chapter 7 provides the conclusion of this study and recommendations for future works.

Chapter 2

Background

Recent years have seen the urban percentage of the global population surpass one-half, and continuing growth in urban areas is projected to add 2.5 billion people to the world's urban population by 2015 (United Nation, 2014). Thus, buildings, one of the most significant assets supporting the urban system, have been considered key areas of research in computer vision, photogrammetry and remote sensing fields over the past few decades. To address the various critical issues caused by rapid urbanization, many researchers have studied computational algorithms to provide 3D photo-realistic building models in an automated manner for supporting effective design, planning and maintenance of urban systems. In this chapter, we review a number of previous research works related to building reconstruction, registration and data fusion methods. The first part of this chapter discusses different data processing strategies (model-driven vs. data-driven approaches) used for building model reconstruction and regularization. The second part reviews existing works addressing data fusion methods to combine the information retrieved from airborne LiDAR and imagery for building modeling process. The last part introduces existing registration methods, a prerequisite process for geometrically co-aligning multi-sensor data, particularly focusing on model-to-image registration.

2.1 Building Reconstruction

Building reconstruction can be recognized as a huge process for the generation of digital representations of physical buildings where raw data without any structured information are converted into highly structured 3D building models with rich semantic information. Since initial efforts for automatically generating 3D building models began in early 1990s, numerous techniques using various remotely sensed data have been explored in computer vision, photogrammetry and remote sensing fields. In this section, we review existing building reconstruction methods in terms of reconstruction strategy (section 2.1.1) and regularization (section 2.1.2).

2.1.1 Model-driven vs. Data-driven

Numerous building reconstruction algorithms have been published for the past two decades. Although it is difficult to clearly classify these various methods into specific categories, there are several ways to categorize the methods: the used data source (single vs. multi-sources), the data processing strategy (data-driven (or generic), model-driven (or parametric)), and the amount of human interaction (manual, semi-automatic, or fully automated) (Vosselman and Mass, 2010). Of those, classifying existing methods into data-driven or model-driven approaches provides a good insight for understanding and developing 3D building model reconstruction algorithms.

In the model-driven approaches, 3D building models are reconstructed by fitting parameterized primitives to data. This is possible due to the fact that many buildings in rural and suburban area have common shapes in whole building or building roof parts.

These common roof shapes such as flat, gable, and hip roof are considered as standard primitives for representing building rooftop structures. Simple buildings can be well represented as regularized building models using pre-defined parameterized primitives even with low density data and presence of missing data. However, complex buildings and arbitrarily shaped buildings are difficult to model using a basic set of primitives. Also, the selection of the proper primitives among a set of primitives is not an easy task. In order to address the limitations, Verma et al. (2006) presented a parametric modeling method to reconstruct relatively complex buildings by combining simple parametric roof shapes that are categorized into four types of simple primitives. In this study, the roof-topology graph is constructed to represent the relationships among the various planar patches of approximate roof geometry. The constructed roof-topology graph is decomposed into sub-graphs, which represents simple parametric roof shapes, and then parameters of the primitives are determined by fitting LiDAR data. Although they decomposed complex buildings into simple building parts, many building parts cannot be still explained by their four simple shape primitives. Similarly, Milde et al. (2008) reconstructed 3D building models by matching sub-graphs of the region adjacency graph (RAG) with five basic roof shapes and then by combining them using three connectors. Kada and McKinley (2009) decomposed the building's footprint into cells which provided the basic building blocks. Three types of roof shapes including basic, connecting, and manual shapes are defined. Basic shapes consist of flat, shed, gabled, hipped, and Berliner roofs while connecting shapes are used to connect the roofs of the sections with specific junction shapes. The parameterized roof shapes of all cells are determined from the normal direction of LiDAR

points. The entire 3D building model is represented by integrating the parameterized roof elements with the neighboring pieces. Although a high level of automation is achieved, the method still requires manual works to adjust cell parameters and to model more complex roof shapes like mansard, cupola, barrel, and even some detail elements. Lafarge et al. (2010) reconstructed building models from a digital surface model (DSM) by combining generic and parametric methods. Buildings are considered as assemblages of 3D parametric blocks from a library. After extracting 2D building supports, 3D parametric blocks are placed on the 2D supports using Gibbs model which controls both the block assemblage and the fitting to data. The optimal configuration of 3D blocks is determined using the Bayesian framework. They mentioned that the optimization step needs to be improved to achieve both higher precision and shorter computing time as future work. Based on a predefined primitive library, Huang et al. (2013) conducted a generative modeling to reconstruct roof models that fit the data. The library provides three groups including 11 types of roof primitives whose parameters consist of position parameters, contour parameters, and shape parameters. Building roofs are represented as one primitive or an assemblage of primitives allowing primitives overlaps. For combining primitives, they derived combination and merging rules which consider both vertical and horizontal intersections. Reversible Jump Markov Chain Monte Carlo (RJMCMC) with a specified jump mechanism is conducted for the selection of roof primitives, and the sampling of their parameters. Although they have shown potential and flexibility of their method, there are issues to be solved: 1) uncertainty and instability of the reconstructed building model,

2) influence of prior knowledge and scene complexity on completeness of the reconstruction, and 3) heavy computation time.

In contrast with model-driven approaches, data-driven approaches do not make any assumptions regarding to the building shapes, thus they can theoretically handle all kinds of buildings. However, the approach may cause considerable deformations due to the sensitivity to surface fluctuations and outliers in the data. Also, it requires a regularization step during the reconstruction process. In general, the generic approach starts by extracting building modeling cues such as surface primitives, step lines, intersection lines, and outer boundary lines followed by reconstructing the 3D building model.

The segmentation procedure for extracting surface primitives divides a given data set into homogeneous regions. Classical segmentation algorithms such as region growing (Rottensteiner et al., 2005, Kada and Wichmann, 2012) and RANSAC (Tarsha-Kurdi et al., 2008) can be used for segmenting building roof planes. Also, Sampath and Shan (2010) conducted eigenanalysis for each roof point within its Voronoi neighbourhood, and then adopted the fuzzy k-means approach to cluster the planar points into roof segments based on their surface normal. Then, they separated the clusters into parallel and coplanar segments based on their distance and connectivity. Lafarge and Mallet (2012) extracted geometric shapes such as planes, cylinders, spheres, or cones for identifying the roof sections by fitting points into various geometric shapes, and then proposed a method for arranging both the geometric shapes and the other urban components by propagating point labels based on MRF. Yan et al. (2014) proposed a global solution for roof segmentation. Initial segmentation is optimized by minimizing a global energy function consisting of the

distances of LiDAR points to initial planes, spatial smoothness between data points, and the number of planes.

After segmenting points or extracting homogeneous surface primitives, modeling cues such as intersection lines and step lines can be extracted based on geometrical and topological relationships of the segmented roof planes. Intersection lines are easily obtained by intersecting two adjacent planes or segmented points while step lines are extracted at roof plane boundary with abrupt height discontinuity. In order to extract step lines, Rottensteiner et. al (2005) detected edge candidate points and then extracted step lines from an adjustment considering edge points within user-specified threshold. Also, Sohn et al. (2008) proposed a step line extractor, called Compass Line filter (CLF), for extracting straight lines from irregularly distributed LiDAR points. Although outer boundary is one type of step line, it is recognized as a separate process in many data-driven approaches. Some researchers delineated initial boundary lines from building boundary points using alpha shape (Dorninger and Pfeifer, 2008), ball-pivoting (Verma et al., 2006), and contouring algorithm (Zhou and Neumann, 2008). Then, the initial boundary was simplified or regularized. The detail reviews for simplification or regularization of boundary will be given in section 2.1.2.

Once all building modelling cues are collected, 3D building models are reconstructed by aggregating the modelling cues. In order to reconstruct topologically and geometrically correct 3D building models, Sohn et al. (2008) proposed the Binary Space Partitioning (BSP) which progressively partitions a building region into homogeneous binary convex polygons. Rau and Lin (2011) proposed a line-based roof model

reconstruction algorithm, namely TIN-Merging and Reshaping (TMR), to reconstruct topology with geometric modeling. Oude Elberink and Vosselman (2009), and Perera and Maas (2014) used a roof topology graph to preserve roof topology. In the latter, roof corners are geometrically modeled using the shortest closed cycles and the outermost cycle derived from the roof topology graph.

As mentioned before, a model-driven approach and a data-driven approach have different characteristics in the modeling process. Thus, Satari et al.(2012) proposed a multi-resolution hybrid approach to combine advantages of model-driven and data-driven approaches. In their study, the data-driven method was applied to reconstruct the main roof planes while the model-driven method was adopted to the models of appended parts such as dormers.

2.1.2 Building Boundary Regularization

Detection of building boundary is an intermediate step for 3D building reconstruction although it is not required in all building reconstruction algorithms. Generally, the initial boundary extracted from irregular LiDAR points have jagged shape with large numbers of vertices. Thus, a simplification or regularization process is required to delineate plausible building boundaries with certain regularities such as orthogonality, parallelism, and symmetry. Various techniques related to the regularization of building boundary have been proposed in the literature. Reviews on boundary detection were given by Brenner (2010). In most methods, the boundary detection process starts by extracting boundary points from segmented points. From extracted boundary points, initial building boundaries are

generated by tracing boundary points followed by a simplification or regularization process which improves the initial boundary. The easiest method to improve initial boundary is to simplify the initial boundary by removing vertices but preserving relevant points. The well-known Douglas-Peucker (DP) algorithm (Douglas and Peucker, 1973) is widely recognized as the most visually effective line simplification algorithm. The algorithm starts by selecting two points which have the longest distance and recursively adding vertices whose distance from the line is less than a given threshold. However, the performance of the algorithm fully depends on the used threshold and is substantially affected by outliers. Another approach extracts straight lines from boundary points using the Hough Transform (Morgan and Habib, 2002) or using RANSAC (Fishcler and Bolles, 1981). The extracted lines are then connected by intersections of the extracted straight lines to generate closed outer boundary lines. However, Brenner (2010) pointed out that the methods require some additional steps due to missing small building edges.

On the other hand, the regularization process imposes certain regularities when the initial boundary is simplified. Vosselman (1999) assumed that building outlines are along or perpendicular to the main direction of a building. After defining the position of a line by the first two boundary points, the line is updated using the succeeding boundary points until the distance of a point to the line exceeds some bound. The next line starts from this point in a direction perpendicular to the previous line. A similar approach was proposed by Sampath and Shan (2007). They grouped points on consecutive edges with similar slopes and then applied a hierarchical least squares solution to fit parametric lines representing the building boundary.

Some methods are based on the model hypothesis and verification approach. Ameri (2000) introduced the Feature Based Model Verification (FBMV) for modification and refinement of polyhedral-like building objects. In their approach, they imposed the geometrical and topological model information to the FBMV process as external and internal constraints which consider linearity for straightening consecutive lines, connectivity for establishing topology between adjacent lines, orthogonality, and co-planarity. Then, the weighted least squares minimization was adopted to produce a good regularized description of a building model. Weidner and Förstner (1995) adopted the Minimum Description Length (MDL) concept to regularize noisy building boundaries. For four local consecutive points, ten different hypothetical models are generated with respect to regularization criteria. Then, MDL, which depends on the mutual fit of the data and model and on the complexity of the model, is used to find the optimal regularity of the local configuration. Jwa et al. (2008) extended the MDL-based regularization method by proposing new implicit hypothesis generation rules and by re-designing model complexity terms where line directionality, inner angle and number of vertices are considered as geometric parameters. Furthermore, Sohn et al. (2012) used the MDL-based concept to regularize topologies within rooftop model. Zhou and Neumann (2012) introduced global regularities in building modeling to reflect the orientation and placement similarities among 2.5D elements which consist of planar roof patches and roof boundary segments. In their method, roof-roof regularities, roof-boundary regularities, and boundary-boundary regularities are defined and then the regularities are integrated into a unified framework.

2.2 Data Fusion

The integration of data and knowledge from several sources is known as data fusion. Hall and Llinas (1997) defined data fusion as follows: "Data fusion techniques combine data from multiple sensors and related information from associated databases to achieve improved accuracy and more specific inferences than could be achieved by the use of a single sensor alone." In the remote sensing community, data fusion combines multiple sources of data acquired with different spatial and spectral resolution to improve the potential values and interpretation performances of the source data and to produce a high-quality visible representation of data. Remote sensing fusion techniques can be classified into three different levels: 1) pixel/data level, 2) feature level, and 3) decision level (Pohl and van Genderen, 1998). Pixel level fusion combines raw data from multiple sources to yield a single resolution datum. The pixel level fusion of optical images is well known as the pan-sharpening technique which improves spatial resolution of panchromatic (PAN) image by injecting structural and textual details of multi-spectral (MS) images or SAR images. Feature level fusion combines features extracted from multiple data sources. Because features are extracted from different characteristics of different sensors, the extracted features can provide additional valuable properties for various applications. Decision or interpretation level fusion combines the results, which are individually processed, to make a final decision. The decision level fusion methods contain voting methods, statistical methods, and fuzzy methods. Zhang (2010) provides reviews on the three different levels of fusion techniques.

In this thesis, our interest is feature level fusion. In particular the fusion of LiDAR and optical images for building reconstruction. Although many building reconstruction algorithms using single data provide some promising results, the integration of two complementary datasets can improve the quality of 3D building models with an increase of available information. In particular, combining LiDAR point clouds and optical images for rooftop modeling have been exploited by many researchers (Haala and Kada, 2010). In previous studies, image information in the fusion approach is mainly used for four different purposes in terms of building reconstruction: 1) extraction of building points while removing non-building points such as tree points, 2) improvement of segmentation, 3) improvement of building boundary, and 4) texture mapping. For building region extraction, Chen et al. (2005) used spectral information and texture of color images. Sohn and Dowman (2007) used Normalized Difference Vegetation Index (NDVI) to discriminate between buildings and trees. Demir and Baltsavias (2012) detected building regions by combining results of four different building detection methods which were respectively derived from combinations of spectral information and NDVI of image data and spatial distribution of LiDAR data.

Awrangjeb et al. (2013) proposed an image line guided technique to robustly segment building points into individual roof planes. Lines extracted from images were classified into ground, tree, roof edge, and roof ridge-lines using the ground mask, colour and texture information of the image. Lines classified as roof edge or roof ridge were used to define robust seed regions for region growing for roof plane segmentation. Cheng et al. (2011, 2013) used images to refine initial roof point segmentation derived from LiDAR

data based on the Shrink-Expand technique. Spectral and texture information (entropy) of images were used as a criterion for judging the reliability of segmentation.

Rottensteiner and Briese (2003) proposed wire frame fitting to improve the geometric quality of the polyhedral models created from LiDAR data. Image edges were matched with LiDAR-driven edges and then the matched image edges were considered in the estimation of model parameters. Hu et al. (2006) proposed a hybrid modeling system where building boundaries and plane surfaces were extracted from image and LiDAR data, respectively. Lee et al. (2008) proposed a method to extract the boundaries of complex buildings from LiDAR and photogrammetric images. Coarse building boundaries generated by LiDAR are simply substituted with image edges to extract precise building boundaries by matching with some constraints such as length ratio, angle and distance. Kim and Habib (2009) similarly replaced initial building boundaries by 3D lines which have the biggest spectral difference between two flanking regions. Sohn et al. (2013) proposed a sequential fusion method to improve the boundary quality of existing building models based on the hypothesis and test (HAT) framework. Image lines were used to propose possible hypotheses. Cheng et al. (2011, 2013) also used image data to extract building boundary and step lines. After establishing relationships between 2D image lines and 3D LiDAR points, 3D lines were determined from multi-view images. Two rectangle boxes along orthogonal directions of a line segment were analyzed to separate step and non-step line segments. 3D building models were reconstructed by segmented roof points, 3D step lines, 3D ridge lines, and 3D boundaries using the Split-Merge-Shape (SMS) method.

In order to achieve photorealistic rendering, Frueh et al. (2004) proposed a way to texture-map a LiDAR-driven 3D building models with oblique aerial images. After registering the oblique image with the existing building model, an optimal image for each triangle of the model was selected for texture by taking into account occlusion, image resolution, surface normal orientation, and coherence with neighbor triangles.

2.3 Registration

Registration is an essential process when multi-data sets are used for various applications such as object recognition, environmental monitoring, change detection, and data fusion. In computer vision, remote sensing, and photogrammetry, this includes registrations of the same source taken from different viewpoints at different times (e.g., image to image), between datasets collected with different sensors (e.g., image and LiDAR), and between an existing model and remotely sensed raw data (e.g., map and image). Numerous registration methods have been proposed to solve the registration problems for given environments and for different purposes (Brown, 1992; Fonseca and Manjunath, 1996; Zitova and Flusser, 2003; Mishra and Zhang, 2012). Regardless of data types and applications, the registration process can be recognized as a feature extraction, and correspondence problem (or matching problem) between datasets. Brown (1992) categorized the existing matching methods into area-based, and feature-based methods according to their nature. Area-based matching methods use image intensity values extracted from image patches. They deal with images without attempting to detect salient objects. Correspondences between two image patches are determined with a moving kernel sliding across a specific size of image

search window or across the entire other image using correlation-like methods (Kaneko et al., 2003), Fourier methods (Castro and Morandi, 1987), mutual information methods (Viola and Wells, 1997), and others. In contrast, feature-based methods use salient objects such as points, lines, and polygons to establish relations between two different datasets. In feature matching processes, correspondences are determined by considering the attributions of the used features. In model-to-image registration, most of the existing registration methods adopt a feature-based method because many 3D building models have no texture information.

In terms of features, point features such as line intersections, corners and centroids of regions can be easily extracted from both models and images. Thus, Wunsch and Hirzinger (1996) applied the Iterative Closest Point (ICP) algorithm to register 3D CAD-models with images. The ICP algorithm iteratively revises the transformation with two sub-procedures. First, all closest point pair correspondences are computed. Then, the current registration is updated using the least square minimization of the displacement of matched point pair correspondences. In a similar way, Avbelj et al. (2010) used point features to align 3D wire-frame building models with infrared video sequences using a subsequent closeness-based matching algorithm. Lamdan and Wolfson (1988) used a geometric hashing method to recognize 3D objects in occluded scenes from 2D grey scale images. However, Frueh et al. (2004) pointed out that point features extracted from images cause false correspondences due to a large number of outliers.

As building models or man-made objects are mainly described by linear structures, many researchers have used lines or line segments instead of points as features. Hsu et al.

(2000) used line features to estimate 3D pose of a video where coarse pose was refined by aligning projected 3D models of line segments to oriented image gradient energy pyramids. Frueh et. al. (2004) proposed a model to image registration for texture mapping of 3D models with oblique aerial images. Correspondences between line segments are computed by a rating function, which consists of slope and proximity. Because an exhaustive search to find optimal pose parameters was conducted, the method is affected by the sampling size of the parameter space, and it is computationally expensive. Eugster and Nebiker (2009) also used line features for real-time geo-registration of video streams from unmanned aircraft systems (UAS). They applied relational matching, which does not only consider the agreement between an image feature and a model feature, but also takes the relations between features into account. Avbelj et al. (2015) matched boundary lines of building models derived from DSM and hyper-spectral images using an accumulator. Iwaszczuk et al. (2013) compared RANSAC and the accumulator approach to find correspondences between line segments. Their results showed that the accumulator approach achieves better results. Yang and Chen (2015) proposed a method to register UAV-borne sequent images and LiDAR data. They compared building outlines derived from LiDAR data with tensor gradient magnitudes and orientation in images to estimate key frame-image EOPs. Persad et al. (2015) matched linear features between Pan-Tilt-Zoom (PTZ) video images with 3D wireframe models based on a hypothesis-verification optimization framework. However, Tian et al. (2008) pointed out several reasons that make the use of lines or edge segments for registration a difficult problem. First, edges or lines are extracted incompletely, and inaccurately, so that ideal edges might be broken into two

or more small segments that are not connected to each other. Secondly, there is no strong disambiguating geometric constraint, whereas building models are reconstructed with certain regularities such as orthogonality and parallelism.

Utilizing a prior knowledge of building structures can reduce the matching ambiguities and the search space. Thus, Ding et al. (2008) used 2D orthogonal corners (2DOC) as a feature to recover the camera pose for texture mapping of 3D building models. The coarse camera parameters were determined by vertical vanishing points that correspond to vertical lines in the 3D models. Correspondences between image 2DOC and DSM 2DOC were determined using Hough transform and generalized M-estimator sample consensus. However, they described their error source as too limited to correct 2DOCs matches, in particular for residential areas. Also, Wang and Neumann (2009) pointed out that 2DOC features are not very distinctive because the features can be extracted from only orthogonal corners. Instead of using 2DOC, they proposed 3 connected segments (3CS) as a feature which is more distinctive, and repeatable. For putative feature matches, they applied a two level RANSAC method, which consists of a local, and a global RANSAC for robust matching. Table 2.1 summarizes the existing model-to-image registration methods.

Table 2.1 Reviews on model-to-image registration methods

Author	Data type	Feature	Matching Method	Application
Lamdan and Wolfson, 1988	3D object model and 2D grey scale image	point	Geometric Hashing	Object recognition
Wunsch and Hirzinger, 1996	CAD-Model and image	point	Iterative Closest Point	Pose estimation
Hsu et al., 2000	3D model and Video	line	RANSAC	Pose estimation and Visualization
Frueh et al., 2004	3D building model and oblique aerial image	line	Rating function (slope and proximity)	Texture mapping
Ding et al., 2008	3D building model and oblique aerial image	2DOC (2D orthogonal corner)	Hough transform and generalized M-estimator sample consensus	Texture mapping
Eugster and Nebiker, 2009	3D building model and video steams	line	Relational matching	Real-time georegistration
Wang and Neumann, 2009	LiDAR and aerial image	3CS(3 connected segments)	Two level RANSAC	Texture mapping
Avbelj et al., 2010	3D building model and infrared video sequences	point	Closeness-based matching	Pose estimation
Iwaszczuk et al., 2013	3D building model and thermal infrared images	line	RANSAC and accumulator	Texture mapping
Avbelj et al., 2015	DSM and hyper-spectral image	line	Accumulator	Image fusion
Yang and Chen, 2015	UAV sequent image and LiDAR	line	Histogram-based matching	Key frame-image registration
Persad et al., 2015	PTZ video images and 3D wireframe model	line	Line-based Randomized RANdom Sample Consensus	Pose estimation

Chapter 3

Dataset and Evaluation Metrics

Evaluation is an essential process to analyze the performance of proposed algorithms. Although many performance evaluation methods have been proposed to assess the quality of 3D building models, the evaluation methods were designed for accommodating specific performance characteristics to be assessed. A new evaluation metric, which is appropriate for our research purpose, needs to be proposed. In this chapter, we describe datasets and an evaluation metric used to assess the performance of our proposed algorithms. In the first part of this chapter, we describe test datasets covering two different sites: 1) Vaihingen in Germany and 2) downtown Toronto in Canada. For each dataset, acquired data types and characteristics are explained in detail. In the second part of this chapter, existing performance evaluation methods are reviewed, and an evaluation metric for our research is proposed to properly evaluate different aspects of our results. The proposed evaluation metric consists of existing evaluation indices used in Rottensteiner et al. (2014) and newly proposed evaluation indices (shape-based indices and angle-based index). Also, conceptual design is introduced to explicitly assess topology accuracy based on the Roof Topology Graph (RTG).

3.1 Introduction

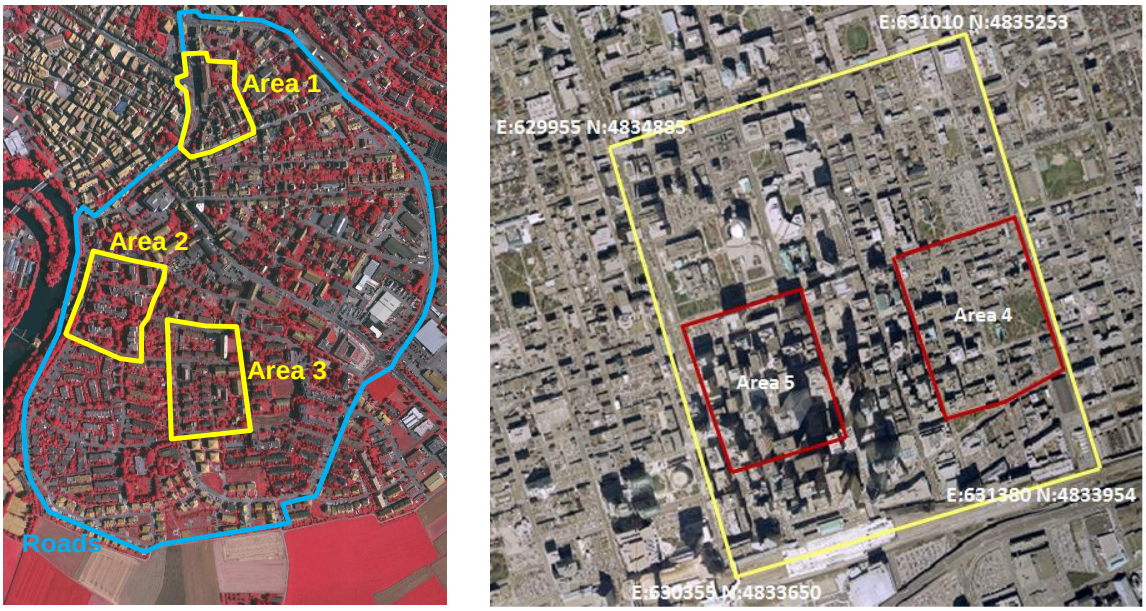
One of the main research objectives pursued in this study is to reconstruct accurate high-quality 3D building models. In this regard, the important question to be answered is how to measure the quality of the building models extracted from our proposed algorithms. In order to answer the question, we propose a novel evaluation metric. In this chapter, we first describe datasets used for our continuous modeling methods and their characteristics in detail (section 3.2). Secondly, we review existing evaluation methods in literature and then present a new evaluation metric to assess the performance of our proposed algorithms (section 3.3).

3.2 Datasets

In 2012, International Society of Photogrammetry and Remote Sensing (ISPRS) Working Group III/4 initiated a benchmark test on urban classification and 3D building reconstruction. This benchmark project supported by ISPRS, the German Society for Photogrammetry, Remote sensing and Geoinformation, and Teledyne-Optech provided state-of-the-art airborne data sets, which can be used by interested researchers in order to test their own data analytic algorithms on urban object classification and building reconstruction (Rottensteiner et al., 2012). By having a common test dataset, and evaluation metrics, researchers can conduct a comparative analysis of their own algorithms against others in a less data-sensitive and metrics-sensitive manner.

In this thesis, the benchmark datasets provided by the ISPRS WGIII/4 were used for evaluating the performance of our proposed methods. Two independent benchmark

datasets were acquired over Vaihingen in German, and downtown Toronto in Canada, respectively, with multi-sensor data including aerial images and airborne laser scanning (ALS) data. In addition, the ISPRS provides reference datasets, which include manually labelled classes, and building models (footprints and rooftop models in 3D) reconstructed by manual stereo plotting method. The ISPRS WGIII/4 also designed evaluation metrics to estimate the accuracy of the results produced by individual participants using the benchmark datasets; if a benchmark participant submits his/her 3D building modeling results to the ISPRS WGIII/4, the modeling performance is measured based on the working group's evaluation metrics and reference data. More detailed explanation on the ISPRS WGIII/4's evaluation metrics can be found in Rottensteiner et al. (2014) and via website (http://www2.isprs.org/commissions/comm3/wg4/detection-and-reconstruction.html). Figure 3.1 shows the coverage of the two datasets.



(a) (b)

Figure 3.1 Test datasets: (a) Vaihingen and (b) downtown Toronto

3.2.1 Dataset 1: Vaihingen

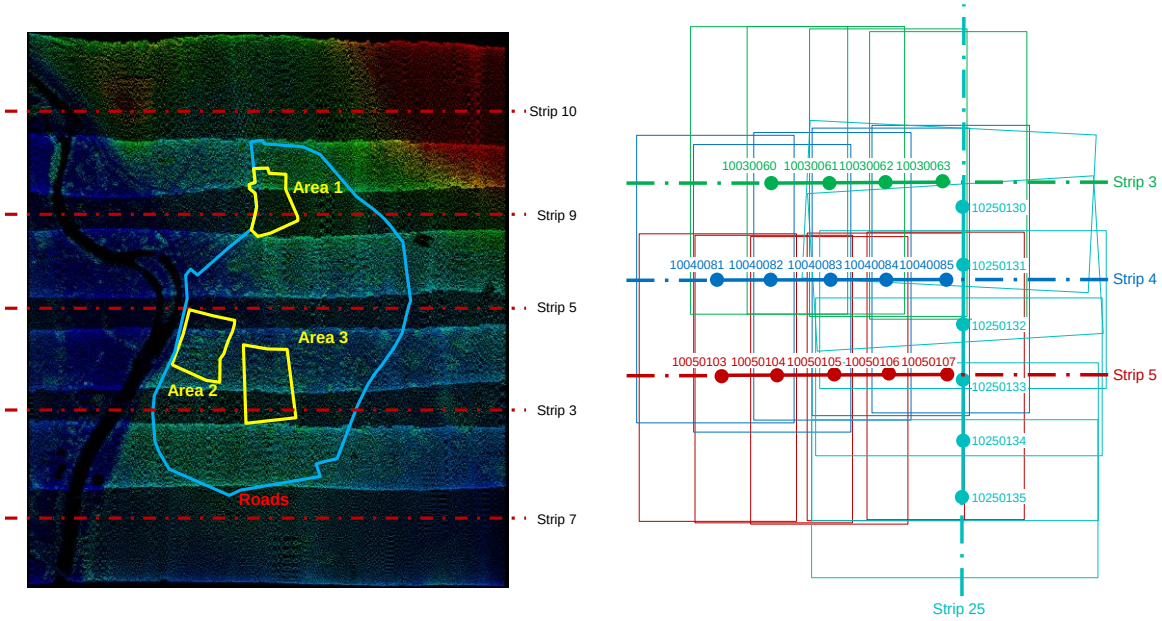
The first dataset consisted of aerial image and ALS data covering Vaihingen in Germany (8°57' E, 48°56' N). The ALS data consisted of 5 strips over the test area acquired by a Leica ALS50 system with 45° field of view at a mean flying height of 500m above ground level. The mean point density for each strip is 4 points/m^2 while the median point density with the overlap is 6.7 points/m^2 (i.e., ~0.39m point spacing). Multiple echoes and intensities were also recorded. The original point clouds were post-processed by strip adjustment to correct for systematic errors. The 3D positional accuracy shows approximately ±10cm. High-resolution pan-sharpened color images were also captured from the Intergraph Z/I imaging's DMC (Digital Mapping Camera) with the ground sampling distance of 8cm and the radiometric resolution of 11 bits. The area is covered by five overlapped strips with two additional cross strips (Figure 3.2(b)). The interior and exterior parameters were estimated at the level of 1 pixel georeferencing accuracy. Table 3.1 gives the interior orientation of the digital images of the Vaihingen area while Table 3.2 shows the flight parameters of the block. Reference building models were generated by manual stereo plotting with a planimetric accuracy of about 10 cm.

Table 3.1 Interior orientation of the digital images of the Vaihingen dataset

Camera	*Image Format*		*Pixel size (mm)*	*Focal length (mm)*	*Principal Point*	
	Row (pixel)	*Col (pixel)*			X_{pp} *(mm)*	Y_{pp} *(mm)*
DMC	7,680	13,824	0.012	120.00	0.00	0.00

Table 3.2 Flight parameters of the Vaihingen 8cm DMC block

Camera	*Focal length*	*Flying height above Ground*	*Forward overlap*	*Side lap*	*GSD*	*Spectral bands*	*Radiometric resolution*
DMC	120mm	900m	60%	60%	8cm	IR-R-G	11 bit



(a) (b)

Figure 3.2 (a) ALS strips and (b) image configuration for the Vaihingen dataset

This dataset is divided into three sub datasets; Area 1 (37 buildings; 125m×200m) contains historic buildings with complex shapes; Area 2 (14 buildings; 170m×190m) is characterized by a few high-rising residential buildings; Area 3 (56 buildings; 150m×220m) is a purely residential area with detached houses. Figure 3.3 shows three sub-datasets in the area of Vaihingen.

(a) (b) (c)

Figure 3.3 Vaihingen dataset: (a) Area1, (b) Area2, and (c) Area3

3.2.2 Dataset 2: Downtown Toronto

This dataset covers an area of about 1.45 km^2 in the central area of the City of Toronto in Canada. ALS data were acquired by Optech's ALTM-ORION M in February 2009 with the aircraft speed of 120 knots at the flying altitude of 650m. The ALTM-ORION M operates at a wavelength of 1064nm and scans the underlying topography with a scan width of 20° and the scan frequency of 50 Hz. The dataset consists of 6 strips and the average point density with the overlap is approximately 6.0 points/m^2 (i.e., ~0.41m point spacing). In addition to the ALS data, digital aerial images were taken by UltraCam-D with the ground sampling distance of 15cm and radiometric resolution of 8 bits (Table 3.4). The image data consist of three overlapping strips with 30% side lap and 60% forward overlap (Figure 3.4). The exterior orientation is estimated by a bundle adjustment method at the level of 1 pixel georeferencing accuracy. Table 3.3 gives a summary of camera parameters of the downtown Toronto dataset. The reference building models were generated by manual

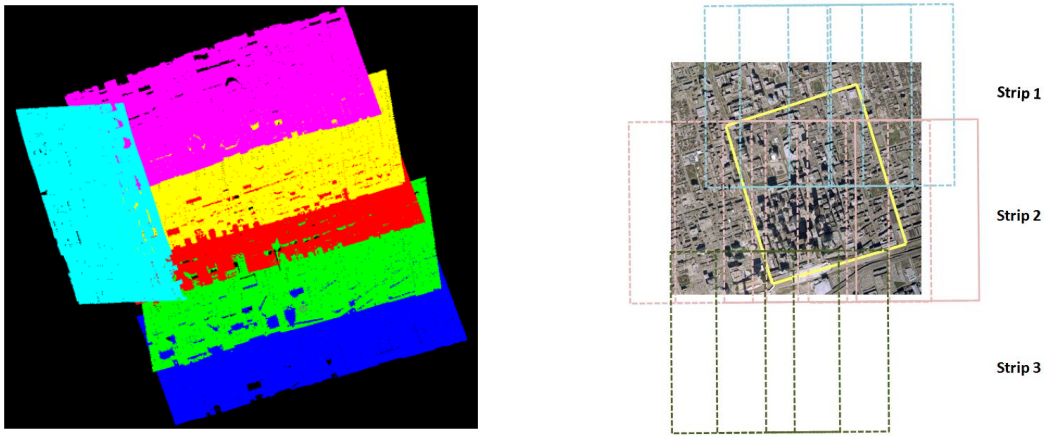
stereo plotting with planimetric accuracy of about 20 cm and height accuracy of about 15cm.

Table 3.3 Interior orientation of digital images of the downtown Toronto dataset

Camera	*Image Format*		*Pixel size (mm)*	*Focal length (mm)*	*Principal Point*	
	Row(pixel)	*Col (pixel)*			X_{pp} *(mm)*	Y_{pp} *(mm)*
UltraCam D	11,500	7,500	0.009	101.40	-0.18	0.00

Table 3.4 Flight parameters of the downtown Toronto

Camera	*Focal length*	*Flying height above Ground*	*Forward overlap*	*Side lap*	*GSD*	*Spectral bands*	*Radiometric resolution*
UltraCam D	101.4mm	1,600m	60%	30%	15cm	R-G-B	8 bit



(a) (b)

Figure 3.4 (a) ALS strips and (b) image configuration for the downtown Toronto dataset

This data contains representative scene characteristics of a modern mega city in North America. This dataset is divided into two sub-datasets; Area 4 (58 buildings; 530m×600m) contains a mixture of low- and high-storey buildings with a wide variety of

rooftop structures; Area 5 (530m×600m) is distinguished by a complex cluster of high-rise buildings. Figure 3.5 shows two sub-datasets in the Downtown Toronto area.

(a) (b)

Figure 3.5 Two test sites in the downtown Toronto area: (a) Area 4 and (b) Area 5

3.3 Performance Evaluation Metrics

Performance evaluation is the process of analysing the performance of a building reconstruction algorithm by comparing its results to the reference models or the results produced by other algorithms. Existing research works suggested different performance variables and objectives to evaluate the quality of boundaries extracted from developed algorithms: by measuring shape similarity in matched contours (Veltkamp, 2001); geometric quality of building boundaries extracted (Song and Haithcoat, 2005; Rutzinger et al., 2009); and 3D building models reconstructed from remotely sensed data (Rottensteiner et al., 2014; Meidow and Schuster, 2005). However, those previous studies pointed out that there is no single optimal performance evaluation method, and evaluation

results, even over the outcome produced by the same algorithm, vary depending on which evaluation metrics are used. This is because a performance evaluation metric is designed for accommodating specific performance characteristics to be assessed, which are subjective to different applications. Therefore, more careful selection of performance evaluation metrics should be taken into account. In this chapter, we will discuss characteristics of existing evaluation methods suggested by previous research works on building reconstruction (section 3.3.1), and propose a new metric which is appropriate for our research purpose of continuous model reconstruction (section 3.3.2).

3.3.1 Existing Evaluation Methods

Existing evaluation methods for the assessment of building models can be roughly divided into two categories; 1) error rates measured based on confusion matrix and 2) shape similarity measuring methods. The former measures the completeness, correctness, and quality to assess the overlapping quality between algorithm results and references, while the latter calculates geometric accuracy and shape similarity between matched model boundaries. The following sections review the two types of evaluation methods in detail.

3.3.1.1 Evaluation Using Confusion Matrix

A confusion matrix, also known as an error metric, has been often used for assessing the performance of an algorithm, typically spatial object detection, and supervised learning. As a quality measure for object reconstruction algorithms, each column of the matrix represents the instances in a reconstructed object (a predicted class in supervised learning),

while each row represents the instances in a reference object (an actual class) or vice-versa. In this confusion matrix with two rows and two columns, we can compute the number of *False Positive (FP), False Negative (FN), True Positive (TP), and True Negative (TN)*. These four performance elements allow more detailed analysis than mere proportion of correct detection. An algorithmic outcome can be positive if an object or image space (evaluation space) is occupied by the reconstructed model or negative if an evaluation space is not occupied by reconstructed model. The algorithm outcome may or may not match the subject's actual status (reference model); the matched case for true, otherwise for false. In this context, we can summarize the definitions of the four performance elements:

- *True Positive*: Total areas of a reconstructed model correctly identified by a reference model;
- *False Positive*: Total areas of a reconstructed model incorrectly identified by a reference model;
- *True Negative*: Total areas of a missing model correctly identified by a reference model;
- *False Negative*: Total areas of a missing model incorrectly identified by a reference model.

Then, using the confusion matrix and the four performance elements, the quantitative values for completeness, correctness and quality criteria can be determined for the results of reconstructed building models (Rutzinger et al., 2009):

$$Comp = \frac{\|TP\|}{\|TP\|+\|FN\|}$$

$$Corr = \frac{\|TP\|}{\|TP\|+\|FP\|} \tag{3.1}$$

$$Quality = \frac{Comp \cdot Corr}{Comp+Corr-Comp \cdot Corr}$$

The completeness refers to the fraction of the reference model which was correctly denoted as "building" by extracted models. The correctness measures how well the extracted model matches the reference model. The quality is a combination metric of completeness and correctness. One question then arises how we can determine the quantitative values of *TP*, *FN*, and *FP* in Eq. (3.1). There are two ways to address this problem: 1) area-based methods, and 2) object-based (or count-based) methods. Area-based methods analyze overlapping areas between reference and extracted models as shown in Figure 3.6.

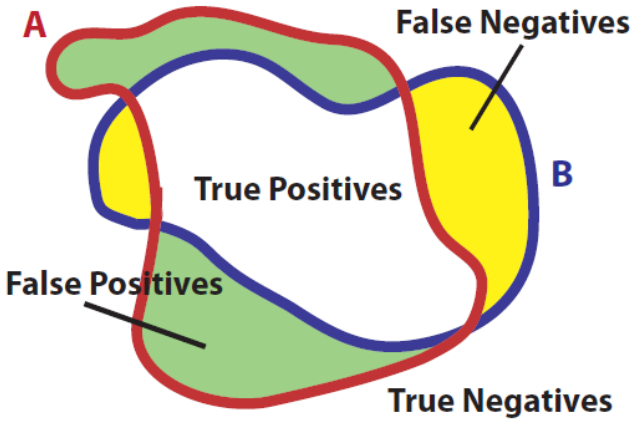


Figure 3.6 Area-based evaluation between extracted model (A) and reference (B) (Movahedi, 2015).

However, Rutzinger et al. (2009) pointed out that the area-based method can mislead the evaluation results, which are sensitive to the quality of building (or roof plane) boundaries extracted. The accuracy of building boundaries are influenced not only by poor performance of an algorithm used, but also many other error sources including data resolution, sensor calibration errors, registration errors, differences in semantic definition and accuracy of reference data (Rutzinger et al., 2009; Foody, 2002). There is no comprehensive way to separate errors caused by the algorithm from the other non-algorithmic errors in order to understand a pure characteristic of an algorithm's performance only.

On the other hand, an object-based evaluation method determines the quantitative values of *TP*, *FN*, and *FP* by counting the number of building objects belonging to each property. Overlapping areas can be used to determine whether two objects in reference and extracted results are correctly or incorrectly identified. This object-based identification is facilitated by introducing a specific threshold over an overlapping area. A fundamental underlying assumption behind this is that the value of overlapping threshold reflects the total influence of errors caused by non-algorithmic mechanisms to the accuracy of extracted object boundaries. However, determining the overlapping threshold is not a trivial task and an ad-hoc value is given. Thus, the object-based evaluation results are sensitive to this hard-constraint; the smaller the threshold used, the more overoptimistic the result may be. In many applications, a certain value between 50% and 70% has been typically used as a threshold value. Instead of using a single threshold, Rutzinger et al.

(2009) proposed investigating a range of system performance of building detection that was evaluated using multi-range thresholds.

The other issue, which should be addressed in object-based evaluation, is how to deal with topology inconsistency between reference models and extracted models. The topology inconsistency mainly occurs due to incomplete segmentation, and different representations of model structure. Thus, topology relations between two models in reference and extracted results may be represented by *1:m*, *n:1*, or *n:m* correspondences. This causes ambiguity in identifying corresponding objects to be compared. In order to address this problem, Rutzinger et al. (2009) proposed a topological clarification method to evaluate the quality of building detection. The method changed building label image by a splitting and merging process where each building in one dataset has only zero or one corresponding buildings in the other dataset. This process reduces the correspondence ambiguity. Then, the completeness, correctness, and quality were computed based on the changed building label image.

Although methods based on the confusion matrix are typically used to assess overlapping quality between two models in reference and extracted results, Song and Haithcoat (2005) recommended to use the confusion matrix-based evaluation method with geometric accuracy such as root mean square error (RMSE) and shape similarity indices.

3.3.1.2 Shape-based Evaluation

Shape-based evaluations measure how two shapes resemble each other. The shape-based evaluations can be used to assess qualities of building models in terms of geometric

accuracy and shape similarity because building models are represented by a combination of planes of various shapes. In this section, we introduce several shape-based evaluation methods and their properties which are appropriate to assess qualities of building models.

Minkowski distance (L_p distance)

One of the most popular indices to measure a degree of shape similarity between two contours is Minkowski distance or L_p distance. Given two points $X = (x_1, \dots, x_n)$ and $Y = (y_1, \dots, y_n)$, the Minkowski distance of order p is defined as:

$$L_p(x, y) = (\sum_{i=0}^{n} |x_i - y_i|^p)^{1/p} \tag{3.2}$$

For $p \geq 1$, Minkowski distance is a metric to quantify a physical space displaced between given vectors. Manhattan distance and Euclidean distance can be considered as special cases of Minkowski distance: p=1 for Manhattan distance, and p=2 for Euclidean distance in Eq. (3.2). A well-known root mean square error (RMSE) uses Euclidean distance to measure the geometric accuracy of point vectors (test vertices) produced by an algorithm where reference point vectors (reference vertices) are used as check points. A shape similarity between test and reference vectors can be estimated by measuring RMSE. However, uniquely identifying a one-to-one correspondence between given two vectors is a challenging problem. This is because different mechanisms are applied for producing respective point vectors and thus resulting physical properties (e.g., numbers of vertices, curvatures, length, etc) differ from each other. Rottensteiner et al. (2014) suggested a

nearest neighborhood method to identify vertex correspondences between roof models generated by an algorithm and manually digitized reference models for evaluating the algorithmic performance (e.g., RMSE). They employed a proximity threshold to identify those corresponding point sets from two model vectors. However, a pre-fixed value for the proximity criterion was employed without considering local variations of point extractions that exist in test and reference vectors. This non-adoptive matching process may lead to errors in determining correspondences at a certain extent which are not ignorable.

Simple shape descriptors

As an alternative to the correspondence-based measurement, one can evaluate the shape similarity by measuring shape descriptors such as area, perimeter, circularity (perimeter^2/area), eccentricity (length of major axis/length of miner axis), and major axis orientation. These shape descriptors can be measured over given entire vectors; measuring their similarities between two vector spaces does not require determining individual vertex correspondence locally and provides a shape characteristics of respective vectors at global scale (object level). However, the shape descriptors are not robust enough to discriminate a subtle difference of vectors, especially if given vectors to be evaluated belong to a similar object category, and shows its weakness to recognizing intra-variations of shapes (Zhang et al., 2004).

Hausdorff Distance

Hausdorff distance is used as a method for determining the resemblance of one point set to another based on a max-min distance (Huttenlocher et al., 1993). Let $A = \{a_1, \dots, a_p\}$ and $B = \{b_1, \dots, b_q\}$ be two finite point sets, a directed Hausdorff distance $h(A, B)$ is defined as the largest distance from any point in *A*, to the closest point in *B* as follows:

$$h(A, B) = sup_{a \in A} inf_{b \in B} d(a, b) \tag{3.3}$$

where *sup* and *inf* represent the supremum and the infimum, respectively; *d* is an underlying norm on the points of *A* and *B* (e.g., L_2 or Euclidean distance). The Hausdorff distance is defined as:

$$H(A, B) = max\left(h(A, B), h(B, A)\right) \tag{3.4}$$

The advantage of the Hausdorff distance is that no correspondence between two shapes to be compared is needed. However, Hausdorff distance is sensitive to noise because a single outlier may determines the distance value. In evaluation of building models, the properties are useful to assess the quality of building models. In contrast to RMSE, which assesses the average difference between two models, the Hausdorff distance can measure the maximum shape difference caused by over-simplification and under-simplification without any pre-defined value for the proximity criterion. Figure 3.7 shows a concept of the Hausdorff Distance.

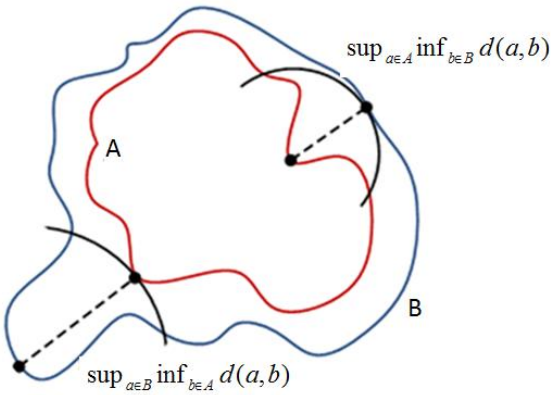


Figure 3.7 Hausdorff distance

Turning Function Distance

A turning function $\Theta_A(s)$ measures the angle of the counter-clockwise tangent as a function of the arc lengths in order to represent a shape *A* (Arkin et al., 1991). It begins in a certain point (reference point) on *A*'s boundary, and firstly measures the counter-clockwise angle between the tangent at the point and the horizontal axis (x-axis). $\Theta_A(s)$ keeps track of the turning that takes place, increasing with left-hand turns and decreasing with right-hand turns as shown in Figure 3.8. The L_p distance between $\Theta_A(s)$ and $\Theta_B(s)$ is applied to measure shape similarity as follows:

$$\|\Theta_A - \Theta_B\|_p = (\int|\Theta_A(s) - \Theta_B(s)|^P ds)^{1/p} \tag{3.5}$$

where $\|\cdot\|_p$ denotes the L_p norm. A turning function distance is invariant under scale, rotation, and translation. Also, the distance can measure a resemblance between two shapes at global scale, and any vertex correspondences do not need to be established.

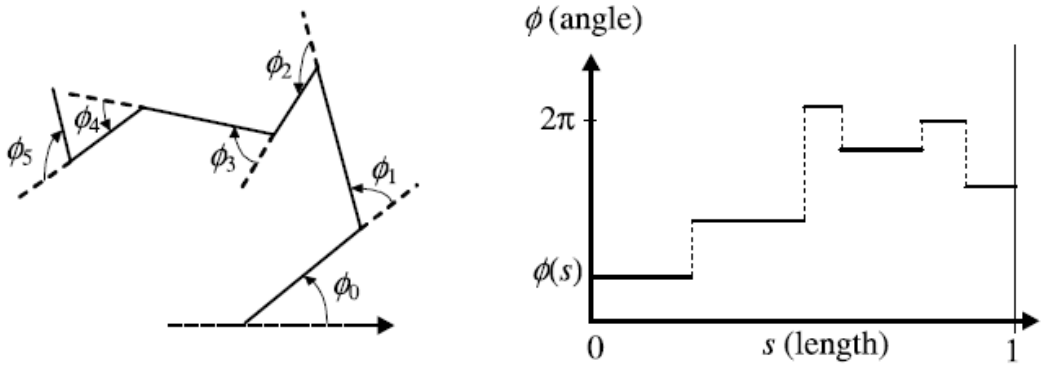


Figure 3.8 Turning function distance (Cakmakov and Celakoska, 2004)

3.3.2 Proposed Performance Evaluation Metrics

The ISPRS benchmark project on urban classification and 3D building modeling led by ISPRS WGIII/4 provides evaluation metrics in order to estimate the results obtained from the latest state-of-the-art algorithms for building detection, and 3D building reconstruction (Rottensteiner et al., 2014). The ISPRS evaluation metrics were designed for measuring the performance characteristics of individual algorithms by combining multiple evaluation indices including confusion matrix (area-based and object-based), topological analysis among roof planes, and geometric accuracy (RMSE). Thus, the ISPRS metrics allow us to evaluate the algorithm performance with many different aspects, rather than relying on a single measure. A summary of the ISPRS evaluation metrics in presented in Table 3.5.

Table 3.5 Evaluation indices used in Rottensteiner et al. (2014)

Evaluation Index	*Description*	*Object to be evaluated*
$Comp_{area}$, $Corr_{area}$, Q_{area}	Completeness, correctness, and quality determined on a per-area level.	• Building detection
$Comp_{obj}$, $Corr_{obj}$, Q_{obj}	Completeness, correctness, and quality determined on a per-building level or a per-roof-plane level.	• Building detection • Building reconstruction
$Comp_{50}$, $Corr_{50}$, Q_{50}	Completeness, correctness, and quality determined on a per-building level but only considering building larger than $50m^2$.	• Building detection
$Comp_{10}$, $Corr_{10}$, Q_{10}	Completeness, correctness, and quality determined on a per-roof-plane level but only considering roof planes larger than $10m^2$.	• Building reconstruction
$N_{1:M}$, $N_{M:1}$, $N_{N:M}$	Difference in the topologies of the extracted roof planes and the reference	• Building reconstruction
RMSXY	Geometrical errors in planimetry; only distances shorter than 3m are considered.	• Building detection • Building reconstruction
RMSZ	Geometrical error in height	• Building reconstruction

Although the ISPRS evaluation metrics provide one of the most extensive sets of indices used for measuring the performance of 3D roof modeling algorithms, they also have some limitations. Firstly, the ISPRS evaluation metrics focus on measuring a local similarity between references and resulting rooftop models produced by an algorithm by assessing overlapping areas or local geometric displacement between two models. However, these measures do not provide a holistic shape similarity such as the differences in main angle of a building object, Hausdorff distance and turning function distance

described in previous sections. As a consequence, the ISPRS metrics might have a tendency to be sensitive to per-roof segmentation accuracy or reconstructing accuracy of local vertices, missing the fact that global shape similarity is equally important to evaluate an algorithmic performance in 3D rooftop modeling. Secondly, the ISPRS evaluation metrics assess the geometric accuracy of rooftop models over the model's vertices only if their proximity to corresponding reference ones fall in a pre-specified error bound. Thus, by excluding model vertices beyond a given proximity threshold, resulting performance measures tend to be overoptimistic. The proximity threshold is determined by considering expected errors involved in rooftop models generated by an algorithm. However, it is difficult to predict this error tolerance in general in advance, and it is sensitive to locality. Lastly, the ISPRS evaluation matrix provides a mean to measure the accuracy of topological relations among adjacent roof planes produced by an algorithm against correspond reference models (*1:M*, *N:1*, and *N:M* relations). However, these ratios implicitly suggest error rates in roof plane generation, but do not provide an explicit understanding of topological errors produced by the algorithm; these indices represent planar segmentation errors, rather than topological errors. Thus, the ISPRS topological ratio indices might not correspond to the errors evaluated by our visual inspection.

In this study, we propose a set of performance measures evaluating the accuracy of rooftop models generated by our method presented in this thesis. For this purpose, a majority of the performance metrics were adopted from the ISPRS benchmark project (Table 3.5). However, in order to address limitations of the ISPRS performance metrics, we adopted additional measures, which include: 1) Hausdorff distance, 2) turning function

distance, and 3) main angle index (Table 3.6). As discussed previously, shape similarity measure is an important indicator evaluating the performance of rooftop model generation, which provides a global perspective in shape matching, and thus compensates the limitation of local similarity measures such as geometric accuracy (e.g., RMSE).

Hausdorff distance is a shape similarity measure between reference models and algorithmic models, which takes the maximum distance among the minimum distances measured between each vertex for two model datasets. Without introducing any threshold, total distance measured over entire shapes identifies a degree of shape similarity between two models matched. It can effectively assess how the reconstructed building model is over-simplified or under-simplified against its reference model. The turning function distance, as the second index of shape-based measures, represents a cumulative measure of the angles through which a polygonal curve turns. In contrast to Hausdorff distance measure (focusing on the measurement of over-simplification or under-simplification), turning function distance enables directly measuring similarity of turning patterns between reference and algorithmic models. However, as discussed in previous sections, applying the shape similarity measures is not a trivial task as it requires finding exact correspondences between reference and algorithmic models. Thus, we use a user-defined threshold for overlapping area to find correspondence between reference models and extracted models, and apply shape-based evaluation methods in two different stages: (1) for outer boundary and (2) roof planes with *1:1* correspondence.

In addition to Hausdorff distance and turning function distance, we assessed main orientation errors in building models generated by an algorithm (i.e., angle-based index).

This angle-based evaluation index measures the difference in main orientations of a building modelled in a reference dataset to the results produced by an algorithm. The main orientation of a building model is determined by analyzing the frequency of building lines for eight direction zones generated by the compass line filter (CLF) proposed by Sohn et al. (2008). A concept of CLF will be explained in Figure 4.7. Table 3.6 summarizes additional evaluation indices. Throughout this thesis, we use a set of performance metrics evaluating the performance of our building modeling methods by combining the indices addressed in Table 3.5 and Table 3.6.

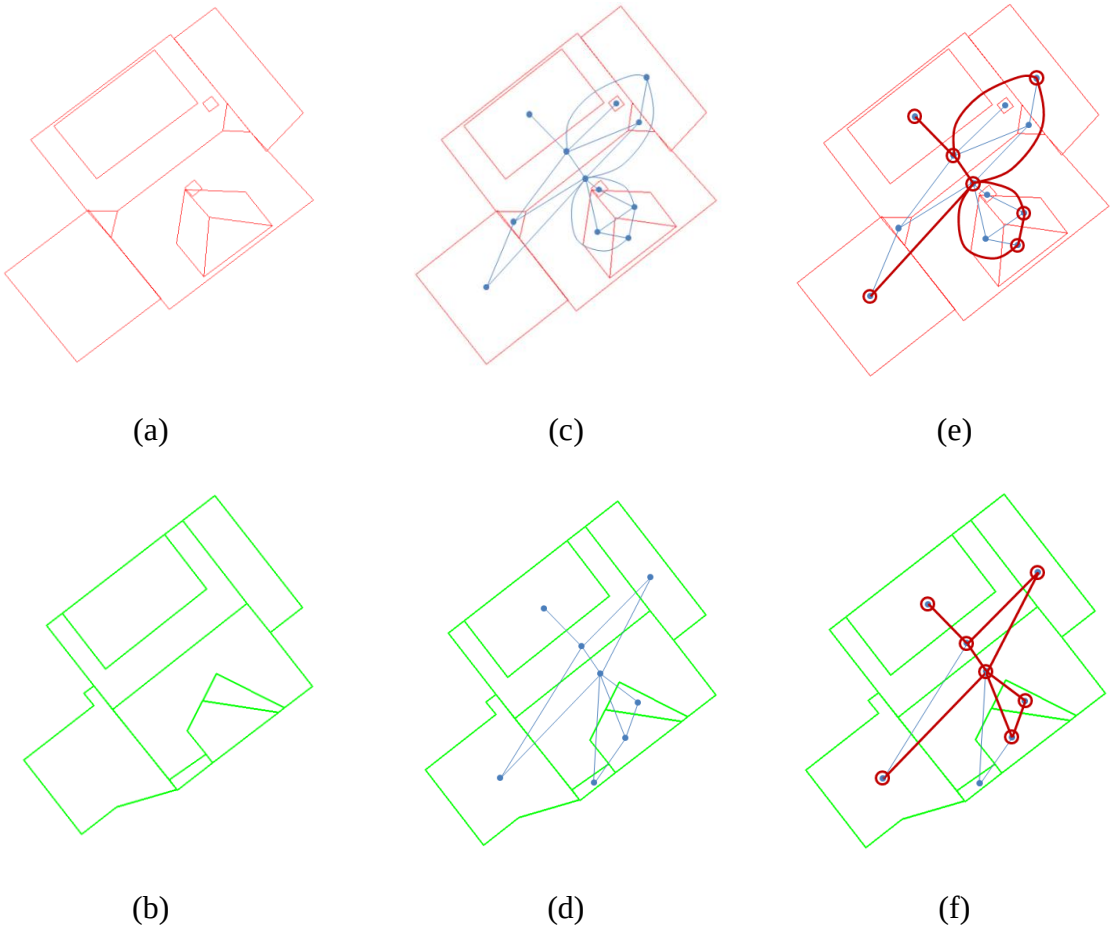
Table 3.6 Additional evaluation indices

Evaluation Index	*Description*	*Object to be evaluated*
Hausdorff Distance	Evaluation for partly deformed shape	• Building detection • Plane with *1:1* correspondence
Turning function Distance	Evaluation for entire shape similarity	• Building detection • Plane with *1:1* correspondence
Angle-based index	Difference in main angle of building model between reference and resulting rooftop models.	• Building detection • Building reconstruction

One remaining problem is how to explicitly measure topology inconsistency between the reference rooftop model and extracted rooftop model. We did not clearly solve the problem. However, we introduce a conceptual design for quantitatively measuring topology accuracy even though the method was not used to evaluate our proposed methods in this thesis. The method is based on the comparison of Roof Topology Graph (RTG) derived from the reference rooftop model and extracted rooftop model, and then assessing

topological accuracy by counting correctly matched edges. In RTG, a node represents a roof face, an edge represents the adjacency relationship of two roof faces (Oude Elberink and Vosselman, 2009). In this study, a graph edge is constructed if two roof faces share a common line (intersection line or step line). Figure 3.9(b) and (e) show RTGs constructed for a reference rooftop model and extracted rooftop model, respectively. We use the constructed RTGs for measuring topology accuracy between two building rooftop models. The process starts by finding correspondence between nodes (roof faces) by checking overlapping area where only *1:1*, *0:1* or *1:0* correspondences are allowed. If other correspondences between two rooftop models (*n:1*, *1:m*, or *n:m*) exist, roof faces, which have the maximum overlapping rate among possible node pairs, are considered to be matched, Once all corresponding nodes are determined, correctly matched edges can be identified by comparing node correspondences. For instance, Figure 3.9(c) and (f) shows matched nodes (red circle), and correctly matched edges (red line) in the reference rooftop model and extracted rooftop model, respectively. Then, a direct topology accuracy from the reference rooftop model to extracted rooftop model, $TA_{R\rightarrow M}$ (or from extracted rooftop model to reference rooftop model, $TA_{M\rightarrow R}$) can be estimated by calculating the number of matched edges over the number of total edges: $TA_{R\rightarrow M} = \frac{\#\ of\ matched\ edges\ in\ reference}{\#\ of\ edges\ in\ reference}$. In the example of Figure 3.9, $TA_{R\rightarrow M}$ and $TA_{M\rightarrow R}$ are 40% and 73%, respectively. The result indicates that 40% of topology relations in the reference rooftop model can be explained by topology relations of the extracted rooftop model while 73% in extracted rooftop model can be described by reference rooftop model. The example shows that topology accuracy can be explicitly assessed. Although the introduced RTG-based accuracy measurement is a

conceptual design level, we believe that the method can be extended to quantitatively assess topology inconsistency in the future.



(a) (c) (e)

(b) (d) (f)

Figure 3.9 RTG-based evaluation: (a) building rooftop model in reference, (d) extracted building rooftop model, (c) topology graph of (a), (d) topology graph of (b), (e) edges matched with (d) and (f) edges matched with (c)

3.4 Summary

In this chapter, we presented datasets and their characteristics, and evaluation metrics for assessing our results. The detailed descriptions of two datasets were given in the first part

of this chapter. The second part summarized existing evaluation methods in the literature by categorizing them into evaluations using confusion matrix and shape-based evaluation methods. Generally, there is no single optimal performance evaluation method, and individual existing evaluation methods assess specific performance characteristics to be assessed. Thus, based on the review of existing performance evaluation methods, we proposed evaluation metrics to test the performance of our proposed algorithms. The proposed evaluation metrics consists of existing evaluation methods, which focus on measuring a local similarity, and newly added shape-based and angle-based evaluation methods to assess different aspects of building models. Hausdorff distance and turning function distance were added to the evaluation metrics as shape-based indices. They can measure shape similarity at a global scale, and any correspondence between vertices derived from two models is not needed. Hausdorff distance is useful to measure over-simplification and under-simplification of building models while turning function distance measures entire shape resemblance of two models to be compared. Also, we added an angle-based index in order to measure the difference of the main orientation of building models. The proposed evaluation metrics were used to assess different quality aspects of 3D building models produced by our algorithms. In addition, we introduced a conceptual design to measure topology accuracy based on RTG even though the method was not used to evaluate our result. Future work is to extend the conceptual topology evaluation methods to explicitly assess topology inconsistency between two rooftop models.

Chapter 4

Implicit Regularization for Reconstructing 3D Building Rooftop Models Using LiDAR Data

In this chapter, we propose a data-driven modeling approach to reconstruct 3D rooftop models from airborne laser scanning (ALS) data. The focus of the developed method is to implicitly impose building regularity on 3D building rooftop models by introducing flexible regularity constraints. This study covers a full chain of 3D building modeling from low level processing to realistic 3D building rooftop modeling. In the element clustering step, building-labelled point clouds are clustered into homogeneous groups by applying height similarity and plane similarity. Based on segmented clusters, linear modeling cues including outer boundaries, intersection lines, and step lines are extracted. Topology elements among the modeling cues are recovered by the Binary Space Partitioning (BSP) technique. The regularity of the building rooftop model is achieved by an implicit regularization process in the framework of Minimum Description Length (MDL) combined with Hypothesize and Test (HAT). The parameters governing the MDL optimization are automatically estimated based on Min-Max optimization and Entropy-based weighting method. The performance of the proposed method is tested over two large-scale datasets using an evaluation metric discussed in Chapter 3. The results show that the proposed method can robustly produce accurate regularized 3D building rooftop models.

4.1 Introduction

A key problem domain that we address in this chapter is to reconstruct a 3D geometric model of building rooftop from remotely sensed data such as airborne laser point clouds. The representation that we follow for 3D rooftop models draws on ideas from geometric modeling used in Photogrammetry and Geographical Information Science (GIS). In this representation scheme, a 3D rooftop is modelled with either primitive geometric elements (i.e., points, lines, planes and objects), or primitive topological elements (i.e., vertices, edges, faces, and edge-groups (rings of edges on faces)). Typically, both primitive geometric and topological elements are used together for representing 3D rooftop models (e.g., CityGML and Esri ArcGIS's shapefile). CityGML is an open data model and XML-based format for the storage and exchange of virtual 3D city models (Kolbe et al., 2005).

In CityGML, 3D rooftop models can be differently represented according to the level-of-detail (LoD). A prismatic model of rooftop that is a height extrusion of a building footprint is defined as LoD 1 in CityGML, while LoD 2 requires a detailed representation of the primitive geometric and topological elements in a 3D rooftop model. An important aspect in GIS-driven 3D model representation is that the reconstructed model elements should correspond to semantically meaningful spatial entities used in architecture, civil and urban planning: for instance, the reconstructed geometric elements represent roof lines (ridges, eaves), roof planes (gables, hips), vents, windows, doors, wall columns, chimneys, etc. Thus, a photo-realistic reconstructed rooftop model can be used for assisting human decisions on but not limited to asset management, renovation planning, energy consumption, evacuation planning, etc. As discussed in Rottensteiner et al. (2014), a city-

scale building model will provide an important mean to manage urban infrastructure more effectively and safely for addressing critical issues related to rapid urbanization. In this thesis, we aim to reconstruct LoD 2 models of the rooftops from remotely sensed data.

Traditionally, 3D rooftop models are derived through interaction with a user using soft photogrammetric tools (e.g., multiple-view plotting or mono-plotting technology). This labour-intensive model generation process is tedious and time-consuming, which is not suitable for reconstructing rooftop models at city-scale. As an alternative method, great research efforts have been made for developing a machine-intelligent algorithm to reconstruct photo-realistic rooftop models in a fully-automated manner for the last two decades (Haala and Kada, 2010). Recently, airborne light detection and ranging (LiDAR) scanners became one of the primary data acquisition tools, which enable rapid capturing of targeted environments in 3D with high density and accuracy. Due to these advantages, state-of-the-art technologies for automatically reconstructing 3D rooftop models using airborne LiDAR data have been proposed by many researchers (Haala and Kada, 2010; Musialski et al., 2012; Wang, 2013; Rottensteiner et al., 2014; Tomljenovic et al., 2015). However, only limited success in a controlled environment has been reported, and the success of developing an error-free rooftop modeling algorithm is not achieved yet (Rottensteiner et al., 2014).

In general, 3D rooftop models are derived automatically from 3D LiDAR point clouds by (1) extracting the primitive geometric elements, namely "modeling cues" and (2) recovering the primitive topological elements among the modeling cues. A critical problem

to hinder the automation of 3D rooftop model generation is that many portions of the object (rooftop) are unknown, and recovered with errors caused by the following reasons:

- Irregular point distribution: Despite the advantages of acquiring highly accurate and dense 3D point clouds over rooftops by airborne LiDAR, the sensor also has its limitations. Airborne LiDAR transmits a packet of collimated laser beams through an electro-optical scanner, and computes a location of scatter, which surface is reflected from the transmitted laser energy, by measuring a range between the transmitter and scatter with known position and orientation of the laser scanner. The size of the beam footprint and space between adjacent laser points on the ground are determined by the flying height of the airborne platform and scanning frequency. In addition, the weak energy reflectance due to absorption and ill-posed surface angle against scanning pose, where the peak is below a pre-defined threshold, are discarded. Thus, all these system variables produce an irregular distribution of laser point clouds over the targeted object surface. As a consequence, the modeling cues are often generated with errors, or are fragmented, or completely missing. These errors have a negative impact on the derivation of the topological elements, and thus the accuracy of rooftop model generation.
- Occlusions: Like other sensors, airborne LiDAR also suffers from difficulties in capturing a complete view of objects due to occlusions. A disadvantageous viewing angle between the laser beam direction and object pose may hinder the illumination of laser beams on certain object surfaces, where no laser points are generated. In theory, airborne LiDAR has an ability to penetrate foliage; however, the amount of

returned laser energy varies depending on tree species, their maturity, seasonal effect and relative viewing angle between the laser beam and the leaf surface angle. A weak reflected energy will be neglected and not be able to produce any laser points over certain areas of roofs where trees grows nearby. These negative effects cause errors in recovering the primitive topological elements for reconstructing the rooftop model.

- Unreliable data analysis: A few of the laser point cloud analytics are applied to detecting building objects, classifying non-roof-related objects (e.g., trees, roof superstructures, etc.), segmenting roof planar patches, extracting corners and line primitives, and other algorithms related to recovering the primitive topological elements (e.g., boundary tracing, edge-linking, etc.). The performance of these algorithms varies depending on data resolution, scene complexity and noise; they may produce some errors, which has a negative effect on recovering both modeling cues and topological elements.

As discussed previously, the aforementioned factors lead to errors in recovering the modeling cues sufficiently well for generating an error-free rooftop model. Typically, knowledge of a rooftop object of interest (e.g., roof type, structure, numbers of roof planes, etc.) is unknown. Thus, recovering all the primitive topological elements accurately with an error-free geometric model is a very challenging vision task. To address this issue, many researchers have introduced some modeling constraints to compensate the limitations of erroneous modeling cues (Vosselman, 1999; Verma et al., 2006; Sampath

and Shan, 2007; Huang et al., 2013). These constraints are used as a prior knowledge on targeted rooftop structures: (1) for constraining the modeling cues to conform with Gestalt law (i.e., parallelism, symmetry, and orthogonality), and linking fragmented modeling cues in the frame of perceptual grouping, and (2) by determining optimal parametric rooftop model fit into part of rooftop objects through a trial-and-error of model section from a given primitive model database. We refer these modeling constraints as an "explicit regularity" imposed on rooftop shape as the definition of regularity is fully and clearly described. However, as discussed in Chapter 2, only a few of the explicit regularity terms can be applicable, and the shapes of rooftops in reality appear too complex to be reconstructed with those limited constraints.

In this thesis, we focus on the data-driven modeling approach to reconstruct 3D rooftop models from airborne LiDAR data by introducing flexible regularity constraints that can be adjusted to given objects in the recovery of modeling cues and topological elements. The regularity terms that are used in this study represent a regular pattern of the line orientations, and the linkage between adjacent lines. In contrast to the term of "explicit regularity", we refer it as an "implicit regularity" because its pattern is not directly expressed, but found with given data and object (rooftop). This implicit regularity is used as a constraint for changing the geometric properties of the modeling cues and topological relations among adjacent modeling cues to conform with a regular pattern found in the given data. This data-adaptive regularity (or regularization process) allows us to reconstruct more complex rooftop models.

In this chapter, we describe a pipeline of 3D rooftop model reconstruction from airborne LiDAR data. First, to gain some computational efficiency, we decompose a rooftop object into a set of homogeneous point clouds based on height similarity and plane similarity, from which the modeling cues of line and plane primitives are extracted. Secondly, the topological elements among the modeling cues are recovered by iteratively partitioning and merging over a given point space with line primitives extracted at a global scale using the Binary Space Partitioning (BSP) technique. Thirdly, errors in the modeling cues and topological elements are implicitly regularized by removing erroneous vertices or rectifying the geometric properties to conform with globally derived regularity. This implicit regularization process is implemented in the framework of Minimum Description Length (MDL) combined with Hypothesize and Test (HAT). The parameters governing the MDL optimization are automatically estimated based on Min-Max optimization and Entropy-based weighting method. The proposed parameter estimators provide optimal weight values that adapt according to building properties such as; size, shape, and the number of boundary points. The proposed pipeline of rooftop model generation was developed based on previous works reported in Sohn et al. (2012) and Jwa (2013). We extended these two works by proposing data-adaptive parameter estimation, conducting an extensive performance evaluation and engineering works to implement a computationally efficient modeling pipeline. The performance of the proposed method is evaluated using ISPRS benchmark data, which was also successfully reported by Rottensteiner et al. (2014).

4.2 3D Building Rooftop Reconstruction

Figure 4.1 shows the overall workflow implemented for generating 3D building rooftop models from airborne LiDAR point clouds, where individual buildings are detected. The method consists of three main parts: 1) modeling cue extraction, 2) topology element reconstruction, and 3) regularization. In the modeling cue extraction, roof element clusters, lines (intersection and step lines), and outer-boundaries are extracted from a set of laser point clouds labelled as single building objects (i.e., building labelled points) (section 4.2.1). Then, the topology relations among the modeling cues are established by BSP (section 4.2.2). Finally, an implicit regularization process is applied to outer-building boundaries and rooftop polygons. The regularization process is based on the framework of MDL in combination with HAT optimization (section 4.3). Note that the regularization process is conducted twice; once for regularizing building outer-boundaries which represent LOD1 models, and then for rooftop models which represent LOD2 models. Two types of weight parameters in the MDL-based objective function are automatically determined by Min-Max optimization and Entropy-based parameter estimation method, respectively (section 4.4).

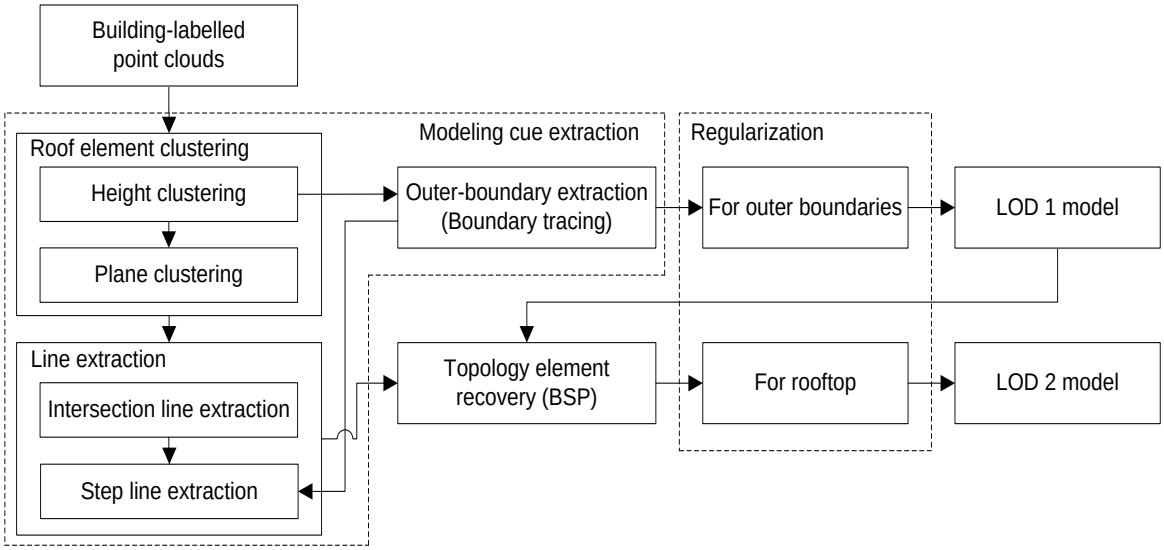


Figure 4.1 The overall workflow developed for reconstructing 3D rooftop models from airborne LiDAR data

4.2.1 Modeling Cue Extraction

The first step towards generating 3D building models using LiDAR data is to gather the evidence of building structures (i.e., primitive geometric elements). Planes and lines are recognized as the most important evidence to interpret building structures due to the fact that 3D building rooftop models can be mainly represented by planar roof faces and edges. The two different modeling cues (planar and linear modeling cues) have different properties and can be separately extracted from LiDAR points. In section 4.2.1.1, building points are sequentially segmented into homogeneous clusters, first based on height similarity and then based on plane similarity. In section 4.2.1.2, linear modeling cues are extracted using boundary points of the homogeneous clusters.

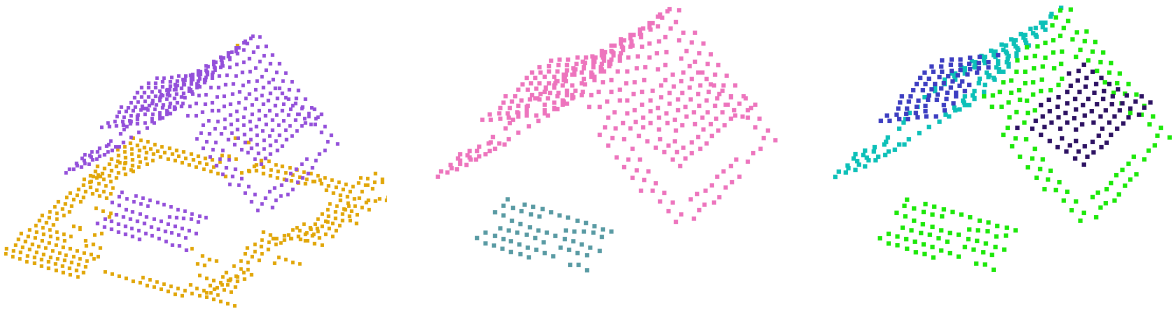
4.2.1.1 Roof Element Clustering

Roof element clustering segments building-labelled points into homogeneous rooftop regions with a hierarchical structure. A building rooftop in an urban area is a combination of multiple stories, each of which consists of various shapes of flat and sloped planes. Directly extracting homogeneous regions from entire building points may result in difficulties due to a high degree of shape complexity. In order to reduce the complexity, the building-labelled points are decomposed into homogeneous clusters by sequentially applying height similarity and plane similarity in order.

In the height clustering step, the rooftop region $R = \{p_i | i = 1, 2, \dots, n\}$ with n numbers of building-labelled points is divided into m height clusters $R = \{S_1, S_2, \dots, S_m\}$. Height similarity at each point is measured over its adjacent neighboring points in Triangulated Irregular Network (TIN). A point with the maximum height is first selected as a seed point, and then a conventional region growing algorithm is applied to add neighbor points to a corresponding height cluster with a certain threshold (δ_h=1m). This process is repeated until all building rooftop points are assigned to one of the height clusters. As a result, the height clusters satisfy the property $R = \bigcup_{i=1}^{M} S_i$, $S_i \cap S_j = \{\}$, $\forall i \neq j$. Note that each height cluster consists of one or more different roof planes.

In the plane clustering step, each height cluster is decomposed into k plane clusters $\Pi = \{\pi_1, \pi_2, \dots, \pi_k\}$ based on a plane similarity criterion. The well-known random sample consensus (RANSAC) algorithm is adopted to obtain reliable plane clusters as suggested in previous studies (Ameri and Fritsch, 2000; Tarsha-Kurdi et al., 2008). The process starts by randomly selecting three points as seed points to generate an initial plane. After a

certain period of random sampling, a plane, which has the maximum number of inliers with a user defined tolerance distance ζ (ζ = 0.1m) from the estimated plane, is selected as a best plane. Points, which are assigned in the previous iteration, are excluded in the next step. The process continues until all points of the height cluster are assigned into certain plane clusters. The results of plane clustering, particularly over the intersecting region between two planes, are negatively affected by plane clustering order due to a so-called winner-take-all effect (Sohn et al., 2008); the first clustered plane has a tendency of over-segmentation compared to the next clustered ones. Thus, segmentation errors may occur in points, which are close to intersecting regions of planes. To avoid this issue, a post-validation process is introduced to refine those segmentation errors. The post-validation process reassigns points nearby plane boundaries by comparing distances between them and their adjacent planes. A final clustering decision is made by selecting the most optimal plane which shows the minimum proximity from the points of interest among adjacent plane candidates. Figure 4.2(b) and (c) show examples of height clusters and plane clusters, respectively, where different colors represent different clusters.



(a) (b) (c)

Figure 4.2 Roof element clustering: (a) building-labelled points (purple), (b) height clustering (pink and green), and (c) plane clustering (black, pink, blue and purple).

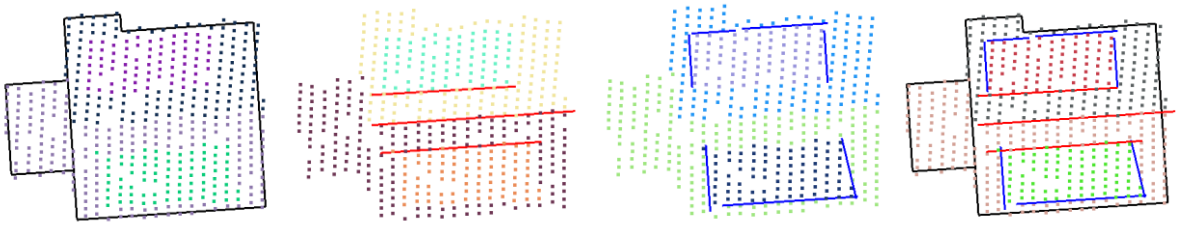
4.2.1.2 Linear Modeling Cue Extraction

Once building-labelled points are segmented into homogeneous clusters with a hierarchical structure, linear modeling cues are extracted from the homogeneous clusters. We divide linear modeling cues into three different types in order to reduce the complexity in the modeling cue extraction process as follows: 1) outer boundaries of height clusters, 2) intersection lines, and 3) step lines within each height cluster.

In boundaries of height clusters, two adjacent planes have a large height discontinuity. Thus, outer boundaries of height clusters can be recognized as step lines. However, distinguishing between outer boundaries of height clusters and step lines within each height cluster can reduce ambiguity in the topology recovering process, which will be described in section 4.2.2. Also, outer boundaries of height clusters can serve to generate the LOD1 model. For these reasons, in this study, we separately extract outer boundaries of height clusters. The process starts by detecting boundary points of height clusters which share neighbour height clusters in a TIN structure. After selecting a starting boundary point, a next boundary point is determined by surveying neighbor boundary points, which are connected with the previous boundary point in TIN structure, and by selecting a boundary point which appears first in an anti-clockwise direction. The process continues until the boundary is closed. Then, the closed boundary is regularized by the MDL-based regularization method which will be described in section 4.3.

An intersection line candidate is extracted by two adjacent roof planes. Candidates are accepted as valid intersection lines if they separate the point sets of the planes and if a sufficient number of points is close to the generated lines.

For step lines, boundary points of plane clusters, which do not belong to outer boundaries or intersection lines, are considered as candidate points for step lines. Given a sequence $D = \{c_1, c_2, \dots, c_l\}$ of l candidate points, step lines are extracted in a similar way to the Douglas-Peucker (DP) algorithm. The process starts with a straight line ($\overline{c_1 c_l}$) connecting the first point and last point of the sequence and then recursively adding candidate points which have a distance larger than a user-defined tolerance (0.5m). Each segment of the line segments is considered a step line. Figure 4.3 gives examples of each type of linear modeling cues.



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

Figure 4.3 Modeling cues extraction: (a) outer boundaries (black), (b) intersection lines (red), (c) step lines (blue), and (d) combined modeling cues

4.2.2 BSP-based Topology Construction

Once all modeling cues are collected, topological relations among the modeling cues are constructed by the BSP technique. In computer science, the BSP is a hierarchical partitioning method for recursively subdividing a space into convex sets with hyperlines. Sohn et al. (2008) used the BSP to recover topological relations of 3D building rooftop planes. We adopt the method to reconstruct a topologically and geometrically correct 3D building rooftop model from incomplete modeling cues. The topology recovery process

consists of a partitioning step and plane merging step. In the partitioning step, a hierarchical binary tree is generated by dividing a parent region into two child regions with hyperlines. The outer boundary is used as the initial building model P_0 (Figure 4.4(a)). Both step and intersection lines $\{l_i\}$ extracted as discussed in section 4.2.1.2 are formulated as hyperlines $\{h_i\}$, each of which will be involved in the subdivision of the given building polygon, which is described by:

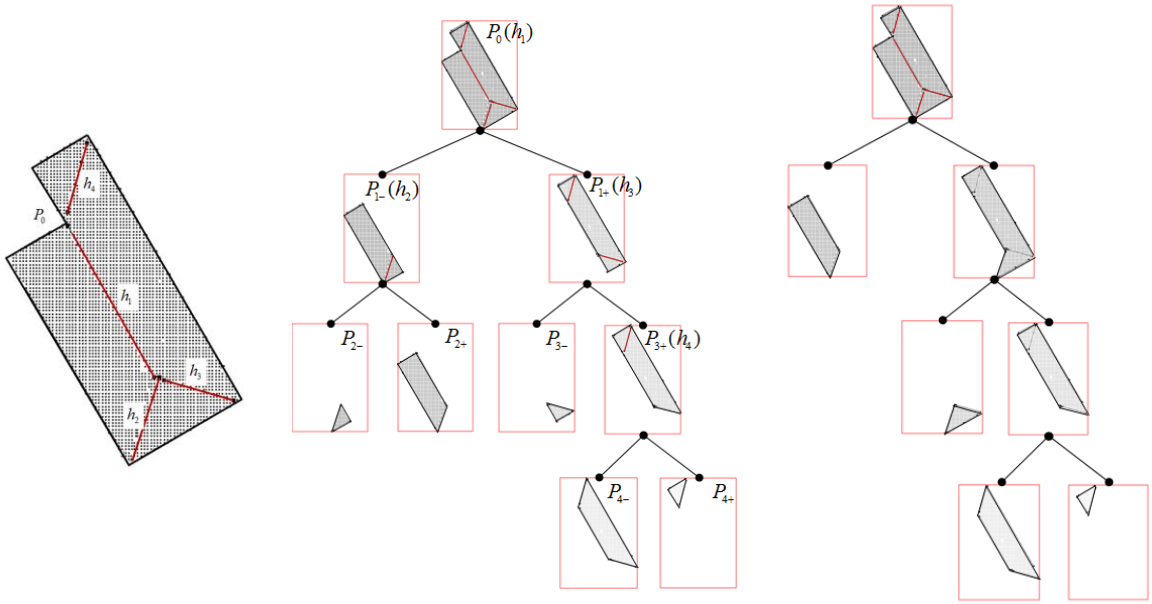
$$h_i(\rho_i, \theta_i) = \{(x, y) \in R | x cos\theta_i + y sin\theta_i - \rho_i = 0\} \tag{4.1}$$

where ρ_i and θ_i represents the distance of the origin from a line segment l_i, and the slope angle between the edge normal and x-axis, respectively. A hyperline h_1 is chosen to partition the parent region P_0 into the positive region P_{1+} and negative region P_{1-} which are expressed by:

$$\begin{aligned} P_{1+}(h_1{:}\,\rho_1, \theta_1) &= \{(x, y) \in R | x cos\theta_1 + y sin\theta_1 - \rho_1 > 0\} \\ P_{1-}(h_1{:}\,\rho_1, \theta_1) &= \{(x, y) \in R | x cos\theta_1 + y sin\theta_1 - \rho_1 < 0\} \end{aligned} \tag{4.2}$$

As shown in Figure 4.4(b), a parent node is divided into two leaf nodes. This process continues until no hyperline exists in all leaf nodes. The partitioning result will be different when a different sequence of line segments is employed. The selection of hyperlines is achieved by a hypothesis and test scheme with a partition scoring function which consists

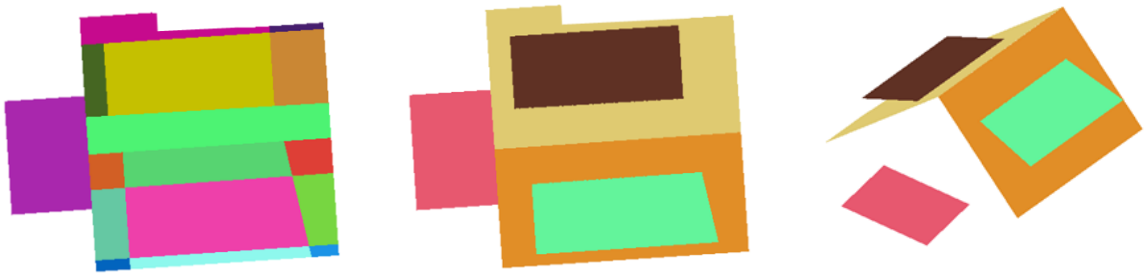
of plane homogeneity, geometric regularity and edge correspondence. Details about the partitioning optimum process can be found in Sohn et al. (2008).



(a) (b) (c)

Figure 4.4 Hierarchical generation of BSP tree: (a) initial region (P_0) and hyperlines (h_i), (b) partitioning process, and (c) merging process

In the plane merging step, starting from the node with the largest area in the generated BSP tree, a simple validation of normal vector compatibility is applied to its adjacent planes. The adjacent roof planes having similar normal vectors are merged. The merging process continues until no plane can be accepted by the co-planar similarity test (Figure 4.4(c)). Once all polygons are merged together, the 3D building rooftop model can be reconstructed by collecting final leaf nodes in the BSP tree. Figure 4.5 shows results of the partitioning step, merging step and the corresponding 3D rooftop model.



(a) (b) (c)

Figure 4.5 Binary Space Partitioning: (a) partitioning step, (b) merging step, and (c) reconstructed model

4.3 Implicit Regularization of Building Rooftop Models

As mentioned in section 4.1, recovering error-free 3D rooftop models from erroneous modeling cues is a challenging task. Geometric constraints such as parallelism, symmetry, and orthogonality can be explicitly used as a prior knowledge on rooftop structures to compensate the limitations of erroneous modeling cues. However, explicitly imposing the constraints has limitations on describing complex buildings that appear in reality. In this study, we propose an implicit regularization where regular patterns of building structures are not directly expressed, but implicitly imposed on reconstructed building models providing flexibility for describing more complex rooftop models. The proposed regularization process is conducted based on HAT optimization in MDL framework. Possible hypotheses are generated by incorporating regular patterns that are present in the given data. MDL is used as a criterion for selecting an optimal model out of the possible hypotheses. The MDL concept for model selection is introduced in section 4.3.1 while section 4.3.2 introduces a method for hypothesis generation.

4.3.1 MDL Principles and Rooftop Modeling

The MDL proposed by Rissanen (1978) is a method for inductive inference that provides a generic solution to the model selection problem (Grünwald, 2005). The MDL is based on the idea of transmitting data as a coded message, where the coding is based on some prearranged set of parametric statistical model. The full transmission has to include not only the encoded data values, but also the coded model parameter values (Davies, 2002). Thus, the MDL consists of model complexity and model closeness as follows:

$$DL = \lambda \mathcal{L}(D|H) + (1 - \lambda)\mathcal{L}(H) \tag{4.3}$$

where $\mathcal{L}(D|H)$ indicates a goodness-of-fit of observations D given a model H while $\mathcal{L}(H)$ represents how complex the model H is. λ is a weight parameter for balancing the model closeness and the model complexity. Assuming that an optimal model representing the data has the minimal description length, the model selection process allows a model H to be converged to the optimal model H^* as follows:

$$H^* = arg\, min_{H \in \Phi} \{\lambda \mathcal{L}(D|H) + (1 - \lambda)\mathcal{L}(H)\} \tag{4.4}$$

The first term in Eq. (4.3) is optimized for good data attachment to the corresponding model. With an assumption that an irregular distribution of data $D = \{x_1, \ldots, x_n\}$ with n measurements caused by random errors follows a Gaussian distribution $x \sim N(\mu, \sigma^2)$ with expectation μ and variance σ^2, its density function can be represented as

$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$. By using a statistical model of the data, the degree of fit between a model and data can be measured by $\mathcal{L}(D|\mu, \sigma^2)$, and then the term of model closeness can be rewritten in a logarithmic form as follows:

$$L(D|\mu, \sigma^2) = -log_2 P(D) = -\left(log_2 e^{-\frac{\sum(x-\mu)^2}{2\sigma^2}} + nlog_2 \frac{1}{\sigma\sqrt{2\pi}} \right)$$

$$= \frac{1}{2ln2} \sum \left(\frac{x-\mu}{\sigma} \right)^2 + nlog_2 \sigma + \frac{n}{2} log_2 2\pi \quad (4.5)$$

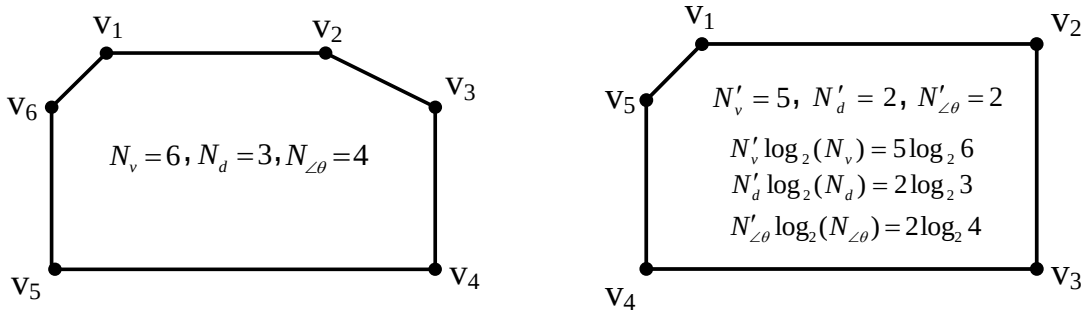
In Eq. (4.5), the last two terms can be ignored with an assumption that all the hypotheses have the same σ. Thus, the equation is simplified as follows:

$$\mathcal{L}(D|H) = \frac{\Omega}{2ln2} \quad (4.6)$$

where Ω is the weighted sum of the squared residuals between a model H and a set of observations D, that is $[D - H]^T [D - H]$ in matrix form.

The second term in Eq. (4.3) is designed to encode the model complexity. In this study, the model complexity is explained by three geometric factors: 1) the number of vertices N_v, 2) the number of identical line directions N_d, and 3) the inner angle transition $N_{\angle\theta}$. By using the three geometric factors, an optimal model is chosen if its polygon has a small number of vertices and a small number of the identical line directions, and if the inner angle transition is smoother or more orthogonal.

Suppose that N_v, N_d, and $N_{\angle\theta}$ are used for an initial model, while $N_v^{'}$, $N_d^{'}$, and $N_{\angle\theta}^{'}$ are used for a hypothetical model generated from the initial model (e.g., Figure 4.6). In order to measure the description length for the number of vertices, we start by deriving the probability that a vertex is randomly selected from a given model, $P(v) = \frac{1}{N_v}$. Then, it can be expressed in bits as $log_2(N_v)$. Since a hypothetic model generated by hypothesis generation process has $N_v^{'}$ vertices, its description length is $N_v^{'}log_2(N_v)$.



(a) (b)

Figure 4.6 An example of three geometric factors (a) for initial model (N_v, N_d, and $N_{\angle\theta}$) and (b) for a hypothetical model generated from the initial model (N_v', N_d', and $N_{\angle\theta}'$)

Similarly, the probability for the number of identical line directions N_d is $P(d) = \frac{1}{N_d}$ and can be expressed in bits as $log_2(N_d)$. By considering the required number of line directions $N_d^{'}$, the description length for identical line direction is measured by $N_d^{'}log_2(N_d)$. In order to define line directions, we adopt compass line filter (CLF) suggested by Sohn et al. (2008) as shown in Figure 4.7. The CLF is determined by the whole set of eight filtering lines with different slopes $\{\theta_i : i = 1, \dots, 8\}$ that is equally

separated in steps of 22.5º. The representative angle for each slope, θ_i^{REP}, is calculated by a weighted averaging of angles that takes the summed line length of each CLF slope into account.

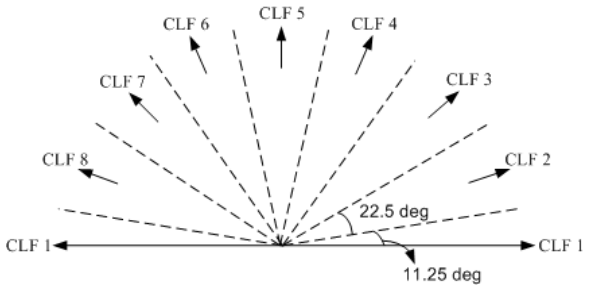


Figure 4.7 Compass line filter

Lastly, the description length for inner angle transition is measured by assigning a certain penalty value to quantized inner angles. As depicted in Eq. (4.7), the penalty values $\gamma_{i=0,1,2}$ are heuristically determined to have the minimum value of 0 (i.e., favour inner angle) if inner angle $\angle\theta$ is close to 90° or 180°, while the maximum value of 2 (i.e., un-favour inner angle) is assigned to very acute inner angles. This is because acute inner angle at two consecutive building vectors rarely appears in reality. Thus, the probability for $N_{\angle\theta}$ can be derived from an inner angle that is located in one of the quantized angles, $P(\angle\theta) = \frac{1}{N_{\angle\theta}}$, and expressed in bits as $log_2(N_{\angle\theta})$. In the optimal model, the cost imposed by penalty values is $\sum_{k=1}^{N_v'} \gamma_{i=0,1,2}$, and its description length is calculated by $N_{\angle\theta}^{'} log_2(N_{\angle\theta})$.

$$\gamma_{\mathrm{i=0,1,2}} = \begin{cases} 0 & \text{if } 78.75° \le \angle\theta \le 101.25° \text{ or } 168.75° \le \angle\theta \le 180° \\ 1 & \text{if } 11.25° < \angle\theta < 78.75° \text{ or } 101.25° < \angle\theta < 168.75° \\ 2 & \text{if } 0° < \angle\theta \le 11.25° \end{cases} \tag{4.7}$$

As a result, the description length for sub-terms of model complexity $\mathcal{L}(H)$ is obtained by the summation of three geometric factors as follows:

$$\mathcal{L}(H) = W_v N_v^{'} log_2 N_v + W_d N_d^{'} log_2 N_d + W_{\angle\theta} N_{\angle\theta}^{'} log_2 N_{\angle\theta} \quad (4.8)$$

where W_v, W_d, and $W_{\angle\theta}$ are weight values for each sub-factor in the model complexity.

4.3.2 Hypothesis Generation

The hypothesis generation process proposes a set of possible hypotheses under certain configurations of a rooftop model (or building boundary). Suppose a rooftop model consists of a polygon $\Pi_A = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$ and a polygon $\Pi_B = \{v_3, v_4, v_5, v_8, v_9, v_{10}\}$, where v_3, v_4 and v_5 are common vertices in both polygons (Figure 4.8(a)). A task is to generate possible hypotheses at a certain vertex considering a given configuration of rooftop model. The hypothesis generation process starts by defining an Anchor Point (AP), Floating Point (FP), and Guide Point (GP) and then by deriving a Floating Line (*FL*=[*AP*, *FP*]) and Guiding Line (*GL*=[*GP*, *FP*]). The role of AP is to define the origin of a line to be changed (FL). FP is a point to be moved while GP is used to generate GL which guides the movement of FP. Hypotheses are generated by moving FP along the GL with AP as an origin of FL. The orientation of FL is determined by representative angles of CLF which consists of eight directions as shown in Figure 4.7. There are different cases for hypothesis generation 1) depending on a relative direction of AP and FP (forward (clockwise) and backward (anti-clockwise)), 2) depending on whether

a vertex is removed (removal or non-removal), and 3) depending on whether FP is a common vertex in more than two adjacent polygons (common vertex or non-common vertex). For the reader's understanding, some cases are explained as follows:

- Case 1 (forward, non-removal, and non-common vertex): As shown in Figure 4.8(b), v_1 and v_2 are assigned as AP (blue circle) and FP (red point), respectively. Hypotheses are generated by moving FP along to the GL where red circles represent new possible positions of v_2.
- Case 2 (backward, non-removal, and non-common vertex): As shown in Figure 4.8(c), v_3 and v_2 are assigned as AP and FP, respectively. In contrast to case 1, FP is located in backward direction of AP.
- Case 3 (backward, removal, and non-common vertex): As shown in Figure 4.8(d), after removing v_2 (green point), v_3 and v_1 are assigned as AP and FP, respectively. New hypotheses are generated by moving v_1.
- Case 4 (forward, non-removal, common vertex): As shown in Figure 4.8(e), v_2 and v_3 are assigned as AP and FP, respectively. v_3 is a common vertex in Π_A and Π_{B}. Because the position of v_3 changes, shapes of both polygons are changed.
- Case 5 (forward, removal, common vertex): As shown in Figure 4.8(f), v_2 and v_4 are assigned as AP and FP, respectively. After v_3 is removed, v_4 is assigned as FP so that the position of v_4 is changed.

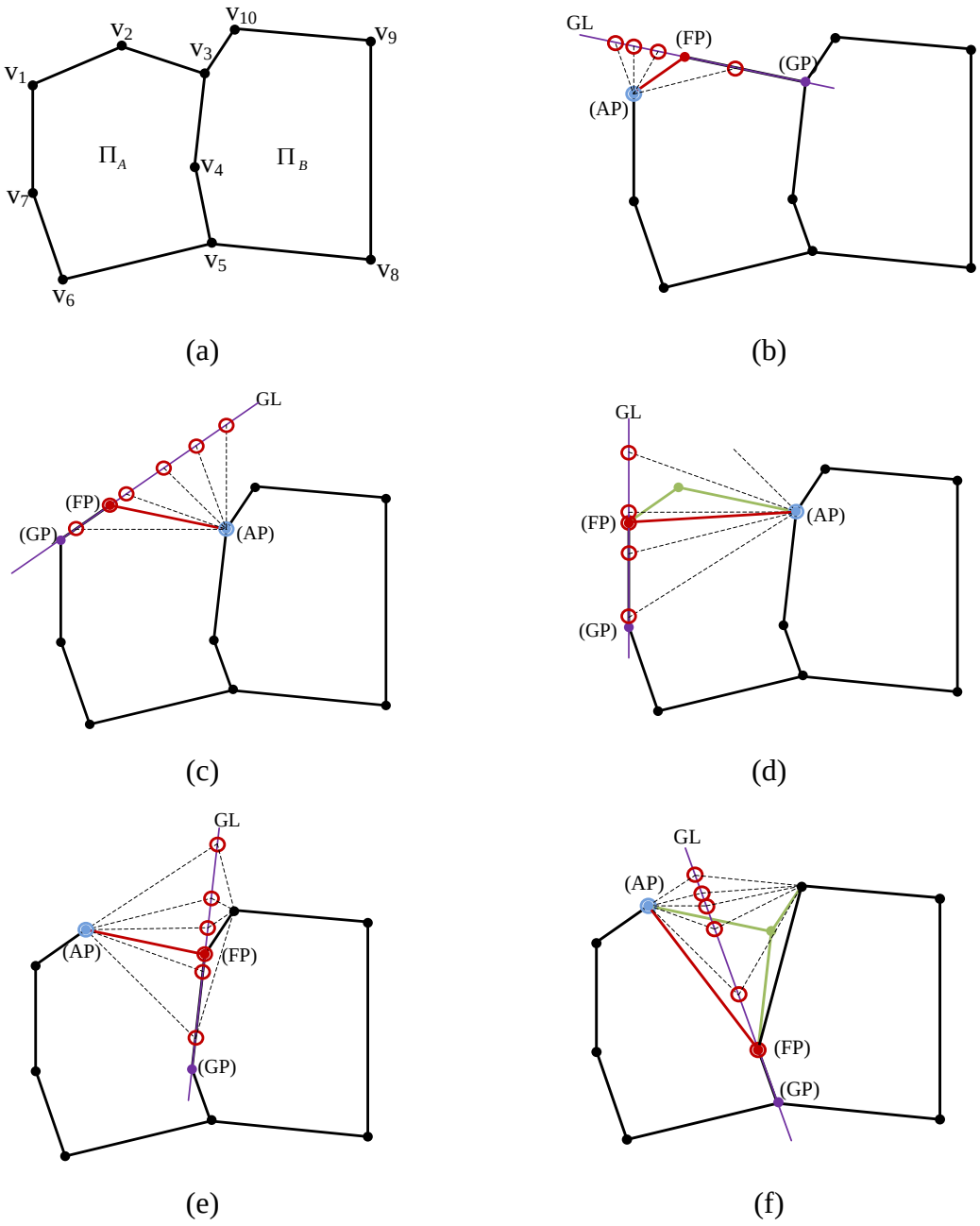


Figure 4.8 Examples of hypothesis generation (blue point: anchor point (*AP*), green point: removed point, purple point: guide point (*GP*), red point: floating point (*FP*), red circle: new possible positions of *FP*, red line: floating line (*FL*) and purple line: guide line (*GL*)): (a) initial configuration, (b) case 1, (c) case 2, (d) case 3, (e) case 4, and (f) case 5

4.4 Parameter Optimization

In the MDL-based objective function, two types of weight parameters are used to evaluate the relative importance of sub-terms. One is a weight parameter (λ) for balancing the model closeness and the model complexity in Eq. (4.3). The other is weight parameters ($W_v, W_d, W_{\angle\theta}$) for the three sub-terms in the complexity term in Eq. (4.8). In previous research (Sohn et al., 2012), these weight parameters were set as constant values, which were empirically determined, for all building models (λ=0.5 and $W_v = W_d = W_{\angle\theta} = 1$). However, buildings have different shapes and sizes in reality. Also, the density of LiDAR points varies on data acquisition settings and flight height. These properties, which vary on individual buildings, may cause unbalanced values in model closeness and model complexity. For instance, when building shape is very simple and the number of observations is significantly large, the closeness value is relatively larger than the complexity value. As a result, optimization process may be dominant to the variation of the model closeness. Thus, the weight parameters have to be appropriately tuned in an automated manner by individually considering the properties of each building. In order to automatically determine proper weight values, we propose two different weighting methods: 1) Min-Max weighting method (section 4.4.1), and 2) Entropy-based weighting method (section 4.4.2). The Min-Max weighting method is used to balance the model closeness and the model complexity while the Entropy-based weighting method is employed to determine the weight values for the three sub-terms in the complexity term.

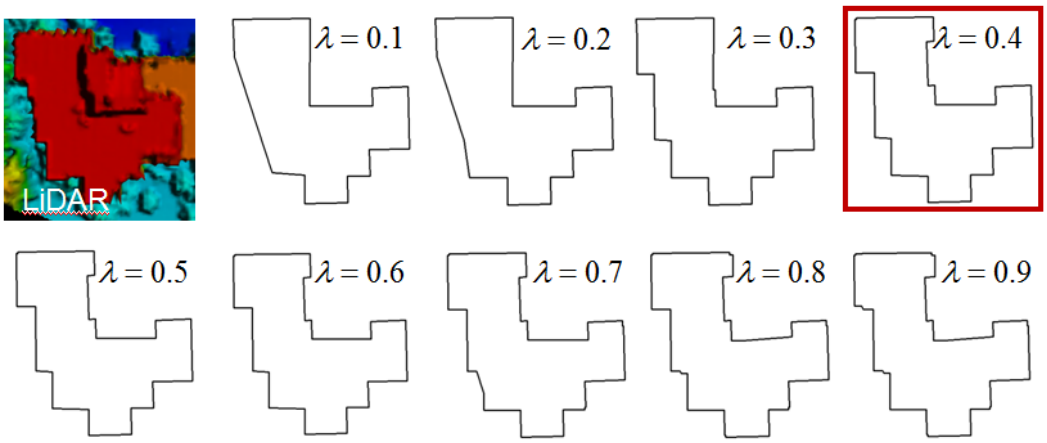
4.4.1 Min-Max Weighting Method

The proposed MDL-based objective function consists of two conflicting terms: the model closeness term $\mathcal{L}(D|H)$ and the model complexity term $\mathcal{L}(H)$ as shown in Eq. (4.3). λ is a weight parameter which affects modeling result. The smaller the value of λ, the simpler the optimal model is. In contrast, a larger value of λ emphasizes goodness-of-fit to data, causing under-simplified model (or over-fitting problem) (see Figure 4.9). In order to automatically estimate an appropriate weight value, we adopt Min-Max criterion (Gennert and Yuille, 1998), which minimizes possible loss while maximizing the gain. In this study, the Min-Max principle is closely related to minimizing the cost value *DL* for each λ and maximizing contributions from both of two terms, thereby finding the optimal $\lambda^* \in [0,1]$. For each term, this leads to avoid the best scenario where one of two terms dominates by having an excessively low or high value of λ. To achieve this goal, the "Min" operator first finds the optimal model for each λ using Eq. (4.4). Considering the boundary conditions, $\mathcal{L}(H)$ at $\lambda = 0$ and $\mathcal{L}(D|H)$ at $\lambda = 1$ corresponds to zero. Then, $\mathcal{L}(D|H)$ and $\mathcal{L}(H)$ are normalized using min-max normalization method, respectively, as follows:

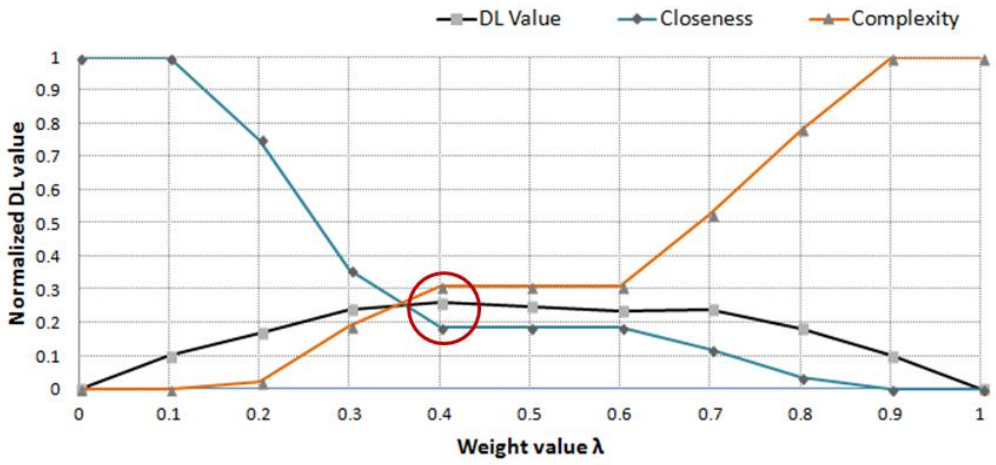
$$z_i = \frac{x_i - min\,(x)}{max\,(x) - min\,(x)} \tag{4.9}$$

where z_i is a normalized value for the i^{th} variable x_i; $min(x)$ and $max\,(x)$ are the minimum value and maximum value for variable *x*. After the total *DL* value is computed from normalized $\mathcal{L}(D|H)$ and $\mathcal{L}(H)$ for each λ, the "Max" operator derives an optimal weight value λ^* by selecting the worst scenario showing the maximum *DL*.

Figure 4.9 shows an example of the Min-Max weighting method. As shown in Figure 4.9(a), as λ is close to 0, a simple model is selected as the optimal model. As λ gets larger, the optimal model is more complex because the *DL* value is more affected by the closeness term. In this example, 0.4 is selected as the best λ because it produces the maximum *DL* value.



(a)



(b)

Figure 4.9 Min-Max based parameter determination: (a) optimal rooftop model for each λ value and (b) corresponding normalized *DL* values where 0.4 is selected as the best λ value.

4.4.2 Entropy-based Weighting Method

Prior to determining the weight parameter λ, we estimate the weight values of geometric parameters forming the complexity term $\mathcal{L}(H)$ in Eq. (4.8). The $\mathcal{L}(H)$ consists of three geometric terms including the number of vertices, the number of identical line directions and the inner angle transition.

In multi-attribute decision making, an entropy weighting method, which is one of the objective methods, is used to determine appropriate weights for attributes (Lotfi and Fallahnejad, 2010). The greater the value of the entropy corresponding to a special attribute, the smaller attribute's weight. We adopt the entropy weighting method to determine the relative importance of three geometric terms in Eq. (4.8). In information theory, entropy is understood as a measure of uncertainty about attributes drawn from data and can be normally characterized as follows:

$$E(X) = -\sum_{i=1}^{n} p(x_i)\, log_2\, p(x_i) \tag{4.10}$$

The basic formulation can be rewritten to calculate entropy in the existence of two possibilities *p* and *q=1-p* as follows:

$$E = -(p\, log_2\, p + q\, log_2\, q) \tag{4.11}$$

where *p* represents the event that a current hypothesized parameter set belongs to a class of optimal model parameters and q indicates the reverse situation of *p*. In this study, a probability for each term in Eq. (4.8) is derived by calculating a probability that each

geometric factor in a given model can converge to the optimal model. The optimal model in terms of model complexity, according to the definition of model complexity discussed in section 4.3.1, is represented by a rectangle where the number of vertices is four, the number of identical line directions is two, and all inner angles have no penalty. Thus, the probability that four vertices are randomly selected from N_v vertices is one over four combinations of N_v, $p(v) = 1/C_4^{N_v}$. Similarly, the probability that two identical line directions are selected from N_d identical line directions is one over two combinations of N_d, $p(d) = 1/C_2^{N_d}$. The probability of inner angle with no penalty in Eq. (4.7) is 3/16. Because all inner angles have no penalty to be optimal model, the probability for $N_{\angle\theta}$ is $p(\angle\theta) = N_v \times 3/16$. The estimated probabilities are converted into entropy using Eq. (4.11). A smaller weight value is assigned to a sub-term with larger uncertainty. Thus, weight parameters for three sub-terms are determined as suggested in previous studies (Zou et al., 2006; Lotfi and Fallahnejad, 2010):

$$W_v = \frac{1-E(v)}{3-\left(E(v)+E(d)+E(\angle\theta)\right)},\ W_d = \frac{1-E(d)}{3-\left(E(v)+E(d)+E(\angle\theta)\right)},\ W_{\angle\theta} = \frac{1-E(\angle\theta)}{3-(E(v)+E(d)+E(\angle\theta))} \quad (4.12)$$

4.5 Experimental Result

The performance of the proposed method was evaluated over the ISPRS benchmark datasets provided by the ISPRS WGIII/4 (Rottensteiner et al., 2014). The ISPRS benchmark datasets consist of three sub-regions (Area 1, Area 2, and Area 3) of the Vaihingen dataset, and two sub-regions (Area 4 and Area 5) of the Toronto dataset. The quality assessments for proposed algorithm were conducted based on the evaluation

metrics proposed in section 3.3. More detailed description of dataset characteristics, and proposed performance evaluation measure can be found in section 3.3.2.

4.5.1 Evaluations Using Confusion Matrix

Evaluations using confusion matrix were applied under three different conditions: (a) by applying area-based method for outer building boundary, and by applying object-based method (b) for all roof planes and (c) for roof planes with more than 10 m^2, respectively (Table 4.1).

In the area-based evaluation (Table 4.1(a)), our proposed rooftop reconstruction algorithm showed that the completeness, correctness, and quality of the reconstructed building models are 91.5%, 97.4%, and 89.2%, respectively. The results indicate that most of resulting building models were properly overlapped to the corresponding reference building models. The error rate for the completeness is larger than the error rate for the correctness. This is due to the fact that the boundary points extracted from irregularly distributed points were not reflected from the real building boundary. The erroneous observations cause boundary displacement which is generally positioned toward the inside of the building. As a result, a building model tends to be shrunken compared to the reference building model. This leads to the increase of *FNs* and the decrease of *TPs*, degenerating the completeness.

Table 4.1 Confusion matrix-based evaluations

Dataset	*Sub-Set*	*# Building*	*# Plane*	*(a) Area-based evaluation*			*Object-based evaluation*					
							(b) All roof planes			*(c) Roof planes ($10m^2$ area)*		
				Comp. (%)	*Corr. (%)*	*Quality (%)*	*Comp. (%)*	*Corr. (%)*	*Quality (%)*	*Comp. (%)*	*Corr. (%)*	*Quality (%)*
Vaihingen	*Area 1*	38	288	90.6	98.8	89.6	88.9	98.2	87.5	93.9	98.5	92.6
	Area 2	15	69	91.3	99.7	91.0	73.9	100	73.9	95.8	100	95.8
	Area 3	57	235	88.6	99.7	88.4	86.4	100	86.4	97.6	100	97.6
	Sub-total	110	592	90.2	99.4	89.7	83.1	99.4	82.6	95.8	99.5	95.3
Toronto	*Area 4*	58	967	93.7	96.9	90.9	82.1	94.8	78.6	92.4	96.2	89.2
	Area 5	38	640	93.1	92.0	86.1	66.1	87.1	60.2	89.5	89.6	81.1
	Sub-total	96	1607	93.4	94.5	88.5	74.1	91.0	69.4	91.0	92.9	85.2
Total		206	2199	91.5	97.4	89.2	79.5	96.0	77.3	93.8	96.9	91.3

In the object-based evaluation methods, a roof plane in one dataset was considered to be a true positive if a certain minimum percentage of its area (50% overlap) is covered by a roof plane in the other dataset. While the completeness, correctness, and quality for all roof planes are 79.5%, 96.0%, and 77.3%, respectively (Table 4.1(b)), the values are increased to 93.8%, 96.9%, and 91.3% if only large roof planes (>$10m^2$) are considered (Table 4.1(c)). The results indicate that small roof planes were not detected as well by our proposed method. This is mainly caused by the small number of points on small building roof planes which made it difficult to extract sufficient modeling cues for reconstructing rooftop models. Figure 4.10 clearly shows the effect of the size of roof plane. When only roof planes with an area smaller than $5m^2$, are considered, the completeness is considerably low for all five datasets. In particular, the completeness for Area 2 (Figure 4.10(b)) and Area 5 (Figure 4.10(e)) were 26.3% and 37.4%, respectively. We observed that buildings in the two regions have many small objects on their roofs which were represented in reference building rooftop models.

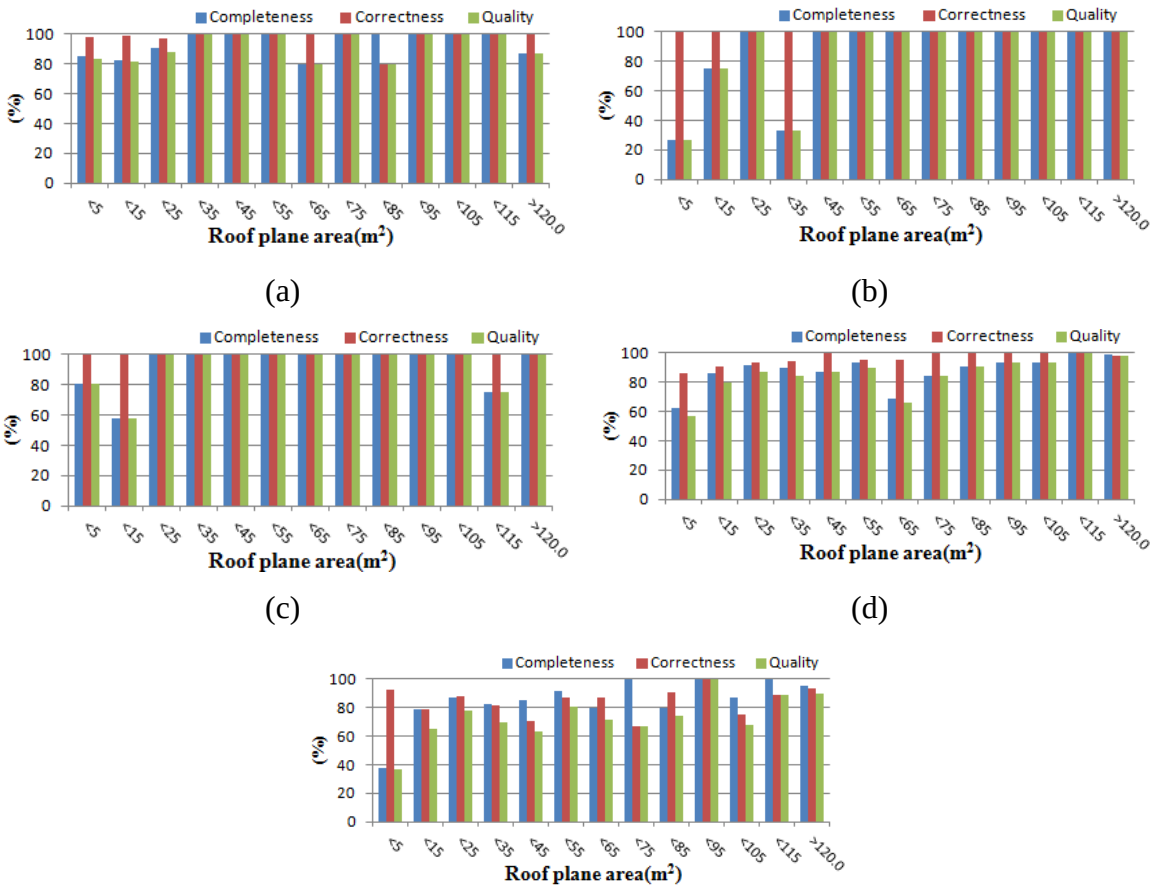


Figure 4.10 Object-based evaluation as a function of the roof plane area: (a) Area 1, (b) Area 2, (c) Area 3, (d) Area 4, and (e) Area 5

As shown in Table 4.1, the area-based evaluations show that similar levels of model quality were achieved for both the Vaihigen dataset and the Toronto dataset. However, the object-based evaluations indicate that the model quality for the Vaihingen dataset is better than one for the Toronto dataset. This is mainly related to segmentation errors which occur more in complex scenes. We observed that many roof planes in the Toronto dataset were under-segmented by merging adjacent clusters. As a result, building

rooftop models generated from under-segmented clusters caused a low success rate of the completeness.

Also, we compared the evaluation results with those assessed for other algorithms that were reported in Rottensteiner et al. (2014) where area-based evaluation results were not reported (Table 4.2). The object-based evaluation results (Table 4.2(a)) demonstrate that our method can outperform other building reconstruction algorithms except for the BNU in terms of the completeness and quality. In particular, when roof planes, whose area is larger than $10m^2$, were considered, our proposed method showed more accurate results. The BNU, which outperform our method, was assessed only for Area 3. With regard to robustness, our proposed method outperforms the BNU. The correctness of our method is better than the average of all other evaluated methods. Considering that the correctness is above 90% for all compared methods except MON and FIE, the correctness of our method is large enough. Also, the superiority of our method can be proven by Toronto dataset which consists of complex buildings. Only three participants submitted their results for Toronto dataset. Out of them, our method achieved the best results for all indices.

Table 4.2 Evaluation results of algorithms reported in Rottensteiner et al. (2014)

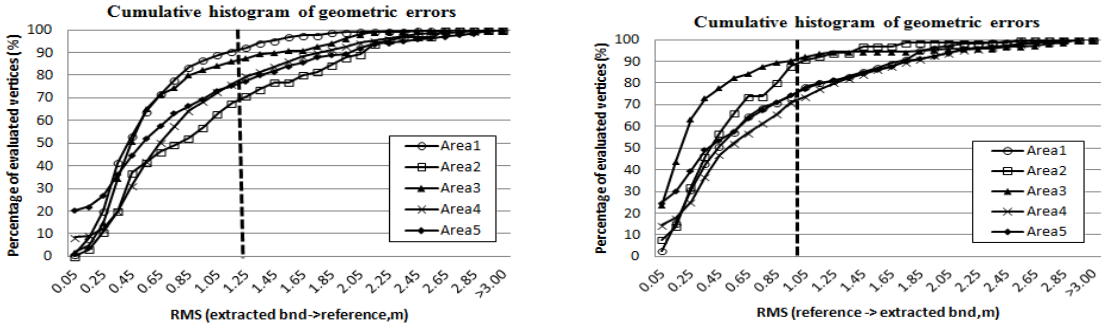
Dataset	*Algorithm*	*(a) Object-based evaluation using confusion matrix*						*(b) RMSE_XY (m)*
		For all roof planes			*For roof planes ($10m^2$ area)*			
		Comp. (%)	*Corr. (%)*	*Quality (%)*	*Comp. (%)*	*Corr. (%)*	*Quality (%)*	
Vaihingen	MON (Awrangjeb and Fraser, 2013)	77.5	89.7	71.2	90.3	91.4	83.5	0.90
	VSK (Dorninger and Pfeifer, 2008)	74.2	98.6	73.5	86.1	98.6	85.2	0.83
	ITCE1 (Elberink and Vosselman, 2009, 2011)	69.4	90.1	63.1	78.4	90.3	69.5	1.00
	ITCE2 (Elberink and Vosselman, 2009, 2011)	69.8	98.3	68.7	76.8	100.0	76.8	1.03
	ITCX1 (Xiong et al., 2014)	69.5	98.1	68.7	74.4	98.0	73.2	0.70
	ITCX2 (Xiong et al., 2014)	82.0	92.9	76.8	91.0	98.1	89.3	0.70
	ITCX3 (Xiong et al., 2014)	82.8	94.9	78.7	93.2	97.8	91.2	0.70
	CAS (Xiao, Y.)	68.5	100.0	68.5	81.2	100.0	81.2	0.75
	TUD (Perera et al., 2012)	70.0	95.8	67.8	78.8	98.6	78.0	0.70
	YOR (Sohn et al., 2012)	79.9	99.5	79.5	91.8	99.7	91.6	0.63
	KNTU (Zarea et al.)	80.4	96.7	78.3	91.9	97.7	90.0	0.90
	FIE (Bulatov et al., 2014)	82.6	83.1	70.7	88.7	93.4	83.5	1.10
	CKU (Rau and Lin, 2011)	82.1	96.8	80.1	91.4	99.4	90.9	0.73
	BNU (Zhang et al., 2011)	87.2	100.0	87.2	96.0	100.0	97.1	0.60
	Proposed method	**83.1**	**99.4**	**82.6**	**95.8**	**99.5**	**95.3**	**0.76**
Toronto	YOR (Sohn et al., 2012)	70.0	91.7	66.2	86.4	92.1	80.4	0.90
	CKU (Rau and Lin, 2011)	69.5	81.8	60.1	79.1	81.4	67.1	1.75
	FIE (Bulatov et al., 2014)	82.3	91.5	49.9	60.4	91.9	57.3	1.40
	Proposed method	**74.1**	**91.0**	**69.4**	**91.0**	**92.9**	**85.2**	**0.96**

4.5.2 Shape-based and Angle-based Evaluations

Geometrical errors in planimetry, and in height were assessed using RMSE. The RMSE measures Euclidean distance in two different ways: (1) from a vertex in the reconstructed rooftop model to its closest vertex in reference model, and (2) from a vertex in the reference model to its closest vertex in the reconstructed rooftop model. Both RMSEs are measured using Eq. (3.2) described in section 3.3.1.2. Note that only distances shorter than

a certain tolerance distance (<3m) were considered as introduced by Rottensteiner et al.(2014).

The average RMSE of distances in planimetry for the Vaihigen dataset and the Toronto dataset are 0.76m and 0.96m, respectively. As shown in Table 4.2(b), the geometric accuracy is better than the average geometric accuracy of building models reconstructed by other algorithms. Figure 4.11 shows the cumulative histogram of geometric accuracy in RMSE over the five sub-regions. Overall, more than 70% of evaluated vertices are located with less than 1.25m RMSE. In most test regions, the results of RMSE of reference vertices (Figure 4.11(b)) are better than those of RMSE of extracted vertices (Figure 4.11(a)). The reason is that the proposed method provides under-simplified models with redundant vertices (i.e., having more numbers of vertices compared to the reference model). Note that the closest vertex within a certain tolerance distance (>3m) was used to calculated RMSE. Thus, RMSE of extracted vertices, which have redundant vertices, tends to be worse than one of reference vertices.



(a) (b)

Figure 4.11 The cumulative histogram of geometrical errors: (a) RMSE of extracted vertices w.r.t reference vertices, and (b) RMSE of reference vertices w.r.t extracted vertices.

Hausdorff distance was applied to 2D outer boundaries and to 3D roof planes with *1:1* correspondence, respectively (Table 4.3(b)). The averages of Hausdorff distance for 2D outer boundaries and for 3D roof planes are 1.81m and 1.17m, respectively. The results show that the maximum distance between the vertices of reference rooftop models and extracted rooftop models is expected to be less than roughly twice the RMSE by our proposed method. Also, the average of the Hausdorff distance for 2D outer boundaries is larger than the value for 3D roof planes. This is mainly caused by topology relations between roof planes. As shown in Figure 4.12, two roof planes, which share a common edge in reference models (or in extracted models), were represented by separated roof planes in extracted models (or reference models). The different topology relations caused a large amount of shape differences in outer boundary representation.

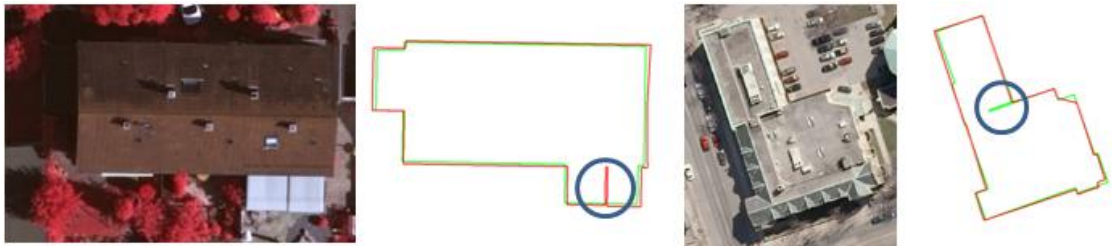
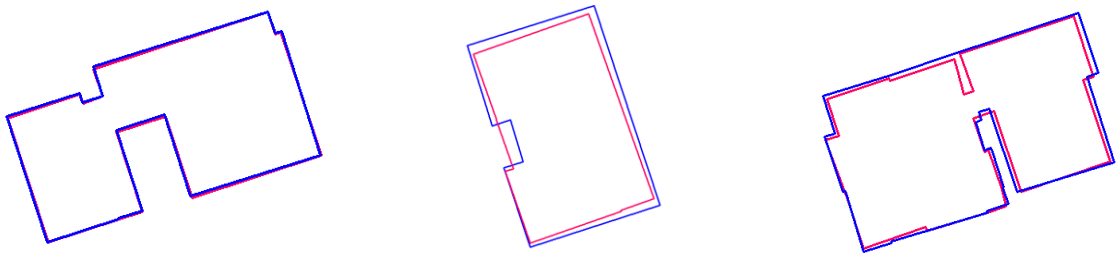


Figure 4.12 Examples of a large amount of Hausdorff distance for 2D outer boundary (Red: Reference, Green: extracted rooftop model)

Turning function distance, which measures how similar two shapes are, was applied to outer building boundaries and to roof planes with 90% overlap, respectively. Roughly, when the value is smaller than approximately 0.03, two corresponding shapes are very similar in terms of visual inspection. However, when the value is larger than

approximately 0.05, the shapes are considerably dissimilar (Figure 4.13). For five sub-regions, the average turning function distances are 0.042 for 2D outer boundaries and 0.033 for 3D roof planes, respectively (Table 4.3(c)). Although turning function distances do not provide a specific range for which value is acceptable for building rooftop models, our results can be compared with examples given in Figure 4.13. The comparison indicates that the building rooftop models reconstructed by the proposed method can achieve acceptable shape similarities compared with reference building rooftop models in terms of visual inspection. Similarly to the results of Hausdorff distance, the turning function distance for 2D outer boundaries is larger than one for 3D roof planes due to different topologies and representations of rooftop models.



(a) (b) (c)

Figure 4.13 Approximate ranges of turning function distance (blue: reference, red: extracted model): (a) 0.016, (b) 0.055, and (c) 0.105

In order to evaluate the quality of model orientation, the angle difference was measured by calculating the difference of dominant orientations between reconstructed rooftop models and reference rooftop models. Table 4.3(a) shows the angle differences for five sub-regions where the averages of angle differences are 1.17º for 2D outer boundaries

and 0.91º for 3D roof planes, respectively. Note that main angles for outer boundary and for 3D roof planes can be different because the main angle is separately determined for outer boundary and 3D roof planes. The orientation error was entirely caused by representative angles of CLF which were used to represent a regular pattern of the line orientation. The representative angles of CLF were calculated from all initial boundary lines connecting boundary points of individual building models without any prior knowledge of building orientations. Thus, a large amount of orientation error in small building models can be accidently caused if angles of the boundary lines were distorted by local distributions of boundary points.

Table 4.3 Angle-based and shape-based evaluations

Dataset	*Sub-Set*	*For 2D outer boundary*			*For 3D roof planes with 1:1 correspondence (90% overlap)*		
		(a) Angle difference (deg)	*(b) Hausdorff distance (m)*	*(c) Turning function distance*	*(a) Angle difference (deg)*	*(b) Hausdorff distance (m)*	*(c) Turning function distance*
Vaihingen	*Area 1*	1.32	1.33	0.049	0.78	0.46	0.020
	Area 2	1.62	1.26	0.040	1.11	1.77	0.041
	Area 3	0.59	0.93	0.031	0.44	0.48	0.016
	Sub-total	1.18	1.17	0.040	0.78	0.90	0.026
Toronto	*Area 4*	1.30	2.44	0.046	1.30	1.38	0.040
	Area 5	1.04	3.10	0.046	0.91	1.75	0.047
	Sub-total	1.17	2.77	0.046	1.11	1.57	0.044
Total		1.17	1.81	0.042	0.91	1.17	0.033

Additionally, topology relations were assessed by comparing overlap area between reference rooftop planes and extracted rooftop planes. Table 4.4 represents the number of instances of *1:1*, *1:M*, N:1, and *N:M* relations. More than 63% of roof planes are matched with *1:1* relations; 22% of roof planes have *N:1* relations; 7% of roof planes have *1:M*

relations; and 8% of roof planes have *N:M* relations. The topology errors are mainly caused by incorrect segmentation and incomplete modeling cues. In particular, relatively higher *N:1* relations are caused by under-segmentations and superstructures on roofs which often occur in complex scene. Thus, the *N:1* relations were observed more in the Toronto dataset.

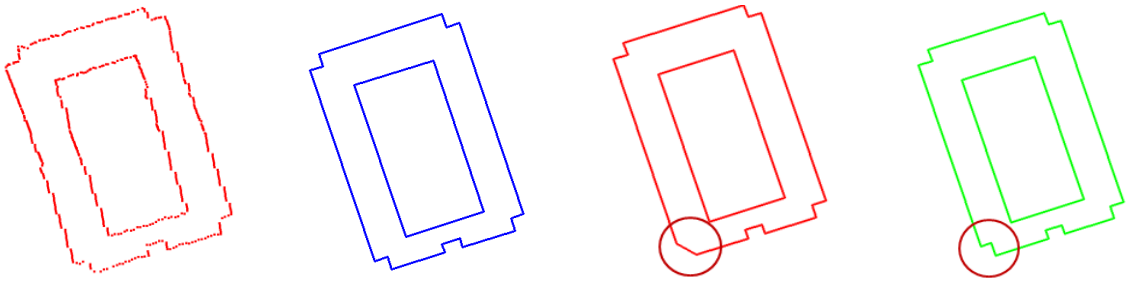
Table 4.4 Topology evaluation

Dataset	*Sub-Set*	*Topology (reference rooftop planes: extracted rooftop planes)*			
		$N_{1:1}$	$N_{N:1}$	$N_{1:M}$	$N_{N:M}$
Vaihingen	*Area 1*	125	36	17	8
	Area 2	29	5	9	1
	Area 3	72	49	6	2
	Sub-total	226	90	32	11
Toronto	*Area 4*	300	89	32	47
	Area 5	147	52	6	33
	Sub-total	447	141	38	80
Total		673	231	70	91

4.5.3 Effects on Weight Parameters

In order to evaluate an effect of weight parameters in MDL-based objective function, we compared building models generated using fixed weight parameters with building models generated using the proposed weighting methods. Area-based evaluations using confusion matrix and shape-based indices were applied. The area-based evaluations using confusion matrix show an increase of 1.3% for the completeness, a decrease of 0.7% for the correctness, and an increase of 0.6% for the quality when the proposed weighting methods were used (Table 4.5). For Hausdorff distance and turning function distance, the improvements of 0.44m and 0.003 were achieved, respectively (Table 4.6). While

evaluation results using confusion matrix and evaluation results for turning function distance show slight improvements, the results for Hausdorff distance show relatively large improvements for all sub-regions except for Area 3. Also, the most improvements for all evaluation methods were achieved by Area 4 where a relatively large number of shape differences at local scale between extracted models and reference models were observed. Figure 4.14 shows an example where shape difference at local scale is reduced by the proposed weighting methods. When fixed weight parameters were used, a lower part of the building model (red circle) were under-simplified (Figure 4.14(c)). This is related to the number of boundary points and a degree of model complexity. A large number of observations produced relatively high closeness value compared with complexity value. This caused imbalance between two values because fixed weight parameters do not consider the property of an individual building model. In contrast, the closeness term and the complexity term were balanced by using flexible weight parameters (Figure 4.14(d)). As shown in Table 4.5, Table 4.6 and Figure 4.14, applying flexible weight values makes positive effects in preserving shapes similar to reference rooftop models.



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

Figure 4.14. Effect on flexible weight parameters: (a) boundary points, (b) reference building model, (c) building model generated with fixed weight parameters, and (d) building model generated with flexible weight parameters.

Table 4.5 Effect on weight parameters in confusion matrix-based evaluation

Dataset	*Sub-Set*	*(a) Fixed weight parameters*			*(b) Weight parameters determined by the proposed method*			*(b)-(a)*		
		$Comp_{Area}$ *(%)*	$Corr_{Area}$ *(%)*	$Qual_{Area}$ *(%)*	$Comp_{Area}$ *(%)*	$Corr_{Area}$ *(%)*	$Qual_{Area}$ *(%)*	$Comp_{Area}$ *(%)*	$Corr_{Area}$ *(%)*	$Qual_{Area}$ *(%)*
Vaihingen	*Area 1*	88.8	99.5	88.4	90.6	98.8	89.6	1.8	-0.7	1.2
	Area 2	90.2	99.8	90.0	91.3	99.7	91.0	1.1	-0.1	1.0
	Area 3	88.8	99.7	88.5	88.6	99.7	88.4	-0.2	0.0	-0.1
	Sub-total	89.3	99.7	89.0	90.2	99.4	89.7	0.9	-0.3	0.7
Toronto	*Area 4*	89.5	98.2	88.1	93.7	96.9	90.9	4.2	-1.3	2.8
	Area 5	93.8	93.5	88.1	93.1	92.0	86.1	-0.7	-1.5	-2.0
	Sub-total	91.7	95.9	88.1	93.4	94.5	88.5	1.8	-1.4	0.4
Total		90.2	98.1	88.6	91.5	97.4	89.2	1.3	-0.7	0.6

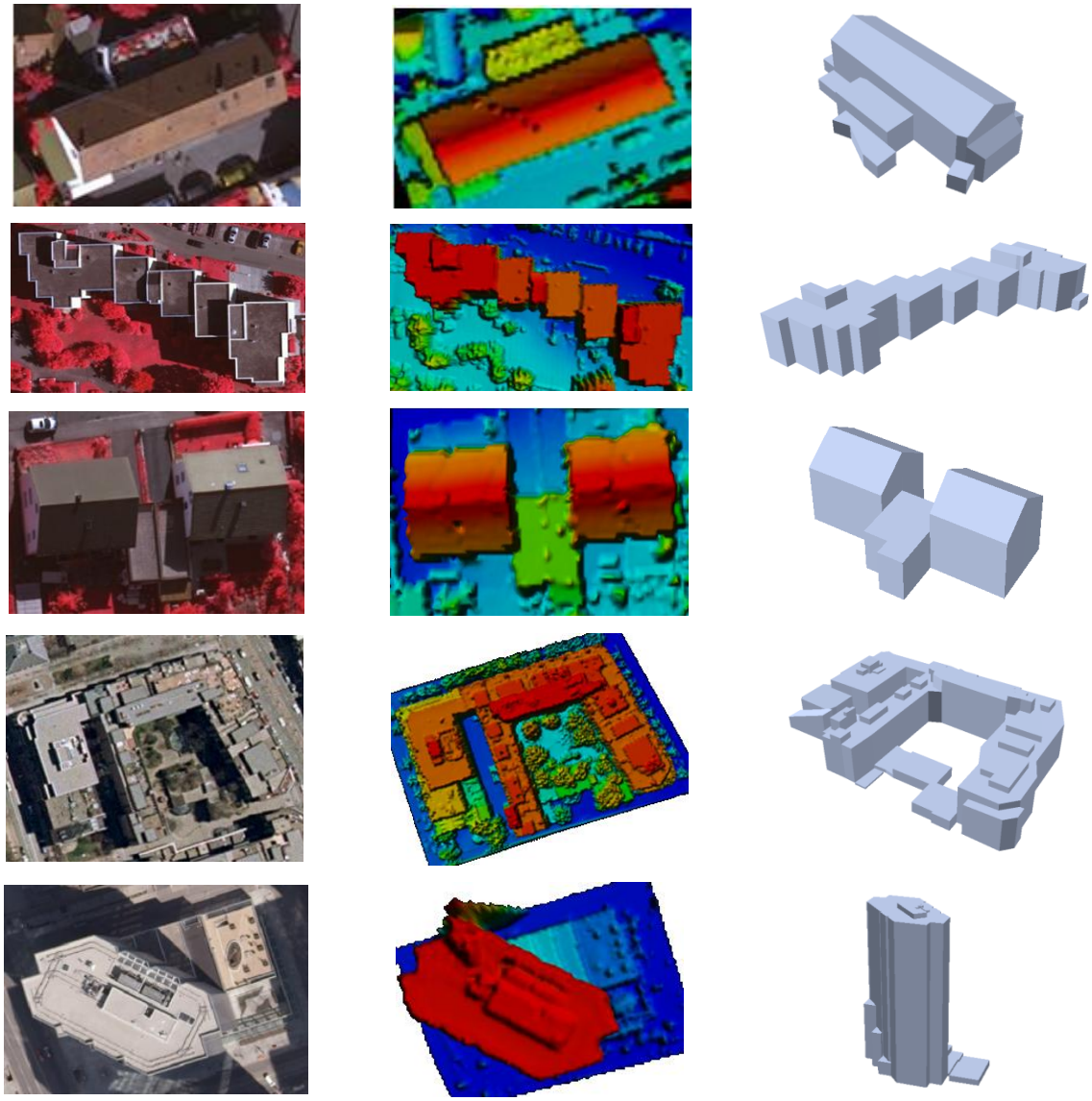
Table 4.6 Effect on weight parameters in shape-based evaluation

Dataset	*Sub-Set*	*(a) Fixed weight parameters*		*(b)Weight parameters determined by the proposed methods*		*(a)-(b)*	
		Hausdorff distance (m)	*Turning function distance*	*Hausdorff distance (m)*	*Turning function distance*	*Hausdorff distance (m)*	*Turning function distance*
Vaihingen	*Area 1*	1.44	0.047	1.33	0.049	0.11	-0.002
	Area 2	1.58	0.041	1.26	0.040	0.32	0.001
	Area 3	0.91	0.036	0.93	0.031	-0.02	0.005
	Sub-total	1.31	0.041	1.17	0.040	0.14	0.001
Toronto	*Area 4*	3.76	0.058	2.44	0.046	1.32	0.012
	Area 5	3.58	0.045	3.10	0.046	0.48	-0.001
	Sub-total	3.67	0.052	2.77	0.046	0.9	0.006
Total		2.25	0.045	1.81	0.042	0.44	0.003

4.5.4 Visual Inspection

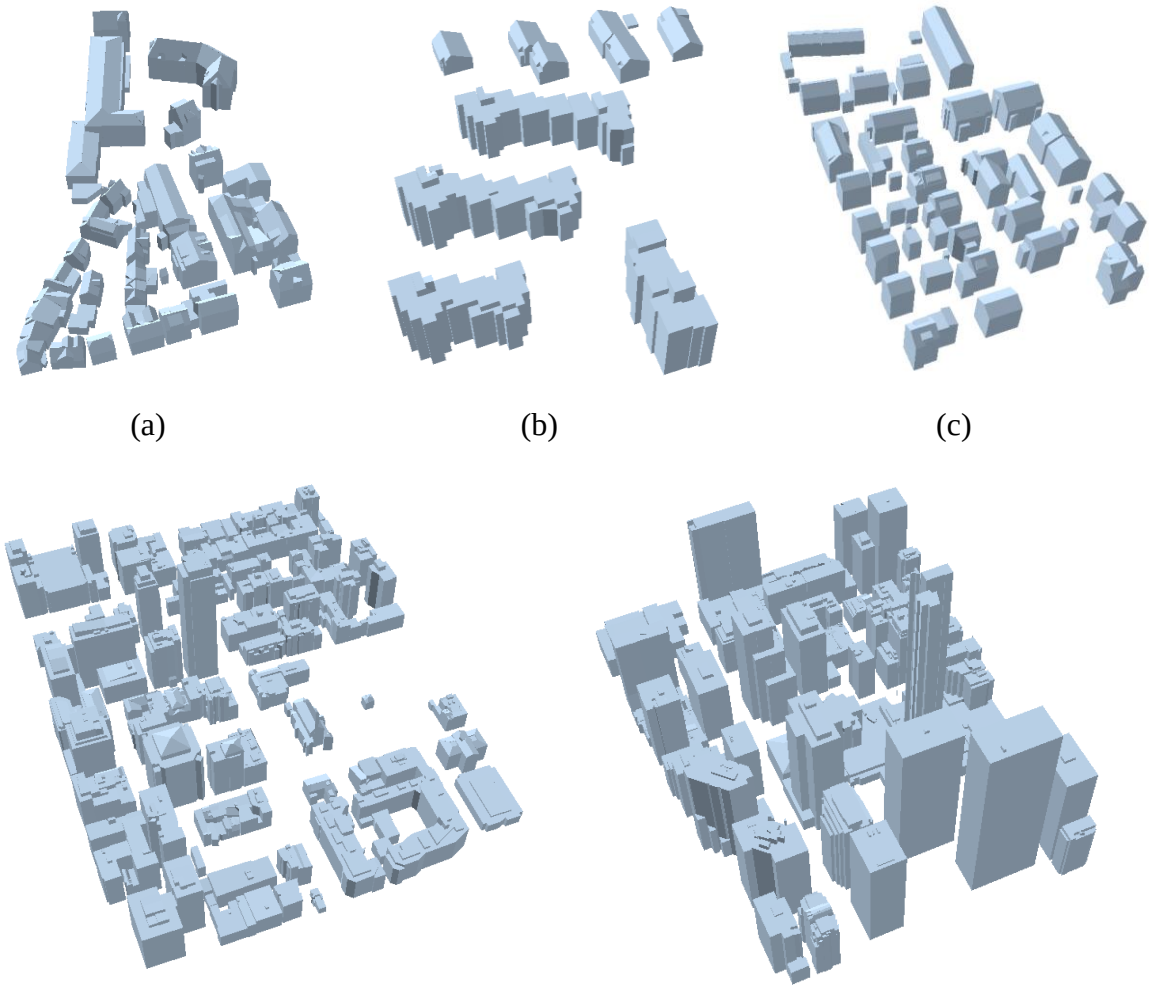
Figure 4.15 visualizes reconstructed building rooftop models which are representative buildings of five sub-regions. Visual inspection indicates that the proposed building reconstruction method can robustly provide accurate regularized 3D building rooftop

models in both simple scenes and complex scenes. Figure 4.16 shows all reconstruction building rooftop models over our test datasets.



(a) (b) (c)

Figure 4.15 Reconstructed building models with complex roof structure: (a) image, (b) LiDAR point clouds, and (c) perspective view of the reconstructed 3D building model.



(a) (b) (c)

(d) (e)

Figure 4.16 Reconstructed building models: (a) Area 1, (b) Area 2, (c) Area 3, (d) Area 4 and (e) Area 5

4.5.5 Limitations

Even though the proposed rooftop modeling method provided promising results, the method has some limitations. The main limitation is caused by plane clustering errors. Figure 4.17(a) shows that roof points were under-segmented by merging adjacent plane clusters. Also, superstructures on the rooftop were not detected due to the small number of

roof points (Figure 4.17(b)). The other limitation is that the method cannot extract polylines with small length as shown in Figure 4.17(c). This is because there are no sufficient observations to support the boundary with small length.

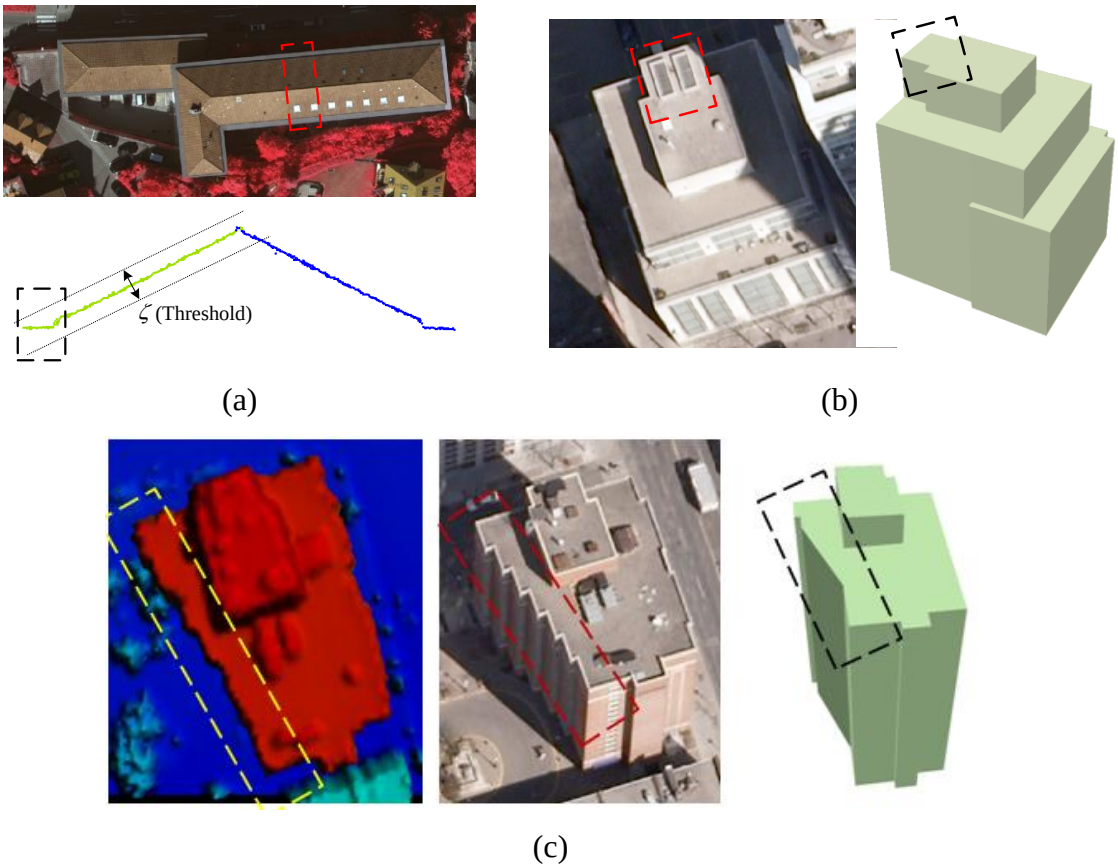


Figure 4.17 Limitations of the proposed rooftop modeling method: (a) incomplete plane clustering, (b) superstructures, and (c) over-simplification.

4.6 Summary

In this chapter, we proposed an automatic 3D building reconstruction method which covers a full chain of rooftop modeling. Building-labelled points were segmented into homogeneous clusters with a hierarchical structure which enables explicit interpretation of

building rooftop configuration. In order to effectively gather evidence of a rooftop structure, three linear modeling cues including intersection line, step lines, and boundaries were separately extracted by considering their characteristics. In the proposed method, regularization is the most important process which implicitly imposes geometric regularities on reconstructed rooftop models based on MDL principle. In the MDL framework, finding a regularized rooftop model was recognized as a model selection problem. The best model was selected by minimizing *DL* values among competing hypotheses generated by a newly designed hypothesis generation process. In order to automatically control weight parameters, a Min-Max based weighting method and Entropy-based weighting method were proposed. The experimental results showed that the proposed method can provide qualitatively and quantitatively well-regularized 3D building rooftop models. More specifically, the results are summarized as follows:

- The proposed method provided a robust solution for 3D rooftop modeling regardless of scene complexity, e.g., typical European style structure with relatively simple building shapes as well as complex clusters of high-rise buildings. This is achieved by the hierarchical clustering of building rooftop points. Even though modeling cues were incompletely extracted, the BSP method produced geometrically and topologically correct rooftop models.
- Evaluation results using confusion matrix showed that the proposed method outperforms other building reconstruction algorithms. However, object-based evaluation results indicated that our method has a limitation on extracting small size rooftops. It is a common problem in data-driven approaches due to the fact it is

difficult to extract modeling cues from the small number of roof points. One possible solution for this problem is to combine the data-driven method and model-driven method by taking their complementary properties.

- The proposed weighting methods have a positive effect on the building regularization process. Results for Hausdorff distance showed that the values are considerably improved when flexible weight parameters in MDL objective function were applied. In particular, shape deformation (under-simplified or over-simplified model) at a local scale were reduced by the proposed method.
- Angle based evaluation shows that the method has 1.17^{o} difference compared to the reference. However, the main orientations of building models in this study were determined without any prior knowledge. Thus, the accidently large amount of orientation error can occur in small size buildings. One possible solution for the problem is to use image data which can explicitly provide the orientation of building model.

Chapter 5

Matching Aerial Images to 3D Building Models Using Context-based Geometric Hashing

A city is a dynamic entity, whose environment is constantly changing over time. Accordingly, its virtual city models also need to be regularly updated to support accurate model-based decisions. A concept of continuous city modeling is to progressively reconstruct city models by accommodating their changes recognized in the spatio-temporal domain, while preserving unchanged structures. A first critical step for continuous city modeling is to coherently register remotely sensed data taken at different epochs with existing building models. In this chapter, we propose a new model-to-image registration method using a context-based geometric hashing (CGH) method to align a single image with existing 3D building models. This model-to-image registration process consists of three steps: 1) feature extraction, 2) similarity measure, and matching, and 3) estimating exterior orientation parameters (EOPs) of a single image. For feature extraction, we propose two types of matching cues: edged corner features representing the saliency of building corner points with associated edges, and contextual relations among the edged corner features within an individual roof. A set of matched corners are found with a given proximity measure through geometric hashing, and optimal matches are then finally determined by maximizing the matching cost encoding contextual similarity between matching candidates. Final matched corners are used for adjusting EOPs of the single airborne image by a least square method based on collinearity equations. The result shows that acceptable accuracy of EOPs of a single image can be achievable using the proposed registration approach as an alternative to labor-intensive manual registration process.

5.1 Introduction

In recent years, a number of mega-cities have built-up detailed 3D city models to support the decisions making process for smart city applications. These 3D models are usually static snap-shots of the environment at the time of their data acquisition. However, cities are dynamic systems that continuously change over time. Accordingly, their virtual representations need to be regularly updated in a timely manner in order to allow for accurate analysis and simulation results that decisions are based upon. In this context, a framework for continuous city modeling by integrating multiple data sources was proposed by Sohn et al. (2013).

A fundamental step to facilitate this task is to coherently register remotely sensed data taken at different epochs with existing 3D building models. Great research efforts have already been undertaken to address the related problem of image registration. For example, Brown (1992) and Zitova and Flusser (2003) give comprehensive literature reviews of relevant methods. Fonseca and Manjunath (1996) conducted a comparative study of different registration techniques for multisensory remotely sensed imagery. Although most of the existing registration methods have shown promising success in controlled environments, registration is still a challenging task due to the diverse properties of remote sensing data related to resolution, spectral bands, accuracy, signal-to-noise ratio, scene complexity, occlusions, etc. (Zitova and Flusser, 2003). These variables have a major influence on the effectiveness of the registration process, and lead to severe difficulties when attempting to generalize it. Still, though a universal method applicable to

all registration tasks seems impossible, the majority of existing methods consist of the following three steps (Brown, 1992; Habib et al., 2005):

- ***Feature extraction***: Salient features such as closed-boundary regions, edges, contour lines, intersection points, corners, etc. are detected in two datasets, and used in the registration process. Special care has to be taken to ensure that these features are distinctive, well distributed and can be reliably observed in both datasets.
- ***Similarity measure and matching***: The correspondences between features that are extracted from two different datasets are then found by a matching process. A similarity measure that is based on the attributes of the features quantifies its correctness. To be effective, the measure should consider the specific feature characteristics in order to avoid possible ambiguities and to be accurately evaluated.
- ***Transformation***: Based on the established correspondences, a transformation function is constructed that transforms one dataset to the other. The function depends on the assumed geometric discrepancies between both datasets, the mechanism of data acquisition, and required accuracy of the registration.

A successful registration strategy must consider the characteristics of the data sources, its later applications, and the required accuracy during the design and combination of the individual steps.

Recent advancements of aerial image acquisition make direct geo-referencing for certain types of applications (coarse localization and visualization) possible. If an

engineering-level accuracy is needed, however, including continuous 3D city modeling, the exterior orientation parameters (EOPs) obtained through these techniques may need to be further adjusted. In indirect geo-referencing of aerial images, accurate EOPs are generally determined by bundle adjustment with ground control points (GCPs). However, obtaining or surveying such points over a large-scale area is labor intensive, and time-consuming. An alternative method is to use other known points instead.

Nowadays, large-scale 3D city models have been generated for many major cities in the world, and are, e.g., available within the Google Earth platform. Thus, the corner points of 3D building models can be used for registration purposes. However, the quality of the existing models is often unknown and varies from building to building, which is the result from different reconstruction methods and data sources being applied. For example, LiDAR points are mostly measured within the roof faces and seldom at their edges, which often results in their boundaries and corner points to be geometrically inexact. Thus, the sole use of corner points from existing building data bases as local features can lead to matching ambiguities and therefore to errors in the registration.

To address this issue for the registration of single images with existing 3D building models, we propose to use two types of matching cues: (1) edged corner features that represent the saliency of building corner points with associated edges, and (2) context features that represent the relations between the edged corner features within an individual roof. Our matching method is based on the Geometric Hashing method, which is a well-known indexing-based object recognition technique, and it is combined with a scoring function that reinforces the context force.

This chapter is organized into four parts. In section 5.2, we address the new model-to-image registration including feature extraction (section 5.2.1) and similarity and matching (section 5.2.2). Section 5.3 deals with the evaluation of the approach, and conclusions are given in section 5.4.

5.2 Registration Method

Figure 5.1 illustrates the proposed registration method for registering a single image with existing 3D building models using extracted edged corner features. It starts by back-projecting the 3D building models to the image using initial (or at later steps updated) EOPs. Then with the help of the similarity measure, the matching process finds corresponding features using a CGH method. Based on the matched feature pairs, the EOPs of the single image are estimated by a least square adjustment. As shown in Figure 5.1, the second and third steps are conducted iteratively to find optimal EOPs until the corresponding matching pairs do not further improve. The three steps of the proposed method are further discussed in the following sub-sections whereat the last two steps are discussed together.

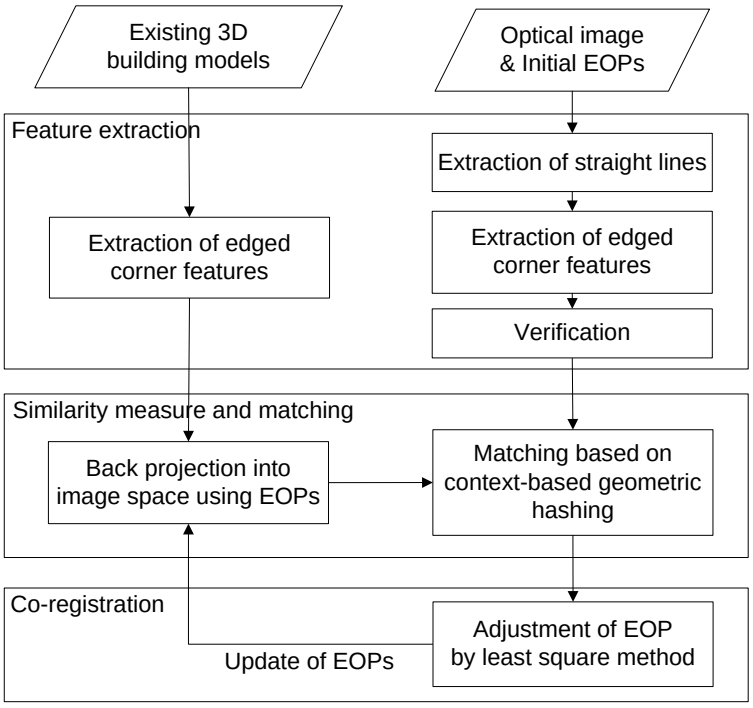


Figure 5.1 Flowchart of the proposed alignment method

5.2.1 Feature Extraction

Feature extraction is the first step of the registration task. As previously mentioned, feature selection should consider the properties of the given datasets, the application, and the required accuracy. In this study, we use two different types of features; edged corner features, and context features. An edged corner feature, which consists of a corner point, and the two associated lines that potentially intersect at this point ("arms"), provides local structure information of a building. In the building models, it is relatively straightforward to extract this feature because each vertex of a building polygon can be treated as a corner and the connected lines as arms. In an image with rich texture information, various corner detectors, and line detectors can be used to extract edged corner features. A context feature is defined as a characteristic spatial relation between two edged corner features selected

within an individual roof. This context feature is used to represent global structure information so that more accurate, and robust matching results can be achieved. Section 5.2.1.1 explains the extraction of edged corner features from an image, and section 5.2.1.2 describes the properties of context features.

5.2.1.1 Edged Corner Feature Extraction from Image

Edged corner features from a single image are extracted by three separate steps; 1) extraction of straight lines, 2) extraction of corners and their arms, and 3) verification. The process starts with the extraction of straight lines from a single image by applying a straight line detector. We use Kovesi's algorithm, which relies on the calculation of phase congruency to localize, and link edges (Korvesi, 2011). Then, corners are extracted by estimating the intersection of the extracted straight lines, considering the proximity with a given distance threshold (T_d = 20 pixels). Afterwards, corner arms are determined by two straight lines used to extract the corner with fixed length (20 pixels). This procedure may produce incorrect corners because the proximity constraint is the only one considered. Thus, the verification process removes incorrectly extracted corners based on geometric, and radiometric constraints. As a geometric constraint, the inner angle between two corner arms is calculated, and investigated to remove corners with sharp inner angles. In general, many of building structures appears in regular shapes following orthogonality and parallelism where small acute angles are found to be uncommon. Through this process, incorrectly extracted corners are filtered out by applying a user-defined inner angle threshold (T_θ = 10^o). For the radiometric constraint, we analyze the radiometric values

(Digital Number (DN) value or color value) of the left, and right flanking regions ($F_1^L, F_1^R, F_2^L, F_2^R$) of each corner arm with a flanking width (ε) as used in Ok et al.(2012). Figure 5.2 shows a configuration of a corner, its arms, and the concept of the flanking regions. In a correctly extracted corner, the average DN (or color) difference between F_1^L and F_2^R, $\|F_1^L - F_2^R\|$, or between F_1^R and F_2^L, $\|F_1^R - F_2^L\|$, is likely to be small, underlining the homogeneity of two regions while average DN difference between F_1^L and F_2^L, $\|F_1^L - F_2^L\|$, or between F_1^R and F_2^R, $\|F_1^R - F_2^R\|$, should be large enough to underline the heterogeneity of two regions. Thus, we measure two radiometric properties: the minimum average DN difference value of two neighbor flanking regions for homogeneity measurement, $D_{min}^{homo} = min\ (\|F_1^L - F_2^R\|,\ \|F_1^R - F_2^L\|)$, and the maximum DN difference value of two opposite flanking regions for heterogeneity measurement, $D_{max}^{hetero} = max\ (\|F_1^L - F_2^L\|,\ \|F_1^R - F_2^R\|)$. A corner is considered as an edged corner feature if the corner has a smaller D_{min}^{homo} than a threshold T_{homo} and if it has a larger D_{max}^{hetero} than a threshold T_{hetero}.

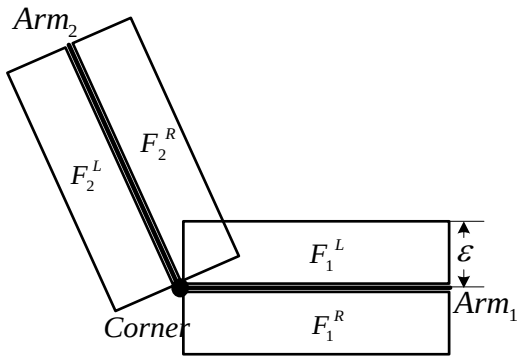


Figure 5.2 Edged corner feature (corner and its arms) and flanking regions

In order to determine thresholds for two radiometric properties, we assume that the intersection points are generated from both correct corners, and incorrect corners; and the two types of intersection points have different distributions with regards to their radiometric properties. Because there are two cases (correct corner and incorrect corner) for the average DN difference values, we can use the Otsu's binarization method (Otsu, 1979) to automatically determine an appropriate threshold value. The method was originally designed to extract an object from its background for binary image segmentation based on histogram distribution. It calculates the optimum threshold by separating the two classes (foreground and background) in such a way that their intra-class variance is minimal. In our study, a histogram of homogeneity values (or heterogeneity values) for the entire selection of points is generated, and the optimal threshold for homogeneity (or heterogeneity) is automatically determined by Otsu's binarization method.

5.2.1.2 Context Features

While an edged corner feature provides only local structure information about a building corner, context features partly impart global structure information related to the building configuration. Context features are set by selecting any two adjacent edged corner features, that is, four angles (θ_i^{left} , θ_i^{right}, θ_j^{left}, *and* θ_j^{right}) between a line (l) connecting the two corners (C_i and C_j) and their arms (Arm_i^{left}, Arm_i^{right}, Arm_j^{left}, and Arm_j^{right}) as shown in Figure 5.3. Note that each angle is determined by the relative line connecting any two corners (l). The context feature, which is invariant under scale, translation, and rotation, is used to calculate contextual similarity in our proposed score function (see section 5.2.2.2).

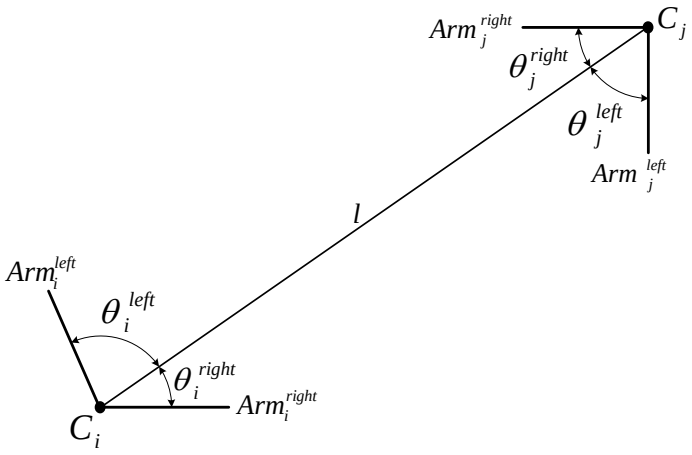


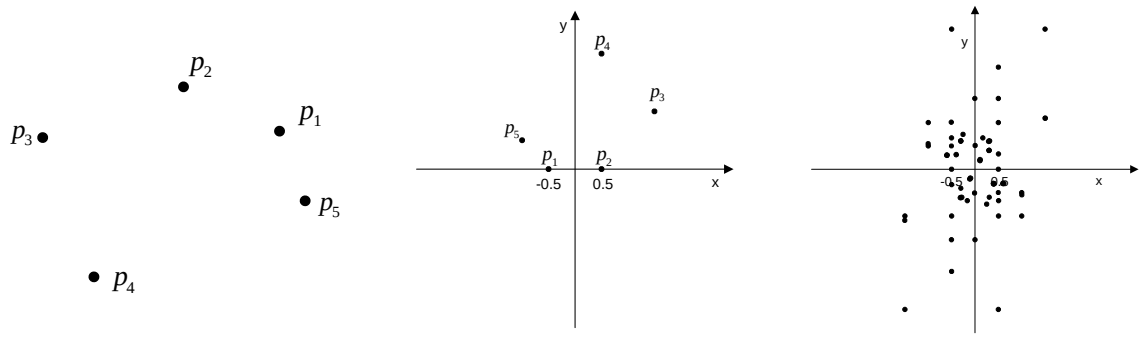
Figure 5.3 Context feature

5.2.2 Similarity Measurement and Primitives Matching

Similarity measurement, and matching process take place in the image space after the 3D building models are back-projected onto the image space using the collinearity equations with the initial EOPs (or updated EOPs). In order to find reliable, and accurate correspondences between features extracted from a single image, and building models, we introduce a CGH method where the vote counting scheme of a standard geometric hashing is supplemented by a newly developed similarity score function. The similarity score function consists of a unary term, and a contextual term. The unary term measures the similarity between edged corner features derived from the image and models while the contextual term measures the geometric property of context features. In the following sections, the standard geometric hashing and its limitations are described (section 5.2.2.1), and our proposed CGH method is introduced (section 5.2.2.2).

5.2.2.1 Geometric Hashing

Geometric hashing, a well-known indexing-based approach, is a model-based object recognition technique for retrieving objects in scenes from a constructed database (Wolfson and Rigoutsos, 1997). In geometric hashing, an object is represented as a set of geometric features such as points, and lines, and by its geometric relations, which are transformation-invariant under certain transformations. Since only local invariant geometric features are used, geometric hashing can handle partly occluded objects. Geometric hashing consists of two main stages: the pre-processing stage, and the recognition stage. The pre-processing stage encodes the representation of the objects in a database and stores them in a hash table. Given a set of object points ($p_k; k = 0, \dots, n$), a pair of points (p_i and p_j) is selected as a base pair (Figure 5.4(a)). The base pair is scaled, rotated, and translated into the reference frame. In the reference frame, the magnitude of the base pair equals 1; the midpoint between p_i and p_j is placed at origin of the reference frame; The vector $\overrightarrow{p_i p_j}$ corresponds to a unit vector of the *x* axis. The remaining points of the model are located in the coordinate frame based on corresponding base pair (Figure 5.4(b)). The locations (to be used as index) are quantized by a proper bin size and recorded with the form (model ID, used base pair ID) in hash table. For all possible base pairs, all entries of points are similarly recorded in the hash table (Figure 5.4(c)).



(a) (b) (c)

Figure 5.4 Geometric Hashing: (a) model points, (b) hashing table with a base pair, (c) all hashing table entries with all base pairs

In the subsequent recognition stage, the invariants, which are derived from geometric features in a scene, are used as indexing keys to assess the previously constructed hash table so that they can be matched with the stored models. In a similar way to the preprocessing stage, two points from a set of points in the scene are selected as the base pair. The remaining points are mapped to the hash table, and all entries in the corresponding hash table bin receive a vote. Correspondences are determined by a vote counting scheme, producing candidate matches.

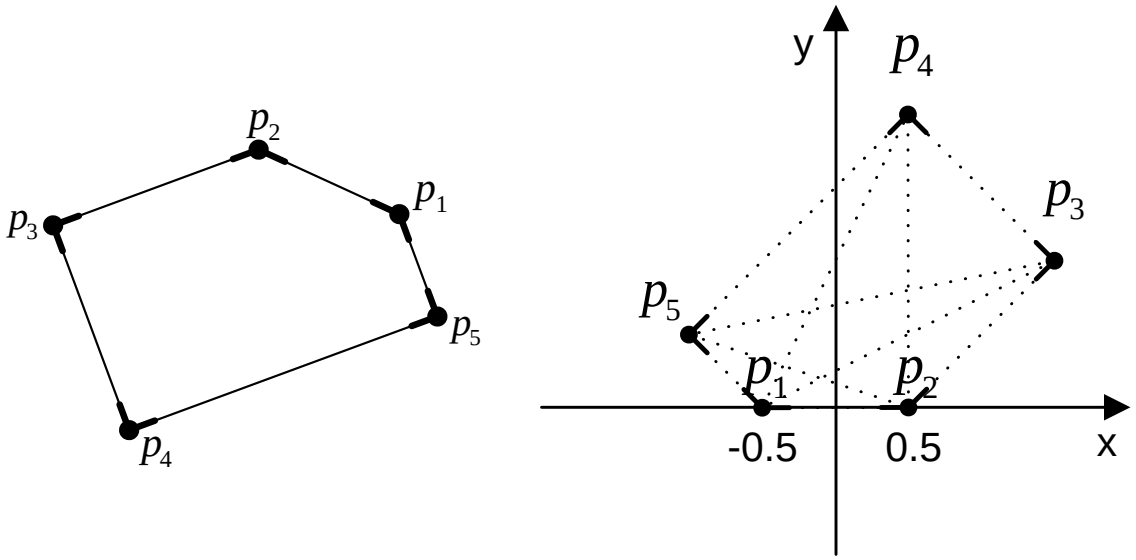
Although geometric hashing can solve matching problems of rotated, translated, and partly occluded objects, it has some limitations. The first limitation is that the method is sensitive to the bin size used for quantization of the hash table. While a large bin size in the hash table cannot separate between two close points, a small bin size cannot deal with the position error of the point. Secondly, geometric hashing can produce redundant solutions due to its vote counting scheme (Wolfson and Rigoutsos, 1997). Although it can significantly reduce candidate hypotheses, a verification step or additional fine matching

step is required to find optimal matches. Thirdly, geometric hashing has a weakness in cases where the scene contains many features of similar shapes at different scales, and rotations. Without any constraints (e.g. position, scale and rotation) based on prior knowledge about the model, geometric hashing may produce incorrect matches due to the matching ambiguity. Fourthly, the complexity of processing increases by the number of base pairs, and the number of features in the scene (Lamdan and Wolfson, 1988). To address these limitations, we enhance the standard geometric hashing by changing the vote counting scheme to a score function, and by adding several constraints such as scale difference of a base and specific selection of bases.

5.2.2.2 Context-based Geometric Hashing (CGH)

In this section, we describe the building model objects, and the scene by sets of edged corner features. Edged corner features derived from input building models are used to construct the hash table in the pre-processing stage while edged corner features derived from the single image are used in the recognition stage. Each given building model consists of several planes. Thus, in the pre-processing stage, we select two edged corner features, which belong to the same plane of the building model as the base pair. It can reduce the complexity of the hashing table, and ensures that the base pair retains the spatial information of the plane. The selected base pair is scaled, rotated, and translated to define the reference frame. The remaining edged corner features which belong to the whole building model are also transformed with the base pair. In contrast to the standard geometric hashing, our hashing table contains model IDs, feature IDs of the base pair, the

scale of the base pair (the rate of real distance of base pair), an index for member edged corner features, and context features generated by combinations with edged corner features. Figure 5.5 shows an example of the information to be stored in a hashing table.



(a) (b)

Figure 5.5 (a) Edged corner features derived from a model, and (b) information to be stored in hashing table (dotted lines represent context features).

Once all possible base pairs are set, the recognition stage tries to retrieve corresponding features based on the designed score function. Two edged corner features from the image are selected as base pair with two constraints: 1) scale constraint, and 2) position constraint. As a constraint on a scale, only those base pairs whose scale is similar to the scale of the base pair in the hash table are considered with an assumption that the initial EOPs provide an approximate scale of the image. Thus, if the scale ratio is smaller

than a user defined threshold ($T_s = 0.98$), the base pair is excluded from the set of possible base pairs. In addition to scale constraint, the possible positions of a base pair can be also restricted with a proper searching space. This searching space can be determined by calculating error propagation with the amount of assumed errors (calculated by the iterative process) for initial EOPs (updated EOPs) of the image, and the models. These two constraints reduce the matching ambiguity and the complexity of processing. After the selection of possible base pairs from the image, all remaining edged corner features in the image are transformed based on a selected base pair. Afterwards, the optimal matches are determined by comparing a similarity score. The process starts by generating context features from the model, and the image in a reference frame. Given a model that consists of five edged corner features (black color), ten context features can be generated as shown in Figure 5.6. Note that all edged corner features derived from the model are not matched with edged corner features derived from the image (red color). Thus, only edged corner features, which have corresponding image edged corner features within the search area (n=4 in Figure 5.6), and their corresponding context features (m=6 in Figure 5.6 (red long-dash)) are considered in the calculation of the similarity score function.

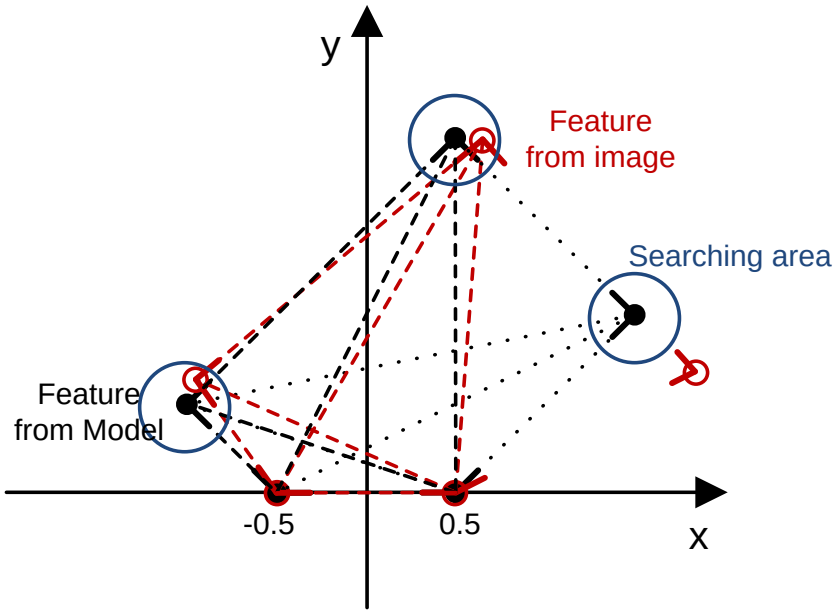


Figure 5.6 Context features to be used for calculating score function

The newly designed score function consists of a unary term, which measures the position differences of the matched points, and a contextual term, which measures length and angle differences of corresponding context features, as follows;

$$Score = \alpha \times \left[w \times \frac{\sum_{i=1}^{n} U(i)}{n} + (1 - w) \times \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} C(i,j)}{m} \right] \tag{5.1}$$

where,

$$\alpha = \begin{cases} 0 & if \ \frac{\#\ of\ matched\ features}{\#\ of\ features\ in\ the\ model} < T_c \\ 1 & else \end{cases} \tag{5.2}$$

α is an indicator function where the minimum number of features to be matched is determined depending on T_c (T_c = 0.5 in this study, at least 50% of corners in the model should be matched with corners from the image) so that all features of the model do not need to be detected in the image; *n* and *m* are the number of matched edged corner features and context features, respectively; *w* is a weight value which balances the unary term and the contextual term; in our case, *w* = 0.5 is heuristically selected.

Unary term: The unary term $U(i)$ measures the position distance between edged corner features derived from the model, and the image in reference frame. The position difference $\left\|P_i^M - P_i^I\right\|$ between an edged corner feature in the model and its corresponding feature in the image is normalized by the distance N_i^P calculated by error propagation.

$$U(i) = \frac{N_i^P - \left\|P_i^M - P_i^I\right\|}{N_i} \tag{5.3}$$

Contextual term: This term is designed to measure the similarity between context features in terms of length and four angles. The contextual term is calculated for all context features which are generated from matched edged corner features. For the length difference, $\left\|L_{ij}^M - L_{ij}^I\right\|$, the difference between lengths of context features in the model, and in the image is normalized by length N_{ij}^L of the context feature in the model. For angle differences, the angle difference $\left\|\theta_{ij}^{M_k} - \theta_{ij}^{I_k}\right\|$ between inner angles of a context feature is normalized by the N_{ij}^θ ($N_{ij}^\theta = \frac{\pi}{2}$).

$$C(i,j) = \frac{N_{ij}^{L} - \left\| L_{ij}^{M} - L_{ij}^{I} \right\|}{N_{ij}^{L}} + \frac{\sum_{k=1}^{4} \left(N_{ij}^{\theta} - \left\| \theta_{ij}^{M_k} - \theta_{ij}^{I_k} \right\| \right)}{4 \times N_{ij}^{\theta}} \quad (5.4)$$

For each model, a base pair and its corresponding corners which maximize the score function are selected as optimal matches. Note that if the maximum score is smaller than a certain threshold ($T_m = 0.6$ in this study), the matches are not considered as matched corners. Once all correspondences are determined, the EOPs of the image are adjusted through space resection using pairs of object coordinates of the existing building models, and newly derived image coordinates from the matching process. Values calculated from the similarity score function are used to weight matched pairs. The process continues until matched pairs do not change.

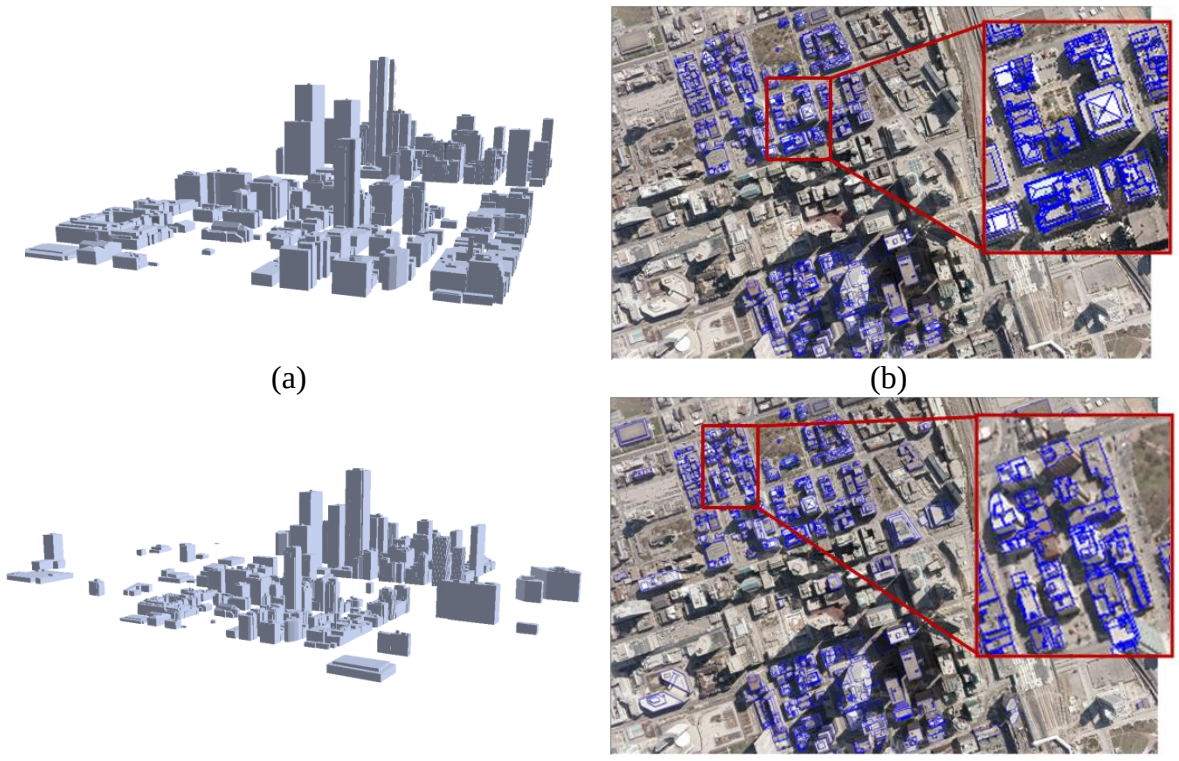
5.3 Experimental Results

The proposed CGH-based registration method was tested on the Downtown Toronto and the Vaihingen datasets. Table 5.1 shows characteristics of reference building models, which were used to determine EOPs. For the Downtown Toronto datasets, two different types of reference building models were prepared by : 1) a manual digitization process conducted by human operators, and 2) using the method proposed in Chapter 4. These two building models were used to investigate their respective effects on the performance of our method (Figure 5.7). For the Vaihingen datasets, LiDAR-driven building models were automatically generated by Kada and Wichmann (2013) and adjusted as described in

Wichmann and Kada (2014) as shown in Figure 5.8. A total of 16 check points for each dataset, which were evenly distributed throughout the image, were used to evaluate the accuracy of the EOPs.

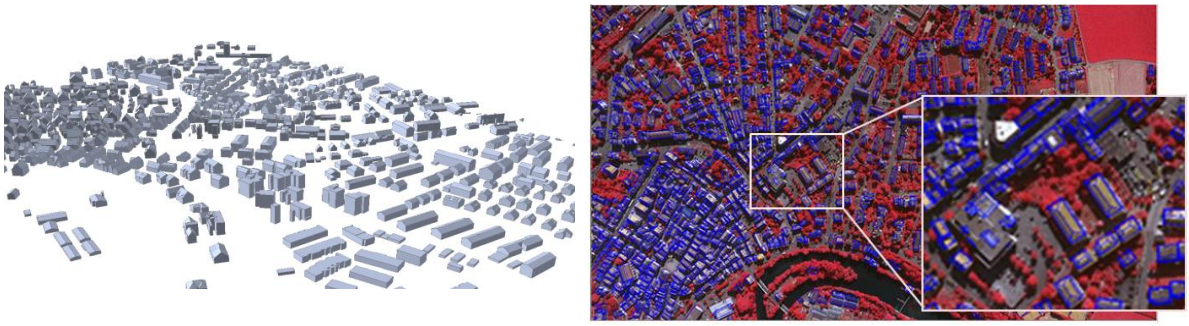
Table 5.1 Characteristics of reference building models

Dataset	*Reconstruction method*	*# of buildings*	*# of planes*	*Description*
Toronto	Manually digitized	159	1,560	· Complex clusters of high-rise buildings · Maximum building height : approximately 290m
	LiDAR-driven	126	1,066	
Vaihingen	LiDAR-driven	894	2,619	· Typical European style structures with simple building shapes · Maximum building height : approximately 32m



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

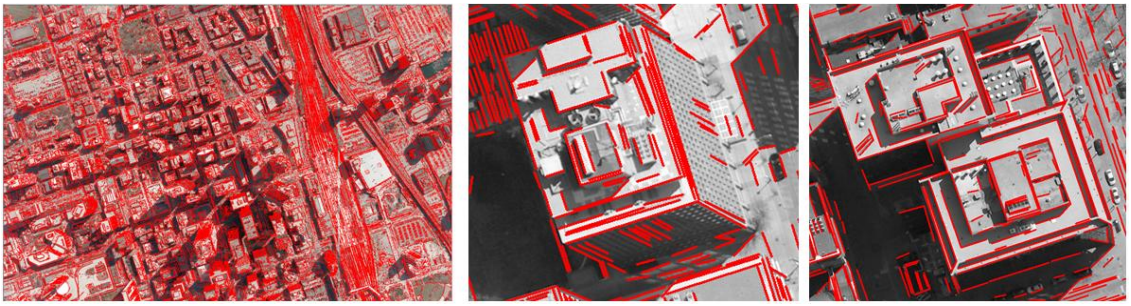
Figure 5.7 Toronto dataset: (a) LiDAR-driven building models, (b) LiDAR-driven building models back-projected to image, (c) manually digitized building models and (d) manually digitized building models back-projected to image.



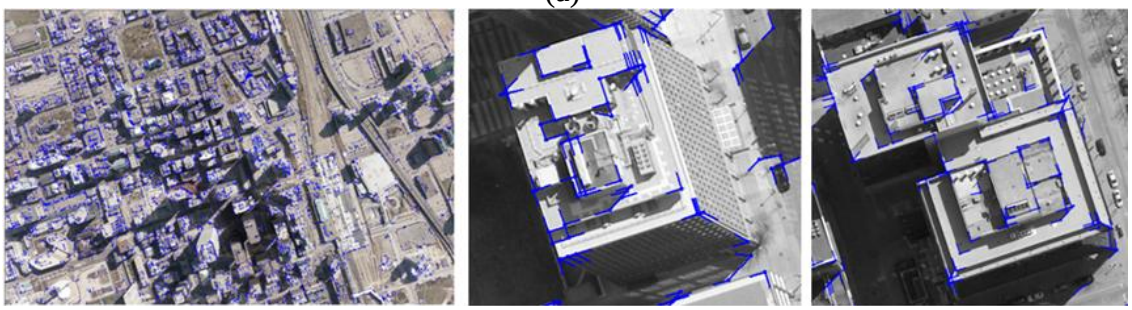
(a) (b)

Figure 5.8 Vaihingen dataset: (a) LiDAR-driven building models, (b) LiDAR-driven building models back-projected to image

For Downtown Toronto dataset, various analyses were conducted to evaluate the performance of the proposed registration method in detail. From the image, a total of 90,951 straight lines were extracted and 258,486 intersection points were derived by intersecting any two straight lines found within 20 pixels of proximity constraint. Out of these, 57,767 intersection points were selected as edged corner features following the removal of 15%, and 60% of intersection points using geometric constraint (T_θ=10º), and radiometric constraints ($T_{homo} = 26$, and $T_{hetero} = 55$), respectively (Table 5.2). The T_{homo} and T_{hetero} were automatically determined by Otsu's binarization method. Figure 5.9 shows edged corner features extracted from the aerial image. As many of the intersection points are not likely to be corners, the majority of them were removed. The method correctly detected corners and arms in most cases even though some corners were visually difficult to detect due to their low contrasts.



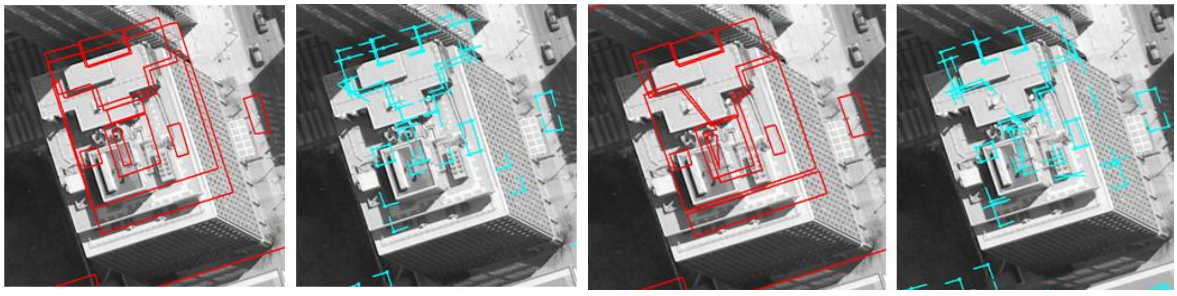
(a)



(b)

Figure 5.9 Edged corner features from image: (a) straight lines (red) and (b) edged corner features (blue)

After the existing building models were back-projected onto the image using error-contained EOPs, edged corner features were extracted from the vertices of the building models in the image space (Figure 5.10). It should be noted that two different datasets were used as the existing building models. Some edged corner features extracted from both existing building models were not observed in the image due to occlusions caused by neighbour building planes. Also, some edged corner features, in particular those extracted from LiDAR-driven building models do not match with the edged corner features extracted from the image due to modeling errors caused by irregular point distribution, occlusion and the reconstruction mechanism. Thus, correspondences between edged corner features from the image and from the existing building models are likely to be partly established.



(a) (b)

Figure 5.10 Features from existing building models: (a) manually digitized building models and their edged corner features and (b) LiDAR-driven building models and their edged corner features

The proposed CGH method was applied to find correspondences between features derived from the image and from existing building models. When manually digitized building models are used as the existing building models, a total of 693 edged corner features (7.8% of edged corner features extracted from the models) were matched using the parameters given in Table 5.3. Only 381 edged corner features (4.9%) were matched using LiDAR-driven building models (Table 5.2). It is noted that the number of matched edged corner features is influenced by the quality of the existing building models, and thresholds used, in particular T_m. As shown in Table 5.2, more edged corner features are matched when manually digitized building models were used as the existing building models than when LiDAR-driven building models were used. If T_m is set as a small value, the number of matched edged corner features increases, but this increases the risk it may contain a large number of incorrect matched edged corner features. The effect on the T_m will be discussed in detail later.

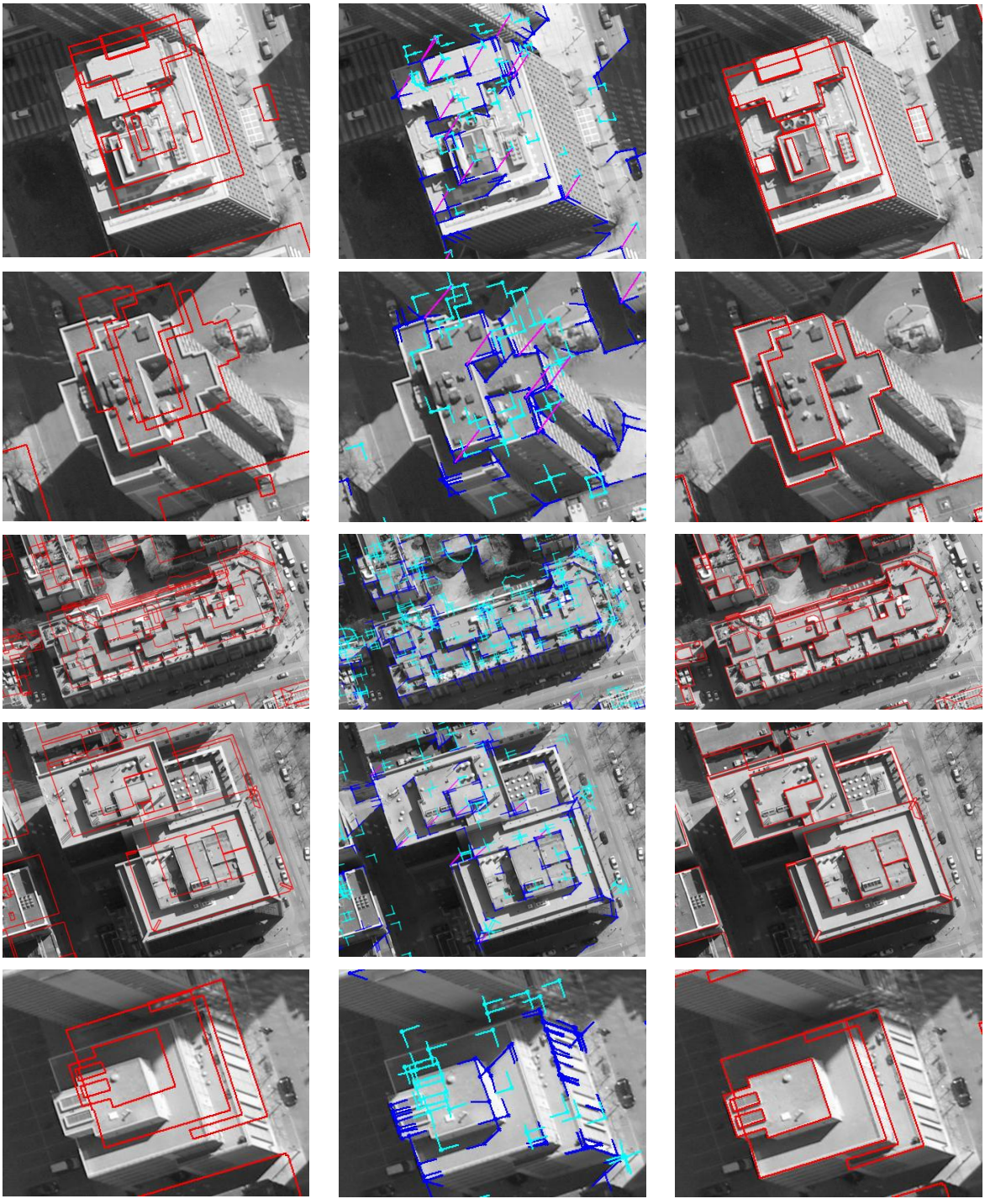
Table 5.2 Extracted features and matched features

	Image		*Existing building models*	
	Intersections	*Corners*	*Manually digitized building models*	*LiDAR-driven building models*
# of features	258,486	57,767	8,895	7,757
# of matched features	-	-	693	381

Table 5.3 Parameters setting

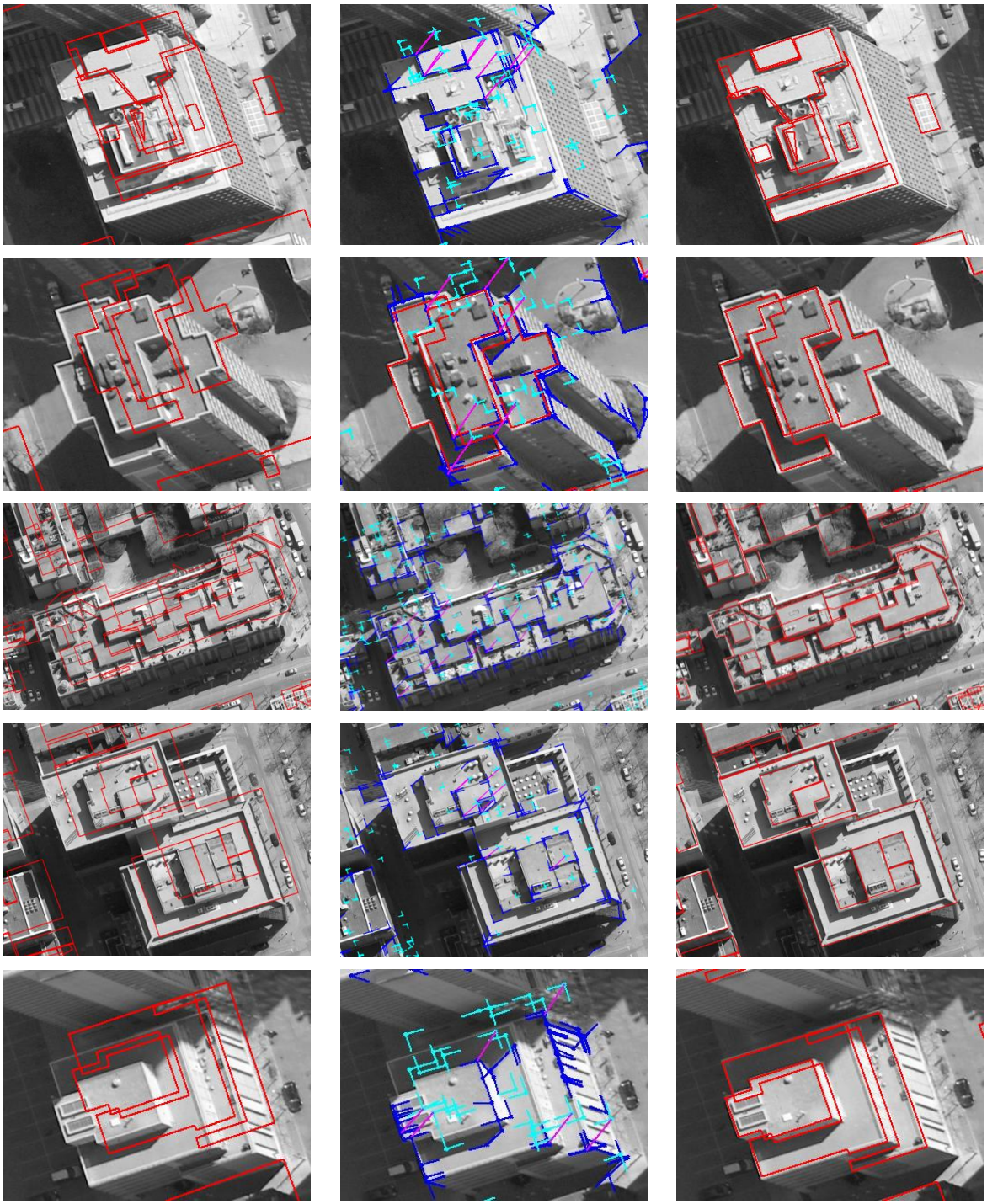
Feature extraction				*Geometric hashing*			
T_d	T_θ	T_{homo}	T_{hetero}	T_s	T_p	T_c	T_m
20 pixel	10º	automatic	automatic	0.98	automatic	50%	0.6

Based on matched edged corner features, EOPs for the image were calculated by applying the least square method based on collinearity equations. For qualitative assessment, the existing models were back-projected to the image with refined EOPs. Each column of Figure 5.11 and Figure 5.12 shows back-projected building models with error-contained EOPs (a), matched edged corner features (b), and back-projected building models with refined EOPs (c). In the figures, boundaries of the existing building models are well matched to building boundaries in the image with refined EOPs.



(a) (b) (c)

Figure 5.11 Manually digitized building models: (a) with error-contained EOPs, (b) matching relations (blue) between edged corner features extracted from the image (blue) and the models (cyan) and (c) with refined EOPs



(a) (b) (c)

Figure 5.12 LiDAR-driven building models (a) with error-contained EOPs, (b) matching relations (blue) between edged corner features extracted from the image (blue) and the models (cyan) and (c) with refined EOPs

In our quantitative evaluation, we assessed the root mean square error (RMSE) of check points back-projected onto the image space using refined EOPs (Table 5.4). When reference building models were used as the existing building models, the results show that the average difference in x and y directions are -0.27 and 0.33 pixels, respectively, with RMSE of ±0.68 and ±0.71 pixels respectively. The results with LiDAR-driven buildings models show that the average differences in x and y directions are -1.03 and 1.93 pixels, with RMSE of ±0.95 and ±0.89 pixels, respectively. Although LiDAR-driven building models are used, the accuracy of the EOPs is less than 2 pixels in image space (approximately 30cm in ground sample distance (GSD)). Considering that the point space (resolution) of the input airborne LiDAR dataset is larger than 0.3m, the refined EOPs provide a greater accuracy for engineering applications.

Table 5.4 Quantitative assessment with check points (unit: pixel)

Error-contained initial EOPs				*Refined EOPs with manually digitized building models*				*Refined EOPs with LiDAR-driven building models*			
Ave.		*RMSE*		*Ave.*		*RMSE*		*Ave.*		*RMSE*	
x	*y*	*x*	*y*	*x*	*y*	*x*	*y*	*x*	*y*	*x*	*y*
20.51	-24.81	±6.64	±8.22	-0.27	0.33	±0.68	±0.71	-1.03	1.93	±0.95	±0.89

In this study, threshold, T_m has an effect on the accuracy of the EOPs. In order to evaluate the effect of T_m, we estimated the matched number of edged corner features and calculated the average error and the RMSE of the check points with different values of T_m. As shown in Table 5.5, the number of matched features is inversely proportional to the value of T_m, regardless of which existing building models are used. However, the effect of T_m on the accuracy is not the same for both building models. We observed T_m affects the

matching accuracy of digitized building models less than it does for LiDAR-driven building models. Furthermore, the matching accuracy tends to get worse with very low or high T_m values. The latter can be explained by the low number of matched features, giving us insufficient data to accurately adjust the EOPs of the image. In the other case, if a low T_m value is selected, the number of matched features increases, but so does the number of incorrect matches if the building models are inaccurate. Thus, we can observe that LiDAR-driven building models, reconstructed with relatively lower accuracy compared to the manually digitized models, produced more sensitive results in the matching accuracy according to T_m. In contrast, the matching accuracy of the manually digitized building models remains high because of high model accuracy. In summary, a higher accuracy of the building models can lead to a higher EOP accuracy, and the value of T_m should be determined by balancing the ratio of correct matched features, and incorrect matched features.

Table 5.5 Effects for T_m (unit: pixel)

T_m	*Manually digitized building models*					*LiDAR-driven building models*				
	# of matched features	*Ave.*		*RMSE*		*# of matched features*	*Ave.*		*RMSE*	
		x	*y*	*x*	*y*		*x*	*y*	*x*	*y*
0.9	67	0.38	0.78	±0.43	±0.42	9	0.49	-1.93	±7.39	±6.99
0.8	268	0.00	0.84	±0.81	±0.97	98	-1.09	1.22	±1.53	±1.52
0.7	505	-0.20	0.31	±0.95	±1.08	273	-1.58	1.56	±0.68	±0.61
0.6	693	-0.27	0.33	±0.68	±0.71	381	-1.03	1.93	±0.95	±0.89
0.5	766	-0.22	0.21	±0.81	±0.66	438	-0.43	3.26	±2.61	±3.52
0.4	796	0.25	-0.08	±1.06	±0.75	499	1.21	2.15	±3.06	±3.66
0.3	800	0.00	-0.09	±0.88	±0.71	502	1.37	2.19	±3.12	±3.93
0.2	800	0.00	-0.09	±0.88	±0.71	502	1.37	2.19	±3.12	±3.93
0.1	800	0.00	-0.09	±0.88	±0.71	502	1.37	2.19	±3.12	±3.93

In order to evaluate the effect on context feature, we set the weight parameter *w* in the score function (Eq. 5.1) as 1 and 0.5, respectively, and then compared the results. When *w*=1, the score function considers only the unary term without the effect of the contextual term so that the contextual force is ignored. As shown in Table 5.6, the results show that registration with only unary terms causes considerably low accuracy in both cases. In particular, with LiDAR-driven models, the accuracy is heavily affected. These results indicate that the use of context features has a positive effect on resolving the matching ambiguity and thus improving the EOP accuracy by reinforcing contextual force.

Table 5.6 Effect on pair-wise feature (unit: pixel)

	Manually digitized building models					*LiDAR-driven building models*				
	# of matched features	*Ave.*		*RMSE*		*# of matched features*	*Ave.*		*RMSE*	
		x	*y*	*x*	*y*		*x*	*y*	*x*	*y*
Unary term only (w = 1)	542	-0.67	-0.39	±1.56	±1.84	361	5.98	1.17	±7.72	±5.31
Unary term and contextual term (w = 0.5)	693	-0.27	0.33	±0.68	±0.71	381	-1.03	1.93	±0.95	±0.89

We also analyzed various impacts of errors in initial EOPs on the matching accuracy by adding different levels of errors to evaluate our proposed method. Each parameter of the EOPs leads different behaviors from back-projected building models: X_0 and Y_0 parameters are related to the translation of back-projected building models; Z_0 is related to scale; ω_0 and φ_0 cause shape distortion; κ_0 is related to rotation (Figure 5.13). In order to assess effects on translation and scale, errors ranging from 0m to 25m were added to three position parameters (X_0, Y_0, Z_0). To assess the shape distortion and rotation effects,

errors ranging from 0º to 2.5º were added to three rotation parameters ($\omega_0, \varphi_0, \kappa_0$). Figure 5.14 shows the accuracies of the refined EOPs with different level of errors for each EOP parameter. Regardless of errors in the initial EOPs, RMSE of under 2 pixels for manually digitized building models, and RMSE of under 3 pixels for LiDAR-driven building models were achieved. The results indicate that the accuracy of the refined EOPs was less affected by the amount of initial EOPs errors. This is due to the fact that the EOPs converge to the optimum solution iteratively.

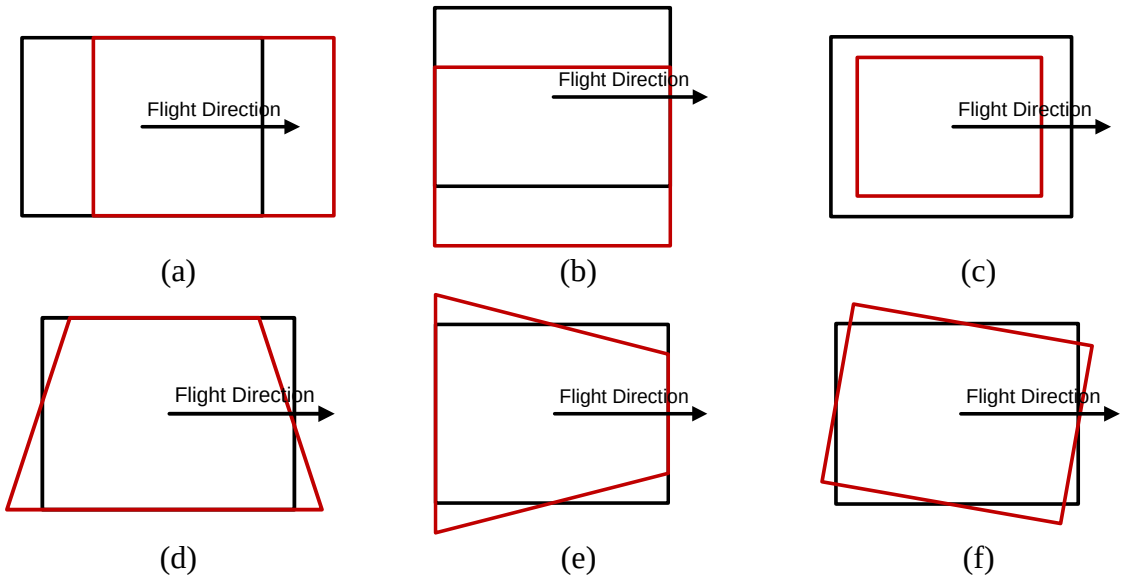
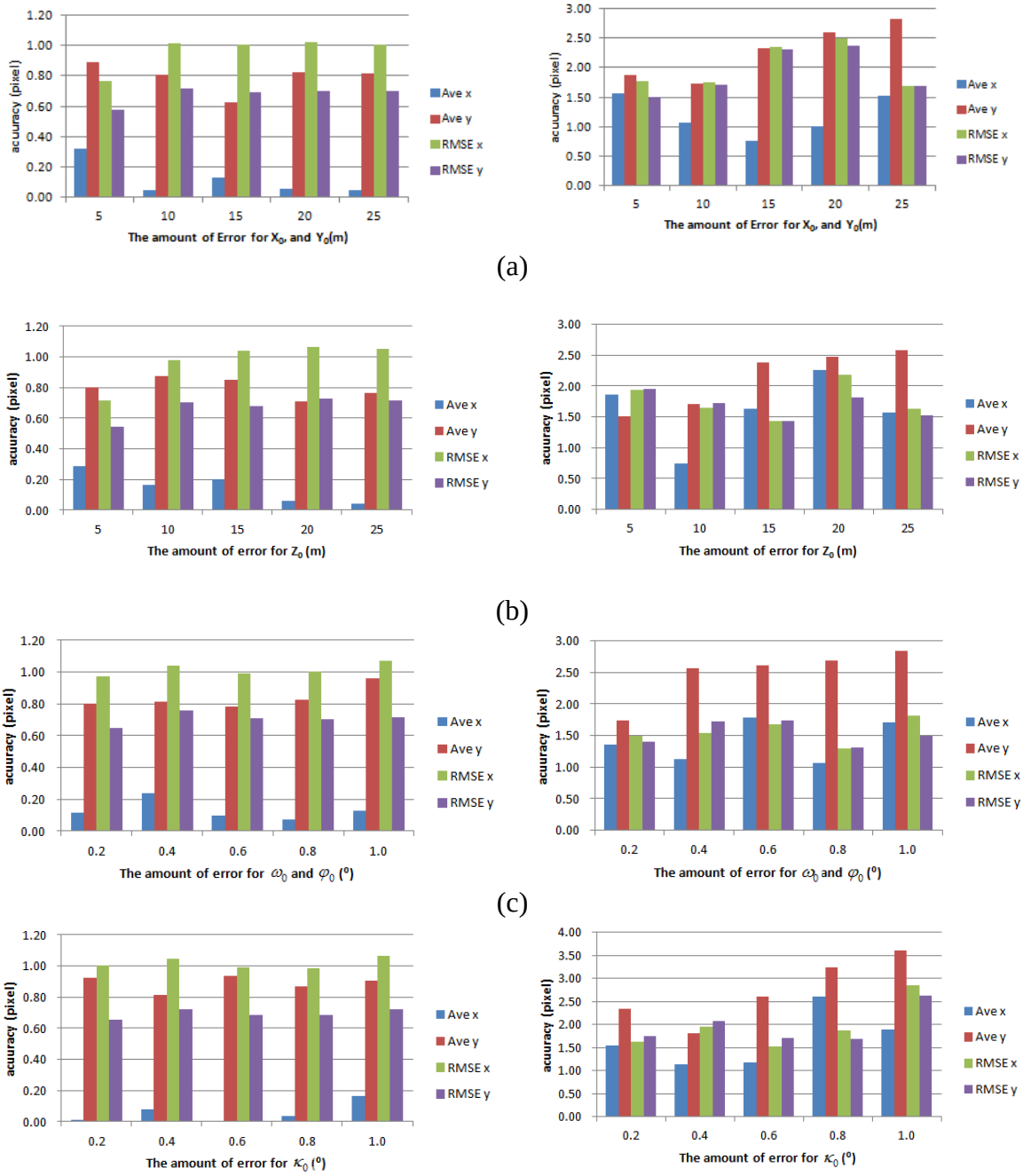


Figure 5.13 The behaviors caused by errors for EOPs: (a) X_0, (b) Y_0, (c) Z_0, (d) ω_0, (e) φ_0, and (f) κ_0.

With manually digitized building models

With LiDAR-driven building models



(a)

(b)

(c)

(d)

Figure 5.14 The impacts of errors in initial EOPs: (a) X_0 and Y_0, (b) Z_0, (c) ω_0 and φ_0, and (d) κ_0

In order to evaluate the robustness of the proposed registration method, the algorithm was applied to the Vaihigen dataset. A total of 31,072 edged corner features from the image and 11,812 edged corner features from the existing building models were extracted using the parameters set in Table 5.3. A total of 379 edged corner features were matched by the CGH method where T_m was heuristically set as 0.7, and other parameters were set by Table 5.3. The results of the extracted and matched features are summarized in Table 5.7. Sixteen check points were evaluated for error-contained EOPs and refined EOPs. The accuracies of the check points with refined EOPs show that the average difference for x and y directions are 0.67 and 0.97 pixels with RMSE of ±1.25 and ±1.49 pixels respectively (Table 5.8). The results suggest that the proposed registration method can achieve accurate and robust matching results even though building models with different error types were used for the registration of a single image.

Table 5.7 Extracted features and matched featrues (Vaihigen dataset)

	Image			*Existing building models*
	Straight lines	*Intersections*	*Corners*	
# of extracted features	276,109	181,200	31,072	11,812
# of matched features	-	-	-	379

Table 5.8 Quantitative assessment with check points (Vaihigen dataset, unit: pixel)

With error-contained initial EOP				*With refined EOPs*							
				Unary term only (w=1)				*Unary term and contextual term (w=0.5)*			
Ave.		*RMSE*		*Ave.*		*RMSE*		*Ave.*		*RMSE*	
x	*y*	*x*	*y*	*x*	*y*	*x*	*y*	*x*	*y*	*x*	*y*
22.92	-19.06	±2.28	±3.90	-1.32	-0.35	±2.45	±2.93	0.67	0.91	±1.25	±1.49

5.4 Summary

In this chapter, we proposed a new model-to-image registration method which can align a single image with 3D building models. Edged corner features, represented by a corner and its associated edges, and context features are proposed as the matching features. Edged corner features are extracted from the image by calculating the intersection of two neighboring straight lines and verified using geometric and radiometric properties. For similarity measurement, and matching, the CGH method was proposed to compensate for the limitations of the standard geometric hashing method. The qualitative assessment showed that the boundaries of the existing building models, back-projected by refined EOPs, are well aligned with boundary lines from the image. Meanwhile, the quantitative assessment showed that both manually digitized building models, and LiDAR-driven building models can be used to evaluate the EOPs of a single image with acceptable and reliable accuracy. More specifically, experimental results are summarized as follows:

- The quality of building models directly affects the accuracy of EOPs. When manually digitized building models were used, the proposed registration method accurately and reliably achieved the EOPs regardless of threshold and assumed error. However, if building models contain more modeling errors, the accuracy of EOPs is reduced, which are more susceptible to threshold, and assumed errors.
- Contextual features employed in geometric hashing enhances matching performance. This is because contextual values provide information about the relation between edged corner features, characterizing geometric properties of individual roof polygon. In particular, the use of context features, which provide

global information of building models, that is at larger scale (object-level) than at using single corners only (point-level), plays a significant role in our enhanced geometric hashing method, and making our matching performance more robust to errors involved in building models used.

- The proposed method can iteratively recover the EOPs of a single image in spite of considerable error in their initial values, which exceed error amounts permitted in commercial aerial image acquisition.

As future works, we will extend the proposed method to arbitrarily acquired images (e.g., UAV image and security camera image).

Chapter 6

Sequential Modeling of Building Rooftop by Integrating Airborne LiDAR Data and Optical Imagery

---

In this chapter, we present a novel fusion method to sequentially reconstruct building rooftop models over time by integrating multi-sensor data. Specifically, the proposed method aims to refine the quality of rooftop models generated by an existing algorithm, in our case the LiDAR-driven method explained in Chapter 4, by integrating modeling cues extracted by an airborne imagery. This fusion method is designed to compensate the limitations caused by respective sensor and thus improve three types of modeling errors (shape deformation, boundary displacement, and orientation error) that are often involved in rooftop building models. An ultimate goal of this research is to develop a research platform for continuously refining the quality of city-scale rooftop models from multi-sensory data acquired over time. In our fusion scheme, a set of new model hypotheses are generated by connecting the lines derived from an existing rooftop model to the lines that are extracted from an airborne imagery around the existing rooftop model. This modeling cue integration process is developed for progressively rectifying geometric errors based on Hypothesize and Test (HAT) optimization using Minimum Description Length (MDL). A stochastic method, Markov Chain Monte Carlo (MCMC), coupled with Simulated Annealing (SA), is employed to generate model hypotheses and perform a global optimization for finding the best solution. In particular, *a prior* knowledge derived from an image is used to propose a new move in Markov Chain. The performance of the proposed fusion method is evaluated by s newly proposed error evaluation matrix. The results show that our proposed method can further refine the three types of modeling errors of LiDAR-driven building rooftop models.

6.1 Introduction

As one of the most prominent objects comprising the virtual city models, automatically reconstructing 3D geometric models of building rooftops has been considered as a central research topic in photogrammetry and computer vision communities for more than two decades. Recently, a few of researchers have reported their success on the automation of the detailed rooftop models (Haala and Kada, 2010). Rottensteiner et al. (2014) reported the latest benchmark experiments of the state-of-the art rooftop modeling algorithms. However, they concluded that current status of the rooftop modeling algorithms is still far from achieving an error-free model generation in an automated manner. In particular, most of the rooftop modeling methods focus on the reconstruction of the roof structure from the information captured by a single data acquisition mode at one particular time epoch. Not many research works have addressed the research problems to reconstruct the rooftop models using multiple sensors over time. In continuously changing city environments, the appearance of the rooftops can be differently viewed by different remote sensors, or with different resolutions of the same sensor, or over time due to the changes of their physical structures. Thus, accurately updating the rooftops is essential to timely provision of accurate building models and to maintaining up-to-date status of a city. This chapter aims to develop a research framework, called “continuous city modeling”, which sequentially refines the quality of the rooftop models by sequentially fusing modeling cues extracted from two different airborne sensors. In this framework, we assume that “existing” rooftop models are reconstructed from airborne LiDAR data, which are sequentially updated by

integrating the information retrieved from the existing model with new modeling cues extracted from an airborne optical imagery.

Many building modeling algorithms using LiDAR data have provided promising results (Rottensteiner et al., 2014). However, LiDAR-driven building models suffer from certain modeling errors caused by inherent characteristics of LiDAR data. Through a thorough analysis, we identified three types of modeling errors, which are often observed in LiDAR-driven building models as follows:

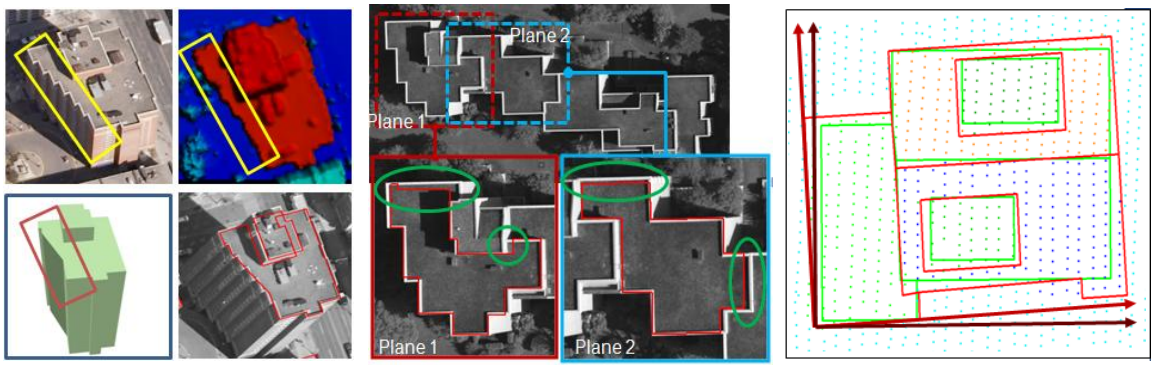
- ***Shape deformation (over- or under-simplified model)***: A shape deformation is defined as an overall difference in shape of the rooftop generated by an algorithm from the reference model. The shape deformation can be caused by various reasons such as scene complexity, data characteristics (resolution, signal-to-noise ratio, occlusion, and incomplete cue extraction), and characteristics of the rooftop modeling algorithms used (model-driven approach or data-driven approach). In general, the building models generated by model-driven approaches tend to be over-simplified due to a limited number of model primitives that are applicable from pre-specified primitive library. The primitive library consists of parameterized simple building shapes such as flat, gable, and hip roof. Detailed local shapes that are not fit by any of the pre-specified model primitives tend to be missed. For instance, the simple primitives cannot delineate details of building rooftops with protruded structure or step-shaped structure even though a combination of primitives is used. Also, the data-driven approach can produce the under-simplified modeling errors when it is over-fitted to error-contained observation, and it also can

cause a loss of small but important parts of building structures as shown in Figure 6.1(a).

- ***Boundary displacement (or shrinkage)***: A boundary displacement is caused by a discrete point distributions of LiDAR, which errors usually do not affect the overall shape of the model, but produce certain errors in model's geometry. In general, building boundary or boundary of roof planes is reconstructed from object boundary points detected from LiDAR data. However, reconstructing exact object boundaries are difficult due to irregular point gaps caused by LiDAR sensor; the irregular gaps ("spacing") shown between adjacent points are produced due to the discrete nature of LiDAR's scanning mechanism or occlusions (or absorptions) casted from illuminated materials. Thus, the building boundary generated from LiDAR data contains the displacement error as much as a sum of the point gap and position error of LiDAR data. The boundary displacement tends to occur toward the inside of the building. As a consequence of this effect, the boundary displacements result in a shrinkage of the reconstructed building boundary as shown in Figure 6.1(b).
- ***Orientation error***: The main orientation of a building model is often used as a modeling cue to regularize noise included in the boundaries of building model. A strong regularization process fit the orientation of noisy model edges to the building's main orientation. The building orientation is determined by analyzing angle distributions of initial rooftop boundaries which are generated by tracking irregularly distributed boundary points. Unlike optical imagery, the building

orientation errors caused by LiDAR data are not uniform across buildings, but are subject to a relative angle between the scanner's flying direction and the orientation of a building of interest. That is, more uniform distribution is expected as the relative angle gets smaller, and vice versa. Due to this reason, the orientation computed by LiDAR data is not accurate and may accidently contain a large amount of error. Thus, enforcing incorrectly determined orientation in regularization process leads to an incorrectly oriented building model as shown in Figure 6.1(c).

These modeling errors observed in LiDAR-driven building models are closely related to the inherent characteristics of LiDAR data. The modeling errors can be rectified by integrating image data which have complementary properties.



(a) (b) (c)

Figure 6.1 Modeling errors observed in LiDAR-driven building rooftop models: (a) shape deformation, (b) boundary displacement, and (c) orientation error.

We propose a sequential fusion method to refine modeling errors contained in LiDAR-driven building models by integrating them with modeling cues extracted from an

airborne imagery. Main contributions of the proposed method can be summarized as follow. First, existing building models are reused and updated in the proposed sequential fusion framework. Whenever a new image is acquired, the existing building models can be effectively updated in an automated manner. Secondly, the proposed method progressively rectifies geometrical errors based on HAT optimization using MDL. The MDL is designed to favor regularized building model by balancing model closeness and model complexity. Thirdly, the method provides a novel concept of "topological lines" that are virtually generated using implicitly derived model regularities (orientation distribution of rooftop model lines) in order to integrate modeling cues extracted from two different information sources (i.e., LiDAR-driven model and image-driven modeling cues). Lastly, we propose a novel kernel process to generate rooftop model hypotheses based on MCMC coupled with SA for determining an optimal rooftop model. Three different types of proposition kernels to govern how to generate building hypotheses are integrated into the MCMC framework.

6.2 Sequential Building Rooftop Modeling

We propose a sequential fusion algorithm to refine an existing rooftop model derived from LiDAR data (L-Model) by integrating it with image features (I-Lines and I-Corners) derived from a single airborne image. Figure 6.2 shows an overall workflow of the proposed method. A LiDAR-driven building rooftop model, which was generated by the method proposed in Chapter 4, is required as an input vector (initial building rooftop model to be refined) of the sequential modeling chain. After extracting image features from a single image, relationships between L-Model and image features are established by

introducing topological lines. The initial rooftop model is gradually refined by integrating image features in MCMC framework. The MDL, which balances model closeness and model complexity favoring regularized building rooftop model, is used to select the optimal model through stochastically competing model candidates.

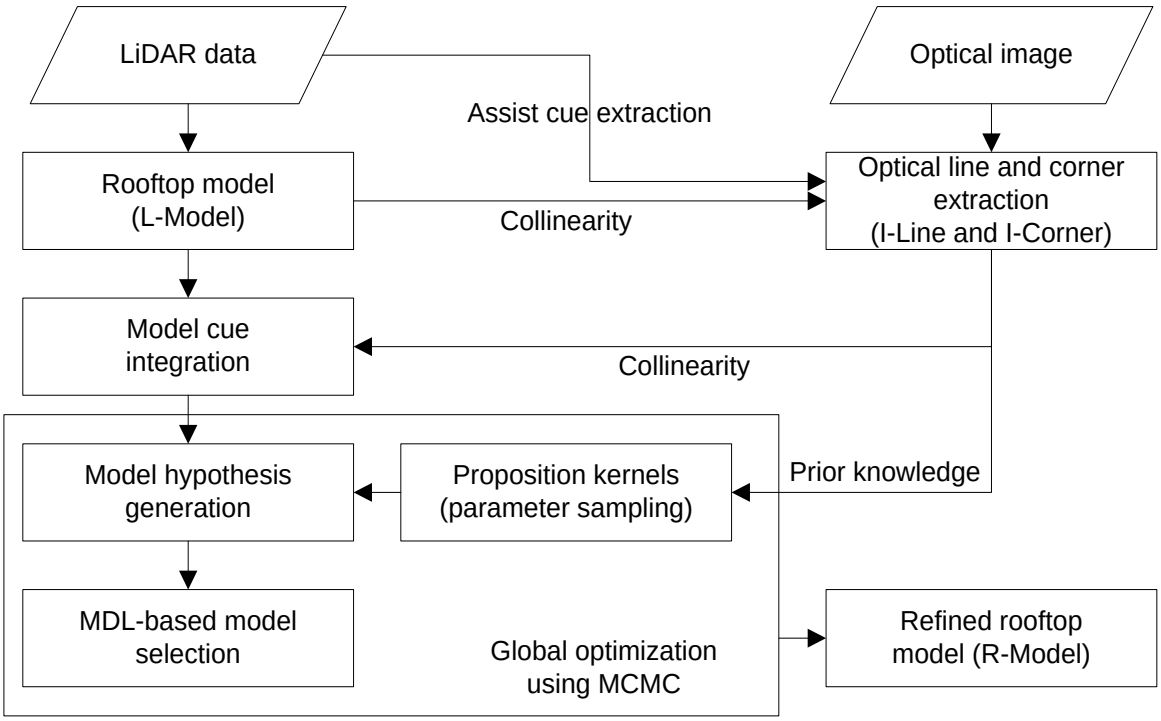


Figure 6.2 Flowchart for the proposed refinement algorithm

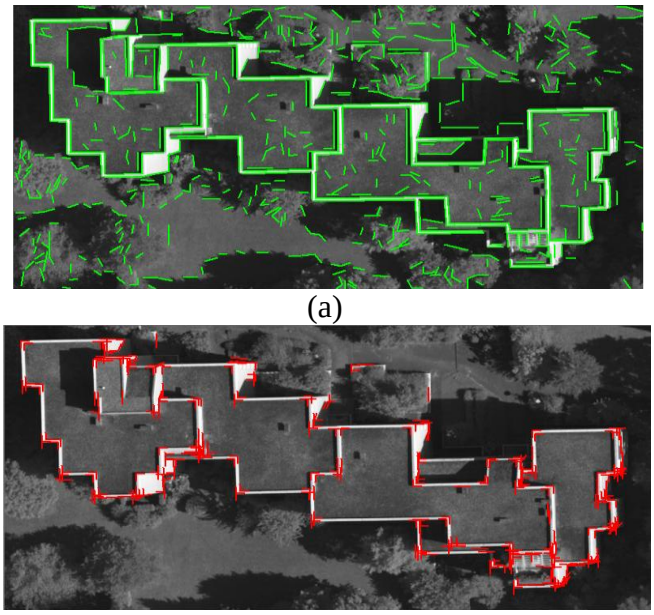
6.2.1 Feature Extraction from Optical Image

As a man-made object that usually contains a certain amount of geometric regularity, the shape of building rooftops can be well described by lines and corners. Images are one of the most appropriate data sources for acquiring the geometrically accurate lines and corners. Thus, we use lines and corners extracted from a single image to rectify modeling

errors of LiDAR-driven building models. In section 6.2.1.1, we extract modeling cues (lines and corners) from a single image. The extracted 2D modeling cues in image space are transformed into 3D object space using collinearity equation (section 6.2.1.2). Then, the 3D modeling cues are quantized with respect to orientation and distance to the origin (section 6.2.1.3).

6.2.1.1 Modeling Cue Extraction

From a single image, straight lines are extracted using Kovesi's algorithm (Kovesi, 2011) that relies on the calculation of phase congruency to localize and link edges (Figure 6.3(a)). Also, we adopted an algorithm proposed by Chatat et al. (1999) to extract corners. The corner detector is based on the analysis of local anisotropism and identifies corners as points with strong gradient without being oriented in a single dominant direction. The advantage of the adopted corner detector is its ability to detect true location of a corner and orientations of its arms (Figure 6.3(b)). The information of corners and their arms provide the local structure information of building shapes (“high-level” primitive geometric elements), facilitating a high-level interpretation of building structures. After extracting corners, each corner is assigned to corresponding lines through a greedy search based on its proximity (20 pixels) and direction. The extracted modeling cues are used as a prior knowledge for generating rooftop model hypotheses.



(a) (b)

Figure 6.3 Modeling cue extraction: (a) straight lines and (b) corners and their arms

6.2.1.2 Transformation between Image Space and Object Space

A transformation between image space and object space is an essential process in establishing a mapping relationship between 3D building models and 2D image features. Given interior orientation parameters (IOPs) and exterior orientation parameters (EOPs) of the image, the well-known collinearity condition between image space and object space is established as follows:

$$x_a = x_0 - f\left[\frac{m_{11}(X_A - X_L) + m_{12}(Y_A - Y_L) + m_{13}(Z_A - Z_L)}{m_{31}(X_A - X_L) + m_{32}(Y_A - Y_L) + m_{33}(Z_A - Z_L)}\right]$$

$$y_a = y_0 - f\left[\frac{m_{21}(X_A - X_L) + m_{22}(Y_A - Y_L) + m_{23}(Z_A - Z_L)}{m_{31}(X_A - X_L) + m_{32}(Y_A - Y_L) + m_{33}(Z_A - Z_L)}\right] \quad (6.1)$$

where x_a and y_a are the image space coordinates of image point a; X_A, Y_A, Z_A are object space coordinates of point A; X_L, Y_L, Z_L are object space coordinates of the exposure station; f is the camera focal length; x_0 and y_0 are the coordinates of the principal point; and the m's are functions of three rotation angles.

A transformation from object space in 3D to image space in 2D is straightforward with given IOPs and EOPs. However, its inverse conversion (from 2D to 3D) is known as an ill-posed problem due to the missing one dimension. One possible solution to address this problem is to use stereo images or multiple images. However, in our research framework, our method is limited to a single image due to constraints of multiple-view data availability. Thus, we use the height information of 3D building models in order to recover the missing third dimension. The process starts by back-projecting L-Model (LiDAR-driven rooftop model) and its associated LiDAR points with attributes including labels (building, non-building) and plane segmentation IDs into the image space (Figure 6.4(1)). I-Lines (line extracted from the imagery) corresponding to L-Model are determined using a proximity criterion; A I-Line is assigned to a line of L-Model as a conjugate line pair if the I-Line is found within a searching space (minimum bounding box) generated from the L-Model line projected onto the image space (Figure 6.4(2)). Then, 3D coordinates of I-Lines are calculated by projecting 2D I-Lines onto their corresponding roof planes (L-Model) using the collinearity equation in Eq. (6.1); starting and ending points of a I-Line is transformed following co-linearity rays, each of which is intersected by the corresponding roof plane in order to calculate the parameters of 3D line in the object space (Figure 6.4(3)). In a similar way, 2D coordinates of I-Corners (i.e., corners and their

arms extracted from the imagery) are computed into 3D object space. It is noted that an I-Line can be shared by multiple roof planes. In this case, multiple 3D lines are calculated in the object space for a single I-Line, all of which will be considered as the modeling cues for generating the rooftop hypotheses.

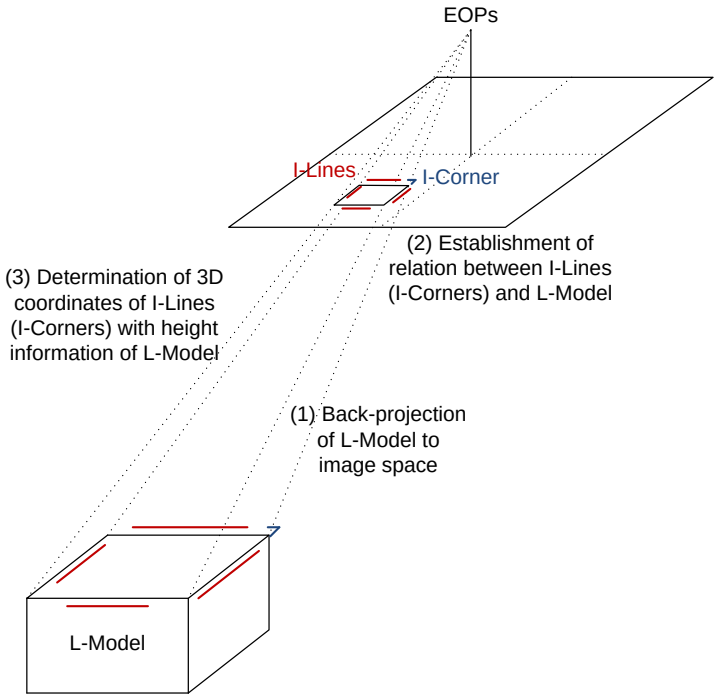


Figure 6.4. Determination of 3D coordinates of I-Lines (or I-Corners)

6.2.1.3 Quantization

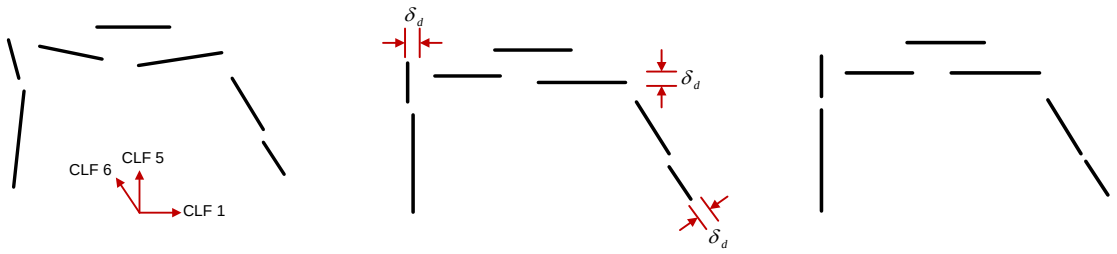
Once 2D I-Lines are converted to 3D I-Lines, the 3D I-Lines are projected into the x-y plane and formulated as:

$$L(\theta, \rho): \rho = x\cos\theta + y\sin\theta \tag{6.2}$$

where θ and ρ are a slope of line and distance to the origin, respectively, while keeping the z value information of each line. Quantization process is conducted to slope θ and distance to the origin ρ in order.

For quantization of θ, we assume that main directions of building models can be represented by representative directions of Compass Line Filter (CLF) proposed by Sohn et al. (2008). As shown in Figure 4.7, the CLF is determined by the whole set of eight filtering lines with different slopes $\{\theta_i : i = 1, \dots, 8\}$ that is equally separated in steps of 22.5º. The representative angle for each slope is calculated by averaging angles weighted by lengths of I-Lines belonging to each slope.

In addition to the orientation quantization, we quantize the distance of I-Lines to the origin ρ. In line extraction process, ideal edges might be broken into two or more small segments that are not connected to each other. The fragmented line segments do not exist on a specific line due to line extraction error and image resolution. Also, a direct use of I-Lines, which are very close to each other, complicates the final rooftop model in an undesired way. In order to address this problem, I-Lines are quantized with a certain threshold (δ_d=0.2m) based on distance to the origin, and then adjusted by averaging distances to the origin of corresponding lines. It is noted that the quantization process is respectively conducted on I-Lines belonging to each CLF. Figure 6.5 shows I-Lines' quantization for angle and distance to the origin.



(a) (b) (c)

Figure 6.5 Quantization: (a) I-Lines in object space, (b) quantization for angle based on CLF and (c) quantization for distance from the origin

6.2.2 Modeling Cue Integration

After all the image modeling cues (I-Lines and I-Corners) are extracted (section 6.1.2.1) and transformed in the object space (section 6.1.2.2), the integration of I-Lines and I-Corners with existing rooftop models (L-Models) is conducted in the object space. This process establishes spatial relationships between modeling cues derived from two different data sources.

Suppose that L-Model, I-Lines and I-Corners are denoted as a set of model lines $L^M = \{L_i^M; i = 0, \dots, l\}$, image lines $L^I = \{L_j^I; j = 0, \dots, m\}$, and image corners $C^I = \{C_k^I; k = 0, \dots, n\}$ where l, m and n represent the number of model lines, image lines and image corners, respectively (Figure 6.6 (a)). The first step of the cue integration is to identify their spatial relations by investigating a spatial proximity and geometric configuration among L^I, C^I and L^M. Given a model line L_i^M, a set of image cues are determined as its conjugate cue pairs if they satisfy following spatial cue relations:

- *Image Cues to Model Line Relations*: As described in section 6.1.2.1, an image cue (i.e., L_j^I or C_k^I) is assigned its membership to a model line L_i^M if a spatial proximity measured between the image cues and a model line in the object space is less than a

pre-specified threshold (2m). A I-Line can belong to multiple model lines if it spans multiple model lines (1 to *N* relation);

- *Image Line to Corner Relations*: The image cues meeting the previous "image cues to model line relation are further filtered in order to determine an "image line to corner pair". A image line L_j^I is paired with an image corer C_k^I if the orientation of L_j^I is co-aligned with the one of orientations of the edged arms C_k^I. Note that C_k^I is excluded for the hypothesis generation if it is not paired with any image line; while L_j^I is accepted even though its corner pair does not exist.

Once all the membership of the image cues to each model line are found, the next cue integration process is to "physically" represent the spatial relations among the paired cues. This topological cue relation is conducted by generating "virtual" lines to connect the paired cues (Figure 6.6). When a model line L_i^M is given and its paired image cues, L_j^I and C_k^I are found, we define two different types of virtual lines as follows:

- *Guide Line (GL)*: Two guide lines, $GL_1^{(L_i^M,s)}$ and $GL_2^{(L_i^M,e)}$, are defined as the infinite lines which line parameters are identical to L_{i-1}^M and L_{i+1}^M (L_i^M's neighboring lines), and are generated from starting and ending points of L_i^M respectively (Figure 6.6 (a)). The starting (s) and ending point (e) of L_i^M is defined through guide points, $GP_1^{(L_i^M,s)}$ and $GP_2^{(L_i^M,e)}$;
- *Topological Line (TL)*: A topological line TL_k is a virtual line to establish spatial relations between image cues and L_i^M. An anchor point *AP* is defined as a starting

point of newly generated topological line. $AP_1^{(L_j^I,s)}$ and $AP_2^{(L_j^I,e)}$ are the anchor points defined from the starting (s) and ending point (e) of an image line L_j^I, while $AP_1^{(C_k^I)}$ is defined from the image corner point C_k^I. Starting from these anchor points, a set of infinite lines are generated with CLF line slope angles determined (see section 6.2.1.3). A line segment generated by intersecting the infinite lines with L_i^M or guide lines ($GL_1^{(L_i^M,s)}$ and $GL_2^{(L_i^M,s)}$) is considered as the topological line to connect the image cues to its paired model line ((Figure 6.6 (b), (c) and (d)).

The generation of topological lines fully depend on a spatial configuration among modeling cues, L^M, L^I and C^I. Three different types of topological lines are defined as follows:

- *Type I*: This topological lines, $TL_k^{(L_j^I,s,)}$ and $TL_k^{(L_j^I,e)}$, are generated for connecting lines between an image line L_j^I and guide lines, $GL_1^{(L_i^M,s)}$ and $GL_2^{(L_i^M,s)}$, that are created from two anchor points of a model line L_i^M (Figure 6.6 (b)). The direction of topological lines is the same as one of L_j^I.
- *Type II*: Type II topological lines are defined for connecting an image line L_j^I to a modeling line L_i^M (Figure 6.6 (c)). A set of Type II topological lines are generated from two anchor points, $AP_1^{(L_j^I,s)}$ and $AP_2^{(L_j^I,e)}$ of an image line L_j^I. Each topological lines are generated using one of the representative angles of CLF. The angle is

determined by quantizing the orientation distribution of lines extracted from a targeted building region captured in an airborne imagery (see section 6.2.1.3).

- *Type III*: Type III topological lines are generated for connecting an image corner C_k^I to its paired modeling line L_i^M. In Type III topological line, the corner point C_k^I serves as an anchor point $AP_1^{(C_k^I)}$and its arms are used as *a priori* knowledge to generate topological line; each topological line is generated following one of orientations of the edged arms C_k^I (Figure 6.6 (d)).

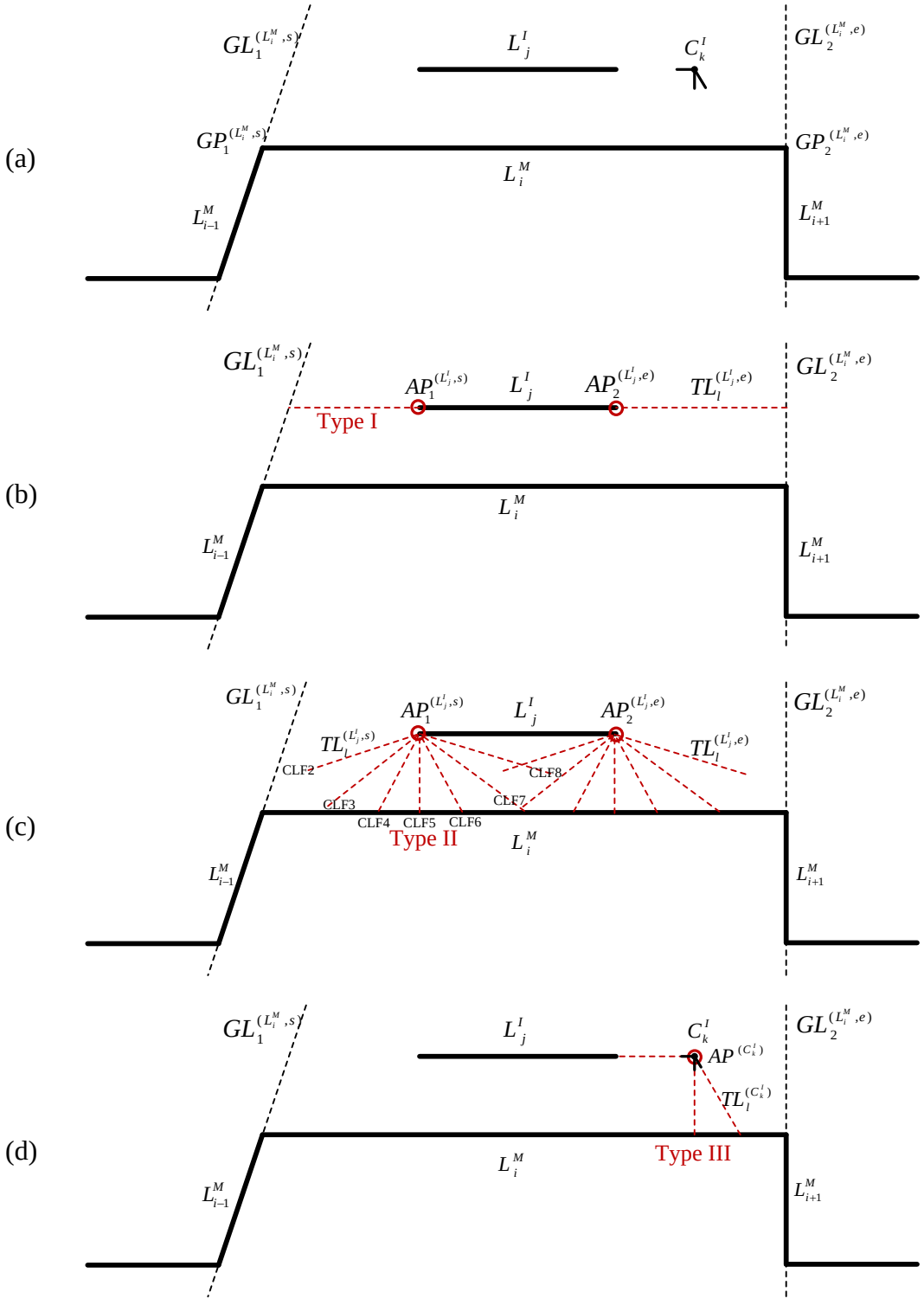


Figure 6.6 Topological lines (red) (Type I: between guidelines and I-Line (b), Type II: between I-Line and L-Model (c), and Type III: prior-guided (d))

6.2.3 Model Hypothesis Generation

In the previous section, multiple topological relations integrating image features with existing model lines were established. The model hypothesis generation is a process to generate probable models reflecting the contribution of image features to improve existing models. Given L-Model lines and I-Lines, I-Corners, guide lines and topological lines, hypotheses are generated by sampling possible combinations (see section 6.2.5.2). A hypothesis with one of possible combinations is generated by finding intersection points between topological line and I-Line, between topological line and L-Model line and/or between guide line and I-Line. Figure 6.7 shows some examples generated with different combination sets for Figure 6.6(a).

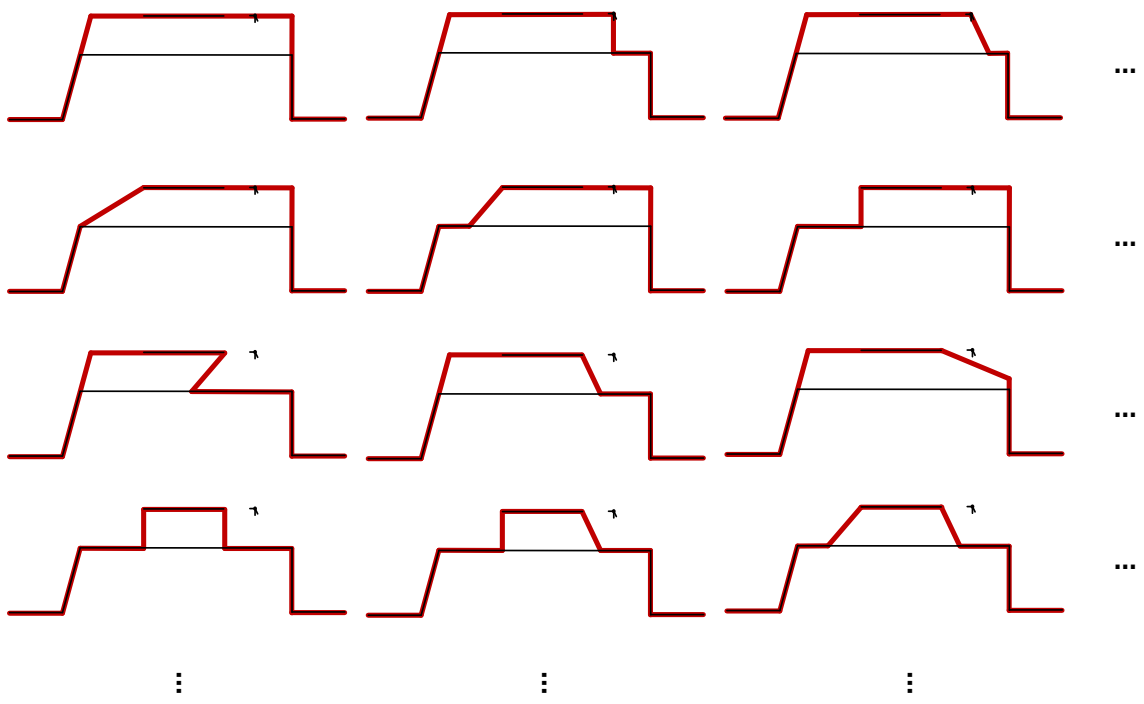


Figure 6.7 Examples of possible hypotheses (red) with respect to a given configuration (black)

6.2.4 MDL-based Model Formulation

In a discriminative modeling approach, the specific model that best explains the given observation is usually not known a priori. Instead, a decision process, called model selection, is adopted for selecting the optimal model through stochastically competing model candidates. Rissanen (1978) introduced MDL for inductive inference that provides a generic solution to the model selection problem (Grünwald, 2005). The MDL provides a flexibility to encode a bias term, which allows us to protect against over-fitting of the model of interest to limited observations. This bias is estimated by measuring the "model complexity", which varies depending on the regularity (similar or repetitive patterns) hidden in observations. Weidner and Förstner (1995) posed building outline delineation as the model selection problem using MDL. Sohn et al. (2012) extended it to rooftop models comprising multiple planes by implicitly generating model hypotheses. In this study, we adopted Sohn et al. (2012)'s MDL framework, whose objective function is described below:

$$DL = \lambda \mathcal{L}(D|H) + (1 - \lambda)\mathcal{L}(H) \tag{6.3}$$

where H and D indicate a building model hypothesis and its boundary associated laser points, respectively. λ is a weight value for balancing between the model closeness and the model complexity. In Eq. (6.3), the model closeness term represents bits encoding the goodness-of-fit between the hypothesis and its associated laser points, while the model complexity term represents bits evaluating the hypothesized model's complexity.

6.2.4.1 Closeness Term

Assuming that the error model between data and hypothesis follows a Gaussian distribution, the closeness term can be rewritten as:

$$\mathcal{L}(D|H) = \frac{\Omega}{2\,ln\,2} \tag{6.4}$$

where Ω is the sum of the squared residuals between a model (H) and a set of observations (D), that is $[D-H]^{\mathrm{T}}[D-H]$. In this research, observations for the model consist of boundary points belonging to the target plane (target points) and boundary points belonging to the non-target planes (non-target points). Ω is a measure described as the shortest length between a point and its corresponding model line using Euclidean distance. However, in order to favor a model hypothesis that maximizes the planar homogeneity, we add a penalized distance for points which hinder planar homogeneity (Figure 6.8). Points to be penalized are determined by checking whether the point is inside or outside of the target polygon. Target points should be inside of the target polygon while non-target points should be outside of the target polygon. For points which do not satisfy this condition, a penalized distance criterion is applied. The amount of penalized distance is the minimum distance between the point and the terminal nodes of its corresponding model line, but cannot exceed a certain given threshold (δ_p=2m in this study). Note that we take the accuracy of LiDAR points (0.2m) into account as a tolerance when measuring homogeneity with buffer.

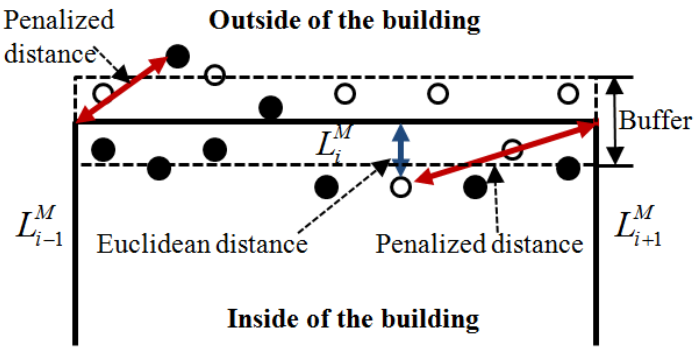


Figure 6.8 Penalized distance (dot: target points and circle: non-target points)

6.2.4.2 Complexity Term

The complexity term is designed to estimate the degree of geometric regularity of the hypothesized model. The geometric regularity is measured depending on 1) polyline simplicity (the number of vertices), 2) directional patterns (the number of different directions), and 3) penalty for inner angles (orthogonality and presence of acute angles). Based on information theory, each term is encoded in bits as follows:

$$\mathcal{L}(H) = W_v N_v' \log_2 N_v + W_d N_d' \log_2 N_d + W_{\angle\theta} Q_{\angle\theta}' \log_2 Q_{\angle\theta} \quad (6.5)$$

where the subscript v, d, $\angle\theta$ indicate vertex, line direction, and inner angle; $(N_{v,}N_d, Q_{\angle\theta})$ indicate the number of vertices, the number of identical line directions, and penalty value for inner angle. $(N_{v,}N_d, Q_{\angle\theta})$ are estimated from the initial model determined at pervious iteration; $(N_v', N_d', Q_{\angle\theta}')$ are computed from model hypotheses that are locally generated as described in section 6.2.3; $(W_v, W_d, W_{\angle\theta})$ are weight values for each factor. Note that

grouping the inner angle and thus estimating $Q_{\angle\theta}$ and $Q'_{\angle\theta}$ was conducted using heuristically determined threshold. Refer to Chapter 4 (section 4.3.1) for details.

6.2.5 Global Optimization

Let $H = \{h_1, \dots, h_m\}$ denote a set of all possible roof hypotheses. The optimal model H^* is selected through the direct comparison of *DL* values for all model candidates, in which H^* has the minimum *DL* as follows:

$$\mathrm{H}^* = arg\, min_{H\in\Phi}\{\lambda\mathcal{L}(D|H) + (1-\lambda)\mathcal{L}(H)\} \tag{6.6}$$

where Φ is the configuration space in which configuration *H* resides. However, as stated in section 6.2.3, it is not possible to explore a large hypothetical space to find the optimal solution. For example, the number of possible hypotheses at a certain configuration, even when new configurations are generated by a model line and an image line using Type II topological lines, is 64 (8 directions for $\mathrm{AP}_1^{(\mathrm{L}_1^{\mathrm{I}},s)}$ and 8 directions for $\mathrm{AP}_2^{(\mathrm{L}_1^{\mathrm{I}},e)}$). Considering all model lines, image lines, and corners, it is too computationally expensive to compare all solutions. Thus, we employ a stochastic method, Monte Carlo Markov Chain (MCMC) coupled with Simulated Annealing (SA), to find a global optimization in the large hypothetical space. Compared to naive hypothesis generation (Chapter 4), the stochastic hypothesis generation can reduce computing time because it does not compare all possible hypotheses. Furthermore, the optimal model H^* is not sensitive to an initial building rooftop configuration because the optimization process is conducted by random sampling.

In section 6.2.5.1, we introduce MCMC for solving our optimization problem. Three types of proposition kernels are proposed to deal with transitions from the current configuration to a new configuration in section 6.2.5.2.

6.2.5.1 Global Optimization in MCMC Framework

MCMC, first introduced by Metropolis et al. (1953), is a method for obtaining a sequence of random samples from a probability distribution for which direct sampling is difficult. As the name suggests, MCMC is a combination of concepts of Monte Carlo sampling and Markov Chain. The Monte Carlo sampling is a method to generate a set of samples from a target density to compute integrals. Markov Chain refers to a sequence of random variables generated by a Markov process whose transition probabilities between different values in the state space depend only on the random variable's current state, $\mathrm{P}(x_t|x_{t-1}, \dots, x_0) = P(x_t|x_{t-1})$. By combining these two concepts, the MCMC sampler can effectively explore a configuration space and approximates a target density. MCMC has been applied to many applications such as Bayesian inference and learning, statistical mechanics, optimization, and penalized likelihood model selection (Andrieu et al., 2003).

In this study, MCMC coupled with SA is used to solve our optimization problem by simulating a discrete Markov Chain $(\mathrm{X_t})$, $\mathrm{t} \in \mathbb{N}$ on the configuration space Φ, which converges towards an invariant measure specified by the *DL*. The MCMC sampler performs transitions from one state to another, which can be managed by kernels Q_m (see section 6.2.5.2). If a rooftop configuration h transits to h' according to proposition kernels

$Q_m(h \to h')$, the move between these configurations is accepted with the following probability as suggested in Zhang (2015):

$$A(h, h') = min\left\{1, \frac{DL^{-1}(h')Q_m(h|h')}{DL^{-1}(h)Q_m(h'|h)}\right\} \tag{6.7}$$

If $A(h, h') = 1$, the new configuration h' is added to the Markov Chain. Otherwise, it remains at *h*.

A simulated annealing is then embedded in the MCMC to find the optimal configuration with the minimum global DL value. To perform the simulated annealing, the description length *DL* is replaced by $DL_{T_t} = DL(h)^{1/T_t}$, where T_t is the temperature parameter, which tends to zero as *t* approaches ∞. The acceptance rate is as follows:

$$A(h, h') = min\left\{1, \frac{\left(DL^{-1}(h')\right)^{1/T_t} Q_m(h|h')}{\left(DL^{-1}(h)\right)^{1/T_t} Q_m(h'|h)}\right\} \tag{6.8}$$

A logarithmic decrease ensures the convergence to the global optimum for any initial configuration h_0. In practice, a geometric cooling scheme is preferred to accelerate the process and to give an approximate solution close to the optimal one as follow:

$$T_t = T_0 \alpha^t \tag{6.9}$$

where T_0, α and t are the initial temperature, the decreasing coefficient and the number of iterations, respectively. A slight adaption of the schedule was made in which the temperature is updated in every k-th iteration of the algorithm.

6.2.5.2 Proposed Kernels

MCMC algorithms typically require the design of proposal mechanisms to propose candidate hypothesis. Appropriately designed proposal kernels let MCMC algorithm quickly converge by proposing configurations of interest. In our approach, three types of kernels (Q_1, Q_2 and Q_3) are defined to perform moves between different configurations as follows:

- ***Kernel Q_1 (with Type I topological line)***: This kernel is designed to replace an L-Model line with an I-Line. The kernel does not add any vertex where an ending point of L-Model line moves to a point generated by intersecting the extension line of I-Line with GL. Thus, L-Model fully contains I-Line's properties (slope and position).
- ***Kernel Q_2 (with Type II topological line)***: This kernel is designed to change the shape at an ending point of an I-Line. This kernel adds vertices to a new configuration to represent the shape changes. The shape changes occur in two intersection points between I-Line and topological line, and between topological line and L-Model. The direction of the topological line is determined by sampling one representative angle of CLF. Thus, a new configuration contains properties of

I-Line, L-Model and topological line. In this Kernel, no prior knowledge for corners is used.

- ***Kernel Q_3 (with Type III topological line)***: This kernel is an advanced form of kernel Q_2, which adds vertices with prior knowledge for corners. A corner, evidence of a sudden change in building structure, serves to represent shape changes. Directions of the corner arms are used to guide the direction of topological line. If there are more than two arms, one direction of the topological lines is determined by randomly sampling the directions of the arms.

L-Model line L_i^M and I-Line L_j^I are randomly selected and topological relations are established as stated in section 6.2.2. The move from a configuration h to h' is realized by sampling Q_m with uniform distribution for each AP. Kernel Q_1 and Q_2 can be selected for all I-Lines while Kernel Q_3 is only selected in the case that corners exist. After selecting the kernel type for each AP, a new building rooftop configuration is generated as explained in section 6.2.3. Figure 6.9 shows the pseudo-code for the optimization process.

Pseudo-code:

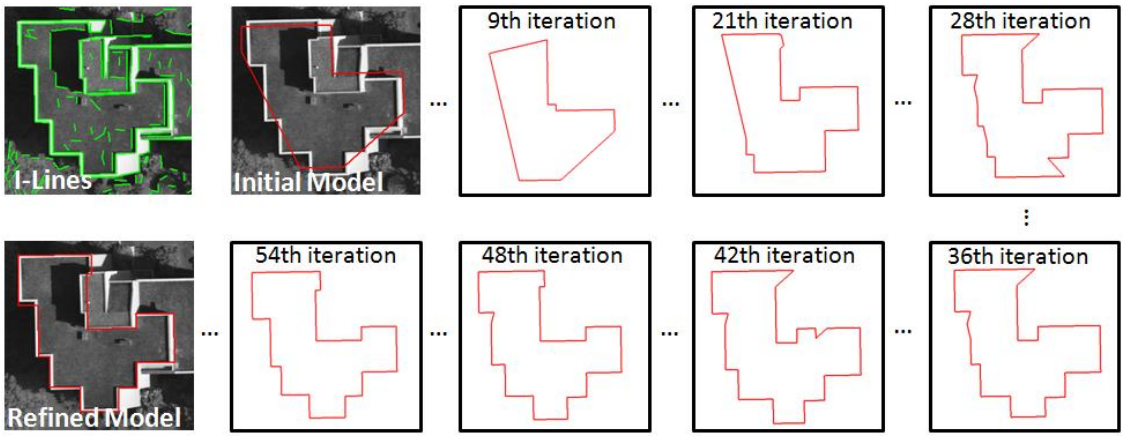
1. Set T_0, α and t ($T_0 = 10$, $\alpha = 0.99$, and $t = 1$)
2. Choose randomly a target L-Model L_i^M and a I-Line L_j^I from the previous L-Model (h)
3. Sample proposition kernels (Q_m) for two ending points of I-Line
4. Propose a new hypothesis (h') based on sampled proposition kernels
5. Compute the acceptance ratio using Eq. (6.8)
6. Add the n^{th} configuration h' to the Markov Chain if the proposed hypothesis is accepted. Otherwise, add the previous model (h).
7. Repeat the steps 2-6 with the same temperature k times ($k = 500$)
8. Decrease the temperature using Eq. (6.9)
9. Repeat the steps 2-8 until the *DL* value has converged.

Figure 6.9 Pseudo-code for MCMC coupled with SA

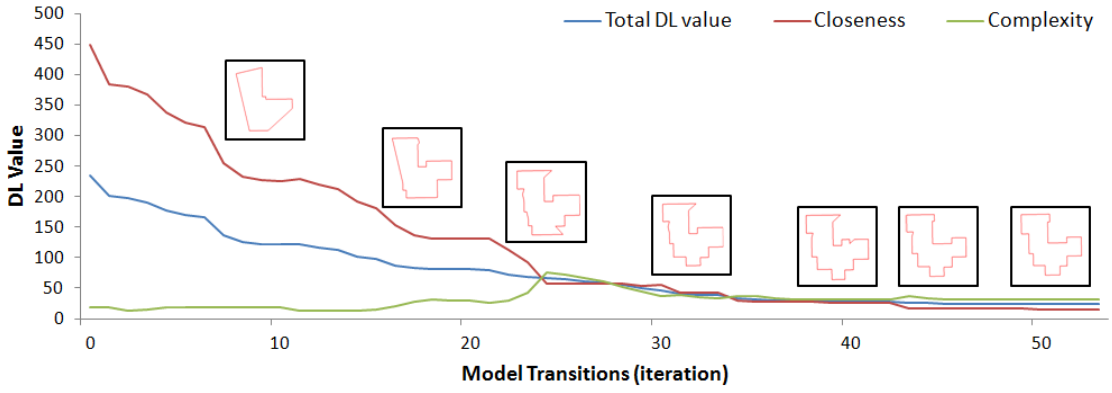
6.3 Experimental Results

The proposed sequential fusion method was tested on the Vaihigen and the Toronto datasets. Before testing the entire real datasets, the proposed method was applied to two simulated building rooftop models, which were produced by manually digitizing vertices around real building boundary. This simulated model-based experiments allow us to investigate the performance of the proposed algorithm in a controlled environment. Figure 6.10 and Figure 6.11 show the simulated over-simplified building model and under-simplified building model, respectively. As shown in Figure 6.10, the simulated over-simplified building model consists of a small number of vertices compared with real building boundary. For the initial building model, the model closeness is large because the boundaries of the model are far away from observations (boundary points) while the model

complexity has a small value with regard to the definition of our model complexity (section 6.2.4.2). As fusion process proceeds, the model closeness gradually decreases; the model complexity increases; total *DL* value decreases. At the end, the model is converged to the optimal building model which has the minimum *DL* value (Figure 6.10(b)). Figure 6.10(a) shows transitions from the initial over-simplified model to the optimal building model where the initial model is progressively rectified by the proposed algorithm.



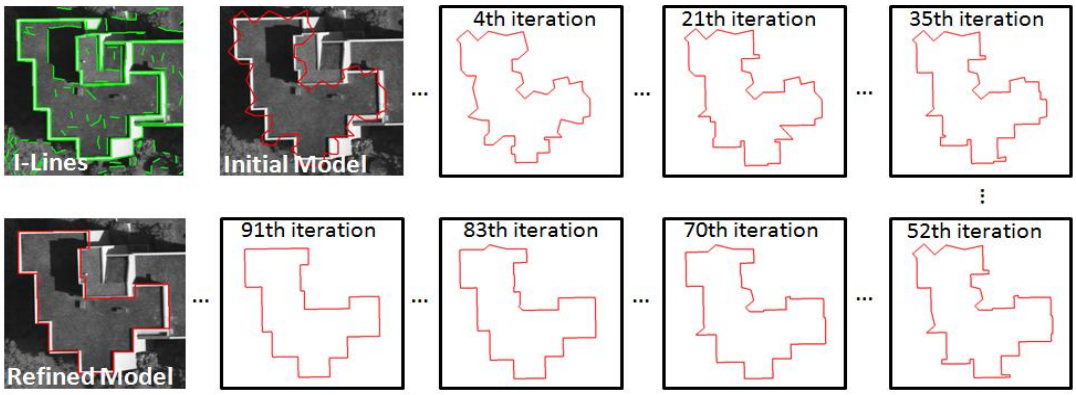
(a)



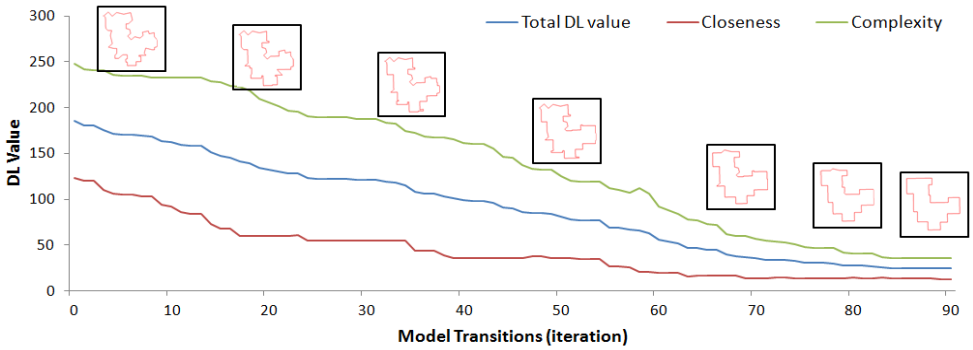
(b)

Figure 6.10 (a) Model transitions and (b) *DL* values for over-simplified model

The simulated under-simplified model represents different patterns compared with the over-simplified model in terms of behavior for model closeness and model complexity (Figure 6.11(b)). Similarly to the over-simplified model, the model closeness has a large value for the initial under-simplified model. However, the model complexity is significantly larger because the initial model has a large number of vertices and directions, and large penalty of inner angles. The model closeness and the model complexity simultaneously decrease during refinement process. At the end of refinement process, the under-simplified model gets closer to the real building. Figure 6.11(a) and (b) show transitions of the under-simplified model and their corresponding *DL* values, respectively.



(a)



(b)

Figure 6.11 (a) Model transitions and (b) *DL* values for under-simplified model

For the above two simulated examples, quantitative assessments were conducted based on an evaluation metric proposed in Chapter 3. Table 6.1 describes results of area-based evaluations using confusion matrix where the average completeness, correctness and quality of the refined model are 96.26%, 97.11% and 93.57%, respectively. Table 6.2 shows results of angle-based and shape-based evaluations where the average angle differences, Hausdorff distance and turning function distance are 0.49º, 0.47m, and 0.031, respectively. The results clearly indicate that arbitrarily drawn initial models can be effectively refined regardless of error types and an amount of modeling errors.

Table 6.1 Confusion matrix-based evaluations

	Initial Model			*Refined Model*		
	$Comp_{Area}$ (%)	*$Corr_{Area}$ (%)*	*$Quality_{Area}$ (%)*	*$Comp_{Area}$ (%)*	*$Corr_{Area}$ (%)*	*$Quality_{Area}$ (%)*
Over-Simplified model	90.58	79.96	73.83	96.92	96.64	93.76
Under-Simplified model	93.62	88.37	83.35	95.59	97.58	93.37
Total	92.10	84.17	78.59	96.26	97.11	93.57

Table 6.2 Angle-based and shape-based evaluations

	Initial Model			*Refined Model*		
	Angle difference (deg)	*Hausdorff distance (m)*	*Turning function distance*	*Angle difference (deg)*	*Hausdorff distance (m)*	*Turning function distance*
Over-Simplified model	3.06	3.06	0.136	0.49	0.55	0.032
Under-Simplified model	1.72	1.36	0.101	0.49	0.39	0.029
Total	2.39	2.21	0.119	0.49	0.47	0.031

Five representative building models of sub-datasets, which have different types of modeling errors, were selected to demonstrate evaluation results in detail. In the selected LiDAR-driven building models, three modeling errors were observed partly or in combinations (Figure 6.12(c)). For example, shape deformations including under-

simplified parts (blue circles) and over-simplified parts (red circles) were observed in all building models; Orientation error was observed in the building model *V* which was rotated by approximately 1.62º compared to the reference building model (Table 6.4); All models were slightly shrunken due to boundary displacements (Table 6.3).

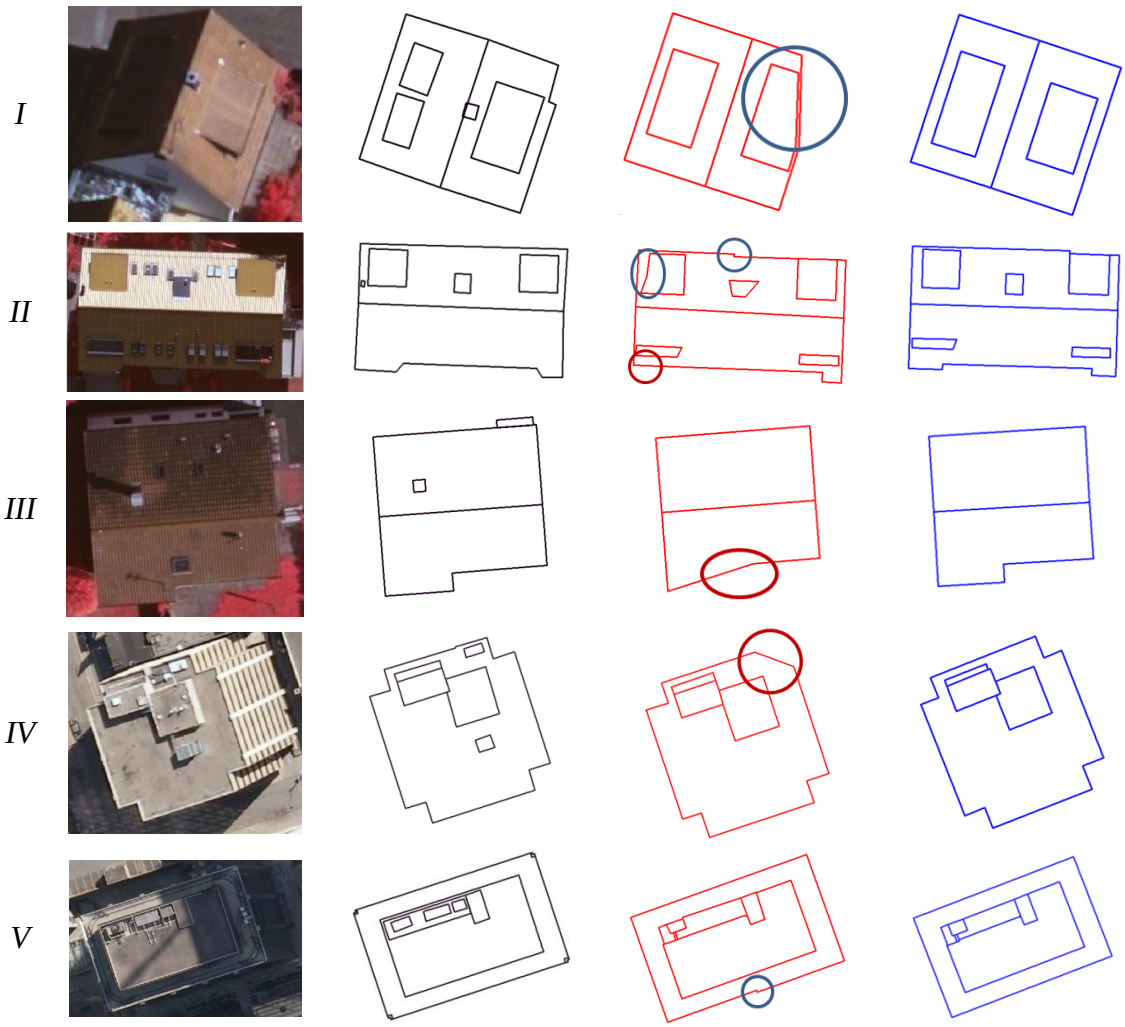


Figure 6.12 Quantitative assessment for selected five building models: (a) input images, (b) reference models (c) initial models, and (d) refined models.

Figure 6.12(d) shows refined building models where most modeling errors observed in the initial building models were rectified in terms of visual inspection. In order to clearly show improvements, quantitative assessments were conducted. Table 6.3 shows evaluation results using confusion matrix. For the initial models, the averages of the completeness, correctness, and quality were 90.91%, 99.84%, and 90.77%, respectively. The correctness was nearly 100% for all building models while the completeness was approximately 91%. These results implicitly indicate that the initial models were shrunken by approximately 9% compared with the reference models. After applying the proposed method, the averages of the completeness, correctness and quality were 96.52%, 98.74%, and 95.33%, respectively. For all selected building models, the correctness was slightly deteriorated by small increase of false positives while higher completeness and quality were achieved by a large decrease of false negatives and an increase of true positives. The results imply that a loss of correctness is inevitable, but the proposed sequential fusion method can improve the overall quality of LiDAR-driven building models.

Table 6.3 Assessment based on confusion matrix for selected building models

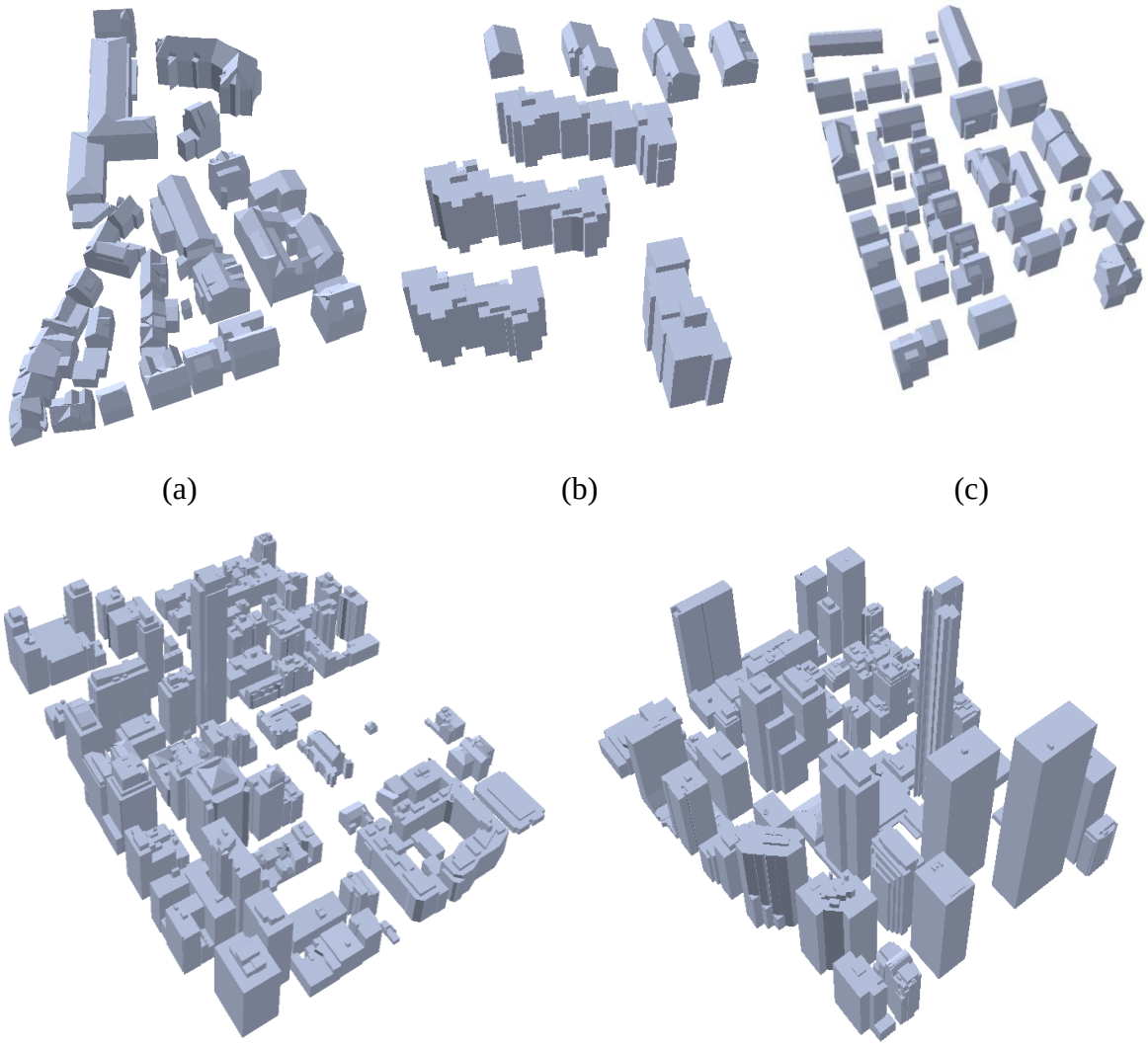
Building Model	*Between reference and initial model*			*Between reference and refined model*		
	$Comp_{Area}$ (%)	$Corr_{Area}$ (%)	$Quality_{Area}$ (%)	$Comp_{Area}$ (%)	$Corr_{Area}$ (%)	$Quality_{Area}$ (%)
I	85.79	100	85.79	93.09	98.77	92.02
II	92.15	99.99	92.14	98.06	98.52	96.63
III	90.47	99.95	90.43	95.68	99.50	95.22
IV	93.03	99.25	92.37	96.66	98.63	95.38
V	93.12	100	93.12	99.10	98.28	97.40
Total	90.91	99.84	90.77	96.52	98.74	95.33

Angle difference, Hausdorff distance and turning function distance were assessed by comparing the reference building models with the initial building models and the refined building models, respectively (Table 6.4). The average angle difference for the initial models was 0.67º while the average angle difference for the refined models was 0.33º. Major improvement was achieved in building model *V* whose initial model contained considerable orientation error. This indicates that an orientation error accidently occurred in LiDAR-driven building models is rectified by accurate orientation derived from image. As shaped-based evaluations, the average Hausdorff distances and the average turning function distance between the reference models and the initial models were 1.43m and 0.037, while the evaluation results for the refined models were 0.61m and 0.017, respectively. The results represent that the proposed method can rectify partly deformed shapes in addition to providing very similar building models with reference models.

Table 6.4 Assessments by angle-based index and shape-based indices for selected building models

Building Model	*Between reference and initial model*			*Between reference and refined model*		
	Angle difference (deg)	*Hausdorff distance (m)*	*Turning function distance*	*Angle difference (deg)*	*Hausdorff distance (m)*	*Turning function distance*
I	0.49	1.28	0.041	0.44	0.49	0.024
II	0.25	1.55	0.024	0.18	0.87	0.024
III	0.08	1.13	0.050	0.72	0.34	0.007
IV	0.89	1.80	0.051	0.24	0.79	0.016
V	1.62	1.41	0.017	0.09	0.56	0.013
Total	0.67	1.43	0.037	0.33	0.61	0.017

The proposed sequential fusion method was applied to entire datasets. Figure 6.13 visualizes the final outcomes. The results for corresponding quantitative assessments are listed in Table 6.5 ~ Table 6.8.



(a) (b) (c)

(d) (e)

Figure 6.13 3D visualization of rooftop models produced by sequential modeling algorithm: (a) Area 1, (b) Area 2, (c) Area 3, (d) Area 4, and (e) Area 5

As shown in Table 6.5, evaluations using confusion matrix show improvements of 2.4% and 1.8% for the completeness and quality while the correctness is deteriorated by 0.6%. Similarly to the results derived from five selected building models, the evaluation results for the entire datasets indicate that true positives were increased while false negatives and false positives were decreased. As a result, model quality was improved by the sequential fusion method.

Table 6.5 Assessment by area-based confusion matrix for entire datasets

Dataset	*Area*	*Between reference and initial model*			*Between reference and refined model*		
		$Comp_{Area}$ (%)	$Corr_{Area}$ (%)	$Quality_{Area}$ (%)	$Comp_{Area}$ (%)	$Corr_{Area}$ (%)	$Quality_{Area}$ (%)
Vaihingen	*Area1*	90.6	98.8	89.6	92.2	98.6	91.0
	Area2	91.3	99.7	91.0	95.8	98.6	94.5
	Area3	88.6	99.7	88.4	92.0	98.5	90.8
	Sub-total	90.2	99.4	89.7	93.3	98.6	92.1
Toronto	*Area4*	93.7	96.9	90.9	95.3	96.4	92.0
	Area5	93.1	92.0	86.1	94.2	91.8	86.9
	Sub-total	93.4	94.5	88.5	94.8	94.1	89.5
Total		91.5	97.4	89.2	93.9	96.8	91.0

Angle-based and shape-based evaluations were conducted for 2D building boundaries (Table 6.6) and for 3D rooftop planes with 50% overlap (Table 6.7). For 2D building boundaries, the average orientation error reduced from 1.17º to 0.63º; The average Hausdorff distance decreased from 1.81m to 0.48m; The average turning function distance decreased from 0.042 to 0.007, respectively. For 3D rooftop planes, the average angle difference, Hausdorff distance, and turning function distance were improved by 0.3º, 0.3m, and 0.004, respectively.

Table 6.6 Assessments by angle-based index and shape-based indices for 2D building boundary

Dataset	*Area*	*Between reference and initial model*			*Between reference and refined model*		
		Angle difference (deg)	*Hausdorff distance (m)*	*Turning function distance*	*Angle difference (deg)*	*Hausdorff distance (m)*	*Turning function distance*
Vaihingen	*Area1*	1.32	1.33	0.049	0.65	1.32	0.043
	Area2	1.62	1.26	0.040	0.82	0.83	0.029
	Area3	0.59	0.93	0.031	0.70	0.72	0.028
	Sub-total	1.18	1.17	0.040	0.72	0.96	0.033
Toronto	*Area4*	1.30	2.44	0.046	0.61	2.14	0.039
	Area5	1.04	3.10	0.046	0.35	2.37	0.035
	Sub-total	1.17	2.77	0.046	0.48	2.26	0.037
Total		1.17	1.81	0.042	0.63	1.48	0.035

Table 6.7 Assessment by angle-based index and shape-based indices for 3D rooftop polygons

Dataset	Area	*Between reference and initial model*			*Between reference and refined model*		
		Angle difference (deg)	*Hausdorff distance (m)*	*Turning function distance*	*Angle difference (deg)*	*Hausdorff distance (m)*	*Turning function distance*
Vaihingen	Area1	0.78	0.46	0.020	0.82	0.43	0.021
	Area2	1.11	1.77	0.041	0.56	0.65	0.030
	Area3	0.44	0.48	0.016	0.58	0.42	0.024
	Sub-total	0.078	0.90	0.026	0.65	0.50	0.025
Toronto	Area4	1.30	1.38	0.040	0.72	1.28	0.041
	Area5	0.91	1.75	0.047	0.39	1.59	0.031
	Sub-total	1.11	1.57	0.044	0.56	1.44	0.036
Total		0.91	1.17	0.033	0.61	0.87	0.029

More specifically, Figure 6.14 shows distributions of angle difference, Hausdorff distance and turning function distance for initial building models and for refined building models, respectively. The figures clearly show that the proposed sequential fusion method

has positive effects for the corrections of orientation errors and shape deformations of LiDAR-driven building models.

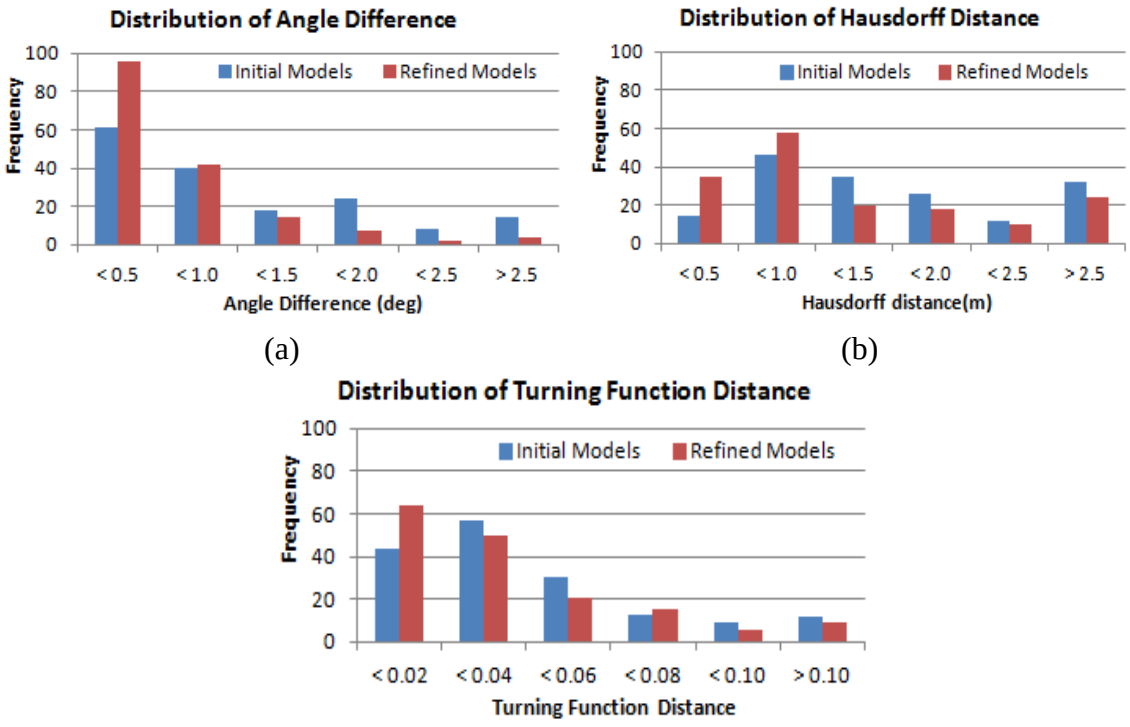


Figure 6.14 Distributions of (a) angle difference, (b) Hausdorff distance, and (c) turning function distance for entire initial models and refined models

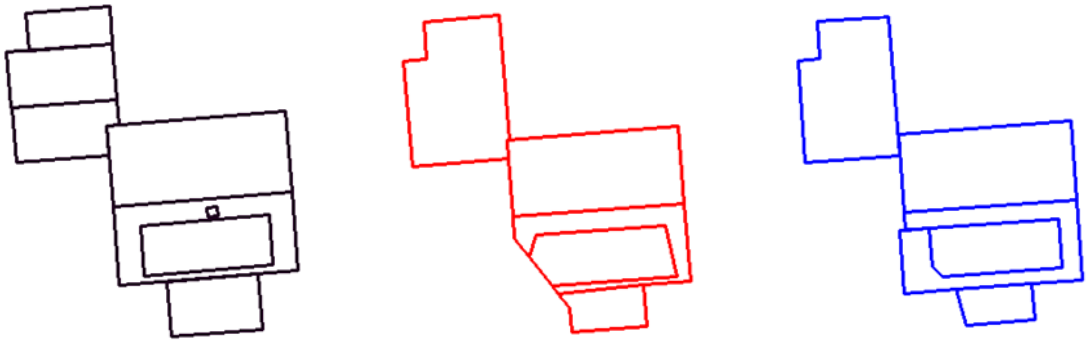
Additionally, geometric accuracies in planimetry and in height were evaluated over 3D rooftop polygons with 50% overlap (Table 6.8). The results show similar levels of geometric accuracies for initial building models and refined building models. This is mainly due to the fact that only the points, which have correspondence between reference building boundary points and extracted building boundary points within user-defined

buffer size, were considered in calculating geometric errors. Thus, points that have larger position errors than the user-defined buffer size do not influence on the geometric accuracy.

Table 6.8 Assessment based on geometric accuracy for rooftop polygons with 50% overlap (unit: m)

Dataset	*Area*	*Between reference and initial model*			*Between reference and refined model*		
		RMSE x	*RMSE y*	*RMSE z*	*RMSE x*	*RMSE y*	*RMSE z*
Vaihingen	*Area1*	±0.76	±0.70	±0.52	±0.76	±0.69	±0.50
	Area2	±0.80	±0.55	±0.40	±0.63	±0.48	±0.40
	Area3	±0.71	±0.73	±0.42	±0.66	±0.75	±0.38
	Sub-total	±0.76	±0.66	±0.45	±0.68	±0.64	±0.43
Toronto	*Area4*	±0.95	±0.76	±1.39	±0.94	±0.75	±1.49
	Area5	±0.97	±0.66	±1.32	±0.95	±0.67	±1.32
	Sub-total	±0.96	±0.71	±1.36	±0.95	±0.71	±1.41
Total		±0.84	±0.68	±0.81	±0.79	±0.67	±0.82

Even though the proposed sequential fusion algorithm can effectively rectify modeling errors of LiDAR-driven building models, the method has some limitations. The main limitation is that topology errors of initial models cannot be ameliorated. Figure 6.15 shows an example where two rooftop polygons in the lower part of reference building model are represented by polygon in polygon. However, in the initial model, the two rooftop polygons are represented as two adjacent polygons. After refinement process, the two rooftop polygons still remain as two adjacent polygons.



(a) (b) (c)

Figure 6.15 Limitation of the proposed method: (a) reference model, (b) initial model, and (c) refined model

6.4 Summary

In this chapter, we proposed a sequential fusion method to refine existing building models (rooftop models reconstructed from LiDAR data) by incorporating modeling cues extracted from an airborne imagery. In this fusion modeling framework, a set of building hypotheses reflecting the contribution of the airborne imagery to existing models are implicitly generated. A progressive regularization of the rooftop models was implemented in MDL as an objective function for determining the most optimal rooftop model. A global optimization was achieved by MCMC coupled with SA where proposition kernels with or without prior knowledge were proposed. The results over both simulated data and entire real datasets demonstrated that the proposed method can effectively and robustly rectify modeling errors caused by inherent characteristics of LiDAR data. More specifically, experimental results are summarized as follows:

- As shown in the simulated data, the method can provide accurate building models regardless of modeling error types such as under-simplified and over-simplified shapes and the amount of errors.
- Evaluations using the confusion matrix showed that the proposed fusion modeling algorithm improves the completeness and quality while correctness was slightly degenerated.
- Angle-based evaluation presented that the orientation error, which occurred in LiDAR-driven building models, can be robustly rectified by the proposed method. This is due to the fact that the errors in the building orientation caused by irregular distribution of LiDAR data were reduced by introducing accurate orientation derived from image.
- Shape-based evaluations showed that refined building models produced similar shapes compared to reference building models. Also, partly deformed building parts were well recovered by the proposed method.

As our future works, we will study on topology error correction which was not covered by the current method. Also, we will extend the proposed method using 3D lines derived from multiple images and using points derived from the structure from motion (SFM) instead of LiDAR points.

Chapter 7
Conclusions and Future Directions

7.1 Conclusions

This thesis aimed to continuously reconstruct 3D building rooftop models using multi-sensor data. In order to achieve the goal, we identified four critical steps and provided reasonable and promising solutions for each topic. The first step towards continuous city modeling was to devise a method to reconstruct robust and accurate regularized building rooftop models, regardless of scene complexity. The second step was to automatically register newly acquired image data with existing building models without any labor-intensive manual process, thus providing a reasonable solution for the sequential fusion process. The third step was to construct a method which can update the existing building models using multi-sensor data in a timely way. Lastly, we addressed the issue of assessing the quality of reconstructed or refined building rooftop models. The following sections provide the conclusions of each critical topic toward continuous city modeling.

Chapter 3 proposed an evaluation matrix which assess various qualities of reconstructed 3D building models. After exhaustive reviews on existing evaluation methods, we added new evaluation indices to measure shape similarity and angle similarity of reconstructed building models. The newly proposed Hausdorff distance provided a way

to quantitatively measure the largest error amount of partly deformed building models. The turning function distance measured how the reconstructed building model is similar to the reference model in terms of overall shape of rooftop models. Angle difference provided information on the orientation error of building models compared with reference models. The developed matrix was used to assess the qualities of LiDAR-driven building models and refined building models. The experiments based on the proposed evaluation matrix showed that the evaluation indices are good indicators for measuring the various qualities of building models. However, the limitation of the proposed evaluation was that the metrics cannot measure the topology accuracy of building rooftop models. Therefore, topology-based evaluation method should be investigated in future works.

In Chapter 4, we proposed an automatic building reconstruction method using LiDAR data which covered low level modeling cue extraction to reconstruction of realistic 3D rooftop models. A hierarchical strategy for modeling cue extraction made it possible to effectively collect modeling evidence from complex building structures. The regularization method proposed by Sohn et al. (2012) was extended by proposing automatic weighting parameter determination methods. The hypothesis generation method generated regular-shaped candidate hypotheses by implicitly designed rules from various configurations of building rooftop. MDL was used as a criterion for model selection, to choose the best model among possible candidate hypotheses. In particular, the proposed implicit regularization provided flexibility for describing more complex rooftop models while preserving building regularity. The experimental results showed that the proposed building

reconstruction method can robustly produce regularized building rooftop models regardless of building complexity. Area-based evaluations using confusion matrix were the average completeness of 91.5%, correctness of 97.4%, and quality of 90%. Object-based evaluation using confusion matrix showed that our method outperforms other building reconstruction algorithms. However, results of object-based evaluations indicated that small size rooftops are not effectively extracted. Also, angle-based index showed that angle difference is approximately 1.17º compared with reference models. These results demonstrated that our proposed building reconstruction method is a reasonable solution. Additionally, the effect for automatic weighting methods was evaluated by comparing building rooftop models with fixed weight parameters and these with adaptive weight parameters determined by the proposed weight methods. Shape-based evaluations indicated that the use of the weighting methods have positive effects for decreasing partly deformed shapes of building models. Three main modeling errors (shape deformation, boundary displacement, and orientation errors) were observed in the rooftop models reconstructed by our proposed method. The modeling errors of the proposed building rooftop modeling was caused by the inherent characteristics of LiDAR data. Thus, a fusion method to integrate complementary data (i.e., image data) with LiDAR data is required to rectify the modeling errors.

Chapter 5 proposed a new registration method, context-based geometric hashing (CGH), to align a single image with existing 3D building models. As an essential step for continuous city modeling, the newly-acquired single image was aligned with large-scale existing building models without any labor-intensive manual processes. We solved the

registration problems by adopting geometric hashing, a well-known model-to-image matching method. In order to compensate the limitations of standard geometric hashing methods, new features, namely the cornered edges and context feature, were proposed. The main contribution of this registration was the development of the new score function in the CGH-based registration method, which reinforce context forces to improve matching performance. Experimental results showed that the overall registration accuracy was under 2 pixels (under 30cm in GSD) over two different scenes which have different complexity. The amount of registration error is reasonable and acceptable for the sequential refinement process considering that airborne LiDAR point space is approximately 30cm and its position accuracy is approximately 15cm. However, the quality of building models directly affected the accuracy of EOPs. When manually digitized accurate building models were used for registering image data, the proposed registration method achieved reliably accurate EOPs of a single image, regardless of used threshold and assumed error amount. In contrast, if the existing building models contained more modeling errors, the accuracy of EOPs degenerated and was more sensitive to the used threshold and assumed error amount. In terms of feature used, the use of the context feature played a significant role in the matching process. It is due to that fact that contextual term made it possible to provide global information of building structures.

Chapter 6 proposed a sequential fusion method to refine existing LiDAR-driven 3D building models. The modeling errors of LiDAR-driven building rooftop models were progressively refined by image features derived from images in a HAT framework. A new

method to connect the existing LiDAR-driven building model with the image cues was proposed to generated possible hypotheses. MDL was again used as a criterion for model selection. MCMC coupled with SA was employed for evaluation of global optimization. Experimental results on simulated data showed that the proposed fusion method can effectively recover the large amount of modeling errors which were often observed in LiDAR-driven building models. Various analyses based on the evaluation matrix proposed in Chapter 3 were conducted to confirm the performance of the proposed fusion method. Confusion matrix-based evaluations showed that the completeness and quality were considerably improved, compared with those of initial LiDAR-driven models, while the correctness is slightly degenerated. Angle-based evaluation showed that orientation error, which accidently occurs in LiDAR-driven building models, can be refined by accurate orientations derived from image. Shape-based evaluations showed that partly deformed shapes in LiDAR-driven models were improved by the sequential fusion method. The proposed refinement process provided a way that existing building models can be effectively reused by accommodating their changes recognized in temporal domain. However, the proposed method should be extended to deal with topology errors of building rooftop models which was not covered by the current method.

7.2 Directions for Future Research

As described in the previous section, this research provided a research platform for continuous cityscape modeling using multi-sensor data. However, the methods mainly focused on airborne LiDAR and image data. The ultimate goal of this research is to fully

describe 3D building models using any type of data. As well as airborne LiDAR and images, laser scanning data, images and video images mounted on mobile and UAV platforms should be able to be imported in our continuous city modeling framework; this could be addressed in future research. Future works of each topic can be summarized as follows:

- In terms of evaluation of building rooftop models, even though the proposed evaluation metrics represented good scores for reconstructed building models, visual inspection indicated that the model still have many modeling errors. The main reason for discrepancies between visual inspection and evaluation matrix was topology difference between the extracted building model and reference. Thus, a method for measuring topology similarity should be studied.

- The proposed building rooftop modeling method provided a promising results. However, the main limitation was that the method cannot detect small size rooftop planes. It is a main disadvantage of data-driven approach which cannot recover all building rooftop structures if enough observations are not taken from the structures in detail. One possible method is to combine model-driven approach to our proposed data-driven approach. Thus, small object such as superstructure can be recovered by model-driven approach.

- The proposed CGH-based registration method required rough initial EOPs of a single image to back-project existing building models to the image and to reduce

search area and matching ambiguity. One prospective goal is to automatically find the rough initial EOPs. Also, we will extend the proposed method to arbitrarily acquired images (e.g., UAV image and security camera image).

- In terms of data fusion, the proposed method used 2D image lines derived from single images as cues for refining existing LiDAR-driven building models. The process made refinement of 3D building models complex, where 2D lines should be converted to 3D lines by adding height information of LiDAR. The direct extraction of 3D lines from stereo images or multi-images would reduce processing complexity and ambiguity in selection of corresponding line cues, and should definitely be considered in future projects.

Bibliography

Ameri, B., 2000. Feature based model verification (FBMV): a new concept for hypothesis validation in building reconstruction. In: *Proceedings of the XIXth ISPRS Congress IAPRS*, 33(B3), Amsterdam, the Netherlands, pp. 24- 35.

Ameri, B., 2000. Automatic recognition and 3D reconstruction of buildings through computer vision and digital photogrammetry. PHD thesis, University of Stuttgart.

Ameri, B. and Fritsch, D., 2000. Automatic 3D building reconstruction using plane-roof structures. *Proceedings of the American Society for Photogrammetry and Remote Sensing Conference*, Washington, D.C.

Andrieu, C., de Freitas, N., Doucet, A., Jordan, M. I., 2003. An introduction to MCMC for machine learning. *Machine Learning*, 50, pp. 5-43.

Arkin, E., Chew, L. P., Huttenlocher, D. P., Kedem. K., Mitchell, J. S. B., 1991. An efficiently computable metric for comparing polygonal shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3), pp. 209-215.

Avbelj, J., Iwaszczuk, D., Müller, R., Reinartz, P., Stilla, U., 2015. Coregistration refinement of hyperspectral images and DSM: an object-based approach using spectral information. *ISPRS Journal of Photogrammetry and Remote Sensing*, 100, 23-34.

Avbelj, J., Iwaszczuk, D., Stilla, U., 2010. Matching of 3D wire-frame building models with image features from infrared video sequences taken by helicopters or UAVs. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38(Part 3B), pp 149-154.

Awrangjeb, M., Fraser, C. S., 2013. Rule-based segmentation of LiDAR point cloud for automatic extraction of building roof planes. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3/W3, Antalya, Turkey, pp. 1-6.

Awrangjeb, M., Zhang, C., Fraser, C. S., 2013. Automatic extraction of building roofs using LIDAR data and multispectral imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 83, pp. 1–18.

Baltsavias, E. P,, 2004. Object extraction and revision by image analysis using existing geodata and knowledge: current status and steps towards operational systems. *ISPRS Journal of Photogrammetry & Remote Sensing*, 58, pp.129-151.

Bay, H., Ess, A., Tuytelaars, T., Van Gool, L., 2008. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110(3), pp. 346-359.

Brenner, C., 2005. Building reconstruction from images and laser scanning. *International Journal of Applied Earth Observation and Geoinformation*, 6(3-4), pp. 187-198.

Brenner, C., 2010. Building extraction. Book chapter. In: *Airborne and Terrestrial Laser Scanning*. George Vosselman, Hans-Gerd Maas (eds.), Whittles Publishing, 2010.

Brown, L.G., 1992. A survey of image registration techniques. *ACM Computing Surveys*, 24, pp. 326-376.

Bulatov, D., Häufel, G., Meidow, J., Pohl, M., Solbrig, P., Wernerus, P., 2014. Context-based automatic reconstruction and texturing of 3D urban terrain for quick-response tasks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93, pp. 157-170.

Cakmakov, D., Celakoska, E., 2004. Estimation of curve similarity using turning function. *International Journal of Applied Math*, 15(4), pp. 403-416.

Castro, E. D., Morandi, C., 1987. Registration of translated and rotated images using finite Fourier transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9, pp. 700-703.

Chabat, F., Yang, G.Z., Hansell, D.M., 1999. A corner orientation detector. *Image and Vision Computing*, 17, pp. 761-769.

Chen, L., Teo, T., Rau, J., Liu, J., Hsu, W., 2005. Building reconstruction from LiDAR data and aerial imagery. In: *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, pp. 2846–2849.

Cheng, L., Gong, J., Li, M., Liu, Y., 2011. 3D building model reconstruction from multi-view aerial imagery and LIDAR data. *Photogrammetric Engineering & Remote Sensing*, 77, pp. 125–139.

Cheng, L., Tong, L., Chen, Y., Zhang, W., Shan, J., Liu, Y., Li, M., 2013. Integration of lidar data and optical multi-view images for 3d reconstruction of building roofs. *Optics and Lasers in Engineering*, 51(4), pp. 493–502.

Davies, R. H., Twining, C. J., Cootes, T. F., Waterton, J. C., Taylor, C. J., 2002. A minimum description length approach to statistical shape modeling. *IEEE Transactions on Medical Imaging*, 21(5), pp. 525-537.

Demir N., Baltsavias, E., 2012. Automated modeling of 3d building roofs using image and lidar data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, I(4), Melbourne, Australia, 2012, pp. 35–40.

Ding, M., Lyngbaek, K., Zakhor, A., 2008. Automatic registrationof aerial imagery with untextured 3D LiDAR models. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Anchorage, Alaska.

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

Douglas, D., and Peucker, T., 1973. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10(2), pp. 112-122.

Economic Dashboard-Annual Summary, 2015. http://www.toronto.ca/legdocs/mmis/2015/ed/bgrd/backgroundfile-76322.pdf

Eugster, H., Neibiker, S., 2009. Real-time georegistration of video streams from mini or micro UAS using digital 3D city models. *6th International Symposium on Mobile Mapping Technology*, Presidente Prudente, Sao Paulo, Brazil.

Fischler, M. A. and Bolles, R. C., 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM,* 24(6), pp. 381-395.

Fonseca L, M. G. and Manjunath B. S., 1996. Registration techniques for multisensor remotely sensed imagery. *Photogrammetric Engineering & Remote Sensing*, 62(9), pp. 1049-1056.

Foody, G., 2002. Status of land cover classification accuracy assessment. *Remote Sensing of Environment*, 80(1), pp. 185-201.

Frueh, C., Russell, S., Zakhor, A., 2004. Automated texture mapping of 3D city models with oblique aerial imagery. *Proceeding of the 2nd International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'04)*.

Fuchs, C., Förstner, W., Gülch, E., Heipke, C., Eder, K., 1998. OEEPE survey on 3D-city models. Bundesamt für Kartographie und Geodäsie.

Gennert, M. A., and Yuille, A. L., 1998. Determining the optimal weights in multiple objective function optimization. *In Proc. IEEE Int. Conf. Computer Vision*, pp. 87-89.

Grün, A., Baltsavias, E., Henricsson, O.(Eds.), 1997. Automatic extraction of man-made objects from aerial and space images (II). Birkhäuser, Basel.

Grün, A., Kübler, O., Agouris, P. (Eds.), 1995. Automatic extraction of man-made objects from aerial and space images. Birkhäuser, Basel.

Grünwald, P., 2005. A tutorial introduction to the minimum description length principle. In P. Grünwald, I. J. Myung, and M. Pitt, editors, Advances in Minimum Description Length: Theory and Applications, pp. 3–81. MIT Press.

Lamdan, Y., Wofson, H. J., 1988. Geometric hashing : a general and efficient model-based recognition scheme. In: *Proceedings of the 2nd International Conference on Computer Vision (ICCV)*, pp. 238–249.

Lee, D. H., Lee, K.M., Lee, S.U., 2008. Fusion of lidar and imagery for reliable building extraction. *Photogrammetric Engineering and Remote Sensing*, 74 (2), pp. 215–225.

Lotfi, F. H., Fallahnejad, R., 2010. Imprecise Shannon's entropy and multi Attribute decision making. *Entropy*, 12, pp. 53-62.

Lowe, D. G., 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), pp. 91-110.

Haala, N., and Kada, M., 2010. An update on automatic 3D building reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65, pp. 570-580.

Habib, A., Ghanma, M., Morgan, M., Al-Ruzouq, R., 2005. Photogrammetric and LiDAR data registration using linear features. *Photogrammetric Engineering & Remote Sensing*, 71(6), pp. 699-707.

Hall, D. L. and Llinas, J., 1997. An introduction to multisensor data fusion. *Proceedings of the IEEE*, 85(1), pp. 6–23.

HERE 360, 2015. HERE updates maps worldwide for Android, iOS and Windows. http://360.here.com/2015/08/04/here-updates-maps-for-android-ios-and-windows/

Hsu, S., Samarasekera, S., Kumar, R., Sawhney, H., S., 2000. Pose estimation, model refinement, and enhanced visualization using video. *Proceedings of IEEE International conference on Computer Vision and Pattern recognition*, Hilton Head, SC, pp. 488-495.

Hu, J., You, S, Neumann, U., 2006. Integrating LiDAR, aerial image and ground image for complete urban building model. *Proceedings of the Third International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT'06)*.

Huttenlocher, D. P., Klanderman, G. A., Rucklidge, W. J., 1993. Comparing images using the Hausdorff distance. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 15(9), pp. 850-863.

Huang, H., Brenner, C., Sester, M., 2013. A generative statistical approach to automatic 3D building roof reconstruction from laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 79, pp. 29-43.

International Herald Tribune, 2008. UN says half the world's population will live in urban areas by end of 2008. https://web.archive.org/web/20090209221745/http://www.iht.com/articles/ap/2008/02/26/news/UN-GEN-UN-Growing-Cities.php

Iwaszczuk, D., Helmholz, P., Belton, D., Stilla, U., 2013. Model-to-image registration and automatic texture mapping using a video sequence taken by a mini UAV. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL(1/W1), pp. 151-156.

Jwa, Y., 2013. 3D reconstruction of building rooftop and power line models in right-of-ways using airborne LiDAR data. PHD thesis, York University.

Jwa, Y., Sohn, G., Cho, W., Tao, V., 2008. An Implicit geometric regularization of 3D building shape using airborne LiDAR data. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37(PartB3A), pp. 69-76.

Kada, M., McKinley, L., 2009. 3D Building reconstruction from lidar based on a cell decomposition approach. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 38(Part 3/W4), pp. 47-52.

Kada, M., Wichmann, A., 2012. Sub-surface growing and boundary generalization for 3D building reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, I(3), pp. 223-238.

Kada, M. and Wichmann, A., 2013. A feature-driven 3D building modeling using planar halfspace. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-2(W3), pp. 37-42.

Kaneko, S., Satoh, Y., Igarashi, S., 2003. Using selective correlation coefficient for robust image registration. *Pattern Recognition*, 36, pp. 1165-1173.

Kim, C. and Habib, A., 2009. Object-based integration of photogrammetric and LiDAR data for automated generation of complex polyhedral building models. *Sensors*, 9, pp. 5679-5701.

Kolbe, T. H., Gröger, G., Plümer, 2005. CityGML-interoperable access to 3D city models. *Geo-information for Disaster Management*, Springer Berlin Heidelberg, pp. 883-899.

Kovesi, P.D., 2011. MATLAB and octave functions for computer vision and image processing. Centre for Exploration Targeting, School of Earth and Environment, The University of Western Australia.

Lafarge, F., Descombes, X., Zerubia, J., Pierrot-Deseilligny, M., 2010. Structural approach for building reconstruction from a single DSM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1), pp. 135-147.

Lafarge, F., Mallet C., 2012. Creating large-scale city models from 3d-point clouds: a robust approach with hybrid representation. *International Journal of Computer Vision*, 99 (1), pp. 69–85.

Lamdan, Y. and Wolfson, H., 1988. Geometric hashing: a general and efficient model-based recognition scheme. *ICCV'88*, pp. 238-249.

Lee, D. H., Lee, K. M., Lee, S. U., 2008. Fusion of lidar and imager for reliable building extraction. *Photogrammetric Engineering and Remote Sensing*, 74(2), pp. 215-225.

Mayer, H., 2008. Object extraction in photogrammtric computer vision. *ISPRS Journal of Photogrammetry & Remote Sensing*, 63, pp. 213-222.

Meidow, J., Schuster, H. F., 2005. Voxel-based quality evaluation of photogrammetric building acquisitions. *International Archives of the photogrammetry, remote sensing and spatial information sciences*, XXXVI (Part 3/W24), pp. 117-122.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E., 1953. Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21, pp. 1087–1091.

Milde, J., Zhang, Y., Brenner, C., Plümer, L., Sester, M., 2008. Building reconstruction using a structural description based on a formal grammar. *International Archives of*

the Photogrammetry, Remote Sensing and Spatial Information Sciences, 37(Part B3b), pp. 227-232.

Mishra, R. K., Zhang, Y., 2012. A review of optical imagery and airborne LiDAR data registration methods. *The Open Remote Sensing Journal*, 5, pp. 54-63.

Morgan, M., Habib, A., 2002. Interpolation of LiDAR data and automatic building extraction. *ACSM-ASPRS 2002 annual conference proceedings*, 12-14 November 2002, Denver, CO, USA.

Movahedi, V., 2015. Automatic extraction of closed contours bounding salient object: new algorithms and evaluation methods. PHD thesis, York University.

Musialski, P., Wonka, P., Aliaga, D. G., Wimmer, M., van Gool, L., Purgathofer, W., 2012. A survey of urban reconstruction. *EUROGRAPHICS 2012*.

Ok, A. O., Wegner, J. D., Heipke, C., Rottensteiner, F., Soergel, U. and Toprak, V., 2012. Matching of straight line segments from aerial stereo images of urban areas. *ISPRS Journal of Photogrammetry and Remote Sensing*, 74, pp. 133–152.

Otsu, N., 1979. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*., 9(1), pp. 62-66.

Oude Elberink, S., Vosselman, G., 2009. Building reconstruction by target based graph matching on incomplete laser data: analysis and limitations. *Sensors*, 9(8), pp. 6101–6118.

Oude Elberink S., Vosselman G., 2011. Quality analysis on 3D building models reconstructed from airborne laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(2), pp. 157-165.

Perera, S. Nalani, H. A., Maas, H. G., 2012. An automated method for 3D roof outline generation and regularization in airborne laser scanner data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, I-3, Melbourne, Australia, pp. 281-286.

Perera, S., Mass, H. G., 2014. Cycle graph analysis for 3D roof structure modelling: concepts and performance. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93, pp. 213–226.

Persad, R. A., Armenakis, C., Sohn, G., 2015. Automatic co-registration of pan-tilt-zoom (PTZ) video images with 3D wireframe models. *Photogrammetric Engineering & Remote Sensing*, 81(11), pp. 847-859.

Pohl, C., van Genderen, J.L., 1998. Multisensor image fusion in remote sensing: concepts, methods and applications. *International Journal of Remote Sensing*, 19(5), pp. 823-854.

Rau, J. Y., Lin, B. C., 2011. Automatic roof model reconstruction from ALS data and 2D ground plans based on side projection and the TMR algorithm. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66 (6), pp.s13-s27.

Remondino, F., El-Hakim, S., 2006. Image-based 3D modelling: a review. *The Photogrammetric Record*, 21(115), pp. 269-291.

Rissanen, J., 1978. Modeling by the shortest data description. *Automatica*,14, pp. 465-471.

Rottensteiner, F., and Briese, Ch., 2003. Automatic generation of building models from LiDAR data and the integration of aerial images, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXIV(3/W13).

Rottensteiner, F., and Jansa, J., 2002. Automatic extraction of building from LIDAR data and aerial images. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 34(Part 4), pp. 295-301.

Rottensteiner, F., Sohn, G., Gerke, M., Wegner, J. D., 2012. ISPRS test project on urban classification and 3D building reconstruction. http://www.commission3.isprs.org/wg4.

Rottensteiner, F., Sohn, G., Gerke, M., Wegner, J.D., Breitkopf, U., Jung, J., 2014. Results of the ISPRS benchmark on urban object detection and 3D building reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93, pp. 256-271.

Rottensteiner, F., Sohn, G., Jung, J., Gerke, M., Baillard, C., Benitex, S. and Breitkopf, U., 2012. The ISPRS benchmark on urban object classification and 3D building reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, I(3), pp. 293-298.

Rottensteiner, F., Trinder, J., Clode, S., Kubik, K., 2005. Automated delineation of roof planes from LiDAR data. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36 (Part 3/W4), pp. 221–226.

Rutzinger, M., Rottensteiner, F., 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(1), pp.11-20.

Sampth, A., and Shan, J., 2007. Building boundary tracing and regularization from airborne lidar point clouds. *Photogrammetric Engineering and Remote Sensing*, 73(7), pp. 805-812.

Sampath, A., Shan J., 2010. Segmentation and reconstruction of polyhedral building roofs from aerial lidar point clouds. *IEEE Transactions on Geoscience and Remote Sensing*, 48 (3), pp. 1554–1567

Satari, M., Samadzadegan, F., Azizi, A., Maas, H. G., 2012. A multi-resolution hybrid approach for building model reconstruction from LIDAR data. *The Photogrammetric Record*, 27(139), pp. 1554-1567.

Sohn, G., and Dowman, I., 2007. Data fusion of high-resolution satellite imagery and lidar data for automatic building extraction. *ISPRS Journal of Photogrametry and Remote Sensing*, 62(1), pp. 43-63.

Sohn, G., Huang, X., Tao, V., 2008. Using a binary space partitioning tree for reconstructing polyhedral building models from airborne LiDAR data. *Photogrammetric Engineering & Remote Sensing*, 74 (11), pp. 1425–1438.

Sohn, G., Jung, J., Jwa, Y, Armenakis, C., 2013. Sequential modelling of building rooftops by integrating airborne LiDAR data and optical imagery: preliminary results. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, II-3(W1), pp. 27-33.

Sohn, G., Jwa, Y., Jung, J., Kim, H. B., 2012. An implicit regularization for 3D building rooftop modeling using airborne data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, I(3), pp. 305-310.

Song, W. and Haithcoat, T. L., 2005. Development of comprehensive accuracy assessment indexes for building footprint extraction. *IEEE Transactions on Geoscience and Remote Sensing*, 43(2), pp. 401-404.

Tarsha-Kurdi, F., Landes, T., Grussenmeyer, P., 2008. Extended RANSAC algorithm for automatic detection of building roof planes from lidar data. *The photogrammetric journal of Finland*, 21(1), pp. 97–109.

Tian, Y., Gerke, M., Vosselman, G., Zhu, Q., 2008. Automatic edge matching across an image sequence based on reliable points. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Science*, 37(Part 3B), pp. 657-662.

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

United Nation, 2014, World urbanization prospects: the 2014 revision. Available online: http://esa.un.org/unpd/wup/highlights/wup2014-highlights.pdf (assessed on 12 March, 2016)

Veltkamp, R. C., 2001. Shape matching: similarity measures and algorithms. In Proc. *Shape Model. Int.*, May 2001, pp. 188-197.

Verma, V., Kumar, R., Hsu, S., 2006. 3D building detection and modeling from aerial lidar data. *The IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR'06. IEEE Computer Society, Washington, DC, pp. 2213-2220.

Viola, P., Wells, W. M., 1997. Alignment by maximization of mutual information. *International Journal of Computer Vision*, 24(2), pp. 137-154.

Vosselman, G., 1999. Building reconsstruction using planar faces in very high density height data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 32(Part 3/2W5), pp. 87-92.

Vosselman, V., Maas, H.-G. (Eds.), 2010. Airborne and terrestrial laser scanning. Taylor & Francis.

Wang, L., Neumann, U., 2009. A robust approach for automatic registration of aerial images with untextured aerial LiDAR data. In Proc. *2009 IEEE Computer Society Conf., CVPR 2009*, pp.2623-2630.

Wang, R., 2013. 3D building modeling using images and LiDAR: a review. *International Journal of Image and Data Fusion*, 4(4), pp. 273-292.

Weidner, U., Förstner, W., 1995. Towards automatic building extraction from high resolution digital elevation models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 50(4), pp. 38-49.

Wichman, A. and Kada, M., 2014. 3D building adjustment using planar half-space regularities. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3, pp. 189-196.

Wolfson, H. J., Rigoutsos, I., 1997. Geometric hashing: an overview. *IEEE Computational Science and Engineering*, 4(4), pp. 10-21.

Wunsch, P. and Hirzinger, G.,1996. Registration of CAD-models to images by iterative inverse perspective matching. *ICPR'96*.

Xiong, B., Oude Elberink, S., Vosselman, G., 2014. A graph edit dictionary for correcting errors in roof topology graphs reconstructed from point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93, pp. 227-242.

Yan, J., Shan, J., Jiang, W., 2014. A global optimization approach to roof segmentation from airborne lidar point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 94, pp. 183-193.

Yang, B., Chen, C., 2015. Automatic registration of UAV-borne sequent images and LiDAR data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 101, pp. 262-274.

Zhang, D., Lu, G., 2004. Review of shape representation and description techniques. *Pattern Recognition*, 37, pp. 1-19.

Zhang, J., 2010. Multi-source remote sensing data fusion: status and trends. *International Journal of Image and Data Fusion*, 1(1), pp.5-24.

Zhang, J., 2015. Single tree detection from airborne laser scanning data: a stochastic approach. PHD thesis, York University.

Zhang, W., Grussenmeyer, P., Yan, G., Mohamed, M., 2011. Primitive-based building reconstruction by integration of LiDAR data and optical imagery. *International*

Archives of Photogrammetry, Remote Sensing and Spatial Information Systems, 38 (Part 5-W12).

Zhou, Q.Y., Neumann, U., 2008. Fast and extensible building modeling from airborne LiDAR data. In: *Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM GIS 2008 (on CD-ROM).

Zhou, Q. Y., Neumann, U., 2012. 2.5D building modeling by discovering global regularities. *The IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 16–21 June, IEEE Computer Society, Providence, RI, USA, pp. 326–333.

Zitova, B. and Flusser, J., 2003. Image registration method: a survey. *Image and Vision Computing*, 21, pp. 977-1000.

Zou, Z.,Yun, Y., Sun, J., 2006. Entropy method for determination of weight of evaluating indicators in fuzzy synthetic evaluation for water quality assessment. *Journal of Environmental Sciences*, 18(5), pp. 1020-1023.